Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie

Wydział Informatyki, Elektroniki i Telekomunikacji

Katedra Informatyki

Rozprawa doktorska

Automatyczna ekstrakcja relacji semantycznych z tekstów w języku polskim

Aleksander Smywiński-Pohl

Promotor:
prof. dr hab. Wiesław Lubaszewski

Kraków 2015

OŚWIADCZENIE AUTORA PRACY

Oświadczam, świadomy odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem osobiście i samodzielnie, i nie korzystałem ze źródeł innych niż wymienione w pracy.

.....

PODPIS

Spis treści

Streszczenie

Praca dotyczy ekstrakcji informacji z polskich tekstów. Zasadniczym jej tematem jest rozpoznawanie relacji semantycznych w oparciu o automatycznie konstruowane wzorce ekstrakcyjne. Przedstawiono w niej również algorytm selekcji zdań, na podstawie których tworzony jest model ekstrakcji oraz algorytmy ujednoznaczniania i semantycznej klasyfikacji wyrażeń języka polskiego.

Wzorce ekstrakcyjne są konstruowane na podstawie przykładowych zdań zawierających wyrażenia połączone relacjami oraz wyposażane są w ograniczenia semantyczne zdefiniowane z wykorzystaniem pojęć ontologii Cyc. Ograniczenia określane są na podstawie trzech metod: ręcznej oceny zdań, predykatów ontologii Cyc oraz danych znajdujących się w DBpedii.

Przeprowadzono szereg eksperymentów weryfikujących skuteczność opisywanych algorytmów, w szczególności dotyczących ekstrakcji relacji *całość-część*. Pokazują one, że użycie ograniczeń semantycznych prowadzi do istotnej poprawy precyzji ekstrahowanych informacji. Porównanie wyników ekstrakcji dla ograniczeń uzyskanych na różne sposoby pozwala obronić tezę pracy o możliwości automatycznej ekstrakcji relacji semantycznych z wykorzystaniem algorytmu hybrydowego, korzystającego z symbolicznych zasobów wiedzy.

Słowa kluczowe: ekstrakcja relacji, wzorce ekstrakcyjne, relacja całość-część, ograniczenia semantyczne, język polski, Cyc, Wikipedia

1. Wstęp

1.1. Teza rozprawy doktorskiej

Tematem niniejszej pracy jest automatyczna ekstrakcja relacji semantycznych z tekstów w języku polskim. Teza pracy jest następująca: **możliwe jest skonstruowanie hybrydowego algorytmu ekstrakcji wybranych relacji semantycznych z tekstów w języku polskim, który:**

1. **dawałby wyniki bardziej precyzyjne niż te, otrzymywane za pomocą algorytmów statystycznych,**
2. **nie byłby ograniczony do pojedynczej dziedziny wiedzy,**
3. **wymagałby mniejszego nakładu pracy ręcznej, niż algorytm wytrenowane na ręcznie oznakowanym zbiorze uczącym.**

Poszczególne elementy tezy wymagają doprecyzowania. W pierwszej kolejności należy wyjaśnić co rozumiemy przez *algorytm hybrydowy* – jest to algorytm, który wykorzystuje elementy charakterystyczne dla dwóch paradygmatów szeroko stosowanych w przetwarzaniu języka naturalnego, tj. paradygmatu *statystycznego* oraz paradygmatu *symbolicznego*.

Cechą charakterystyczną pierwszego paradygmatu jest wykorzystywanie dużych zbiorów danych, najczęściej dużych korpusów tekstów. Stosując proste algorytmy statystyczne (np. opierające się na prawdopodobieństwie warunkowym występowania różnych zdarzeń językowych) lub bardziej zaawansowane algorytmy uczenia maszynowego oczekuje się, że odpowiednie modele zjawisk językowych, zostaną zbudowane automatycznie. Takie podejście stosowane jest np. przez firmę Google w jej systemie tłumaczenia maszynowego[1]. Zaletą systemów tego rodzaju jest to, że nie wymagają dostosowywania do konkretnego języka naturalnego, natomiast wadą, że nie zawsze dostępne są zasoby pozwalające na wytrenowanie odpowiednio precyzyjnych modeli.

W przeciwieństwie do paradygmatu statystycznego, w paradygmacie symbolicznym podstawowym zasobem wykorzystywanym przez algorytmy są bazy wiedzy, opisujące zjawiska językowe w sposób symboliczny. Najczęściej bazy te konstruowane są ręcznie, co wymaga dużych nakładów finansowych. Ponadto, przy ich konstrukcji zakłada się określoną teorię funkcjonowania języka, która może okazać się niekompletna lub niedokładna. Z drugiej jednak strony, działanie algorytmu jest łatwiejsze do zrozumienia, gdyż np. pośrednie wyniki jego działania można zinterpretować w kontekście wykorzystywanej teorii. Przykładami zasobów wykorzystywanych w algorytmach tego rodzaju są WordNet [41], FrameNet [8] oraz ontologia Cyc [66].

Pierwszy podpunkt tezy, mówiący o tym, że wyniki algorytmu hybrydowego powinny być lepsze niż wyniki otrzymywane przez algorytm statystyczny, nie oznacza, że algorytm ten nie może posługiwać

[1] Dostępne `http://translate.google.com`.

się danymi statystycznymi. Teza to ma na celu podkreślenie charakteru algorytmu – jego hybrydowości, tzn. uwzględniania zarówno danych korpusowych, jak i symbolicznych. Aby obronić tę część tezy należy pokazać, że uwzględnienie cech symbolicznych prowadzi do poprawy precyzji ekstrakcji. Jeśli taka poprawa nie następuje, oznaczałoby to, że dodatkowy nakład związany z analizą symboliczną nie jest uzasadniony.

Nie należy jednak przyjmować, że teza ta jest trywialna – badania pokazują bowiem, że wykorzystanie analizy symbolicznej, która wymaga między innymi ujednoznacznianie sensu wyrażeń językowych, wcale nie musi poprawić wyników algorytmów przetwarzania tekstu. Wynika to przede wszystkim z problemów jakie pojawiają się przy ujednoznacznianiu – jego jakość może być na tyle niska, że dodatkowa poprawa wyników, otrzymywana dzięki wykorzystaniu zasobów symbolicznych, jest niwelowana przez błędy pojawiające się na tym etapie [3].

Drugi podpunkt tezy wymaga, aby algorytm sprawdzał się w możliwie najszerszym spektrum zastosowań. Chodzi tu przede wszystkim o pokazanie, że analiza semantyczna prowadzona w trakcie ekstrakcji informacji nie jest trywialna. Przyjmując bowiem, że algorytm przeznaczony byłby do ekstrakcji informacji z wąskiej dziedziny wiedzy, możliwe byłoby uzyskanie znacznie lepszych wyników, bowiem problem wieloznaczności danych jest w takich warunkach istotnie ograniczony. Ponadto, zakładając określoną dziedzinę wiedzy można wykorzystać ontologię lub słownik dziedzinowy, które definiują i klasyfikują wyłącznie te pojęcia oraz relacje, które występują w danej dziedzinie. Konstrukcja algorytmu niezależnego od dziedziny jest o tyle utrudniona, że konieczne jest wykorzystanie bardzo obszernych zasobów, które obejmują swoim zakresem jak największą liczbę pojęć, należących do różnych dziedzin. Biorąc pod uwagę ten fakt, problemy ujednoznaczniania sensu, klasyfikacji wyrażeń oraz rozpoznawania relacji semantycznych muszą być faktycznie rozwiązane.

Ostatnie wymaganie dotyczy nakładów pracy ręcznej, które są konieczne do realizacji założonego przedsięwzięcia. Wypracowana metoda powinna go minimalizować, choć trudno oczekiwać, aby można ją było całkowicie wyeliminować. Aby wykazać prawdziwość tej części tezy, konieczne jest zaimplementowanie różnych wariantów algorytmu – jednego opierającego się w istotnej mierze na danych pozyskanych ręcznie oraz drugiego, w którym przeważająca ilość danych, pozyskana byłaby w sposób automatyczny. Jeśli wyniki drugiego wariantu okażą się lepsze od wyników wariantu pierwszego, ta część tezy zostanie również obroniona.

1.2. Struktura rozprawy doktorskiej

Rozwiązanie tak złożonego problemu jakim jest automatyczna ekstrakcja relacji nie jest zadaniem prostym, dlatego też opis jego rozwiązania jest dość obszerny. Chcąc ułatwić czytelnikowi poruszanie się po niniejszym tekście, treść pracy podzielona została na szereg rozdziałów. I tak w rozdziale 2 przedstawione zostało wprowadzenie do zagadnienia ekstrakcji informacji. W szczególności zdefiniowano w nim samo pojęcie *ekstrakcji informacji*, tak jak jest ono rozumiane przez autora, przedstawiono w nim szereg problemów, które stanowią o złożoności tego zagadnienia oraz omówiono zadania definiowane w ramach tej dziedziny wiedzy, z szczególnym uwzględnieniem *ekstrakcji relacji semantycznych*.

W rozdziale 3 przedstawiona jest terminologia wykorzystywana w dalszych częściach pracy. Rozdział ten jest niezbędny, ponieważ praca niniejsza należy do dziedziny informatyki, ale nie może abstrahować od terminów wykorzystywanych w językoznawstwie. Szczególny nacisk został położony na omówienie *relacji semantycznych*, gdyż to one są przedmiotem ekstrakcji oraz *sieci semantycznych*, gdyż stanowią one podstawowy zasób wykorzystywany do rozwiązania postawionego problemu.

Rozdział 4 zawiera omówienie historii badań nad ekstrakcją informacji. W pierwszej kolejności omówiono w nim rozwój tej dziedziny na przykładzie języka angielskiego, gdyż, podobnie jak w innych problemach podejmowanych w dziedzinie przetwarzania języka naturalnego, najwcześniej zajęto się tym problemem w kontekście tego języka. W drugiej części tego rozdziału omówione są również postępy jakie zostały dokonane w tej dziedzinie dla języka polskiego. Stanowią one zasadniczy punkt odniesienia dla badań przedstawionych w niniejszej rozprawie.

Zasadnicza część pracy rozpoczyna się w rozdziale 5 – w nim opisana jest struktura głównego algorytmu służącego do ekstrakcji relacji semantycznych. Zostały w nim przedstawione również algorytmy pomocnicze niezbędne do realizacji tego zadania oraz zasoby wykorzystywane przez algorytm. Zasoby te są szczegółowo omówione w rozdziale 6, natomiast algorytmy pomocnicze w rozdziale 7.

Rozdział 8 zawiera szczegółowy opis algorytmu służącego do konstrukcji *wzorców ekstrakcyjnych* oraz sposób ich wykorzystania do ekstrakcji relacji semantycznych. Jest to niewątpliwie najważniejszy rozdział niniejszej pracy, a fragment, który zasługuje na największą uwagę dotyczy różnych metod określania *ograniczeń semantycznych* wykorzystywanych do ekstrakcji relacji. Przedstawione są w nim zasady ręcznej oraz automatycznej konstrukcji tych ograniczeń.

Kolejne dwa rozdziały, tj. 9 oraz 10, opisują proces konstrukcji wzorców ekstrakcyjnych dla relacji *całość-część* oraz wyniki eksperymentów przeprowadzonych z użyciem tych wzorców. Eksperymenty te stanowią podstawę obrony tezy przedstawionej na początku niniejszego rozdziału.

Rozdział 11 zawiera wnioski wynikające z niniejszej pracy. Przede wszystkim omówione są wnioski płynące bezpośrednio z przeprowadzonych eksperymentów. Przedstawiony jest również szereg zagadnień związanych z ekstrakcją relacji semantycznych, które mogłyby poprawić jakość uzyskanych wyników, ale nie zostały zbadane przez autora.

1.3. Podziękowania

Przygotowanie niniejszej pracy nie byłoby możliwe bez pomocy licznej grupy osób. W pierwszej kolejności chciałbym podziękować swojej rodzinie, w szczególności żonie Annie, za wyrozumiałość oraz cierpliwość. Chciałbym bardzo serdecznie podziękować swojemu promotorowi, prof. dr. hab. Wiesławowi Lubaszewskiemu, za nieustanne wsparcie merytoryczne udzielane na wszystkich etapach powstawania tej pracy. Ponadto chciałbym podziękować dr. hab. inż. Markowi Kisielowi-Dorohinickiemu, dr. inż. Bartoszowi Ziółce, Mike'owi Bergmanowi, pracownikom Małopolskiego Centrum Przedsiębiorczości oraz pracownikom Wydziału Zarządzania i Komunikacji Społecznej Uniwersytetu Jagiellońskiego za możliwość udziału w projektach badawczych, które rozwinęły moje umiejętności naukowe i dzięki którym uzyskałem wsparcie materialne w trakcie wielu lat powstawania niniejszej pracy. Osobne podziękowanie kieruję do Jakuba Perlińskiego, za przetłumaczenie znacznej ilości pojęć ontologii Cyc na język polski, Krzysztofa Wróbla, z którym współpracowaliśmy w projekcie ekstrakcji informacji z Wikipedii oraz Sabinie Prajsner-Szatyńskiej, która oceniała wyniki ekstrakcji algorytmu. Podziękowania pragnę również złożyć Sebastianowi Zontkowi, prezesowi firmy Wisdio S.A., za szereg uwag dzięki którym prezentowany algorytm będzie miał większe zastosowanie praktyczne.

2. Wprowadzenie do zagadnienia ekstrakcji informacji

2.1. Pojęcie ekstrakcji informacji

Ekstrakcja informacji (ang. *information extraction – IE*) jest jednym z problemów podejmowanych w ramach przetwarzania języka naturalnego (ang. *natural language processing – NLP*). Obejmuje ona obszar badań leżący na styku prostych metod opierających się na dopasowaniu wzorców formalnych (np. wyrażeń regularnych) do tekstu oraz metod uwzględniających semantykę języka [58, s. 725-727]. Ambicją badaczy zajmujących się ekstrakcją informacji nie jest jednak stworzenie algorytmu, który dokonywałby pełnej interpretacji analizowanego tekstu, gdyż ten problem należy do obszaru badań określanego mianem automatycznego rozumienia tekstu. Niemniej badania te istotnie przekraczają proste metody, w których wydobycie określonej informacji sprowadza się do rozpoznania charakterystycznego układu tekstu (jak np. w systemach wydobywających tytuł, autorów oraz słowa kluczowe z publikacji naukowych) czy określonego układu znaczników HTML (jak to ma miejsce w systemach służących do automatycznej konwersji stron internetowych posiadających sztywną strukturę do postaci danych tabelarycznych). Tym, co odróżnia ekstrakcję informacji, od tego drugiego podejścia i co zbliża do automatycznego rozumienia tekstu jest centralna rola semantyki w tej metodzie przetwarzania danych.

W historii badań nad ekstrakcją informacji można spotkać się z wieloma jej definicjami. Moens [92] przytacza kilka spośród nich, poczynając od Riloff i Lorenzena [134], którzy omawiając system AutoSlog-TS, przedstawiają następującą definicję:

> IE systems extract domain-specific information from natural language text. The domain and types of information to be extracted must be defined in advance. IE system often focus on object identification, such as references to people, places, companies, and physical objects. [...] Domain-specific extraction patterns (or something similar) are used to identify relevant information.[1]

Moens zwraca uwagę, że definicja ta reprezentuje tradycyjne rozumienie tego, czym jest ekstrakcji informacji. W szczególności występuje w niej odwołanie do wzorców ekstrakcyjnych (*domian-specific extraction patterns*), które dostosowane są do ściśle określonej dziedziny wiedzy. Definicja ta również zakłada, że domena i rodzaj informacji, które mają zostać wyekstrahowane, są z góry znane, a zadanie systemu polega na identyfikacji obiektów takich jak ludzie, miejsca czy przedsiębiorstwa.

Definicja ta jest uznawana przez Moens za zbyt wąską, bo choć w praktyce system ekstrakcji informacji zakłada istnienie schematu danych wykorzystywanego do organizacji danych pozyskiwanych z analizowa-

[1] Systemy ekstrakcji informacji ekstrahują informacje należące do określonej dziedziny z tekstów w języku naturalnym. Dziedzina oraz typ informacji, które mają zostać wyekstrahowane muszą być zdefiniowane z góry. System ekstrakcji informacji często koncentruje się na identyfikacji elementów, takich jak odniesienia do ludzi, miejsc, przedsiębiorstw oraz obiektów fizycznych. [...] Dostosowane do wybranej dziedziny wzorce ekstrakcyjne (lub coś podobnego) są używane do identyfikacji istotnych informacji. (tłum. aut.)

nego zbioru tekstów, to ten schemat nie powinien być dostosowany wyłącznie do jednej dziedziny wiedzy. Innymi słowy, oczekujemy aby system ekstrakcji informacji był uniwersalny w swym sposobie działania. Podobnie, podkreślenie roli wzorców ekstrakcyjnych jest ważne, ale obecnie nie oczekuje się, że będą one budowane dla każdej dziedziny wiedzy z osobna.

Moens następnie analizuje definicję[2] Cowiego i Lehnert [30]:

> [information extraction] isolates relevant text fragments, extracts relevant information from the fragments, and then pieces together the extracted information in a coherent framework. [...] The goal of information extraction research is to build systems that find and link relevant information while ignoring extraneous and irrelevant information.[3]

Definicja ta pojawia się w artykule *Information extraction*, podsumowującym rozwój tej dziedziny do roku 1996, ze szczególnym uwzględnieniem *Message Understanding Conference* (MUC), której głównym celem była stymulacja rozwoju oraz ewaluacja systemów ekstrakcji informacji.

Moens zauważa, że definicja ta jest bliska współczesnym definicjom ekstrakcji informacji. W szczególności brakuje w niej odwołań do wzorców ekstrakcyjnych – jest ona więc definicją uniwersalną. Kładzie również nacisk na charakterystyczne dla ekstrakcji informacji ignorowanie tych treści, które nie są istotne z punktu widzenia prowadzonej analizy. Moens zauważa jednak, że ekstrakcja informacji nie musi ograniczać się do tekstów w języku naturalnym. Pojęcie to można rozszerzyć na informacje zawarte w dokumentach dźwiękowych i audiowizualnych, dlatego autorka proponuje następującą definicję [92, s. 4]:

> Information extraction is the identification, and consequent or concurrent classification and structuring into semantic classes, of specific information found in unstructured data sources, such as natural language text, making the information more suitable for information processing tasks.[4]

Definicja zaproponowana przez Moens jest skonstruowana w ten sposób, aby mogła obejmować swoim zasięgiem również wydobywanie informacji z danych takich jak materiały audiowizualne czy pliki graficzne, przez co jest bardziej uniwersalna. Wprowadza ona jednak pojęcie nieustrukturyzowanych źródeł wiedzy, które choć jest intuicyjnie zrozumiałe, nie posiada precyzyjnej definicji. Cel ekstrakcji informacji również nie jest zbyt jasny w tym kontekście – chodzi o sprawniejsze wykorzystanie wyekstrahowanej wiedzy, ale sprawność ta nie jest określona (chodzi o większą wydajność, czy większą precyzję?). Definicja ta staje się zatem zbyt szeroka.

Intencja Moens może zostać wyrażona w inny sposób. W językoznawstwie od dawna znane jest rozróżnienie na *język przedmiotowy* oraz *metajęzyk* [73, s. 14-17]. Alfred Tarski w artykule *The semantic conception of truth and the foundations of semantics* [151] wykorzystuje to rozróżnienie w celu zdefiniowania semantycznej definicji prawdy:

> [...] we have to use two different languages in discussing the problem of the definition of truth and, more generally, any problems in the field of semantics. The first of these languages is the language which is „talked about" and which is the subject matter of the whole discussion;

[2] Definicja ta chronologicznie jest wcześniejsza niż definicja Riloff i Lorenzena.

[3] [Ekstrakcja informacji] izoluje istotne fragmenty tekstu, ekstrahuje istotne informacje z tych fragmentów, a następnie łączy wyekstrahowane informacje w spójną całość. Celem badań w zakresie ekstrakcji informacji jest zbudowanie systemów, które znajdują i łączą istotne informacje, ignorując uboczne oraz nieistotne informacje. (tłum. aut.)

[4] Ekstrakcja informacji polega na identyfikacji oraz sekwencyjnym, bądź współbieżnym klasyfikowaniu oraz strukturyzowaniu w klasy semantyczne specyficznych informacji znalezionych w nieustrukturyzowanych źródłach wiedzy, takich jak teksty w języku naturalnym, w celu sprawniejszego wykorzystania tej wiedzy w zadaniach przetwarzania informacji. (tłum. aut.)

> [...]. The second is the language in which we „talk about" the first language [...]. We shall refer to the first language as „the object language", and to the second as „the meta-language".[5]

Zwraca on również uwagę, że przedstawione rozróżnienie przydatne jest w każdym kontekście, w którym mowa jest o semantyce języka. Zamiast odwoływać się do pojęcia nieustrukturyzowanych danych, które sugeruje, że dane tekstowe, czy jakiekolwiek inne dane, z których chcemy ekstrahować informacje nie posiadają struktury, możemy odwołać się do koncepcji meta-języka. Możemy wtedy powiedzieć, że dane takie posiadają strukturę (np. w kontekście tekstu są to zdania i słowa, w kontekście materiałów audiowizualnych ramki dźwięku i obrazu, etc.), ale struktura ta jest wyrażona wyłącznie w terminach meta-języka (zdania, słowa, ramki, etc.). Istotę ekstrakcji informacji stanowi przejście od opisu w terminach meta-języka do opisu z użyciem języka przedmiotowego. A tylko informacje wyrażone w tym drugim języku mogą być bezpośrednio wykorzystane w systemach informatycznych stworzonych do ich przetwarzania.

Opis danych zawartych np. w artykule dotyczącym Banku Japonii w terminach meta-języka wyglądałby następująco: tekst składa się z 12 paragrafów, 85 zdań i 900 słów, 30% zdań zawiera ponad 20 słów, słowo „ jest" występuje 24 razy, a słowo „waluta" 3 itd. Natomiast ten sam artykuł opisany w języku przedmiotowym będzie mówił jaki jest cel inflacyjny Banku, jaki jest aktualny poziom deflacji, jaka jest aktualna i planowana wartość jena w stosunku do dolara, itp. Informacje wyrażone w meta-języku nie mogą być wykorzystane bezpośrednio np. w aplikacji dokonującej automatycznych inwestycji walutowych, gdyż język ten nie zawiera pojęć takich jak „waluta" czy „kurs". Co prawda w języku tym można odnieść się do słowa „waluta" ale jest ono jedynie cytowane. W przeciwieństwie do meta-języka, język przedmiotowy zawiera te terminy.

Biorąc pod uwagę tradycję wykorzystania terminów „ język przedmiotowy" oraz „meta-język" w językoznawstwie i filozofii proponujemy następującą definicję ekstrakcji informacji:

> *Ekstrakcja informacji* jest procesem nadawania znaczenia (interpretacji), w którym przechodzi się od opisu danych w terminach meta-języka, do opisu w terminach języka przedmiotowego, dzięki czemu uzyskane informacje mogą być bezpośrednio wykorzystane w zadaniach przetwarzania informacji. Ekstrakcja informacji zwykle ogranicza się do interpretowania pewnego podzbioru dostępnych informacji, istotnych z punktu widzenia realizowanego zadania.

W dalszej części niniejszej pracy będziemy przyjmować tę definicję jako obowiązującą. W szczególności prezentowany algorytm ekstrakcji relacji został skonstruowany w taki sposób, aby ograniczenie, o którym mowa w drugiej części definicji, miało jak najmniejszy zasięg, tzn. tak aby opracowany algorytm mógł być wykorzystywany niezależnie od dziedziny zastosowania.

2.2. Uzasadnienie potrzeby badań

Automatyczne przetwarzania języków naturlanych jest aktywnym obszarem badań naukowych. W 2011 roku program Watson stworzony przez IBM wygrał konkurs Jeopardy![6], w którym uczestnicy odpowiadali (a właściwie zadawali pytanie zawierające odpowiedź) na pytania wyrażone w języku naturalnym. Pokonał on dwóch ludzi, którzy wcześniej wielokrotnie wygrywali ten konkurs, wykazując się

[5] [...] musimy więc używać dwóch języków w kontekście dyskusji nad problemem definicji prawdy i szerzej, każdego problemu w obszarze semantyki. Pierwszym z tych języków jest język „o którym jest mowa" i który jest przedmiotem całej dyskusji; [...] Drugi jest językiem „w którym mówimy o" pierwszym języku [...]. Do pierwszego języka będziemy odnosić się mianem „ języka przedmiotowego", a do drugiego mianem „meta-języka".

[6] `http://www.nytimes.com/2011/02/17/science/17jeopardy-watson.html?pagewanted=all&_r=0`

niezwykłą wiedzą oraz znakomitą pamięcią obejmującą szeroki zakres dziedzin życia. Wygrana ta porównywana była do wcześniejszego tryumfu IBMa, którego Deep Blue pokonał arcymistrza szachowego Gariego Kasparowa. Podobnie niejedna osoba miała do czynienia z wirtualnym asystentem Siri, wbudowanym w telefony marki Apple, który wydaje się rozumieć bardzo szeroki zakres poleceń głosowych i reagować stosowanie do zamierzeń jego użytkownika. Czy zatem nie osiągnęliśmy już celu wyznaczonego przez Alana Turinga [154], jakim było zbudowanie myślącej maszyny?

Analizując wynik badań publikowanych w czasopismach poświęconych przetwarzaniu języka, można jednak dojść do innych wniosków. Cząstkowe zadania, takie jak znakowanie morfosyntaktyczne, rozstrzyganie wieloznaczności, parsowanie zdań, czy wreszcie ekstrahowanie informacji, nadal nie są realizowane w sposób całkowicie satysfakcjonujący. Przykładowo – aktualnie najlepszy polski tagger morfosyntaktyczny cechuje się 90% precyzją w przypisywaniu wartości kategorii gramatycznych do słów [158]. Najlepsze algorytmy ekstrahujące relacje z angielskich tekstów osiągają jakość (F_1) na poziomie 70-80% [21, 40, 39], zbyt niską aby uzyskiwane wyniki można było wykorzystywać praktycznie. Natomiast poprawne rozpoznawanie mowy ogranicza się do krótkich wypowiedzi zawierających co najwyżej kilka słów [164]. Dlatego też wyniki osiągane przez komercyjne programy należy traktować z pewną rezerwą – niewątpliwie przyczyniają się one do popularyzacji osiągnięć z dziedziny przetwarzania języków naturalnych, lecz daleko im do kompetencji językowej ludzi.

Niniejsza praca jest próbą podniesienia jakości uzyskiwanych rezultatów w dziedzinie ekstrakcji informacji w języku polskim. Choć nie przynosi ona ostatecznych rozstrzygnięć poruszanych problemów, pokazuje jednak i weryfikuje jeden z możliwych kierunków rozwoju systemów ekstrahujących informacje. Istotny nacisk położony został również na automatyzację tego procesu, co nie można być zweryfikowane w odniesieniu do systemów komercyjnych.

2.2.1. Wyszukiwanie informacji w sieciach rozległych

Jednym z obszarów, w których ekstrakcja informacji może przynieść istotną poprawę jest wyszukiwanie informacji w sieciach rozległych. Nieustannie rosnąca ilość danych tekstowych powoduje, że standardowe mechanizmy wyszukiwania, oparte na dopasowaniu słów kluczowych oraz strukturze odnośników hipertekstowych dobrze sprawdzają się jedynie przy wyszukiwaniu pojedynczych informacji (tzn. znajdujących się na jednej stronie internetowej lub kilku powiązanych stronach). Jeśli informacja jest rozproszona w wielu dokumentach, ze względu na liczbę zwracanych rezultatów, ich ręczne przeglądanie jest nieefektywne.

Zastosowanie mechanizmów ekstrakcji informacji pozwoliłoby na znalezienie poszukiwanej informacji bez potrzeby przeglądania wielu dokumentów, zawierających jedynie cząstkowe odpowiedzi na zadane pytanie. W szczególności wynikiem algorytmu mogłaby być precyzyjna odpowiedź zbudowana na podstawie wielu cząstkowych wyników.

To zastosowanie ekstrakcji informacji można zilustrować następującym przykładem. Przypuśćmy, że firma produkująca nawozy zamierza zainwestować w nową fabrykę w jednym z krajów azjatyckich. Swoją decyzję uzależnia jednak od wielu czynników, w tym obowiązujących w danym kraju regulacji prawnych dotyczących sposobu transportu oraz przechowywania substancji groźnych dla środowiska. O ile wiele czynników makroekonomicznych dotyczących krajów świata dostępnych jest np. w rocznikach statystycznych, o tyle szczegółowe regulacje prawne nie posiadają takiego zestawienia. Odpowiedź na pytanie, jakie regulacje obowiązują w danym kraju, wymaga zbadania szczegółowych przepisów prawa obowiązujących w tym kraju. Zastosowanie mechanizmów ekstrakcji informacji umożliwiłoby odnalezienie konkretnych przepisów regulujących to zagadnienie oraz sformułowanie poprawnej odpowiedzi.

2.2.2. Analiza informacji zawartych w danych tekstowych

Kolejnym obszarem, w którym ekstrakcja informacji może być bardzo przydatna jest analiza informacji np. na potrzeby gospodarki, kryminalistyki czy bezpieczeństwa wewnętrznego. Analiza danych pochodzących z wielu niezależnych źródeł tekstowych (nie tylko internetowych) może być istotnie usprawniona, jeśli system komputerowy pozwala na wydobywanie (ekstrahowanie) z wielu dokumentów tylko tych informacji, które istotne są z jej punktu widzenia. Precyzyjne, ustrukturyzowane i skondensowane informacje, posiadające uzasadnienie w postaci odnośników do źródeł, pozwalają znacznie przyśpieszyć podejmowanie trafnych decyzji.

Podobnie jak w przypadku wyszukiwania informacji w sieciach rozległych, tak i w analizie dokumentów tekstowych ekstrakcji informacji, może istotnie przyczynić się do poprawy efektywności pracy osób rozwiązujących dany problem. Różnica pomiędzy tymi scenariuszami polega na tym, że w pierwszym wypadku poszukiwana jest odpowiedź na konkretne pytanie, która może być zawarta w wielu dokumentach. W przypadku analizy dokumentów np. w kontekście bezpieczeństwa wewnętrznego, często konieczne jest nie tylko znalezienie odpowiedzi na określone pytanie, ale automatyczne uporządkowanie wielu powiązanych informacji. Zadanie to nie może być zrealizowane, jeśli nie przyjmiemy pewnego schematu (ontologii) uporządkowywania tych informacji. Wiele metod z zakresu ekstrakcji informacji zakłada istnienie takiego schematu, a ich celem jest właśnie wypełnienie tych schematów danymi odnalezionymi w dokumentach.

Przykładem zastosowania mechanizmu ekstrakcji informacji, dającego w rezultacie analizy dokumentów tekstowych ujednolicone dane, może być system służący do automatycznej oceny CV kandydatów do pracy. System oparty o mechanizmy ekstrakcji informacji nie wymagałby przy rejestracji wypełniania długiej ankiety, badającej umiejętności oraz dotychczasową karierę kandydata, ale akceptowałby CV napisane w języku naturalnym, zgodnie z ogólnymi zasadami, bez narzucania jednego, wcześniej ustalonego szablonu. W efekcie analizy informacje zawarte w dokumencie trafiałyby do bazy danych, która pozwalałaby wyszukiwać kandydatów według ujednoliconych kryteriów. System tego rodzaju akceptowałby na wejściu zwykłe dokumenty tekstowe, a na wyjściu produkowałby informacje w ujednoliconym formacie, które można by przetwarzać zarówno za pomocą zapytań *ad-hoc* jak i wyspecjalizowanych algorytmów analitycznych.

2.3. Natura danych a problem ekstrakcji informacji

2.3.1. „Nieprzezroczystość" semantyczna danych tekstowych

Dane tekstowe są semantycznie „nieprzezroczyste" dla algorytmów, tzn. w tekście algorytm ma do czynienia wyłącznie z formą symboli językowych lub nawet tylko z ciągami liter (więcej informacji na temat opozycji *symbol – ciąg znaków* znajduje się w rozdziale 3). O ile więc wyszukiwanie oparte o dopasowanie słów kluczowych może zostać zrealizowane z całkowitym pominięciem znaczenia symboli językowych, o tyle ekstrakcja danych wymaga zidentyfikowania kategorii semantycznych przetwarzanych słów i wyrażeń (porównaj [89, s. 545-546]). Np. jeśli poszukiwane są informacje gospodarcze na temat krajów azjatyckich, w pierwszym rzędzie trzeba określić jakie są nazwy tych krajów. Wprowadzenie frazy „gospodarka krajów azjatyckich" w zwykłej wyszukiwarce nie przyniesie pożądanego rezultatu, jeśli nie jest dostępny dokument, w którym pojawiłoby się takie zestawienie. Charakter danych tekstowych można przeciwstawić danym ustrukturyzowanym, dostępnym np. w relacyjnej bazie danych. W tym drugim przypadku nie powinno być wątpliwości, z których tabel należy pobrać dane, aby uzyskać pożądaną informację, ponieważ

ich struktura jest jawna. Dane tekstowe również posiadają strukturę, ale nie jest ona jawna i musi zostać odtworzona w procesie ekstrakcji informacji.

2.3.2. Wieloznaczność danych tekstowych

Dane tekstowe są wieloznaczne, co powoduje, że na każdym poziomie analizy językowej można uzyskać wiele wyników, spośród których najczęściej tylko jeden jest właściwy w określonym kontekście (porównaj [3]). Poczynając od identyfikacji przynależności formy do określonej jednostki leksykalnej (np. forma `goli`, może przynależeć do jednostek o formach podstawowych: `goły`, `golić`, `gol`, itp.), poprzez określenie wartości kategorii gramatycznych (np. forma `goły` odpowiada *mianownikowi liczby pojedynczej* oraz *wołaczowi liczby pojedynczej* wszystkich *rodzajów męskich* przymiotnika `goły`), oraz znaczenie jednostki (np. wyraz `zamek` może reprezentować **`budowlę`**[7] lub **`mechanizm zamykający drzwi`**), skończywszy na drzewie rozbioru syntaktycznego, na wszystkich poziomach analizy występują wieloznaczności.

Pominięcie problemu rozstrzygania wieloznaczności prowadzi do wyników, które mogą być wieloznaczne lub takich, których nie da się sensownie zinterpretować. Jest to szczególnie istotne, jeśli na podstawie wyników generowanych przez system ekstrakcji, informacje są dalej przetwarzane z wykorzystaniem mechanizmów automatycznego wnioskowania.

Problem ten można bardzo dobrze zilustrować na przykładzie serwisu Google Trends[8]. Jeśli będziemy w nim chcieli porównać popularność dwóch języków programowania: Rubiego i Pythona, szybko okaże się, że ze względu na wieloznaczność tych wyrażeń, prezentowane wyniki nie są wiarygodne. Przykładowo w trendzie Rubiego można zauważyć odnośniki do wydarzeń związanych z aferą Berlusconiego[9]. Użytkownik serwisu może oczywiście wybrać mniej wieloznaczne terminy, np. wpisując *Ruby language* oraz *Python language*, uniknie wieloznaczności, ale wtedy wyniki również nie będą do końca wiarygodne, gdyż użytkownicy wyszukiwarki Google, na bazie której prezentowane są wyniki w Google Trends, znacznie rzadziej korzystają z tego rodzaju jednoznacznych zapytań poszukując informacji na określony temat.

Gdyby w systemie tym zastosowano mechanizmy ekstrakcji informacji oraz uwzględniono problem wieloznaczności, uzyskany wynik mógłby być zdecydowanie bardziej precyzyjny. Inteligentny system posiadałby wiedzę na temat wieloznaczności tych terminów i w tym konkretnym kontekście przedstawił wyniki uwzględniające znaczenie posiadające wspólny nadrzędnik semantyczny – tj. *język programowania*.

2.3.3. Wyrazy pospolite a nazwy własne

Kolejnym problemem, który musi zostać uwzględniony w procesie ekstrakcji informacji są nazwy własne (porównaj [71, s. 19-20]). Z jednej strony nazwy są szczególnie istotne w procesie analizy informacji, gdyż ze względu na swoją wąską dystrybucję najczęściej zawierają istotne informacje, które powinny być uwzględnione przy ekstrakcji. Z drugiej strony, ich liczba jest istotnie większa niż liczba wyrazów pospolitych, przez co w zasadzie nie konstruuje się słowników, które pretendowałyby do obejmowania wszystkich nazw własnych (z wyjątkiem słowników cząstkowych, zawierających np. imiona, nazwiska, czy nazwy geograficzne). Nazwy własne często posiadają wiele wariantów (np. `AMD`: `Advanced Micro Devices`, itp.), niekiedy także synonimów (np. `Cracowia`: `Pasy`), co dodatkowo utrudnia ich analizę. W językach flek-

[7] W pracy przyjęto konwencję, zgodnie z którą napisy, czyli składniki meta-języka, pisane są pismem o stałej szerokości, a symbole językowe, czyli elementy języka przedmiotowego, pogrubionym pismem o stałej szerokości. Rozróżnienie to jest szczegółowo omówione w rozdziale 3.

[8] `http://www.google.com/trends/`

[9] Bohaterka seks-skandalu miała na imię *Ruby*.

syjnych (wliczając w to język polski) problem komplikowany jest również przez fakt, że odmiana tych nazw musi zazwyczaj zostać odgadnięta przez algorytm ujednoznaczniania morfosyntaktycznego, właśnie ze względu na brak odpowiednich słowników.

Problem można łatwo zilustrować na przykładzie systemu, który dobiera reklamy do treści artykułów w portalu internetowym. W najprostszym przypadku taki system mógłby wyszukiwać nazw reklamowanych produktów w treści artykułu i jeśli taka nazwa pojawiłaby się, algorytm dodawałby w treści artykułu odnośnik do reklamowanego produktu. System tego rodzaju może działać całkiem nieźle dla języka angielskiego (choć oczywiście istnieją lepsze metody dobierania reklam), ale ze względu na fleksję języka polskiego algorytm ten nie rozpozna odmienionych form reklamowanych produktów. Jeśli w tekście pojawi się np. wyrażenie *Najnowsza recenzja **Tomb Raidera***, a w bazie produktów będzie występowała gra `Tomb Raider`, to produkt ten nie zostanie wybrany. Użycie słowników fleksyjnych również nie pomaga w tym kontekście, ponieważ żaden z dostępnych w języku polskim słowników fleksyjnych [112, 163, 161] nie zawiera odmiany obcego wyrazu `Raider`.

2.3.4. Wyrażenia wielosegmentowe

Istotną kwestią, który wiąże się z nazwami własnymi, jest analiza wyrażeń wielosegmentowych, takich jak `panna młoda`, czy `Rawa Ruska` (porównaj [71, s. 23-25]), które składają się z wielu słów. Nazw własne oraz inne wyrażenia wielosegmentowe, nie zachowują zasady kompozycjonalności, w myśl której znaczenie wyrażenie złożonego jest sumą znaczeń jego składowych. Z tego względu wyrażenia tego rodzaju muszą być rozpoznane jako całość, w przeciwnym bowiem razie ich analiza będzie co najmniej niedokładna, a w skrajnych przypadkach (np. `Zielona Góra`) może prowadzić do zupełnie błędnych wniosków.

2.3.5. Wyrażenia metaforyczne

Metaforyzacja jest procesem polegającym na tworzeniu nowego znaczenia za pomocą przekształceń dokonywanych na znaczeniach już istniejących w języku. Potocznie przyjmuje się, że metafora to zjawisko należące do języka artystycznego. Jednak XX-wieczne językoznawstwo pokazuje, że metafory występują we wszystkich odmianach języka, w tym także w języku potocznym, a nawet języku nauki (porównaj [64]). Występowanie metafor w tekstach języka naturalnego sprawia ogromny kłopot algorytmom przetwarzania tekstu, gdyż bardzo trudno jest reprezentować nieliteralne (przenośne) znaczenie w systemach formalnych. Na szczęście, wiele z często używanych metafor podlega skostnieniu, przez co można umieścić je w słowniku wraz z wyrażeniami wielosegmentowymi, np. *panna młoda*, *analiza koszykowa*, *teoria względności*, itp.

2.4. Zadania definiowane w ramach ekstrakcji informacji

Ekstrakcja informacji jest złożonym procesem. Kompletny system ekstrakcji informacji wymaga rozwiązania przynajmniej niektórych problemów omówionych w punkcie 2.3. Proces ekstrahowania informacji można jednak podzielić na etapy, które stanowią odrębne problemy badawcze. Zwykle w ramach ekstrakcji informacji wyróżnia się następujące zagadnienia [58, s. 725-727]:

– rozpoznawanie jednostek referencyjnych (ang. *named entity recognition*),

– rozpoznawanie wyrażeń współodnoszących się (ang. *coreference resolution*),

– ekstrakcja relacji semantycznych (ang. *relation extraction*),

– rozpoznawanie wyrażeń temporalnych (ang. *temporal expression recognition*),

– ekstrakcja zdarzeń (ang. *event extraction*),

– wypełnianie szablonów (ang. *template filling*).

2.4.1. Rozpoznawanie jednostek referencyjnych

Rozpoznawanie jednostek referencyjnych[10] polega na określeniu, które spośród wyrażeń występujących w tekście odnoszą się do specyficznych obiektów najczęściej posiadających własną nazwę oraz jaka jest kategoria semantyczna obiektów, do których odnoszą się te wyrażania. Przykładowo w zdaniu:

Korea Północna zagroziła wystrzeleniem pocisku balistycznego w kierunku **USA**.

występują dwie nazwy własne: `Korea Północna` i `USA`. Każde z tych wyrażeń odnosi się do obiektu, któremu moglibyśmy przypisać kategorię semantyczną **`kraju`**. Przypisanie określonej kategorii semantycznej uzależnione jest zwykle od sposobu dalszego wykorzystania ekstrahowanych informacji oraz od dostępnego schematu klasyfikacyjnego. O ile w systemach opisywanych w literaturze liczba tych kategorii może być bardzo niewielka i obejmować tylko zgrubny podział, o tyle zastosowania praktyczne mogą wymagać szczegółowej klasyfikacji.

Przykładowo Jurafsky i współpracownicy [58, s. 728] wymienia następujące kategorie semantyczne dla jednostek referencyjnych:

– ludzie (ang. *people*),

– organizacje (ang. *organizations*),

– miejsca (ang. *locations*),

– podmioty geopolityczne (ang. *geo-political entitites*),

– obiekty użyteczności publicznej (ang. *facilities*),

– pojazdy (ang. *vehicles*).

Podział ten jest jednak bardzo ogólny. Dla kontrastu warto przywołać zadania definiowane w ramach konferencji MUC (Message Understanding Conferenc) [30], gdzie konkurujące systemy ekstrahowały informacje na temat zdarzeń terrorystycznych w Ameryce Południowej. Jednym z warunków zaklasyfikowania danego zdarzenia jako aktu terrorystycznego było to, że celem ataku był **`obiekt cywilny`** lub **`cywile`** (w przeciwieństwie do **`obiektów militarnych`**). Zastosowanie takiej definicji wymagało wprowadzenia istotnego rozgraniczenia pomiędzy **`obiektami cywilnymi`** i **`militarnymi`**. Jeśli system rozpoznawania jednostek referencyjnych nie stosowałby tego rozróżnienia, uzyskiwane przez niego wyniki byłyby mało precyzyjne.

Należy również zauważyć, że problem ten nie ogranicza się wyłącznie do rozpoznawania nazw własnych, ale wszystkich wyrażeń, które w sposób jednoznaczny odnoszą się do obiektów rzeczywistych, bądź dobrze zdefiniowanych obiektów abstrakcyjnych. Często rozpoznawanie jednostek referencyjnych obejmuje również wartości procentowe, daty, godziny czy odniesienia do aktów prawnych.

[10] W polskiej literaturze funkcjonuje również termin *rozpoznawanie jednostek nazewniczy*, porównaj [127].

Tablica 2.1: Przykładowa tabela w relacyjnej bazie danych zawierająca wyniki ekstrakcji relacji *bycia prezydentem państwa*.

prezydent	**państwo**
Park Geun-hye	Korea Południowa
Bronisław Komorowski	Polska
François Hollande	Francja
Evo Morales	Boliwia
Giorgio Napolitano	Włochy

2.4.2. Rozpoznawanie wyrażeń współodnoszących się

Zadanie rozpoznawania *wyrażeń współodnoszących się* (inaczej *koreferencji*) polega na określeniu, które wyrażenia występujące w tekście odnoszą się do tych samych obiektów. Omawiając to zagadnienie najczęściej wskazuje się na zjawiska anafory i katafory, to jest zastępowanie (najczęściej zaimkiem) wyrażenia, które odpowiednio już w tekście wystąpiło, bądź dopiero się pojawi.

> „Groźby Korei Północnej są nierealne. **Jej** zdolność bojowa jest zerowa. **Ja** stoję osobiście na straży integralności **naszego państwa**" – powiedziała prezydent Korei Południowej Park Geun-hye.

W powyższym przykładzie wyrażenie `jej` odnoszące się do **`Korei Północnej`** jest przykładem anafory, natomiast wyrażenie `ja` odnoszące się do **`prezydent Korei Południowej`** przykładem katafory, podobnie jak wyrażenie `naszego państwa`, które odnosi się do **`Korei Południowej`**.

W zadaniu rozpoznawania wyrażeń współodnoszących się można wyróżnić dwa aspekty: pierwszy dotyczący wiązania zaimków z wyrażeniami, które zastępują oraz drugi polegający na rozpoznawaniu innych wyrażeń, w szczególności wariantów nazwy własnej, które posiadają wspólne odniesienie. O ile w obu przypadkach cel jest ten sam, to znaczy zidentyfikowanie i przypisanie wszystkich wyrażeń współodnoszących się do pojedynczego obiektu, o tyle metody stosowane do realizacji tych zadań będą odmienne. Rozpoznawanie odniesień zaimków musi odbywać się poprzez analizę dyskursu i wymaga przynajmniej powierzchniowej analizy syntaktycznej. Natomiast rozpoznawanie wariantów nazwy własnej może być zrealizowane przez zastosowanie słownika wyrażeń wielosegmentowych, w którym poszczególne warianty zgrupowane są razem.

2.4.3. Ekstrakcja relacji semantycznych

Ekstrakcja relacji semantycznych z tekstów polega na identyfikacji relacji semantycznych, które występują pomiędzy wyrażeniami w tekście. Identyfikacja ta obejmuje zarówno rozpoznanie argumentów relacji, ich kolejności oraz rozpoznanie typu relacji. Przykładowo w zdaniu:

> Prezydent *Korei Południowej* **Park Geun-hye** odwiedzając koszary, zagrzewała żołnierzy do walki.

pomiędzy symbolami **`Korea Południowa`** oraz **`Park Geun-hye`**, odpowiadającym wyrażeniom `Korei Południowej` oraz `Park Geun-hye`, występuje relacja *bycia prezydentem państwa*. Celem algorytmu ekstrahującego tę relację mogłoby być wypełnienie tabeli, która zawierałaby pary: (*prezydent*, *państwo*), tak jak zostało to przedstawione w tabeli 2.1.

Wypełnianie tabeli w bazie danych jest typowym zastosowaniem mechanizmu ekstrakcji relacji. Przedstawiony przykład posiada jednak pewne założenia, które choć występują dość powszechnie, nie muszą być spełnione. Pierwsze założenie dotyczy argumentów relacji – w przytoczonym przykładzie są nimi jednostki referencyjne. Choć często w praktycznych zastosowaniach ekstrakcji informacji to założenie jest prawdziwe, algorytmy ekstrakcji relacji mogą również operować na wyrażeniach nominalnych, które nie są nazwami własnymi, lecz rzeczownikami pospolitymi. Na przykład algorytm rozpoznający relację *całość-część*, mógłby określić, że częścią terytorium **`Polski`** są **`obszary nizinne`**, bez wskazania o jakie niziny chodzi. Przypadek wpisuje się bardzo dobrze w to, co rozumiane jest pod pojęciem ekstrakcji relacji.

Drugie założenie występujące w przytoczonym przykładzie dotyczy możliwości wielokrotnego potwierdzenia zachodzenia ekstrahowanej relacji. Aby podwyższyć jakość otrzymywanych wyników, do bazy danych mogłyby trafiać tylko te krotki, których wystąpienie zostało kilkukrotnie potwierdzone. W ogólności założenie to może nie być spełnione, tzn. możemy wymagać aby algorytm rozpoznawał relacje za każdym razem gdy pojawia się ona w tekście.

Trzecie założenie, które nie zostało dość wyraźnie uwypuklone, dotyczy typu relacji. W przytoczonym przykładzie typ relacji został z góry założony. Ostatnio jednak coraz częściej opisywane są systemy ekstrahujące dowolne relacje semantyczne z tekstu [10, 12, 21]. Są to tak zwane *otwarte systemy ekstrakcji relacji*. W systemach tych nie określa się *a priori* zbioru ekstrahowanych relacji, lecz stara się rozpoznać wszystkie występujące relacje semantyczne. Należy jednak zwrócić uwagę, że tego rodzaju rozwiązania nie w pełni odpowiadają przyjętej tutaj definicji ekstrakcji informacji, gdyż ekstrahowane relacje nie podlegają interpretacji (w szczególności, relacja posiadająca wiele reprezentacji tekstowych będzie zwykle traktowana jak wiele odrębnych relacji).

Przedmiotem niniejszej pracy jest ekstrakcja relacji z polskich tekstów. Biorąc pod uwagę wielość dostępnych wariantów ekstrakcji relacji, szczegółowe omówienie wariantu przyjętego w niniejszej pracy zostało przedstawione w punkcie 5.1.

2.4.4. Rozpoznawanie wyrażeń temporalnych

Rozpoznawanie wyrażeń temporalnych zwykle nie stanowi celu samego w sobie, lecz jest istotnym składnikiem w ekstrakcji zdarzeń. Wyrażenia temporalne to wyrażenia odnoszące się do czasu. Można je zaklasyfikować do jednego z trzech typów [58, s. 743]:

– bezwzględne wyrażenia temporalne,

– względne wyrażenia temporalne,

– wyrażenia określające czas trwania.

Bezwzględne wyrażenia temporalne określają czas zajścia jakiegoś zdarzenia w bezwzględnej skali odniesienia (w obszarze kultury europejskiej będzie to kalendarz gregoriański). Na przykład w zdaniu:

> Manewry na Morzu Japońskim odbędą się **15 kwietnia 2013 roku**.

wyrażenie `15 kwietnia 2013 roku` jest bezwzględnym wyrażeniem temporalnym, gdyż określa dokładną datę zdarzenia. Względne wyrażenia temporalne określają wystąpienie określonego zdarzenia jedynie względem innego wydarzenia lub daty:

> Nie wszyscy historycy uważają, że zrzucenie drugiej bomby atomowej na Nagasaki, **3 dni po zbombardowaniu Hiroszimy**, było przyczyną zakończenia wojny z Japonią.

W przytoczonym zdaniu wyrażenie `3 dni po zbombardowaniu Hiroszimy` jest względnym wyrażeniem temporalnym. W tym konkretnym przypadku można ustalić bezwzględną datę jego wystąpienia, ponieważ wyrażenie `zbombardowanie Hiroszimy` jest jednoznaczne. Nie wszystkie wyrażenia temporalne względne posiadają tę własność. Czasami jednak tym czego oczekujemy od systemu jest uszeregowanie wydarzeń w czasie, a wtedy wystarczające są informacje o względnych relacjach czasowych.

Wyrażenia określające *czas trwania* wskazują odcinek czasu, w którym określone wydarzenie miało miejsce, np.

> Ostatnia podróż pociągiem z Krakowa do Warszawy zajęła **ponad 4 godziny**.

Są one szczególnie istotne w kontekście zdarzeń długotrwałych, gdyż umożliwiają identyfikację początku oraz końca występowania określonego zdarzenia, co również jest istotne w kontekście szeregowania zdarzeń.

2.4.5. Ekstrakcja zdarzeń

Ekstrakcja zdarzeń z tekstu polega na rozpoznaniu opisywanych zdarzeń i określeniu najistotniejszych atrybutów tych zdarzeń. W poniższym przykładzie[11] :

> W nocy ze środy na czwartek **zmarł** po długiej i ciężkiej chorobie minister kultury i dziedzictwa narodowego Andrzej Zakrzewski. Miał 59 lat. Z wykształcenia prawnik, był historykiem, badaczem historii m.in. II Rzeczypospolitej. Przez wiele lat **pracował** w Instytucie Historii PAN.

możemy zidentyfikować następujące zdarzenia:

- *śmierć* Andrzeja Zakrzewskiego,
- *pracę* Andrzeja Zakrzewskiego w Instytucie Historii PAN.

W pierwszym zdarzeniu zidentyfikowany został podmiot zdarzenia, czyli **`Andrzej Zakrzewski`** natomiast w drugim również przedmiot zdarzenia, czyli **`Instytut Historii PAN`**, w którym pracował historyk.

W odniesieniu do pierwszego zdarzenia można również określić względny czas jego wystąpienia: `ze środy na czwartek`, ale bez dodatkowej informacji obejmującej datę powstania tej notatki nie mamy możliwości określenia kiedy dokładnie to nastąpiło. W odniesieniu do drugiego zdarzenia można również określić przybliżony czas jego trwania, tj. `wiele lat`. Odwołując się do ogólnej wiedzy o świecie można również określić, że drugie zdarzenie poprzedzało pierwsze, jednakże tego rodzaju inferencje raczej nie są przeprowadzane przez systemy ekstrakcji informacji, gdyż wymagają wykorzystania rozbudowanych ontologii zdarzeń, z którymi stowarzyszone są odpowiednie reguły wnioskowania.

W tekście występuje również opis stanu – **`choroba`** Andrzeja Zakrzewskiego – która pod wieloma względami przypomina zdarzenie. W szczególności stan ten posiada swój podmiot, początek oraz koniec, który w tym wypadku zbiega się z wystąpieniem zdarzenia **`śmierci`**. W zależności od przyjętych założeń ekstrakcja zdarzeń może również obejmować ekstrakcję informacji dotyczących zmiany stanu przedmiotów.

Efektem ekstrakcji zdarzeń powinno być przede wszystkim określenie typu zdarzenia, jego podmiotu oraz przedmiotów biorących w nim udział (o ile występują). Ważnym aspektem jest również określenie czasu oraz miejsca wystąpienia zdarzenia, a także chronologiczne uporządkowanie występujących zdarzeń. Nie zawsze jednak jest to możliwe, na co wskazuje analizowany przykład.

[11] Przykład pochodzi z korpusu notatek PAP opisanego w punkcie 6.1.2.

```
numer dokumentu: 1234
data dokumentu: 11/02/2010
źródło~dokumentu: PAP
innowacja:
  podmiot:
    nazwa: AMD
    rodzaj: przedsiębiorstwo
  przedmiot:
    typ: trawienie
    rodzaj podłoża: aluminium
    urządzenie:
      nazwa: SSZ 77
      producent: AMD
      typ: AB7
      stan: w~użyciu
      grubość podłoża: 100 nm
```

Rysunek 2.1: Przykładowy szablon ekstrakcyjny

2.4.6. Wypełnianie szablonów

Ostatnim zadaniem definiowanym w ramach ekstrakcji informacji jest wypełnianie szablonów. Pomimo tego, że zadanie to wydaje się najbardziej skomplikowanym, jest ono jednym z problemów, które najwcześniej były podejmowane w obrębie ekstrakcji informacji. W trakcie kolejnych edycji konferencji MUC [30] koncentrowano się m.in na uzupełnianiu szablonów dotyczących ataków terrorystycznych w Ameryce Południowej oraz innowacji w procesie wytwarzania urządzeń półprzewodnikowych. Podjęcie tak skomplikowanego problemu zaowocowało wypracowaniem metod opartych o szablony ekstrakcyjne oraz identyfikacją wymienionych wcześniej prostszych zadań, które muszą być zrealizowane w celu stworzenia uniwersalnego systemu ekstrakcji informacji.

Zadanie wypełniania szablonów podobne jest do zadania ekstrakcji zdarzeń – w istocie szablony definiowane w ramach MUC dotyczyły m.in. ogłoszeń o postępach w badaniach nad procesem produkcji układów scalonych. Zadanie to było jednak bardziej skomplikowane, gdyż elementami szablonu mogły być pod-szablony. Przykładowy szablon musiał obejmować nie tylko informacje na temat daty ogłoszenia innowacji, przedsiębiorstwa które jej dokonało, ale również jej szczegółów. Przykładowy szablon ekstrakcyjny mógł wyglądać jak na rysunku 2.1 (na podstawie [30]).

Jak widać na tym przykładzie zadanie to wymaga bardzo precyzyjnej identyfikacji obiektów, które mają zostać umieszczone w szablonie, rozpoznania relacji łączących te obiekty, a także dopasowania rozpoznanych relacji do odpowiednich elementów szablonu. Wymagało to ścisłego dopasowania systemów ekstrahujących informacje do dziedziny, dla której system ten był budowany. Trudność adaptowania systemów tego rodzaju do nowych dziedzin okazała się istotną wadą tego podejścia i zaowocowała uproszczeniem zadań definiowanych w ramach ekstrakcji informacji.

3. Reprezentacja wiedzy – relacje i sieci semantyczne

Ekstrakcja informacji jest zagadnieniem, które łączy dwie odrębne dziedziny wiedzy: językoznawstwo oraz informatykę. Podstawowym materiałem, na którym operują algorytmy ekstrakcji informacji jest tekst. W doskonałym systemie ekstrakcji informacji nie powinny występować ograniczenia co do charakteru danych językowych, dlatego należy przyjąć, że znaczna część zjawisk językowych opisanych w poprzednim rozdziale będzie miała wpływ na skuteczność algorytmu ekstrakcyjnego. Jednakże językoznawstwo i informatyka posługują się inną terminologią oraz sposobem reprezentacji wiedzy. Zjawiska są opisywane przez językoznawców językiem naukowym zbudowanym na bazie języka naturalnego. W informatyce natomiast konieczne jest definiowanie wykorzystywanych struktur danych oraz algorytmów na bazie języka matematyki. Nie zawsze jednak jest możliwe przełożenie, czasami nieostrych pojęć językoznawstwa, na precyzyjne pojęcia informatyczne.

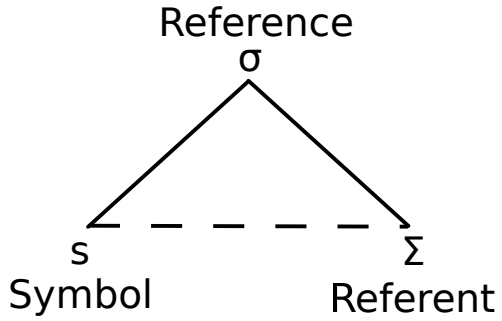
Z drugiej jednak strony warto zauważyć olbrzymi wpływ jaki wiedza językoznawcza wywarła na sposób reprezentacji wiedzy. Pierwsze filozoficzne rozważania na temat tak istotnych w logice (a w konsekwencji w informatyce) pojęć jak np. *alternatywa* czy *wartość logiczna* rozpoczęły się od analizy języków naturalnych. Pojęciami zaczerpniętymi z językoznawstwa szczególnie istotnymi z punktu wiedzenia ekstrakcji informacji w ogólności, a ekstrakcji relacji semantycznych w szczególności są *symbol językowy*, *relacja semantyczna* oraz *sieć semantyczna*.

Konieczność wprowadzenia technicznego pojęcia *symbolu językowego* wynika przede wszystkim z niejednoznaczności pojęć takich jak *słowo*, czy *wyraz*. Nie sposób jednak pisać o ekstrakcji informacji z tekstu nie odwołując się do jego podstawowych budulców. *Relacje semantyczne* są istotne z dwóch powodów: w pierwszym rzędzie tematem tej pracy jest ekstrakcja relacji semantycznych – bez ich definicji nie można opisać na czym miałaby polegać ich ekstrakcja. Natomiast *sieci semantyczne* stanowią niezwykle istotne narzędzie wykorzystywane w różnych algorytmach, w szczególności operujących na danych językowych. Składnikami sieci semantycznych mogą być różne elementy, ale w tych najczęściej stosowanych sieciach węzłami są symbole językowe, a relacjami – relacje semantyczne, bądź ontologiczne. W niniejszym rozdziale przedstawiamy zatem definicje pojęć takich jak: symbol językowy, relacja semantyczna, relacja ontologiczna, itd. Przedstawione definicje nie pretendują do statusu definicji uniwersalnych. Są one opracowane na potrzeby realizacji zadania jakim jest ekstrakcja relacji semantycznych.

3.1. Symbol językowy

3.1.1. Trójkąt semiotyczny

Jedną z najczęściej cytowanych prac analizujących znaczenie symboli językowych jest praca Ogdena i Richardsa *The Meaning of Meaning* [100]. W pracy tej przedstawiona jest koncepcja *trójkąta semiotycznego* – konstrukcji teoretycznej służącej do wyjaśnienia relacji pomiędzy zjawiskami językowymi takimi



Rysunek 3.1: Trójkąt semiotyczny według Ogdena i Richardsa [100, s. 11].

jak słowa, występujące w mowie oraz wyrazy, występujące w tekście (*symbol* – s – w terminologii Ogdena i Richardsa), a obiektami pozajęzykowymi (*referent* – Σ – w ich terminologii), do których zjawiska te się odnoszą. Ogden i Richards podkreślają[1], że związek pomiędzy wskazanym zjawiskiem językowym, a obiektem pozajęzykowym nie jest bezpośredni, lecz zapośredniczony w myśli (*thought* lub *reference* – σ – w ich terminologii). Zależności te są zilustrowane na rysunku 3.1. Linia ciągła określa relacje kauzalne, a linia przerywana zapośredniczenie relacji między zjawiskiem językowym, a obiektem rzeczywistości pozajęzykowej. Przykładowo, wyraz `pies` w zdaniu *Ten pies zagryzł dziś kurę*, nie odnosi się bezpośrednio do *zwierzęcia*, które zagryzło inne *zwierzę*, lecz zapośrednicza to odniesienie w pojęciu lub myśli **`pies`**, które dostarcza kognitywnych kryteriów identyfikacji danego obiektu.

Lyons [73, s. 9-28] wskazuje, że terminy takie jak *wyraz*, *pojęcie* i *znaczenie* są wieloznaczne. Tym niemniej przywołanie i szczegółowa analiza wszystkich znaczeń związanych z tymi terminami wykracza daleko poza ramy niniejszej pracy. Dlatego jako podstawowe, techniczne terminy przyjmiemy *napis* oraz *symbol językowy* – Lubaszewski wskazuje bowiem[71, s. 15], że

> [...] z komputerowego punktu widzenia, wystarczy tylko rozróżnić *napis* (ciąg liter), reprezentujący wyrazy w tekście, oraz wyraz (*symbol*) reprezentowany przez przechowywany w słowniku *opis*.

Odnosząc się do koncepcji oraz terminologii Ogdena i Richardsa przyjmujemy zatem, że nasz ***napis*** odpowiada ich pojęciu *symbol*, a nasz ***symbol językowy*** ich pojęciu *reference*. *Referent* jako twór pozajęzykowy nie pojawia się bezpośrednio w systemie komputerowym – może być co najwyżej reprezentowany, np. jako wiersz w bazie danych. Taki wiersz jest jednak składnikiem kolejnego trójkąta semiotycznego.

To rozróżnienie zgodne jest z definicją ekstrakcji informacji przedstawioną w punkcie 2.1: *napis* stanowi składnik meta-języka, zaś *symbol językowy* jest składnikiem języka przedmiotowego. Istotnym zadaniem algorytmu jest odtworzenie relacji łączącej napisy z symbolami, co umożliwia przejście od opisu w terminach meta-języka, do terminów języka przedmiotowego i w konsekwencji przetwarzanie informacji na temat obiektów pozajęzykowych.

W treści pracy często termin *napis* będzie używany zamiennie z terminami *słowo*, *wyraz* oraz *wyrażenie*. Podobnie termin *symbol językowy* może być używany zamiennie z terminami *pojęcie* oraz *znaczenie*. Taka swoboda używania terminów nie powinna jednak prowadzić do niejednoznaczności, gdyż odpowiednie znaczenie zwykle wynika z kontekstu. Za każdym razem należy mieć również na uwadze, że rozważania te prowadzone są w kontekście ekstrakcji informacji, czyli zadania praktycznego, dla którego punktem odniesienia są systemy informatyczne, za pomocą których implementowane są odpowiednie algorytmy

[1] Choć nie są oni pierwszymi naukowcami dostrzegającymi tę zależność, koncepcja ta była już bowiem znana w średniowieczu – porównaj [71, s. 26].

ekstrakcji informacji. Toteż rozróżnienie na *napis* oraz *symbol językowy* powinno być wystarczające. Nie zmienia to faktu, że natura danych językowych jest bardziej złożona i językoznawcy oraz filozofowie zwykle posługują się bardziej szczegółowymi rozróżnieniami, nadając wymienionym terminom nierzadko odmienne znaczenie.

3.1.2. Definicja symbolu językowego

Bardzo ważnym elementem w ekstrakcji relacji staje się zatem *symbol językowy* reprezentowany za pomocą *opisu*. Nie zakładamy bowiem innych kognitywnych metod konstrukcji symboli językowych niż poprzez opis w wybranym formalizmie. Opis ten znajduje się w odpowiednim *słowniku* (dalej nazywanym *słownikiem semantycznym*), w którym zdefiniowane są wszystkie symbole językowe, istotne z punktu widzenia ekstrakcji informacji. Taki sposób określenia znaczeń symboli nie jest jedyny i w praktyce wykorzystywane są również inne (porównaj [10, 12, 21]). Pozwala on jednak na ekstrahowanie relatywnie spójnych informacji z tekstu, dlatego posługujemy się rozróżnieniem Lubaszewskiego przedstawionym w punkcie 3.1.1. Ma ono tę zaletę, że założenie istnienia odpowiedniego słownika uwalnia nas od problemu odróżniania symboli – przyjmujemy istnienie tylko tych symboli, których opisy dostępne są w słowniku oraz przyjmujemy, że dwa odrębne symbole reprezentują dwa odrębne znaczenia. Oczywiście nie zwalnia nas to z konieczności ujednoznaczniania *napisów*, gdyż jeden napis może odnosić się do wielu symboli. Natomiast przyjmujemy, że liczba znaczeń napisów jest dokładnie określona.

Rozstrzygnięcia te możemy sformalizować w następującej definicji symbolu językowego σ_a

$$\sigma_a \ \in \boldsymbol{Q} \stackrel{def}{\Leftrightarrow} (desc(\sigma_a) \in \boldsymbol{D} \wedge (desc(\sigma_a) = desc(\sigma_b) \Leftrightarrow \sigma_a \ = \sigma_b)) \ , \tag{3.1}$$

gdzie:

- $\boldsymbol{Q}$ – zbiór wszystkich symboli językowych,
- $\boldsymbol{D}$ – zbiór wszystkich opisów symboli językowych – *słownik semantyczny*,
- $desc(\sigma)$ – funkcja zwracająca opis symbolu σ.

3.1.3. Zbiory form fleksyjnych

Powiązanie pomiędzy *symbolem językowym* a *napisem* nie jest jednak tak proste – zwykle w słowniku nie umieszcza się wszystkich napisów, które mogą reprezentować dany symbol. W języku polskim, który jest językiem fleksyjnym, pojedynczy symbol może mieć bardzo wiele odpowiadających mu napisów. Chcąc uprościć opis symbolu, często zakłada się, że zawiera on odniesienie wyłącznie do *formy podstawowej* (inaczej *hasłowej*), czyli tradycyjnie wyróżnionej formy symbolu. Np. symbol **`Pies`**, posiada między innymi następujące formy tekstowe: `psa`, `psu`, `psie`, `psy`, `psom`, ale formą podstawową jest forma `pies`, czyli forma będąca mianownikiem liczby pojedynczej. Umieszczając w opisie symbolu jedynie formę podstawową, zakłada się istnienie odpowiedniego *słownika fleksyjnego*, w którym opisane są związki pomiędzy formami fleksyjnymi.

Wprowadzenie pojęcia zbioru form fleksyjnych pozwala nam na powiązanie elementów tekstu (napisów) z abstrakcyjnymi jednostkami jakimi są zbiory form fleksyjnych $\boldsymbol{F}$

$$\begin{aligned} \boldsymbol{F} \in \boldsymbol{FS} \ \stackrel{def}{\Leftrightarrow} \ & (\forall s \ \forall p \ : (s,p) \in \boldsymbol{F} \Rightarrow s \ \in \boldsymbol{N} \wedge p \ \in \boldsymbol{P}) \wedge \\ & (\forall p : p \ \in positions(\boldsymbol{F}) \Rightarrow \exists! s \ : (s,p) \in \boldsymbol{F}) \wedge \\ & (\exists s \ : s \ \in \boldsymbol{N} \wedge base(\boldsymbol{F}) = s) \end{aligned} \tag{3.2}$$

gdzie:

- $\boldsymbol{FS}$ – zbiór wszystkich zbiorów form fleksyjnych – *słownik fleksyjny*,
- $\boldsymbol{N}$ – zbiór wszystkich napisów niezawierających spacji, występujących w polskich tekstach – *słownik form tekstowych*,
- $\boldsymbol{P}$ – zbiór wszystkich *pozycji fleksyjnych*, tzn. poprawnych kombinacji wartości kategorii gramatycznych,
- $positions(\boldsymbol{F})$ – funkcja zwracającą zbiór pozycji fleksyjnych wymaganych dla zbioru form fleksyjnych $\boldsymbol{F}$; wartość tej funkcji zależy od klasy gramatycznej (części mowy) zbioru form fleksyjnych,
- $base(\boldsymbol{F})$ – funkcja zwracająca *formę podstawową (hasłową)* zbioru form fleksyjnych.

Powyższa formalizacja wprowadza pojęcie *pozycji fleksyjnych* – są to dopuszczalne dla danej klasy gramatycznej kombinacje wartości kategorii gramatycznych, które mogą (ale nie muszą) wymagać odmiennych form fleksyjnych (porównaj [71, s. 17-19]). Przykładowo dla zwykłego rzeczownika[2] występuje 14 pozycji fleksyjnych, będących kombinacjami dopuszczalnych wartości *przypadka* (7 wartości) oraz *liczby* (2 wartości). Wymagamy zatem aby w zbiorze form fleksyjnych wszystkie pozycje fleksyjne posiadały dokładnie jeden napis odpowiadający jednej pozycji fleksyjnej[3]. Oczywiście nie jest wymagane aby jeden napis mógł występować tylko na jednej pozycji fleksyjnej (porównaj forma `psa`, która odpowiada *dopełniaczowi* oraz *biernikowi liczby pojedynczej* zbioru form odpowiadających symbolowi **`pies`**). Co więcej nie wymagamy również aby jeden napis przynależał wyłącznie do jednego zbioru form fleksyjnych – w ten sposób odzwierciedlone jest zjawisko wieloznaczności napisów. Dodajmy również, że forma podstawowa jest wyróżniona w zbiorze form fleksyjnych poprzez wyróżnienie odpowiedniej kombinacji kategorii gramatycznych, np. dla zwykłych rzeczowników, jest to forma odpowiadająca pozycji: (*mianownik*, *liczba pojedyncza*).

Zdefiniowawszy pojęcie *symbolu językowego* oraz *zbioru form fleksynych* należy zwrócić uwagę, że sama forma podstawowa, ani też forma podstawowa uzupełniona o klasę gramatyczną nie wystarczają do jednoznacznego zidentyfikowania odpowiedniego zbioru form fleksyjnych (porównaj [71, s. 18], [162, s. 43-44]). Dlatego chcąc jednoznacznie zidentyfikować odpowiedni zbiór form, należy każdy zbiór opatrzyć etykietą – np. numeryczną, bądź literową, jak to ma miejsce w przypadku Słownika Fleksyjnego Języka Polskiego [44]. Wtedy w słowniku semantycznym można zarejestrować wyłącznie formę podstawową oraz etykietę pozwalającą w sposób jednoznaczny zidentyfikować odpowiedni zbiór form. Para (*forma podstawowa*, *etykieta fleksyjna*) jest zwracana przez funkcję identyfikującą *id* określoną na zbiorze zbiorów form fleksyjnych

$$\begin{aligned} &id\colon \boldsymbol{FS} \to \boldsymbol{N} \times \boldsymbol{L}\,: \\ &id(\boldsymbol{F}) = (base(\boldsymbol{F}), itag(\boldsymbol{F})) \wedge \\ &\forall \boldsymbol{F} : id^{-1}(base(\boldsymbol{F}), itag(\boldsymbol{F})) = \boldsymbol{F}\ , \end{aligned} \tag{3.3}$$

[2] To znaczy np. nie rzeczownika *plurale tantum*.

[3] Czasami zbiory te określa się mianem *leksemów* bądź *fleksemów* (porównaj [162, s. 44-45]). W tym miejscu nie będziemy jednak wprowadzać tych pojęć, zwracając jedynie uwagę, że podział symboli językowych na klasy gramatyczne może odbywać się na kilka różnych sposobów – dla nas istotne jest jedynie to, aby w obrębie określonej klasy gramatycznej obsadzona była każda pozycja fleksyjna. Nie wymagamy zatem aby pozycje fleksyjne różniły się jedynie wartościami ściśle określonych kategorii gramatycznych. Dopuszczamy możliwość, że np. w obrębie jednego zbioru form występowała forma odpowiadająca (nieodmiennemu) bezokolicznikowi oraz form 1 osoby, liczby pojedynczej, czasu teraźniejszego, czyli formy których opis morfologiczny odwołuje się do różnych zbiorów kategorii gramatycznych. Ponadto, występowanie form *wariantywnych* [71, s. 20-21], np. `kawiarń`, `kawiarni`, rozwiązane jest w ten sposób, że każdy z wariantów przynależy do osobnego zbioru form fleksyjnych, które powiązane są z tym samym symbolem językowym.

gdzie $itag(\boldsymbol{F})$ jest funkcją zwracającą etykietę fleksyjną (ang. *inflection tag*) dla zbioru form $\boldsymbol{F}$. Powyższa definicja wprowadza również funkcję odwrotną id^{-1}, która na podstawie odpowiedniej pary (*base*,*itag*) pozwala jednoznacznie zidentyfikować odpowiadający jej zbiór form fleksyjnych.

3.1.4. Powiązanie napisów z symbolami językowymi

Powiązanie symbolu z formą podstawową nie rozwiązuje jednak wszystkich problemów związanych z przyporządkowaniem symboli do odpowiadających im napisów. Należy bowiem zwrócić uwagę, że znaczna część symboli, w szczególności nazwy własne, mogą być reprezentowane za pomocą wielu zbiorów form tekstowych, ponadto mogą być one reprezentowane przez napisy składające się ze spacji oraz napisy, które w tekście mogą zwierać nieciągłości. W niniejszej pracy przywiązujemy dużą wagę do spójność ekstrahowanych informacji. Dlatego też różne napisy reprezentujące ten sam symbol, np. `NIP`, `Numer Identyfikacji Podatkowej`, `Numerem identyfikacji podatkowej` powinny być powiązane z tym samym symbolem językowym: **`Numer Identyfikacji Podatkowej`**. Zagadnienie to jest rozwiązywane poprzez powiązanie jednego symbolu z wieloma zbiorami form fleksyjnych, a także wieloma ciągami napisów. Nie rozwiązuje to wprawdzie wszystkich problemów (w szczególności możliwości zmiany pozycji poszczególnych napisów w obrębie ciągu napisów), ale określa podstawowe ramy konstrukcji słownika semantycznego, przydatnego w ekstrakcji relacji semantycznych.

Do powiązania symboli językowych z napisami służy funkcja mapująca *strings*

$$\begin{aligned} strings : \boldsymbol{Q} &\rightarrow 2^{\boldsymbol{N'}} : \\ strings(\sigma) &= \{s \ : (\exists p \ : (s,p) \in \boldsymbol{F} \wedge \boldsymbol{F} \in \boldsymbol{FS}) \vee \\ &\quad (s \ = s_1 \oplus s_2 \oplus s_3 \oplus \ldots \wedge s_1 \in \boldsymbol{N} \wedge s_2 \in \boldsymbol{N} \wedge s_3 \in \boldsymbol{N} \ldots)\} \ . \end{aligned} \tag{3.4}$$

gdzie:

- $\boldsymbol{N'}$ – zbiór wszystkich napisów zawierających spacje,
- $\oplus$ – operator konkatenacji napisów wstawiający spację, pomiędzy łączone napisy.

Funkcja *strings* dopuszcza powiązania symbolu z określonym zbiorem napisów za pośrednictwem zbioru form fleksyjnych, co upraszcza przechowywany opis. Ponadto funkcja ta dopuszcza powiązanie z napisem powstałym przez połączenie, za pomocą spacji, dowolnego ciągu napisów. To rozwiązanie nie jest najbardziej ekonomiczne, ale pozwala zachować względną prostotę reprezentacji. Warto też zwrócić uwagę, że implementacja tej funkcji nie musi zwracać pełnego zbioru wyrazów, lecz jedynie parę $(base, itag)$, która dzięki istnieniu funkcji odwrotnej id^{-1} pozwala jednoznacznie odnaleźć odpowiedni zbiór form fleksyjnych i odpowiadający mu zbiór napisów.

Odnosząc funkcję *strings* do koncepcji trójkąta semiotycznego (porównaj rys. 3.1) można umiejscowić ją pomiędzy elementami oznaczonymi symbolami s oraz σ – tzn. reprezentuje ona relację jaka w terminologii Ogdena i Richardsa występuje pomiędzy *symbol* a *reference*. Należy jednak mieć na uwadze, że ich *symbol* (czyli w naszej terminologii *napis*) jest tylko jednym z wielu napisów, który może reprezentować dany symbol. Dzieje się tak, ponieważ napis może być jedną z wielu form fleksyjnych stowarzyszonych z danym symbolem językowym, bądź jednym z wielu wariantów nazwy własnej, bądź wyrażenia wielosegmentowego.

Ponieważ napisy są wieloznaczne, nie istnieje funkcja odwrotna dla funkcji *strings*. Przy ekstrakcji informacji konieczne jest ujednoznacznienie napisów względem słownika semantycznego. W uproszczeniu ujednoznacznienie to polega na wyborze dla napisu występującego w tekście, jednego spośród (najczęściej) wielu symboli językowych, dla których napis ten należy do zbioru napisów zwracanych przez

funkcję *strings*. W praktyce problem ten jest jeszcze bardziej skomplikowany, gdyż ze względu na występowanie symboli, którym odpowiadają wyrazy wielosegmentowe, algorytm ujednoznaczniania musi uwzględniać ich wykrycie i dokonać wyboru właściwych mapowań w obrębie napisów, które mogą na siebie zachodzić.

3.1.5. Powiązanie symboli językowych z obiektami rzeczywistymi

Przyjęcie, że wszystkie symbole językowe muszą być zdefiniowane w słowniku semantycznym stanowi uproszczenie. Niemniej jednak, biorąc pod uwagę konkretną dziedzinę zastosowań, to założenie nie musi być dalekie od rzeczywistości. Zupełnie inaczej wygląda sprawa po stronie obiektów rzeczywistych (w terminologii Ogdena i Richardsa *referent*). Skonstruowanie bazy danych, czy słownika, który zawierałby identyfikatory wszystkich obiektów rzeczywistych jest z praktycznych powodów niemożliwe, dlatego w ogólnym przypadku ekstrakcji informacji nie zakłada się istnienia odpowiedniej bazy danych, w której skatalogowane byłyby obiekty rzeczywiste.

Nie oznacza to jednak, że tożsamość obiektów rzeczywistych jest zawsze nieustalona. W pierwszym rzędzie dzięki technikom rozpoznawania koreferencji (porównaj 2.4.2) możliwe jest ustalenie, że dwa napisy odnoszą się do *tego samego* obiektu rzeczywistego, co pozwala na integrowanie różnych informacji wyekstrahowanych na temat tego obiektu. W niniejszej pracy jednak techniki te są stosowane w ograniczonym zakresie. Drugim, ważniejszym z naszego punktu widzenia przypadkiem są sytuacje, w których w tekście występują *nazwy własne*. Dzięki istnieniu encyklopedii elektronicznych, w szczególności Wikipedii[4], możliwe jest wykorzystanie zawartych w nich informacji w celu stworzenia *częściowego rejestru* obiektów rzeczywistych. Jeśli wiemy, że określony artykuł w encyklopedii opisuje indywiduum, np. Billa Clintona, możemy oczekiwać, że algorytm ekstrakcji informacji będzie w stanie rozpoznać odniesienia do tej osoby, przynajmniej w sytuacji, w której w tekście występuje odpowiednia nazwa własna (w przeciwieństwie do wyrażenia ogólnego, np. *bohater afery rozporkowej*, albo *ostatni prezydent USA poddany procedurze impeachmentu*). Dzięki temu możliwe jest zdefiniowanie dwóch pokrewnych, lecz odmiennych zadań w ramach ekstrakcji informacji: ujednoznaczniania (ang. *word sense disambiguation*) oraz linkowania (ang. *entity linking*). Tylko w drugim przypadku oczekujemy, że określone napisy reprezentujące nazwy własne powiązane zostaną z odpowiednimi pozycjami w bazie danych, bądź bazie wiedzy.

Chcąc reprezentować związek jaki występuje pomiędzy symbolami językowymi, a obiektami rzeczywistymi możemy zdefiniować funkcję interpretacji *int*, która zwraca obiekt rzeczywisty Σ (*referent*) odpowiadający symbolowi językowemu

$$\begin{aligned} &int : \boldsymbol{D} \times \boldsymbol{C} \times \mathbb{W} \rightarrow \boldsymbol{\Sigma} : \\ &int(\sigma, c_\sigma, \boldsymbol{W}) = \Sigma \Rightarrow \Sigma \in \boldsymbol{W} \end{aligned} \tag{3.5}$$

gdzie:

- $\boldsymbol{C}$ – zbiór wszystkich kontekstów użycia słów,
- c_σ – kontekst użycia symbolu σ; może obejmować miejsce, czas oraz użytkownika tego symbolu,
- $\mathbb{W}$ – zbiór wszystkich światów możliwych,
- $\boldsymbol{\Sigma}$ – zbiór wszystkich obiektów,
- $\boldsymbol{W}$ – świat możliwy.

[4]Szczegółowe omówienie własności tej encyklopedii elektronicznej, w kontekście przetwarzania języka naturalnego przedstawione jest w punkcie 6.3.

Funkcja *int* określa warunek konieczny odnoszenia się symbolu σ do obiektu Σ: istnienie obiektu Σ w *świecie możliwym* $\boldsymbol{W}$ (porównaj p. 3.3.2). Funkcja *int* w tej ogólnej postaci wymaga doprecyzowania np. tego czym jest kontekst użycia symbol c_σ oraz jak jest rozumiemy świat możliwy.

Z naszego punktu widzenia bardziej użyteczna jest jednak funkcja int', która zdefiniowana jest wyłącznie dla nazw własnych – przyjmiemy bowiem, że interpretacja symbolu językowego będącego nazwą własną nie wymaga uwzględnienia ani kontekstu użycia, ani świata możliwego, w którym jest ona interpretowana. Należy bowiem zwrócić uwagę, że choć nazwy własne mogą być wieloznaczne, ponieważ jeden napis może odnosić się do dwóch lub większej liczby symboli językowych (np. napis `Kraków` może odnosić się do *Krakowa* w Polsce oraz w USA), to po rozstrzygnięciu wieloznaczności *napis – symbol językowy*, związek pomiędzy symbolem językowym będącym nazwą własną, a obiektem rzeczywistym jest jednoznaczny. Ponadto przyjmujemy za Kripkem [59], że nazwy własne są tzw. *sztywnymi desygnatorami*, tzn. wskazują obiekty jednoznacznie, niezależnie od możliwego świata, w którym dokonuje się ich interpretacji. Funkcja int' określona jest następująco

$$\begin{aligned} &int' : \boldsymbol{D} \to \boldsymbol{\Sigma} : \\ &int'(\sigma) = \Sigma \Rightarrow \sigma \in \boldsymbol{PN} \end{aligned} \tag{3.6}$$

gdzie $\boldsymbol{PN}$ jest zbiorem wszystkich nazw własnych. W tym miejscu pomijamy również szereg innych problemów związanych z tożsamością obiektów (np. rozstrzygnięcie takich problemów jak tożsamość autora *Iliady* i *Odysei*) oraz ich istnieniem (np. status ontologiczny obiektów sprzecznych takich jak np. *kwadratowe koło*), przyjmując, że punktem odniesienia jest dla nas odpowiedni słownik semantyczny, w którym nazwy własne są odpowiednio oznaczone, pozostawiając rozstrzygnięcie problemów tożsamości oraz różnych sposobów istnienia filozofom.

3.2. Relacje semantyczne

3.2.1. Relacje semantyczne a relacje morfologiczne i ontologiczne

Zanim przejdziemy do szczegółowego omówienia różnych typów relacji semantycznych musimy wskazać na kilka istotnych rozróżnień terminologicznych. Symbol językowy może wchodzić w rozmaite relacje – nie wszystkie jednak są relacjami semantycznymi.

W pierwszej kolejności należy zwrócić uwagę na fakt, szczególnie jaskrawy w językach fleksyjnych, dotyczący związku pomiędzy różnymi formami fleksyjnymi jednego symbolu językowego. Związki, które zachodzą pomiędzy formami należącymi do jednego zbioru form fleksyjnych, nie są relacjami semantycznymi, gdyż mają ona charakter wyłącznie formalny (dotyczą ich kształtu).

Również na poziomie samych symboli językowych występują relacje morfologiczne, które pozwalają nam rozpoznać podobieństwo wyrazów takich jak *pisać*, *piszący*, *pisany* oraz *pisanie*, w których pierwszy należy do kategorii czasownika, drugi – rzeczownika, trzeci to imiesłów, a czwarty – rzeczownik odczasownikowy. W języku występują całe zespoły relacji tego rodzaju (porównaj [71, s. 22]). Zależności tych nie nazywamy jednak relacjami semantycznymi, lecz relacjami morfologicznymi, gdyż w pierwszym rzędzie opierają się nie na pokrewieństwie znaczeń (choć w istocie ono tutaj występuje), lecz na podobieństwie formy. Pokrewieństwo znaczenia jest tylko efektem ubocznym podobieństwa morfologicznego. Dlatego nie wszystkie relacje występujące pomiędzy symbolami językowymi nazywać będziemy relacjami semantycznymi.

Z drugiej strony, relacje semantyczne warto skontrastować z relacjami ontologicznymi. Relacjami ontologicznymi nazwiemy te relacje, które występują pomiędzy obiektami rzeczywistymi, natomiast relacjami

semantycznymi nazwiemy relacje występujące pomiędzy symbolami językowymi (warunek konieczny relacji semantycznych). Charakterystyczną cechą relacji ontologicznych jest to, że są one niezależne od języka, natomiast relacje semantyczne zawsze są zrelatywizowane do konkretnego języka. Przykład relacji ontologicznej można podać *lokalizację* – relacja ta odnosi przedmiot fizyczny do miejsca, w którym przedmiot ten się znajduje. Odnosząc relacje ontologiczne do trójkąta semiotycznego (porównaj rys. 3.1), powiemy, że występują one pomiędzy elementami oznaczonymi w trójkącie symbolem Σ, czyli ogdenowskim *referent*.

Natomiast jako przykład typowej relacji semantycznej można podać *synonimię* – terminy takie jak *buty* i *obuwie* są synonimiczne, nie można jednak sensownie mówić o tej relacji, w oderwaniu od faktu, że oba te terminy należą do języka polskiego. W innych językach mogą występować analogiczne pojęcia, ale nic nie przesądza o tym, że np. w języku angielskim w ogóle będą występowały dwa terminy na określenie zbioru obiektów, do którego odnosimy się za pomocą wyżej wymienionych słów. Interpretując relacje semantyczne w terminach ogdenowskich powiemy, że relacje te występują pomiędzy *reference*, czyli elementami oznaczonymi za pomocą symbolu σ. Ponieważ na rysunku 3.1 występuje schemat odpowiadający tylko jednemu symbolowi, nie zostały na nim uwzględnione ani relacje semantyczne, ani relacje ontologiczne.

Rozstrzygnięcia te możemy sformalizować następująco:

$$\begin{array}{lll} \boldsymbol{R_s} \in \boldsymbol{RS} & \Rightarrow & (\forall \sigma_a \ \forall \sigma_b \ : (\sigma_a, \sigma_b) \in \boldsymbol{R_s} \Rightarrow \sigma_a \ \in \boldsymbol{Q} \wedge \sigma_b \ \in \boldsymbol{Q}) \\ \boldsymbol{R_o} \in \boldsymbol{RO} & \Rightarrow & (\forall \Sigma_a \ \forall \Sigma_b \ : (\Sigma_a, \Sigma_b) \in \boldsymbol{R_o} \Rightarrow (\exists \boldsymbol{W} : \Sigma_a \ \in \boldsymbol{W} \wedge \Sigma_b \ \in \boldsymbol{W})) \end{array} \tag{3.7}$$

gdzie:

- $\boldsymbol{R_s}$ – relacja semantyczna,
- $\boldsymbol{R_o}$ – relacja ontologiczna,
- $\boldsymbol{RS}$ – zbiór relacji semantycznych,
- $\boldsymbol{RO}$ – zbiór relacji ontologicznych.

Formalizacja ta obejmuje wyłącznie warunki konieczne istnienia odpowiednich relacji. Wymagamy zatem aby dla relacji semantycznych oba człony relacji były symbolami językowymi, a dla relacji ontologicznych, wymagamy aby człony relacji były obiektami występującymi w tym samym świecie możliwym $\boldsymbol{W}$.

Warto jednak podkreślić, że odróżnienie relacji semantycznych od relacji ontologicznych opiera się na wyodrębnieniu spośród ogółu ludzkiej wiedzy specjalnego podtypu – wiedzy językowej. Oczywiście takie spojrzenie na wiedzę jest nieco naiwne, bowiem zakłada, że istnieje jakiś niezależny od języków naturalnych sposób opisywania wiedzy. W istocie, wiele dziedzin nauki (w szczególności nauki ścisłe) wypracowało swoje specyficzne języki, którymi można bardziej precyzyjnie ujmować relacje ontologiczne. Zwykle oczekujemy również, że relacje pomiędzy informacjami, które przechowywane są w informatycznych bazach danych są ściśle zdefiniowane. Dlatego jako ich podstawę często wykorzystuję się relacje ontologiczne. Nie zmienia to jednak następującego faktu: wszelkie teorie naukowe ostatecznie opisywane i interpretowane są w językach naturalnych. Trudno nie zgodzić się z tezą, że ich rozumienie możliwe jest dzięki rozumieniu odpowiednich relacji semantycznych. Ponadto, wiele z relacji oraz atrybutów szczególnie istotnych dla człowieka np. imię i nazwisko, ma charakter konwencjonalny.

Dlatego w dalszych rozważaniach, choć będziemy posługiwać się tym rozróżnieniem – mając na myśli jako pierwotne kryterium przynależność, bądź brak przynależności odpowiednich elementów (symboli językowych) do języka – to podział ów będzie traktowany pragmatycznie. O relacjach semantycznych mowa będzie na etapie przetwarzania tekstu – o relacjach ontologicznych zaczniemy mówić w momencie, w którym wyekstrahowane dane trafiać będą do ustrukturyzowanego repozytorium informacji albo gdy na

podstawie tych relacji prowadzone będzie wnioskowanie. Pomimo to, podział ów jest umowny i nie daje się ściśle zdefiniować. Przyjmując słownik semantyczny jako punkt odniesienia, nie można zapominać, że dopuszczając występowanie w nim nazw własnych, musimy również dopuścić występowanie w nim relacji ontologicznych.

Trudność odróżnienia relacji semantycznych od relacji ontologicznych ma szczególne znaczenia dla dwóch typów relacji: *generalizacji/specjalizacji* oraz *całość-część* – oba typy relacji możemy bowiem zaliczyć do relacji semantycznych i ontologicznych. W odniesieniu do pierwszej relacji dzieje się tak dlatego, że jest ona podstawową relacją porządkującą terminologię w danej dziedzinie wiedzy. Nie sposób zdefiniować w szczególnej ontologii np. terminu *człowiek*, nie odwołując się do pojęć takich jak *ssak* czy *zwierzę*. W systemach ontologicznych relacja ta zaś jest wprowadzana przede wszystkim w celu zwiększenia ekonomii reprezentacji informacji – pozwala bowiem wyrażać pewne fakty w sposób bardziej zwięzły – na wyższym poziomie abstrakcji. W istocie jednak jest to relacja semantyczna. Podobnie relacja *całość-część* konieczna jest do zdefiniowania pojęć takich jak *rękaw*, czy *klapa* (porównaj [73, s. 300]), ale trudno byłoby jednocześnie uznać, że relacja ta występuje wyłącznie pomiędzy symbolami językowymi.

3.2.2. Relacje paradygmatyczne i syntagmatyczne

Relacje semantyczne były jednym z najważniejszych pojęć eksploatowanych przez strukturalistyczny nurt językoznawstwa. De Saussure [139] jako podstawową relację służącą wyjaśnieniu zjawisk językowych uznawał *przeciwstawienie*. Relacja ta w ujęciu de Saussure'a jest dwojakiego rodzaju [139]: z jednej strony symbole językowe mogą wchodzić w *relacje paradygmatyczne*, ze względu na to, że mogą występować w tym samym miejscu określonej struktury językowej; z drugiej zaś strony mogą wchodzić w *relacje syntagmatyczne*, ze względu na to, że występują w tej samej strukturze językowej.

W obu przypadkach mamy do czynienia z relacjami kontrastywnymi, lecz o innym charakterze. W pierwszym można mówić o relacjach pionowych: tak np. pomiędzy głoskami *a*:*o* występuje relacja paradygmatyczna widoczna w słowach takich jak *bak* i *bok*. Różnica pomiędzy tymi słowami wynika według de Saussure'a wyłącznie z faktu istnienia kontrastywnej relacji paradygmatycznej, gdyż pozostałe elementy tej struktury są identyczne. W drugim przypadku mamy do czynienia z kierunkową relacją poziomą, którą można zilustrować opozycją słów *od* i *do* – w obu przypadkach występują identyczne symbole, ale na innych pozycjach. Różnica w znaczeniu tych słów może być wyjaśniona poprzez odwołanie się do istnienia relacji syntagmatycznej.

Atrakcyjność podejścia de Saussure polega na tym, że za pomocą tych samych typów relacji można również tłumaczyć zjawiska językowe występujące na innych poziomach opisu oraz w innych dziedzinach niż fonologia. W szczególności wiele współczesnych teorii opisu znaczenia opiera się na rozróżnieniu relacji syntagmatycznych i paradygmatycznych. Jak zauważa jednak Lyons [73, s. 262-263], trudno jest opisywać pełnię zjawisk semantyki leksykalnej odwołując się wyłącznie do pojęcia przeciwstawności znaczeń. Warto również zwrócić uwagę, że podstawową relacją wykorzystywaną obecnie do opisu znaczenia jest synonimia (porównaj WordNet [41, s. 23-24]). Od niej też rozpoczniemy bardziej szczegółowe omówienie relacji semantycznych.

3.2.3. Synonimia, podobieństwo i pokrewieństwo znaczeń

Synonimia jest relacją semantyczną znaną już w starożytności [71, s. 27] i zachodzi pomiędzy symbolami, które posiadają to samo znaczenie. Najczęściej zjawisko synonimii eksplikuje się jako możliwość zastępowania jednego symbolu innym symbolem w dowolnym kontekście językowym, bez zmiany znacze-

nia wyrażenia, w którym zamiana ta występuje. Synonimia jest zatem szczególną relacją paradygmatyczną – różnica pomiędzy symbolami dotyczy wyłącznie ich formy.

Tak ściśle zdefiniowana synonimia występuje jednak dość rzadko w języku, dlatego też jako podstawę konstrukcji WordNetu – semantycznego słownika języka angielskiego – jego twórcy przyjęli nieco szerszą definicję: dwa symbole połączone są relacją synonimii (czyli w terminologii WordNetu należą do jednego *synsetu*), jeśli mogą być zastępowane *w pewnym kontekście* [41, s. 23-24].

Taka definicja nie jest jednak zbyt precyzyjna, bowiem na podstawie przykładu ***Chłopiec*** *zachowywał się niegrzecznie/****On*** *zachowywał się niegrzecznie* należałoby przyjąć, że *chłopiec* i *on* są synonimami, wszak istnieje pewien kontekst, w którym mogą być zastąpione bez zmiany znaczenia. Dlatego też w trakcie konstrukcji polskiego WordNetu, jego twórcy przyjęli inną definicję synonimii [108, s. 24]: A jest synonimem B, jeśli można powiedzieć, że „A jest rodzajem B" oraz „B jest rodzajem A". Przykładowo oba zdania *Buty są rodzajem obuwia* oraz *Obuwie jest rodzajem butów* są prawdziwe, dlatego *buty* oraz *obuwie* są synonimami. Co więcej twórcy polskiego WordNetu przyjmują, że przynależność określonego symbolu do synsetu uzależniona jest od posiadania identycznych relacji semantycznych[5], jak pozostałe elementy danego synsetu. Jest to dość oczywista konsekwencja tego jak skonstruowany jest WordNet – większość relacji semantycznych określanych jest na poziomie synsetu, w związku z czym należy zakładać, że obowiązują one względem wszystkich jego elementów. Ta koncepcja jest również zgodna z definicją symboli językowych przedstawioną we wzorze 3.1, gdzie symbole posiadające identyczny opis są utożsamiane. Tym niemniej twórcy angielskiego WordNetu nie przyjęli tej własności jako *warunku* przynależności do synsetu.

Obok synonimii, która wymaga pełnej, bądź wysokiej zgodności znaczenia symboli, często definiuje się relację *podobieństwa semantycznego* (ang. *semantic similarity*). Jest to relacja, która może być określona dla pary symboli językowych należących do tej samej klasy gramatycznej. Stosując odpowiednie algorytmy (porównaj [58, s. 652-667]) możliwe jest określenie miary tej relacji, a dzięki temu wyróżnienie symboli, które choć nie są synonimami, są najbardziej podobne do określonego symbolu językowego. Przykładami wyrazów podobnych są np. *statek* i *okręt*, które w potocznym języku są często stosowane zamiennie[6].

Relację, która pozwala określić znaczeniową bliskość dla słów należących do różnych kategorii gramatycznych jest zaś *pokrewieństwo znaczenia* [58, s. 652-653]. Tak np. *benzyna* i *samochód* oraz *jechać* i *samochód* są semantycznie spokrewnione, choć nie są semantycznie podobne, w przeciwieństwie do *samochodu* i *pojazdu*, pomiędzy którymi występuje wysokie podobieństwo znaczenia.

Wszystkie te relacje mają szczególne znaczenie w algorytmach ekstrakcji informacji. Synonimia pozwala rozpoznać w tekście różne wyrażenia, które reprezentują ten sam symbol, a co za tym idzie ekstrahować spójną wiedzę. Miara podobieństwa semantycznego znajduje zastosowanie w algorytmach służących do *ujednoznaczniania* wyrażeń (porównaj p. 3.1.4), podobnie jak miara semantycznego pokrewieństwa znaczenia.

3.2.4. Antonimia

Antonimia, czyli przeciwieństwo paradygmatyczne symboli językowych, jest relacją, która najlepiej wpisuje się w strukturalistyczną teorię znaczenia. Należy zwrócić uwagę, że podobnie jak synonimia i podobieństwo znaczeń, relacja ta występuje pomiędzy symbolami należącymi do wszystkich otwartych klas gramatycznych (tj. rzeczownika, czasownika, przymiotnika i przysłówka), więc istotnie może zostać

[5] Z wyjątkiem antonimii, która definiowana jest dla samodzielnych symboli.

[6] Ścisła definicja *statku* wskazuje, że może on poruszać się również pod wodą, w powietrzu oraz w przestrzeni kosmicznej, zatem jest pojęciem znacznie szerszym niż *okręt*.

uznana za relację podstawową. Problem polega jednak na tym, że symbole językowe mogą różnić się pod wieloma względami, dlatego też relacja ta nie jest jednorodna i językoznawcy definiują wiele relacji semantycznych, których podstawową cechą jest różnica występująca pomiędzy symbolami.

Tak np. Lyons [73, s. 262-280] jako najbardziej ogólny typ przeciwieństwa definiuje *opozycję* (ang. *contrast*)[7]. Wyróżnia on również następujące podtypy opozycji:

- *przeciwstawność* (ang. *opposition*), która obejmuje ścisłe opozycje dwuczłonowe, np. *kręgowy/bezkręgowy*, odpowiadające logicznej relacji sprzeczności (negacja jednego członu powoduje prawdziwość drugiego członu),
- *antonimię*, która obejmuje opozycje stopniowalne, np. *lodowaty*, *zimny*, *ciepły* i *gorący*,
- *uzupełnialność*, która obejmuje nieścisłe opozycje dwuczłonowe, np. *kobieta/mężczyzna*, odpowiadające logicznej relacji przeciwieństwa (negacja jednego członu nie oznacza prawdziwości drugiego członu – oba człony mogą być jednocześnie fałszywe, ale nie mogą być jednocześnie prawdziwe),
- *przeciwstawienie kierunkowe*, którego paradygmatycznym przykładem są relacje przestrzenne, takie jak np. *wschód/zachód/północ/południe*,
- *niezgodność*, która obejmuje uszeregowane wartości np. *poniedziałek/wtorek/środa/czwartek/piątek/sobota/niedziela*.

Lyons określa jeszcze wiele dodatkowych podtypów tych relacji, np. przeciwstawniki *prywatywne* i *ekwipolentne*.

Na tych rozróżnieniach bazują twórcy angielskiego WordNetu, ale ich typologia jest znacznie ograniczona. W pierwszym rzędzie zastrzegają [41, s. 39-40] (podobnie jak twórcy polskiego WordNetu), że antonimia nie występuje pomiędzy synsetami, lecz pomiędzy symbolami językowymi. Definiując antonimię odwołują się do obserwacji, która występuje w teście skojarzeniowym: użytkownicy języka często jako pierwsze skojarzenie danego słowa przywołują jego antonim. Ponadto zauważają oni, że relacja antonimii nie występuje pomiędzy wszystkimi słowami w tym samym stopniu – często istnieją klastry (w szczególności przymiotników), w których tylko centralne elementy są swoimi antonimami. Przykładowo w parze *szybki/wolny*, po lewej stronie może występować np. *błyskawiczny* a po prawej *ślamazarny*, ale nie uznamy, że te dwa słowa są swoimi bezpośrednimi antonimami.

Biorąc pod uwagę złożoność zjawiska antonimii nie dziwi fakt, że nie doczekało się ono jeszcze szerszego wykorzystania w algorytmach ekstrakcji informacji z tekstu. Czasami informacje na temat antonimii wykorzystywane są jako jeden z elementów składających się na rozpoznawanie negacji i sprzeczności w tekście [51]. W niniejszej pracy antonimia wykorzystywana jest wyłącznie pośrednio – negatywna informacja o przeciwstawności pojęć wykorzystywana jest przy rozpoznawaniu konfliktów w określaniu klasyfikacji symboli językowych (porównaj p. 7.2.7).

3.2.5. Hiponimia i hiperonimia

Relacja *hiponimii*, zwana również *subsumpcją*, *zawieraniem*, *subordynacją* oraz *relacją specjalizacji* jest jedną z najszerzej eksploatowanych relacji, zarówno w algorytmach przetwarzania tekstu jak i algorytmach operujących na bazach wiedzy (por. [72, 29, 73, 66, 146, 67, 41, 7, 71, 108]). Relacja ta występuję pomiędzy symbolami językowymi – powiemy, że pomiędzy symbolem σ_a oraz σ_b występuje hiponimia (tzn. σ_b jest

[7] Tłumaczenia na j. polski angielskich terminów przywołuję za tłumaczem Adamem Weinsbergiem. Nie jest do końca jasne, czy pozostawienie tłumaczeń bliższych oryginalnym terminom nie byłoby bardziej uzasadnione.

hiponimem σ_a), jeśli σ_b *jest pewnym typem* σ_a, np. na podstawie prawdziwego zadania *pies jest pewnym typem zwierzęcia*, możemy powiedzieć, że *pies* jest hiponimem *zwierzęcia*.

Relacja ta jest podobna do relacji inkluzji zbiorów znanej w matematyce, ale trzeba dokładnie określić to, co stanowi elementy tych zbiorów – jeśli utożsamimy je z egzemplarzami odpowiednich pojęć (*ekstensją*, czyli zakresem tych pojęć), to hiponim σ_b *zawiera się w* σ_a, np. wszystkie psy należą do zbioru obejmującego wszystkie zwierzęta. Jeśli natomiast elementami tymi byłyby znaczenia tych pojęć (*intensja*, czyli treść), to byłoby odwrotnie, bowiem to znaczenie σ_a (symbolu bardziej ogólnego) *zawiera się w* znaczeniu σ_b, dlatego, że każdy pies jest jednocześnie zwierzęciem, zatem składnikiem znaczenia symbolu *pies* jest symbol *zwierzę* (por. [73, s. 281]). Warto również zwrócić uwagę na fakt, że symbole, które posiadają ten sam zakres nie zawsze są synonimami – znanym przykładem są symbole: *zwierzęta posiadające nerkę* oraz *zwierzęta posiadające serce*, które choć mają ten sam zakres, posiadają odmienne znaczenie.

Relacją symetryczną względem hiponimii jest *hiperonimia*. Hiponimia oraz hiperonimia zasadniczo są relacjami przechodnimi. Dzięki temu możliwe jest bardziej zwięzłe reprezentowanie wiedzy – hiponim dziedziczy od hiperonimu jego cechy semantyczne takie jak atrybuty, relacje oraz funkcje [29], [146], [85, s. 26-27], [71, s. 28-29] dlatego możliwe jest zdefiniowanie pewnych cech semantycznych na wyższym poziomie i ich odtwarzanie, w odniesieniu do hiponimów, za pomocą mechanizmu inferencji.

Realność psychologiczna relacji hiponimii była wpierw badana przez Collinsa oraz Quillian [29] w znanym eksperymencie dotyczącym różnicy w czasach odpowiedzi na pytanie: „Czy kanarek potrafi śpiewać?" i „Czy kanarek ma skórę?". Na podstawie dłuższego czasu wymaganego do odpowiedzi na drugie pytanie, autorzy wnioskowali na temat realności odpowiedniej struktury hierarchicznej. Niemniej jednak dalsze badania pokazały, że problem ten jest bardziej złożony, czego efektem było powstanie alternatywnej teorii językoznawczej – teorii prototypu [135].

Hiponimię omawia się najczęściej w odniesieniu do rzeczowników, ale może być również zdefiniowana dla czasowników. Wtedy jednak jej definicja musi być nieco zmodyfikowana: czasownik σ_a jest hiponimem czasownika σ_b, jeśli „robienie σ_a to robienie σ_b *w określony sposób*", np. *gapienie się* to *bezmyślne patrzenie się* (por. [41, s. 79]). Wiele przymiotników również nie ma swoich hiperonimów – Lyons definiuje jednak relację pseudo-hiponimii, która łączy przymiotniki z atrybutem, którego są one wartościami, np. *czerwony* jest pseudo-hiponimem *koloru*, a *kwaśny* jest pseudo-hiponimem *smaku* [73, s. 290].

Warto również wskazać na różnicę pomiędzy relacją hiponimii a relacją *typ-okaz*, która jest pokrewna relacji hiponimii. O relacji typ-okaz mówimy wtedy, gdy „σ_a (okaz) jest σ_b (typ)", np. „Burek jest psem", a o relacji hiponimii gdy „σ_a jest [pewnym] typem σ_b", np. „Pies jest typem zwierzęcia". Często relacje te są utożsamiane, gdyż w drugim przypadku można powiedzieć, że „pies jest zwierzęciem". Czasami jednak utożsamienie tych relacji może prowadzić do błędnych rozumowań (porównaj [79]). Wiele zależy jednak od tego, w jaki sposób dalej wykorzystywana jest ta zależność.

Hiponimia ma bardzo istotne znaczenie w algorytmach ekstrakcji informacji. W pierwszym rzędzie wykorzystywana jest ona do uporządkowania pojęć, wykorzystywanych do klasyfikacji wyrażeń podlegających ekstrakcji (więcej na temat struktur hierarchicznych budowanych na bazie tej relacji można znaleźć w punkcie 3.3.1). Schemat klasyfikacji traktowany jest jako punkt odniesienia dla mechanizmów ekstrakcyjnych i pierwsze, podstawowe zagadnienie w trakcie ekstrakcji informacji polega na określeniu typu obiektów, do których odnoszą się wyrażenia ekstrahowane z tekstu. Zwykle klasyfikacja ta obejmuje również rozpoznanie relacji *typ-okaz*. Relacje te są również wykorzystywane przy ekstrakcji innych relacji, ponieważ pozwalają w sposób bardziej ekonomiczny określać ograniczenia semantyczne pozostałych relacji (porównaj p. 8.9).

3.2.6. Meronimia

Meronimia zwana także relacją *całość-część* zaliczana jest również do podstawowych relacji semantycznych (porównaj [73, 66, 67, 41, 71, 108]). Relacja ta wiąże ze sobą obiekty oraz ich części. Paradygmatycznym przykładem tej relacji są związki pomiędzy *ciałem* i jego elementami, np. *ręką*. Powiemy zatem, że jednym z meronimów *ciała* jest *ręka*. Symetryczną względem niej jest relacja *holonimii – ciało* jest holonimem *ręki*, ponieważ ręka jest częścią ciała.

Przykład opierający się na obiektach fizycznych nie wyczerpuje jednak wszelkich wariantów relacji całość-część. Choć można wyróżnić ich całkiem sporo (porównaj [73, s. 298-303]), w angielskim WordNecie wyróżniono tylko trzy podtypu: *kompozycję*, np. *samochód – kierownica* ; *bycie członkiem*, np. *klasa – uczeń* oraz *materiał*, np. *krzesło – drewno* [41, s. 39]. W polskim WordNecie zestaw ten został rozszerzony o następujące podtypu: *element taksonomiczny*, np. *rośliny nasienne – okrytozalążkowe*; *miejsce*, np. *pustynia – oaza* oraz *porcja*, np. *pieczywo – kromka* [108, s. 31-32]. Nieco zaskakujący jest tutaj brak podtypu obejmującego relacje czasowe, wszak *chwila* może być częścią *godziny* ale trudno zaklasyfikować ten związek do któregoś spośród wymienionych podtypów.

Wagę tej relacji w semantyce języków naturalnych można dość łatwo zilustrować na podstawie pojęć takich jak *mankiet*, *klapa* czy *element*, które trudno byłoby zdefiniować nie odwołując się do tej relacji. Dlatego też, pomimo faktu, że relacja ta charakteryzuje się cechami typowo ontologicznymi (np. relacja między *kołem* i *samochodem* może być zaobserwowana bez pośrednictwa języka), zaliczana jest ona również do relacji semantycznych.

Pewnej uwagi wymagają problemy zwrotności i przechodniości meronimii. W pierwszym rzędzie – to, że jedno pojęcie dla swej definicji wymaga użycia tej relacji, nie oznacza automatycznie, że pojęcie stojące po drugiej stronie musi być zdefiniowane z użyciem symetrycznej relacji holonimii. Przykładem tutaj może być *mankiet* – jest on częścią *koszuli*, ale definicja *koszuli* nie wymaga odwołania się do *mankietów*.

W odniesieniu do przechodniości meronimii przywołać można przykład Lyonsa [73, s. 298-300]: *klamka* jest częścią *drzwi*, zaś *drzwi* są częścią *domu*, ale sformułowanie „klamka domu" jest co najmniej dziwne. Z drugiej jednak strony bez wątpienia klamka jest w sensie ontologicznym częścią domu.

Problemy te powodują, że wnioskowanie na temat relacji meronimii jest znacznie trudniejsze niż wnioskowanie na temat relacji hiponimii. Co więcej niejednorodność tej relacji, którą dostrzegli twórcy różnych wersji WordNetu, dodatkowo utrudnia przetwarzanie informacji pozyskanych z tekstu w oparciu o tę relację. Brak jednoznacznych rozstrzygnięć teoretycznych jest prawdopodobnym powodem znacznie mniejszej liczby publikacji na temat ekstrakcji tej relacji z tekstów w języku naturalnym. Tym niemniej doniosłość tej relacji dla rozumienia treści tekstu zadecydowała o tym, że jest ona podstawową relacją ekstrahowaną z tekstu z użyciem opisywanego algorytmu.

3.2.7. Pozostałe relacje semantyczne

Relacje takie jak synonimia, antonimia, hiponimia i meronimia (oraz odpowiadające im relacje symetryczne) nie wyczerpują całego zbioru relacji semantycznych. Relacje te zaliczają się do relacji paradygmatycznych[8], trudno jednak podejrzewać, że opierając się wyłącznie na relacjach paradygmatycznych można skonstruować skuteczne algorytmy rozumienia tekstu. Nie ulega wątpliwości, że relacje syntagmatyczne odgrywają równie doniosłą rolę, w szczególności w problemach takich jak ujednoznacznianie znaczenia wyrażeń. Tym niemniej nie istnieje kanoniczny zbiór relacji paradygmatycznych.

[8] Z zastrzeżeniem, że w przypadku meronimii możliwość zastępowania wymaga użycia *metonimii*.

Lubaszewski [71, s. 30-31, s. 246-247] wzorując się na pracach Schanka [143] oraz twórcach FrameNetu [8] definiuje np. relacje takie jak: *akcja*, *przeznaczenie*, *rola*, *stan*, *sprawca*, *przedmiot*, *instrument*, *konceptualne źródło* oraz *konceptualny kierunek* zdarzenia, *czas*, *miejsce* oraz *sposób*. Twórcy angielskiego i polskiego WordNetu powstrzymali się od relacji międzykategorialnych (z zastrzeżeniem relacji pomiędzy przymiotnikami i rzeczownikami), dlatego relacje paradygmatyczne nie występują w tym źródle wiedzy. Z kolei twórcy ontologii Cyc [66, 67] nie wyróżniają relacji paradygmatycznych jako specjalnego rodzaju relacji, ponieważ relacje w tym zbiorze wiedzy mają charakter ontologiczny, a nie semantyczny.

W tym miejscu rezygnujemy zatem z bardziej szczegółowego omówienia innych relacji semantycznych, ponieważ pomimo tego, że mają one bardzo istotne znaczenie dla procesu rozumienia tekstu, będą miały one wyłącznie akcydentalne znaczenie dla konstruowanego algorytmu. Podstawową przyczyną tego stanu rzeczy jest fakt, że ich rozpoznawanie wymaga pełnej analizy syntaktycznej tekstu (porównaj [58, s. 670-674]), co, ze względu na brak parsera języka polskiego o jakości wystarczającej dla zastosowań praktycznych, istotnie wykracza poza ramy niniejszej pracy.

3.3. Sieci semantyczne

Elementy łączone za pomocą relacji semantycznych nie są odseparowane od siebie nawzajem, lecz tworzą złożone struktury zwane *sieciami semantycznymi*. Sieci semantyczne stanowią grafową reprezentacją wiedzy, składającą się z węzłów oraz krawędzi. Zwykle węzły reprezentują symbole językowe a krawędzie (semantyczne) relacje zachodzące pomiędzy tymi węzłami. Zastosowanie reprezentacji grafowej w obszarze sztucznej inteligencji pozwala przeprowadzać wnioskowania w obrębie tak reprezentowanej wiedzy z wykorzystaniem dobrze znanych algorytmów grafowych.

Warto zwrócić uwagę, że niektórzy językoznawcy, w szczególności de Saussure [139] oraz kontynuatorzy zainicjowanej przez niego szkoły strukturalnej, np. Lyons [72, 73], twierdzą, że znaczenie danego symbolu językowego może być definiowane jedynie poprzez relacje do innych symboli. W konsekwencji węzły, czyli nośniki znaczenia, nie mogłyby funkcjonować w oderwaniu od sieci relacji semantycznych, a zatem nie można by mówić o znaczeniu symboli językowych w oderwaniu od sieci do której one należą. Wskazują tym samym na prymat całości względem poszczególnych części, w definiowaniu tego co rozumiemy przez znaczenie.

Istnieje wiele typów sieci semantycznych wykorzystywanych w sztucznej inteligencji. Bardzo dobrą ich typografię przedstawia John F. Sowa w haśle *Semantic networks* zamieszczonym w *Encyclopedia of Artificial Intelligence* oraz rozwiniętym i zaktualizowanym na jego stronie internetowej[146]. Typografia ta obejmuje następujące typy sieci semantycznych:

1. sieci definicyjne,
2. sieci asercyjne,
3. sieci implikacyjne,
4. sieci wykonywalne,
5. sieci uczące się,
6. sieci hybrydowe.

Z naszego punktu widzenia najbardziej interesujące są trzy pierwsze typu sieci semantycznych: definicyjne, asercyjne oraz implikacyjne. Pozostałe sieci mają istotne zastosowanie w sztucznej inteligencji

Rysunek 3.2: Fragment sieci definiującej człowieka oraz zwierzę.

i szerzej – w ogólnie rozumianej informatyce – lecz wykorzystywane są jedynie pośrednio w ekstrakcji informacji. Np. sieć neuronowa może być wykorzystana do ujednoznaczniania kategorii gramatycznych słów w zdaniu – jednym z etapów ekstrakcji informacji. Jednakże trzy pierwsze typy sieci mają swoje szczególne miejsce w tej dziedzinie, gdyż mogą z jednej strony być bezpośrednio wykorzystywane do ekstrakcji informacji, np. do reprezentowania oraz rozpoznawania kategorii semantycznej wyrażeń językowych, a z drugiej strony mogą stanowić rezultat ekstrakcji informacji. Dlatego omówienie sieci semantycznych zawęzimy do tych trzech typów.

3.3.1. Sieci definicyjne

Sieci definicyjne są najszerzej znanymi sieciami semantycznymi. Większość opracowań dotyczących reprezentacji wiedzy omawia w ten lub inny sposób sieci definicyjne. Nawet tak prosta reprezentacja wiedzy, jaką stanowią grafy związków-encji, zawiera podstawowy element sieci tego rodzaju, relację *ISA*, która w tym kontekście oznacza relację generalizacji [155].

Podstawą sieci definicyjnych jest relacja hiponimii omówiona w punkcie 3.2.5. Historycznie sieci definicyjne biorą początek w koncepcji *definicji* wypracowanej przez Arystotelesa – poprawnie skonstruowana definicja zawiera dwa kluczowe elementy: *genus proximum* (pol. *rodzaj najbliższy*) oraz *differentia specifica* (pol. *różnicę gatunkową*) [6]. Przykładowo *człowieka* to *zwierzę rozumne* . *Zwierzę* stanowi rodzaj najbliższy, a *rozumność* cechę odróżniającą człowieka od innych zwierząt. Przechodząc do definicji *zwierzęcia*, które jest *istotą żywą obdarzoną zmysłami* możemy zbudować niewielki fragment sieci semantycznej przedstawiony na rysunku 3.2. Strzałki występujące na diagramie pokazują kierunek generalizacji, tzn. pojęcie bardziej ogólne (rodzaj najbliższy) wskazywane jest przez grot strzałki.

Z punktu widzenia zastosowań tego rodzaju sieci semantycznych w sztucznej inteligencji najistotniejszą ich cechą jest to, że relacja generalizacji jest relacją przechodnią, a w konsekwencji cechy przyporządkowane pojęciom bardziej ogólnym, są dziedziczone przez wszystkich ich potomków. Dzięki temu wiedza dotycząca bytów określonego rodzaju nie musi być powtarzana dla każdego symbolu z osobna, lecz może zostać umieszczona na najbardziej ogólnym poziomie, co pozwala uniknąć duplikacji – problemu niezwykle istotnego we wszelkich systemach informatycznych (porównaj p. 3.2.5).

Sieci definicyjne posiadają bardzo długą historię – o ile sam Arystoteles opisał „jedynie" fundament ich konstrukcji, o tyle Porfiriusz w komentarzu do jego *Kategorii* zastosował graficzną reprezentację podobną do tej przedstawionej na rysunku 3.2 zwanej *drzewem Porfiriusza* [146]. Jego komentarz był podstawą nauczania logiki w średniowieczu i wywarł bardzo istotny wpływ na średniowieczną scholastykę.

Metoda zaproponowana przez Arystotelesa i rozwinięta przez Porfiriusza została z wielkim powodzeniem zaadaptowana w biologii przez Karola Linneusza [70], który uczynił ją podstawą klasyfikacji organizmów żywych. Opracowana przez niego system taksonomiczny choć uległ istotnym przekształceniom w sferze materialnej, w sferze formalnej wykorzystywany jest do dnia dzisiejszego. Taksonomia organizmów żywych jest zatem jedną z największych i najlepiej znanych definicyjnych sieci semantycznych.

W XX wieku wraz z rozwojem logiki zaczęto formalizować sieci definicyjne, m.in. na potrzeby sztucznej inteligencji. W tym celu opracowano logiki deskrypcyjne, a jednym z pierwszych języków reprezentacji wiedzy opartym o sieci definicyjne był KL-ONE [146]. Najnowszym, szeroko rozpowszechnionym językiem służącym do budowania sieci definicyjnych jest, zbudowany na potrzeby Semantic Web[9], OWL-DL[10].

Sieci definicyjne wzbudziły również duże zainteresowanie wśród językoznawców, w szczególności tych o nastawieniu kognitywnym. Badania Milera i współpracowników [41] zaowocowały utworzeniem w latach dziewięćdziesiątych XX wieku jednej z najszerzej znanych semantycznych sieci definicyjnych, tj. WordNetu. Istotnym *novum* zastosowanym w konstrukcji tej sieci było oparcie się o pojęcie *synsetu* (porównaj p. 3.2.3). Twórcy WordNetu uznali, że to synsety, a nie poszczególne symbole językowe, wchodzą w określone relacje definicyjne, w związku z czym to te pierwsze były połączone odpowiednimi relacjami. Podstawowymi (definicyjnymi) relacjami w WordNecie są hiponimia i hiperonimia.

Pierwsza wersja WordNetu opisywała angielskie symbole językowe i powstała na uniwersytecie w Princeton. Pomysł ten był jednak na tyle interesujący, w szczególności w kontekście komputerowego przetwarzania języka, że powstało wiele WordNetów stworzonych dla innych języków. Obecnie instytucje zajmujące się rozwijaniem WordNetów zrzeszone są w Global WordNet Association[11]. Wśród nich znajduje się również Politechnika Wrocławska, której powstaje polski WordNet czyli Słowosieć[12] [108]. Obecnie jest to jedna z największych leksykalnych sieci semantycznych na świecie.

Jednym z istotniejszych problemów związanych z sieciami definicyjnymi jest rodzaj wykorzystywanej logiki. Z jednej strony w sieciach tych można wykorzystywać logiki monotoniczne, które charakteryzują się tym, że wnioski wyprowadzone na podstawie zbioru przesłanek nie mogą zostać unieważnione przez dodanie nowych faktów. Cecha ta jest wysoce pożądana z punktu widzenia zastosowań tych sieci, lecz często uniemożliwia reprezentowanie przybliżonych rozumowań, którymi posługują się ludzie w codziennym życiu. Przykładowo, w sieci definicyjnej można by zawrzeć informację, że ssaki są zwierzętami lądowymi. Fakt ten jednak nie obowiązuje bez wyjątków – niektóre ssaki, np. walenie, żyją w wodzie. W sieci opartej o logikę monotoniczną nie można by zatem zawrzeć tej przybliżonej wiedzy. Rozwiązaniem problemu jest użycie logik niemonotonicznych, tzn. logik, w których dodanie kolejnych faktów może unieważnić wcześniej wyciągnięte wnioski. W sieci takiej można by zatem zawrzeć informację, że ssaki zwykle są zwierzętami lądowymi, ale np. dodać informację, że walenie choć są ssakami, są również zwierzętami wodnymi [17, s. 193].

Zastosowanie logik niemonotonicznych prowadzi jednak do istotnych problemów, w szczególności jeśli pozwolimy, aby jedno pojęcie mogło posiadać wiele generalizacji. Klasyczny przykład to Nixon, który stanowił instancję dwóch pojęć: kwakra oraz republikanina. Kwakrzy, w przeciwieństwie do republikanów, są pacyfistami – dlatego w odniesieniu do Nixona nie można było określić, czy jest, czy też nie jest on pacyfistą [17, s. 192]. Istnieją co prawda rozwiązania tego problemu, ale dość istotnie komplikują prowadzenie wnioskowań na podstawie sieci definicyjnych.

[9] Celowo nie tłumaczymy tej nazwy jako „sieć semantyczna" aby nie wprowadzić niepotrzebnej wieloznaczności, wynikającej z polskiego tłumaczenia terminów *semantic network* oraz *semantic web*.

[10] `http://www.w3.org/TR/owl-features/`

[11] `http://www.globalwordnet.org`

[12] `http://www.plwordnet.pwr.wroc.pl/main/`

Sieci definicyjne mają bardzo istotne znaczenie w ekstrakcji informacji. Ponieważ oczekujemy, że informacje ekstrahowane z tekstu będą posiadały przyporządkowaną odpowiednią kategorię semantyczną, są one najczęściej organizowane w sieć definicyjną, która stanowi punkt odniesienia w interpretacji informacji. Z drugiej strony sieci definicyjne mogą być wykorzystywane do rozstrzygania wieloznaczności, a także do uogólniania ograniczeń semantycznych argumentów ekstrahowanych relacji semantycznych. W konsekwencji sieci te nie tylko stanowią referencyjną bazę wiedzy dla algorytmów ekstrakcyjnych, ale również pozwalają na poprawienie wyników ekstrakcji oraz bardziej efektywne ich stosowanie. Słownik semantyczny występujący w definicji 3.1, stanowi w opisywanym systemie właśnie taką referencyjną sieć definicyjną.

3.3.2. Sieci asercyjne

Podstawową różnicą występującą pomiędzy sieciami definicyjnymi a sieciami asercyjnymi jest rodzaj opisywanej wiedzy. Sieci definicyjne służą do wyrażania zależności pomiędzy symbolami językowymi, tzn. stworzono je po to by opisywać semantykę języków naturalnych. Dlatego zwykle ograniczają się one do relacji *hiponimi-hiperonimii* oraz relacji *typ-okaz*. W przeciwieństwie do nich sieci asercyjne mogą być wykorzystywane do wyrażania dowolnych faktów, w szczególności faktów przygodnych, w odróżnieniu od (przynajmniej w pewnym stopniu) koniecznych faktów językowych. Z tego względu charakteryzują się one większym stopniem komplikacji, gdyż notacje tego rodzaju chcą odzwierciedlać wszelkie fakty, które można wyrazić za pomocą języka, bądź przynajmniej fakty, które dają się wyrazić w logice pierwszego rzędu.

Notację tego rodzaju rozwijali w XIX wieku niezależnie od siebie Gottlob Frege oraz Charles Sanders Peirce. Ważnym osiągnięciem Peirce'a było opracowanie notacji algebraicznej, która po zmodyfikowaniu przez Peano używana jest do dnia dzisiejszego w rachunku predykatów [146]. Tym niemniej Peirce nie był do końca zadowolony z tej notacji i w 1909 opublikował pracę [104], w której zaproponował grafową reprezentację w postaci sieci asercyjnej. Jego przełomowym osiągnięciem było zastosowanie jawnej notacji do reprezentowania *zakresu* kwantyfikacji, dzięki czemu możliwe było odwzorowanie wszystkich klasycznych spójników logicznych w postaci grafowej.

Innym ważnym osiągnięciem na drodze rozwoju sieci asercyjnych było opracowanie przez Luciena Tesniére'a notacji dla jego *gramatyki zależności* [152]. W przeciwieństwie do Noama Chomskiego, Tesniére nie koncentrował się na syntaktyce, lecz na semantyce. Koncepcja ta została wykorzystana w jednych z pierwszych systemów służących do automatycznego tłumaczenia tekstów [146]. Jednym z najważniejszych rozwinięć tej koncepcji była jednak teoria *pojęciowej zależności* Rogera Schanka [142, 143] (porównaj p. 4.1.1), która została zastosowana w jednym z pierwszych systemów ekstrahujących informacje.

Notacja Tesniére'a posiadała jednak istotny mankament, uniemożliwiający wyrażanie relacji pomiędzy zdaniami, które są niezbędne do opisywania wielu faktów językowych, w szczególności kontekstów modalnych (np. ***Możliwe, że*** *jutro będzie padał deszcz.*) oraz intensjonalnych (np. ***Bożena wie, że*** *Andrzej nie lubi fasoli.*). Rozwiązaniem tego problemu było zastosowanie *reifikacji*, tj. wprowadzenia bezpośredniej reprezentacji dla stwierdzeń. Dzięki temu możliwe było łączenie stwierdzeń za pomocą relacji odzwierciedlających ich zależności, np. wiedzę jakiejś osoby na określony temat. Pomysł ten został zastosowany między innymi w pierwszym systemie wykorzystującym sieci asercyjne w sztucznej inteligencji, tj. systemie MIND stworzonym przez Stuarta Shapiro [144].

Podobnie jak w przypadku sieci definicyjnych, dokładne odwzorowanie bogactwa języków naturalnych w systemach formalnych narażona jest na istotne problemy. W przypadku sieci asercyjnych najwięcej problemów generują konteksty modalne, gdyż nie zachowują one tzw. *referencjalnej przezroczystości*, tzn.

w tych kontekstach nie można stosować zasady zastępowalności wyrażeń. Problem ten można zilustrować następującym przykładem:

1. (*założenie*) Elektra wie, że za kotarą stoi mężczyzna.
2. (*założenie*) Mężczyzna za kotarą to Orestes.
3. (*teza*) Elektra wie, że za kotarą stoi Orestes (*na podstawie zasady zastępowalności*).

Ale Elektra nie wie, że za kotarą stoi Orestes, ponieważ nie wie jak wygląda mężczyzna za kotarą. Zasada zastępowalności nie może być zatem stosowana bez ograniczeń w kontekstach modalnych.

Jednym z rozwiązań tego problemu jest zastosowanie opracowanej przez Saula Kripkego semantyki światów możliwych [60]. Stosując semantykę światów możliwych prawdziwość zdania nie jest oceniana na podstawie jednego, „aktualnego" świata, lecz zbioru światów możliwych, powiązanych relacją osiągalności. Jeden świat jest osiągalny z innego świata dla określonego podmiotu, jeśli podmiot ten nie posiada wiedzy, która wykluczałaby takie przejście. Przykładowo, jeśli Jan nie wie, jaka jutro będzie pogoda, to świat możliwy, w którym jutro jest słonecznie, jest dla niego osiągalny ze świata, w którym jutro pada deszcz. Problemem, który pozostaje jednak nierozwiązany w tym kontekście jest założenie, że podmioty wiedzy i przekonań znają wszystkie wnioski z nich wynikające – założenie bardzo dalekie od tego, jak faktycznie rozumują ludzie [136, s. 453].

Przechodząc do zastosowań sieci asercyjnych w ekstrakcji informacji należy wskazać na charakter wiedzy, dla której zostały one stworzone – jest to wiedza epizodyczna, która ma odzwierciedlać jak najdokładniej semantykę języka naturalnego. Z tego względu sieci asercyjne wydają się najlepszym mechanizmem reprezentacji informacji ekstrahowanych z tekstu. Co prawda często w tym celu wykorzystuje się prostszą reprezentację, tzn. algebrę relacyjną stosowaną powszechnie w bazach danych. Jednakże posiada ona istotne ograniczenia w zakresie reprezentacji kontekstów modalnych, a także w zakresie wydajności, w przypadku wyszukiwania informacji w grafie wiedzy, który powstaje w wyniku ekstrakcji informacji.

Ontologia Cyc jest ciekawym przykładem sieci asercyjnej. Chociaż jej zawartość, jest zdominowana przez wiedzę ogólną o charakterze sieci definicyjnej, to zawiera ona również opis faktów o charakterze jednostkowym. Ta część ontologii tworzy zatem sieć asercyjną.

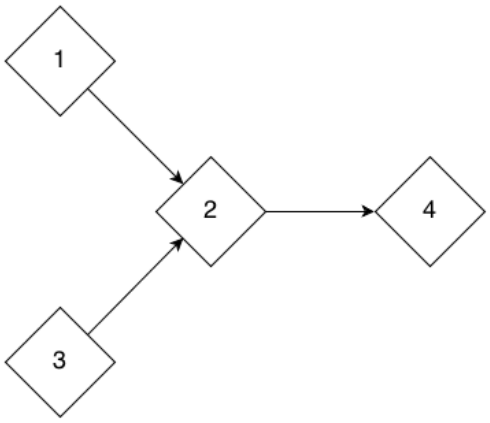
Jednym z ostatnich osiągnięć w dziedzinie sieci asercyjnych jest opracowany na początku XXI język RDF[13], którego przeznaczeniem jest reprezentowanie wiedzy w Semantic Web. Struktura tego języka jest bardzo prosta – stwierdzenia wyrażane w RDF składają się zawsze z trzech elementów:

1. podmiotu (ang. *subject*),
2. predykatu (ang. *predicate*),
3. przedmiotu (ang. *object*).

Podmiot stwierdzenia nie musi być obiektem, ale np. innym stwierdzeniem, do którego odwołujemy się poprzez globalnie rozpoznawalny identyfikator, tj. jednoznaczny adres URL. Dzięki temu w języku RDF możliwe jest opisywanie dowolnych, pozytywnych faktów wykorzystując nomenklaturę, która dzięki globalnemu charakterowi adresów URL, przynajmniej w założeniu, jest jednoznaczna i niezależnie od kontekstu zastosowania.

Globalna uniwersalność notacji RDF zdecydowała o jej dużej popularności – również w kontekście ekstrakcji informacji. Najlepszym przykładem zastosowanie tej notacji jest DBpedia [7] – baza wiedzy

[13] `http://www.w3.org/RDF/`



Rysunek 3.3: Prosta sieć implikacyjna reprezentująca zależności pomiędzy czterema zdaniami. Kwadraty reprezentują zdania, zaś grot strzałki wskazuje następnik implikacji.

wyekstrahowanej z różnych wersji językowych Wikipedii. Istnieją również systemy ekstrakcji informacji takie jak DBpedia Spotlight [82], czy Típalo [45], które przekształcają wyekstrahowane informacje bezpośrednio do formatu RDF.

3.3.3. Sieci implikacyjne

Sieci implikacyjne są specjalnym typem sieci asercyjnych, w których krawędzie odpowiadają relacji implikacji, natomiast węzły stwierdzeniom. Innymi słowy sieci implikacyjne służą do reprezentacji wiedzy odwzorowującej dedukcję. Przykładowo, sieć tego rodzaju może służyć do reprezentacji następujących implikacji:

– *Jeżeli* system zraszania trawy jest uruchomiony (1), *to* trawa jest śliska (2).

– *Jeżeli* pada deszcz (3), *to* trawa jest śliska (2).

– *Jeżeli* trawa jest śliska (2), *to* piłka gorzej toczy się po trawie (4).

Sieć implikacyjna, którą można zbudować na podstawie tych implikacji przedstawiona jest na rysunku 3.3. Na rysunku tym poszczególne zdania reprezentowane są przez kwadraty, natomiast zależności implikacyjne pomiędzy nimi wskazywane są przez strzałki. Np. $1 \rightarrow 2$ oznacza, że zdanie 1 implikuje zdanie 2. Strzałki mogą być również uzupełnione indeksem wskazującym na prawdziwość poprzednika. Wtedy z jednego zdania mogą wychodzić dwie strzałki, jedna wskazująca na jego prawdziwość, druga na jego fałszywość, podobnie jak ma to miejsce w blokach decyzyjnych schematów blokowych wykorzystywanych do graficznej reprezentacji algorytmów. Sieci tego rodzaju pozwalają na prowadzenie wnioskowań zarówno wprzód – wtedy celem jest np. określenie skutków określonego zdarzenia oraz wstecz – wtedy celem jest np. określanie przyczyn jakiegoś zdarzenia.

Implikacja wykorzystywana w sieciach tego rodzaju nie musi być tożsama z logicznym operatorem implikacji występującym w klasycznym rachunku zdań. Dlatego też sieci tego rodzaju mogą być również nazywane [146]:

– sieciami przekonań (ang. *belief network*)

– sieciami kauzalnymi (ang. *causal network*)

– sieciami Bayesa (ang. *Bayesian network*)

– systemami utrzymywania prawdziwości (ang. *truth-maintenance system*, w skrócie TMS)

W szczególności zatem możliwe jest zastosowanie zamiast zwykłego operatora implikacji prawdopodobieństwa (jak ma to miejsce w sieciach przekonań i sieciach Bayesa) albo niemonotonicznego odpowiednika implikacji (jak ma to miejsce w systemach utrzymywania prawdziwości).

W kontekście ekstrakcji informacji sieci implikacyjne mogą mieć podwójne zastosowanie. W pierwszym rzędzie mogą one reprezentować reguły, z użyciem których prowadzone są wnioskowania na bazie wyekstrahowanych informacji. Innymi słowy – mogą one reprezentować wiedzę uzupełniającą, która pozwala wyciągać wnioski wykraczające poza informacje bezpośrednio występujące w analizowanych tekstach. Z drugiej strony sieci implikacyjne mogą również powstawać na podstawie informacji zawartych w tekstach. Tym niemniej ekstrahowanie reguł z tekstu jest tematem dość słabo rozpoznanym w literaturze przedmiotu. W szczególności w porównaniu do innych metod wydobywania wiedzy regułowej, np. z baz danych.

Podsumowanie

W niniejszym rozdziale przedstawione zostały podstawowe pojęcia, które będą występowały w dalszej części pracy. W szczególności omówiono pojęcie *symbolu językowego*, które stosowane będzie wszędzie tam gdzie odwoływać będziemy się do znaczenia słów. Jest one przeciwstawione *napisom*, które stanowią niezinterpretowane składniki tekstu. To przeciwstawienie wynika wprost z definicji ekstrakcji informacji przedstawionej w punkcie 2.1, gdyż opis tekstu z wykorzystaniem odwołania do napisów jest opisem metajęzykowych. Relacje pomiędzy symbolami językowymi a napisami muszą również uwzględniać zjawiska takie jak odmian wyrazów oraz występowanie symboli, którym odpowiadają wyrazy wielosegmentowe.

Ponadto przedstawiony został warunek konieczny bycia *relacją semantyczną*, tj. występowanie pomiędzy dwoma symbolami językowymi. Zastrzeżono jednak, że nie wszystkie relacje występujące pomiędzy symbolami językowymi są relacjami semantycznymi. Wskazano również na podobieństwa oraz różnice pomiędzy relacjami semantycznymi i ontologicznymi oraz omówiono kilka podstawowych relacji semantycznych, szczególnie istotnych z punktu widzenia ekstrakcji informacji.

W ostatniej części omówiono różne rodzaje sieci, które przydatne są w algorytmach sztucznej inteligencji, w szczególności zaś w algorytmach ekstrakcji informacji. Wśród nich najważniejsze są sieci definicyjne – w szczególności słowniki semantyczne – które zawierają definicje znaczeń symboli językowych, stanowiące podstawę dla algorytmów ekstrakcji informacji, a także sieci asercyjne, w których może być przechowywana wiedza wydobywana w trakcie ekstrakcji informacji.

4. Historia i stan badań nad ekstrakcją informacji

4.1. Ekstrakcja informacji w języku angielskim

4.1.1. Początki ekstrakcji informacji

Historia badań nad ekstrakcją informacji sięga lat siedemdziesiątych poprzedniego wieku. Cowie i Lehnert [30] a także Grishman i Sundheim [50] wskazują pracę grupy pod przewodnictwem Naomi Sager [137] jako pierwsze badania w dziedzinie ekstrakcji informacji. Były one sponsorowane przez Amerykańskie Towarzystwo Medyczne i koncentrowały się na zamianie pisemnych raportów o stanie pacjentów na wpisy w elektronicznej bazie danych.[1]

Drugą ważną pracą, która wskazywana jest w historycznych przeglądach badań nad ekstrakcją informacji [30, s. 81], [92, s. 25-26], [58, s. 762-763] jest system FRUMP (*Fast Reading Understanding and Memory Program*) stworzony i opisany przez DeJonga w pracy *Prediction and Substantiation: A New Approach to Natural Language Processing* [34].

System FRUMP analizował elektroniczne notatki prasowe dostarczane przez United Press International, które obejmowały szeroki zakres informacji z całego Świata. Na podstawie treści tych wiadomości generował jednozdaniowe podsumowania zawierające najistotniejsze informacje zawarte w notatkach. Podsumowania te mogły być generowane zarówno w języku angielskim, jak i kilku innych językach, gdyż wewnętrzny format reprezentacji informacji oparty był na teorii *pojęciowej zależności* Schanka (*Conceptual Dependency Theory* – [142, 143]).

Charakterystyczną cechą systemu FRUMP były dwa moduły „predyktor" oraz „substancjator", które współpracowały ze sobą w trybie sprzężenia zwrotnego. W przeciwieństwie do innych systemów, w których analiza syntaktyczna poprzedzała przetwarzanie informacji, FRUMP integrował analizę syntaktyczną z analizą semantyczną. Naczelną rolę w systemie odgrywał predyktor, który generował przewidywania na temat tego, co pojawi się w nieprzeanalizowanej części notatki. Przewidywania te były przekazywane do substancjatora, który wypełniał je konkretnymi wartościami odnalezionymi w tekście, wykorzystując w tym celu powierzchniową analizę syntaktyczną, bądź zwracał informację, że oczekiwanie te nie zostały spełnione. W drugim przypadku predyktor dokonywał modyfikacji swoich przewidywań i ponownie przekazywał je do substancjatora.

Przewidywania predyktora generowane była na podstawie *uproszczonych skryptów* – rozwiązania bazującego na koncepcji Schanka, ale wprowadzającego istotne uproszczenia. Teoria Schanka [143] zakładała, że rozumienie tekstu może być modelowane za pomocą zbioru podstawowych, niezależnych od języka pojęć pierwotnych, dzięki którym można budować złożone pojęcia występujące w języku. Do pojęć podstawowych należały *generatory wyobrażeń* (ang. *picture producers*) oraz *akty* (ang. *acts*), które wspomagane były przez modyfikatory odpowiednio generatorów wyobrażeń oraz aktów. Najważniejszym elementem tej

[1] Praca Sager niestety nie jest dostępna w wersji elektronicznej. Nie posiada jej również żadna z polskich bibliotek.

teorii były akty, wśród których Schank zidentyfikował 11 aktów podstawowych. Według Schanka, za pomocą aktów podstawowych, używając reguł składni semantycznej, można było opisać dowolne zdarzenie rzeczywistości fizycznej[2].

Sposób konstrukcji aktów podstawowych można zilustrować na następującym przykładzie [143, s. 21]:

Jan pojechał do Nowego Jorku.

Formalna reprezentacja odpowiadająca temu zdaniu jest następująca:

```
         actor: Jan
        action: PTRANS
        object: Jan
  direction TO: Nowy Jork
direction FROM: unknown
```

Typ zdarzenia `PTRANS` wskazuje na pierwotny akt *fizycznego przemieszczenia się*. W opisie zdarzenia występuję podmiot, czyli sprawca zdarzenia (`actor`), przedmiot zdarzenia (`object`) oraz fizyczny kierunek zdarzenia (`direction FROM` oraz `direction TO`). Zdanie to można skontrastować ze zdaniem:

Jan poleciał do Nowego Jorku.

w którym oprócz informacji o tym co stało się z Janem, mamy również dostępną informację na temat tego jaki instrument został wykorzystany do realizacji tego zdarzenia:

```
         actor: samolot
        action: PROPEL
        object: samolot
  direction TO: Nowy Jork
direction FROM: unknown
```

W tym zdarzeniu mamy do czynienia z aktem pierwotnym `PROPEL` polegającym na wprawianiu obiektów w ruch za pomocą siły. Pełna analiza zdarzenia wymaga również uwzględnienia faktu, że Jan dostał się w jakiś sposób do samolotu:

```
         actor: Jan
        action: PTRANS
        object: Jan
  direction TO: samolot
direction FROM: unknown
```

Ponieważ analiza taka mogłaby ciągnąć się w nieskończoność (czemu nie uwzględnić w niej tego co działo się podczas lotu, np. faktu, że Jan czytał gazetę oraz spożywał posiłek?), Schank w celu ograniczenia poziomu analizy wprowadził pojęcie *skryptu* – stereotypowego ciągu aktów, wymagane do realizacji określonego zamierzenia. Obejmowały one zdarzenia typowe oraz warianty występujące w specyficznych kontekstach. Przykładowy skrypt dla zakupów obejmowałby wybór towaru, podejście do kasy oraz zapłatę za towar. Skrypt taki mógł zawierać elementy opcjonalne, np. pakowanie towaru oraz warianty, np. dla sprzedaży wysyłkowej, w której klient nie podchodzi do kasy ale zamawia towar przez telefon,

[2]Schank w pracy [143, s. 17-26] zastrzega, że jego prace nie definiują pełnego zbioru aktów podstawowych, wystarczającego do opisu dowolnego zdarzenia. Jest to fakt dość często ignorowany przez krytyków jego teorii.

a sprzedawca nadaje towar na poczcie. Istotnym aspektem skryptów było częściowe uporządkowanie aktów w ramach skryptu (klient nie podchodzi do kasy zanim nie wybierze towaru, oczywiście przy założeniu, że wybór towaru nie odbywa się przy kasie).

Skrypty Schanka wykorzystywane np. w programie SAM [143, s. 75-119] byłby bardzo szczegółowe i ich konstrukcja była długotrwała. Dlatego DeJong w swoim programie zastosował skrypty uproszczone, które obejmowały jedynie najistotniejsze zdarzenia. Pomimo tych ograniczeń system FRUMP wykazywał zaskakująco wysoką precyzję działania.

Pierwsze wyniki badań nad ekstrakcją informacji wyglądały obiecująco. Najistotniejszym ich aspektem było to, że zaczęły one przynosić obserwowalne efekty, które można było łatwo porównać z kompetencją przeciętnych użytkowników języka. O ile inne zadania w obszarze przetwarzania języka naturalnego, takie jak tagowanie morfosyntaktyczne, czy analiza syntaktyczna zakładały istnienie pewnej teorii językoznawczej, o tyle systemy ekstrakcji informacji, choć opierały się na różnych teoriach językoznawczych, mogły być testowane w całkowitej izolacji od nich. Był to istotny impuls do rozwoju systemów tego rodzaju oraz zobiektywizowania kryteriów ich oceny.

4.1.2. Rozwój metod symbolicznych

Message Understanding Conference (w skrócie MUC) była serią konferencji organizowanych w latach 1987-1995, których celem było zbadanie możliwości stworzenia systemów ekstrahujących informacje z angielskich[3] tekstów [50]. Pierwsza edycja MUC-1 (1987) została poświęcona głównie na ustalenie przedmiotu badań. Każdy z zespołów stosował swój własny format danych, co uniemożliwiło porównanie osiąganych przez nie wyników. W trakcie MUC-2 (1989) zostało określone podstawowe zadanie, tj. wypełnianie szablonów, na podstawie którego porównywana była skuteczność systemów. Szablony składały się z 10 pozycji, a teksty z których ekstrahowano informacje dotyczyły raportów marynarki wojennej na temat incydentów morskich i obserwacji. Istotnym osiągnięciem drugiej edycji było wypracowanie miar służących do oceny skuteczności systemów: wykorzystano znane z wyszukiwania informacji miary precyzji (*precision*) oraz pokrycia (*recall*), ale definiowane były one w odniesieniu do pozycji, które należało uzupełnić w szablonach ekstrakcyjnych.

W kolejnych edycjach – MUC-3 (1991) oraz MUC-4 (1992) – zaczęto ekstrahować informacje z dziedziny aktów terrorystycznych obserwowanych w Ameryce Środkowej oraz Południowej. Edycje te przyniosły również dalszą komplikację szablonów, posiadały one odpowiednio 18 oraz 24 pozycje. W trakcie piątej edycji MUC-5 (1993) kolejny raz zmieniono dziedzinę ekstrakcji informacji (obejmowała ona przedsięwzięcia *joint venture* oraz proces produkcji półprzewodników), a także wprowadzono szablony składające się z pod-szablonów (porównaj p. 2.4.6).

W trakcie przygotowań do kolejnej edycji konferencji przyszedł czas na refleksję. O ile wyniki uzyskiwane przez najlepsze systemy osiągały blisko 70% precyzji oraz pokrycia, o tyle ich przygotowanie zajmowało bardzo dużo czasu – kilka miesięcy. Uświadomiono sobie, że taki nakład pracy jest zdecydowanie zbyt duży, jeśli systemy te miałyby być wykorzystywane powszechnie. Ponadto opracowanie systemu wymagało wiedzy zarówno w dziedzinie przetwarzania języka naturalnego, jak i w dziedzinie, z której ekstrahowano informacje. Dlatego też celem kolejnej edycji było zademonstrowanie możliwych usprawnień w następujących obszarach:

– konstrukcji elementów systemu, które byłyby niezależne od dziedziny ekstrakcji,

– procesu dostosowywania systemów ekstrakcji do określonej dziedziny wiedzy,

[3] W późniejszych edycjach występowały również teksty japońskie.

– głębszego rozumienia przetwarzanych informacji.

W odniesieniu do pierwszego zagadnienia uznano, że w każdym zadaniu ekstrakcji informacji konieczne jest rozpoznanie jednostek referencyjnych oraz wyrażeń temporalnych i innych wyrażeń odnoszących się do wartości liczbowych (np. ceny, temperatury, itp.) (porównaj p. 2.3.3, 2.4.1, 2.4.4).

Bliższe przyjrzenie się zagadnieniu przenośności systemów skutkowało zdefiniowaniem uproszczonego zadania wypełniania szablonów (mini-MUC). Zauważono bowiem, że niektóre szablony (np. dotyczące obiektów będących podmiotami i przedmiotami zdarzeń) mogą być wykorzystywane w wielu dziedzinach. Dlatego też istotnie uproszczono wykorzystywane szablony.

Głębsze zrozumienie przetwarzanych informacji doprecyzowano jako rozwiązanie następujących podproblemów:

– rozpoznawanie wyrażeń współodnoszących się (porównaj p. 2.4.2),

– rozstrzyganie wieloznaczności (porównaj p. 2.3.2),

– rozpoznawanie struktur predykatywnych (porównaj p. 2.4.3).

W trakcie ewaluacji systemów okazało się, że wyodrębnienie tych podproblemów dało dobre rezultaty. W szczególności rozpoznawanie jednostek referencyjnych mogło być realizowane z bardzo dobrą precyzją oraz pokryciem, sięgającymi ponad 90%. Problem wypełniania uproszczonych szablonów również dał dobre rezultaty, ale mimo dużego podobieństwa do zadania rozpoznawania jednostek referencyjnych, nadal pozostawał problemem trudnym. Szczególnie istotne okazało się wyodrębnienie problemu rozpoznawania wyrażeń współodnoszących się. O ile wyniki w tej dziedzinie nie były w pełni zadowalające, zespoły uczestniczące w konferencji uznały, że w tym obszarze można uzyskać znaczną poprawę.

Przykładem systemu uczestniczącego w ewaluacjach prowadzonych w ramach konferencji MUC jest FASTUS [53]. System ten, wykorzystywał kaskadę niedeterministycznych automatów skończonych do analizy tekstów języka angielskiego, w szczególności do rozpoznania wcześniej zdefiniowanych zdarzeń i wypełnienia powiązanych z nimi szablonów ekstrakcyjnych.

System ten składał się z następujących warstw realizowanych w postaci automatów skończonych:

1. warstwy rozpoznawania złożonych słów, tj. nazw własnych oraz wyrażeń wielosegmentowych,

2. warstwy rozpoznawania fraz prostych, obejmującej trzy typy fraz: grupy nominale, grupy czasownikowe oraz partykuły,

3. warstwy rozpoznawania fraz złożonych, tj. złożonych grup nominalnych oraz złożonych grup czasownikowych,

4. warstwy rozpoznawania i ekstrahowania zdarzeń oraz ich składników,

5. warstwy odpowiadającej za łączenie informacji dotyczących identycznych obiektów i zdarzeń.

Wyniki produkowane przez warstwę niższego poziomu stawały się wejściem dla warstwy poziomu wyższego. Taka konstrukcja systemu pozwalała na powtórne użycie warstw 1-3 niezależnie od dziedziny, z której ekstrahowano informacje. System ten mógł również wykrywać nieznane mu nazwy obiektów, dzięki zastosowaniu ogólnych reguł w warstwie 1, a także generował wiele wariantów szablonów ekstrakcyjnych wykorzystywanych w warstwie 3. Tym samym istotnie ograniczał ręczną pracę niezbędną do dostosowania go do wybranej dziedziny wiedzy. Najistotniejszą warstwę z punktu widzenie ekstrakcji informacji stanowiła warstwa 4. Korzystając z wyrażeń regularnych definiowano w niej dziedzinowe

wzorce ekstrakcji informacji, które były bezpośrednio powiązane z uzupełnianymi przez system pozycjami szablonu wyrażonego na wysokim poziomie abstrakcji.

Dzięki zastosowaniu kaskady automatów skończonych system działał bardzo szybko, co umożliwiło jego łatwo testowanie i dostosowywanie do nowych dziedzin wiedzy. System osiągał precyzję w przedziale 52-62% oraz pokrycie w przedziale 34-44%.

Najistotniejszym ograniczeniem tego i innych systemów uczestniczących w ewaluacji w ramach konferencji MUC była konieczność ręcznego konstruowania wzorców ekstrakcyjnych, na podstawie których dokonywana była ekstrakcja. Wymagało to wiedzy eksperckiej zarówno z dziedziny, z której pochodziły analizowane teksty, ale również z zakresu gramatyk formalnych, co istotnie utrudniało stosowanie go na szeroką skalę. Podobne obserwacje dotyczyły innych systemów biorących udział w konferencji MUC. Z tego względu w trakcie dalszej pracy nad systemami ekstrakcji informacji zaczęto kłaść nacisk na możliwość automatycznego pozyskiwania wzorców ekstrakcyjnych.

4.1.3. Zwrot ku metodom statystycznym

Długi czas potrzebny na ręczne skonstruowanie wzorców ekstrakcyjnych wykorzystywanych w systemach opartych o paradygmat symboliczny, które zdominowały MUC spowodował, że naukowcy zaczęli poszukiwać rozwiązań, w których zadanie to mogłoby zostać przynajmniej częściowo zautomatyzowane. Ponadto poszukiwano rozwiązań pozwalających ekspertom dziedzinowym, nieposiadającym wiedzy w zakresie programowania i przetwarzania języka naturalnego, na adaptowanie systemu tego rodzaju do własnych potrzeb. Zaowocowało to zainteresowaniem badaczy systemami, w których wzorce ekstrakcyjne były konstruowane na podstawie zbioru uczącego. W zbiorze tym każdy tekst posiadał odpowiadający mu zbiór informacji, które powinny zostać z niego wyekstrahowane. Ciężar konstrukcji odpowiednich wzorców spoczywałby na algorytmach wykorzystujących statystyczną analizę tak uzyskanych danych.

Tego rodzaju podejście zastosowane zostało m.in. w systemie WHISK [145]. Podobnie jak system FASTUS, konstrukcja WHISKa opierała się o reguły ekstrakcyjne, które można były wyrazić w języku regularnym. Podstawowa zasada konstruowania reguł opierała się na następującej formule:

$$E_i = \frac{e+1}{n+1}, \tag{4.1}$$

gdzie E_i, to przybliżona wartość błędu reguły o numerze i, e to liczba przykładów w zbiorze uczącym, dla których ta reguła generowała niepoprawne dane, a n to liczba wszystkich przykładów uczących, pasujących do reguły i. Miara ta była wykorzystywana do wyboru reguł, które najlepiej odpowiadałyby danym uczącym.

Algorytm WHISKa rozpoczynał swoje działanie z regułą, która nie była ograniczona w żaden sposób (tzn. pasowała do wszystkich przykładów), ale ze względu na jej postać, która rozpoczynała się od uniwersalnego, nieograniczonego dopasowania, nie ekstrahowała żadnych informacji. Dopiero dzięki sprecyzowaniu reguły, tzn. zamianie elementów o dopasowaniu uniwersalnym na konkretne wartości, generowała ona niepuste dopasowania. Następnie tak określone reguły były uogólniane, poprzez zastępowanie napisów kategoriami semantycznymi określonymi przez użytkownika lub przez parser semantyczny. Jeśli reguła uzyskana w ten sposób posiadała niższy współczynnik błędu, oryginalna reguła była zastępowana regułą bardziej ogólną.

Istotnym aspektem działania systemu był sposób doboru przykładów, na podstawie których konstruowane były reguły. Algorytm nie wymagał od użytkownika znakowania przykładów „na ślepo", ale wybierał te przykłady, które pozwoliłyby uogólnić istniejące reguły lub zbudować reguły dla przykładów pozbawionych dopasowań. Przykładowe teksty dzielone były na trzy grupy:

1. teksty pasujące do jednej z istniejących reguł,
2. teksty częściowo pasują do jednej z istniejących reguł,
3. teksty nie pasujące do żadnej z reguł.

Pierwsza grupa przykładów wykorzystywana była do poprawy precyzji algorytmu – w szczególności wyszukiwania kontrprzykładów dla istniejących reguł. Druga grupa, w której zawarte były przykłady w przybliżeniu pasujące do którejś z istniejących reguł, wykorzystywana była do zwiększenia pokrycia algorytmu. Natomiast trzecia grupa przykładów wykorzystywana była przede wszystkim w pierwszej fazie algorytmu, kiedy trzeba było określić początkowy zbiór reguł ekstrakcyjnych.

Dla pełnego zbioru przykładów uczących (6900) zdefiniowanego w ramach konferencji MUC-6 wyniki uzyskiwane przez WHISKa były lepsze od tych uzyskiwanych przez systemy z ręcznie definiowanymi wzorcami ekstrakcyjnymi, osiągając 72% precyzji oraz 55% pokrycia. Wykorzystując heurystykę doboru przykładów WHISK uczył się szybciej niż gdyby przykłady dobierane były losowo. Tym niemniej dla mniejszego zbioru obejmującego 800 przykładów, precyzja wynosiła 47% a pokrycie 35% – daleko poza obszarem praktycznej stosowalności tego rozwiązania.

Inne podejście zastosowano w systemie AutoSlog-TS [134]. W przeciwieństwie do WHISKa, który wymagał aby użytkownik podał dokładne wyniki ekstrakcji dla przykładów prezentowanych przez system, AutoSlog-TS wymagał jedynie oznaczenia, czy przykład zawiera informacje, które mają zostać wyekstrahowane. Dzięki temu złożona odpowiedź użytkownika była zastępowana odpowiedzią tak/nie, co istotnie przyspieszało proces przygotowywania przykładów uczących.

Posiadając dwa zbiory przykładów – tekstów relewantnych oraz nierelewantnych dla danego problemu, algorytm generował wszystkie potencjalne wzorce ekstrakcyjne zawierające frazę nominalną. Następnie wzorce, które posiadały tylko jedno dopasowanie były odrzucane. Dla pozostałych wzorców określano współczynnik relewancji zdefiniowany jako iloczyn logarytmu z częstości dopasowania wzorca oraz prawdopodobieństwa warunkowego należenia danego tekstu zawierającego wzorzec do grupy przykładów relewantnych, pod warunkiem dopasowania wzorca. Tak uzyskane wzorce były przeglądane w kolejności malejącej relewancji i przypisywano im (ręcznie) odpowiednie zdarzenia oraz pozycje w szablonie ekstrakcyjnym.

Zastosowanie metod opartych o zbiory uczące pozwoliło przezwyciężyć najistotniejsze ograniczenie związane z konstrukcją systemów ekstrahujących informacje, tj. długi czas potrzebny na dostosowanie systemu do nowej dziedziny wiedzy. Niemniej jednak systemy tego rodzaju ciągle wymagały dość dużego nakładu pracy ręcznej, a jakość uzyskiwanych wyników, choć uległa poprawie, nie pozwalała na całkowitą automatyzację procesu ekstrakcji informacji.

4.1.4. Wykorzystanie danych zarodkowych

Zastosowanie opisanych wcześniej metod statystycznych przyczyniło się do przyspieszenia procesu adaptowania systemów ekstrakcji wiedzy do nowych dziedzin. Przede wszystkim adaptacja systemów mogła być wykonywana przez ekspertów dziedzinowych, którzy zwykle nie posiadają wiedzy z zakresu przetwarzania języka. W połączeniu z konstrukcją przyjaznych interfejsów użytkownika, proces ten mógł być realizowany przez system w trakcie jego adaptacji, poprzez zadawanie prostych pytań użytkownikowi.

Jednak pomimo tak istotnego przyspieszenia i uproszczenia, procesu ten był nadal dość żmudny. Konieczność oznakowania setek, bądź tysięcy przykładów uczących była ciągle wąskim gardłem, które istotnie utrudniało wykorzystywanie systemów ekstrakcji informacji. Dlatego też poszukiwano metod, które pozwoliłyby znacząco ograniczyć ilość informacji potrzebną do tego by dostosować system do nowej

dziedziny. Ponadto rosnąca popularność Internetu jako źródła informacji, a także dostępność dużych baz zawierających kolekcje tekstów (korpusów) pozwoliła na stworzenie metod, które minimalizowały nakład pracy ręcznej.

Podstawowy pomysł zastosowany w systemach ekstrakcji informacji powstałych pod koniec poprzedniego wieku, polega na wykorzystaniu tzw. *danych zarodkowych* (ang. *seed data*) oraz *wzorców zarodkowych* (ang. *seed patterns*). Praca ręczna niezbędna do dostosowania systemu do nowej dziedziny sprowadzała się do wprowadzenia zaledwie kilku przykładowych informacji, charakterystycznych dla analizowanego problemu. Informacje te mogły być wypełnionymi szablonami ekstrakcyjnymi lub parami elementów połączonych określoną relacją semantyczną. Przykładowo: przy budowie systemu ekstrahującego informacje na temat autorów książek, wystarczyło podać kilka par, np.:

– Adam Mickiewicz – Pan Tadeusz,

– William Szekspir – Romeo i Julia,

– Henryk Sienkiewicz – W pustyni i w puszczy.

Zadaniem systemu było odnalezienie fragmentów tekstu, w których te pary występują oraz automatyczne zbudowanie wzorców ekstrakcyjnych, które pozwoliłyby na wyekstrahowanie innych par: *autor – dzieło literackie*. Tak uzyskane pary mogły być powtórnie wykorzystane do odkrycia nowych wzorców ekstrakcyjnych, na podstawie których można było uzyskiwać nowe dane. Proces ten mógł zatem powtarzać się wielokrotnie, co pozytywnie wpływało na zwiększenie pokrycia.

Jeden z pierwszych systemów tego rodzaj został opisany w pracy Riloff i Jones *Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping* [133]. W artykule tym opisany jest system, który służy do automatycznego, jednoczesnego pozyskiwania wzorców ekstrakcyjnych oraz słownika terminów, które mogą występować na określonych pozycjach w szablonie ekstrakcyjnym.

Działanie systemu było następujące: w pierwszej fazie system ekstrahował wszystkie frazy nominalne, które występowały w korpusie tekstów zawierającym dane treningowe. Następnie we frazach wyszukiwano wystąpień jednego ze słów zarodkowych przypisanych do określonej kategorii semantycznej. Przykładowo, dla kategorii *location* (miejsce) autorki użyły następujących słów: *australia, canada, china, england, france, germany, japan, mexico, switzerland, united_states*. Frazy zawierające jedno z wymienionych słów, po usunięciu tego słowa, dodawane były do zbioru wzorców ekstrahujących miejsca. W kolejnej fazie wzorce te były sortowane ze względu na metrykę uwzględniającą liczbę unikalnych słów, do których one pasowały. Na tej podstawie można było również określić, które słowa należące do danej kategorii semantycznej, są najbardziej wiarygodne.

Istotnym elementem całego procesu było to, że po kilku iteracjach na podstawie których pozyskiwano jednocześnie nowe wzorce ekstrakcyjne oraz nowe słowa należące do określonej kategorii semantycznej, określano najbardziej wiarygodne słowa i cały proces powtarzano od początku (w szczególności odrzucano wszystkie wzorce ekstrakcyjne).

Algorytm Riloff i Jones został przetestowany m.in na danych z konferencji MUC-4 [50]. Uzyskiwane wyniki zależały od liczby iteracji w zewnętrznej pętli algorytmu. Choć z rosnącą liczbą iteracji precyzja algorytmu zmniejszała się, to i tak utrzymywała się na dość wysokim poziomie: po 10 iteracjach średnia precyzja wynosiła 75%, po 20: 76,8%, a po 50: 57,6%, natomiast pokrycie osiągało średnio 38,3%.

Inną pracą, która wywarła istotny wpływ na rozwój algorytmów ekstrakcji informacji był eksperyment opisany w artykule Sergeya Brina *Extracting Patterns and Relations from the World Wide Web* [18]. Praca ta była o tyle istotna, że jako źródło wiedzy służące do budowy wzorców ekstrakcyjnych użyto danych zgromadzonych w Internecie. Pomysł ten zapoczątkował tendencję, która utrzymuje się do dzisiaj,

tzn. wykorzystywanie olbrzymiego zbioru danych tekstowych dostępnych w sieciach rozległych. Podobnie jak Riloff i Jones, Brin zastosował zbiór danych zarodkowych w celu zbudowania wzorców ekstrakcyjnych, jednakże inaczej postawił cel konstrukcji algorytmu. W tym przypadku chodziło o automatyczne zbudowanie bazy danych zawierającej pary *autor – tytuł książki*. Zadanie to było więc dużo prostsze niż uzupełnianie rozbudowanych szablonów ekstrakcyjnych, tak jak to miało miejsce w eksperymentach realizowanych w ramach serii MUC.

Algorytm Brina był następujący:

1. $R' \leftarrow Sample$
   W zbiorze krotek R' umieść krotki zarodkowe,
2. $O \leftarrow FindOccurrences(R', D)$
   Znajdź wystąpienia O krotek ze zbioru R' w zbiorze dokumentów D,
3. $P \leftarrow GenPatterns(O)$
   Zbuduj wzorce ekstrakcyjne P na podstawie zbioru wystąpień krotek O,
4. $R' \leftarrow M_D(P)$
   Przeszukaj bazę dokumentów w celu wykrycia krotek pasujących do wzorców P,
5. Jeśli R' jest wystarczająco duże, zakończ, jeśli nie wróć do punktu 2.

Najważniejszą innowacją tego algorytmu był punkt 3, tzn. budowa wzorców ekstrakcyjnych. W algorytmie tym użyto predefiniowanych wyrażeń regularnych, do których musiało pasować nazwisko autora oraz tytuł książki. W ten sposób eliminowano dopasowania, które nie wyglądały jak pary *autor – książka*. Sam wzorzec był zaś 5-elementową krotką: (*kolejność, prefiks_url, prefiks, środek, sufiks*). *Kolejność* określała czy autor występował przed tytułem, czy po nim, *prefiks_url* zawierał prefiks adresu URL strony, na której znaleziono daną parę, zaś *prefiks*, *środek* i *sufiks* były wyrażeniami, które występowało odpowiednio przed, pomiędzy oraz po dopasowanych elementach (w zależności od porządku: autorze i tytule, bądź tytule i autorze), tzn. stanowiły lewy, środkowy oraz prawy kontekst dopasowania.

Do tak uzyskanych wzorców przypisywana była miara specyficzności zdefiniowana jak iloczyn liczby liter występujących w prefiksie adresu URL, prefiksie, środku i sufiksie wzorca oraz liczbie wyekstrahowanych krotek. Wzorce poniżej ustalonego progu t oraz wzorce generujące mniej niż 2 krotki były odrzucane.

Początkowo planowano, że eksperymenty z wykorzystaniem tego algorytmu przeprowadzone będą na zbiorze 24 milionów dokumentów, lecz proces ten był zbyto powolny, dlatego kolejne iteracje były wykonywane na pewnym ich podzbiorze. Zbiór danych zarodkowych obejmował 5 par *autor – tytuł*. Ich wystąpień poszukiwano w podzbiorze zawierającym 5 milionów dokumentów, gdzie odnaleziono ich 199. Na tej podstawie zbudowano 3 wzorce ekstrakcyjne. Użycie tak uzyskanych wzorców na tym samym podzbiorze dało w wyniku 4047 unikalnych par *autor – tytuł*. Poszukiwanie tych par w kolejnym zbiorze 5 milionów dokumentów dało 3972 wystąpienia, które pozwoliły na zbudowanie 105 wzorców. Użycie tych wzorców z kolei dało 9369 unikalnych par *autor – tytuł*. Finalna iteracja wyprodukowała zaś 346 wzorców, które pozwoliły wykryć 15257 unikalnych par *autor – tytuł*. Analiza 20 losowo wybranych wyników wykazała, że tylko jeden z nich był niepoprawny – zamiast pary *autor – tytuł książki* była to para *autor – tytuł artykułu*.

Idea wykorzystana przez Riloff, Jones oraz Brina znalazła swoje rozwinięcie w pracy Agichteina i Gravano. W artykule *Snowball: Extracting Relations from Large Plain-Text Collections* [2] opisali oni system ekstrakcji relacji semantycznych, w którym wprost powołują się na pracę Brina, do której wprowadzili jednak istotne modyfikacje, skutkujące istotną poprawną precyzji działania algorytmu. Pierwsza modyfikacja polegała na odejściu od wyrażeń regularnych, stosowanych do wykrywania argumentów relacji na

rzecz kategorii semantycznych przypisywanych za pomocą odrębnego narzędzia – w tym wypadku systemu Alembic [32]. Dzięki temu system mógł pominąć te dopasowania wzorca ekstrakcyjnego, w których kategorie semantyczne argumentów relacji nie były zgodne z kategoriami występującymi we wzorcu.

Druga modyfikacja polegała na odejściu od ścisłego dopasowania lewego, środkowego oraz prawego kontekstu występującego we wzorcu na rzecz metryki opartej o wektorowy model języka wykorzystywany w systemach wyszukiwania informacji. Słowom występującym w kontekście dopasowania przypisywano wagę określoną na podstawie dopasowań krotek zarodkowych (a w późniejszych iteracjach – krotek uzyskanych w iteracjach wcześniejszych). Decyzja o tym, czy dany wzorzec pasuje do określonego fragmentu tekstu była zaś określana na podstawie wartości metryki cosinusowej – jeśli wartość iloczynu skalarnego przekraczała określony próg, uznawano, że wzorzec został dopasowany.

Trzecia, według autorów najważniejsza, modyfikacja dotyczyła sposobu wybierania krotek oraz wzorców, które przechodziły do kolejnej iteracji algorytmu. Ponieważ opisywany system był wykorzystywany do ekstrakcji krotek o postaci *przedsiębiorstwo – główna siedziba*, możliwe było wykorzystanie zależności funkcyjnych[4] do określania, czy nowa krotka wyekstrahowana przez określony wzorzec ekstrakcyjny jest zgodna z wcześniej wyekstrahowanymi krotkami, o ile pierwsza pozycja tej krotki (tzn. przedsiębiorstwo) znalazła się już w bazie danych. Na tej podstawie można zdefiniować poziom pewności danego wzorca, jako stosunek liczby krotek zgodnych do liczby krotek zgodnych oraz niezgodnych z wcześniejszymi wynikami. Ponadto wszystkie wzorce, które miały wsparcie (tzn. liczbę dopasowań do krotek z poprzedniego kroku ekstrakcji) poniżej określonego progu były odrzucane.

Podobny mechanizm został zastosowany do oceny ekstrahowanych krotek – aby określona krotka została dodana do bazy, musiała pasować (biorąc pod uwagę wcześniej zdefiniowany poziom dopasowania) do odpowiedniej liczby wzorców posiadających odpowiedni stopień pewności.

System *Snowball* został przetestowany na dużym zbiorze artykułów prasowych, obejmującym ponad 100 tysięcy pozycji. Wyniki uzyskiwane przez system zależały od tego ile razy określona para *przedsiębiorstwo – siedziba* występowała w tych tekstach. Dla par posiadających tylko jedno wystąpienie, system uzyskiwał pokrycie na poziomie niespełna 80%, a precyzję na poziomie 85%. Tak wysoka jakość wyników w połączeniu z bardzo niewielkim zbiorem danych zarodkowych obejmującym jedynie 5 pozycji, świadczyła o istotnej przewadze tego rozwiązania nad wcześniej opracowanymi systemami. Pokazywała również, że ekstrakcja informacji jest problemem, który może doczekać się praktycznego rozwiązania, bez użycia olbrzymich nakładów finansowych.

4.1.5. Zastosowanie strukturalnych i quasi-strukturalnych źródeł wiedzy

Wraz z rozwojem systemów zdolnych do ekstrakcji informacji nie ograniczających się do wybranej dziedziny wiedzy zaczęto rozumieć, że tradycyjne problemy z dziedziny przetwarzania języka naturalnego, w szczególności problem ujednoznacznianie sensu, mają również istotne znaczenie dla problemu ekstrakcji informacji. O ile jednak w tradycyjnym NLP istniały słowniki takie jak WordNet [41], które starały się wyodrębniać i opisywać zbiory znaczeń dla poszczególnych słów, niezbędne do rozstrzygania wieloznaczności, o tyle w przypadku ekstrakcji informacji, ze względu na jej charakter, obejmujący przede wszystkim nazwy własne, stworzenie słowników tego rodzaju byłoby niezmiernie kosztowne i długotrwałe.

Dlatego też naukowcy szybko dostrzegli potencjał Wikipedii jako źródła wiedzy bardzo przydatnego w procesie ekstrakcji informacji. Jedna z pierwszych prób wykorzystania tej encyklopedii do rozstrzygania wieloznaczności opisana jest w artykule Mihalcea'y [84]. Algorytm ten został użyty w narzędziu Wikify!

[4] Każde przedsiębiorstwo posiada tylko jedną główną siedzibę, porównaj [155]

[83], które służyło do wzbogacania tekstów stron internetowych o automatycznie generowane odnośniki do istotnych artykułów w Wikipedii.

Na bazie Wikipedii zdefiniowano wiele metryk semantycznego podobieństwa między pojęciami, które poza pracą Mihalcea'y zostały również opisane m.in. w pracach Gabrilovicha i Markovitcha [43] oraz Milnego i Wittena [160]. Ulepszony algorytm Milnego i Wittena jest wykorzystywany również w niniejszej pracy, dlatego po jego omówienie odsyłamy do punktu 7.3 oraz do pracy [123].

Obok algorytmów ujednoznaczniających na bazie Wikipedii zaczęto również tworzyć strukturalne źródła wiedzy, które mają istotne znaczenie w procesie ekstrakcji informacji. Do najbardziej znanych systemów tego rodzaju należą YAGO [149] oraz DBpedia [7]. Autorzy YAGO powiązali dane Wikipedii z ontologią SUMO [98] oraz angielskim WordNetem [33]. Wykorzystali w tym celu system kategorii strukturyzujący Wikipedię bowiem zauważyli, że jeśli główny rzeczownik w nazwie kategorii występuje w liczbie mnogiej, to taka kategoria zwykle stanowi kategorię semantyczną pojęć, które do niej należą. Wiążąc te rzeczowniki z pojęciami w WordNecie mogli przypisać artykułom Wikipedii kategorie semantyczne wzięte z tego słownika.

DBpedia wykorzystuje natomiast inną cechę Wikipedii – ustrukturyzowane informacje występujące w znacznej liczbie artykułów nazywane *infoboksami* (patrz rys. 6.1). Zawierają one informacje w formie tabelarycznej dzięki czemu można łatwo przekształcić je do postaci nadającej się do przechowywania w relacyjnej bądź semantycznej bazie danych. Co więcej – na podstawie nazwy infoboksu można również określić kategorię semantyczną, do której przynależy dane pojęcie. Typy infoboksów zostały ujednolicone i uporządkowane w wyniku czego powstała niewielka ontologia obejmująca ponad 300 klas[5]. Należy jednak zwrócić uwagę, że ontologia ta obejmuje swoim zasięgiem jedynie około połowy artykułów występujących w angielskiej wersji Wikipedii – znaczna ich część nie posiada infoboksu, który pozwalałby określić ich kategorię semantyczną.

Na podstawie YAGO oraz DBpedii powstały systemy, takie jak Sofie [150] oraz DBpedia Spotlight [82], których celem jest ekstrakcja informacji. Obecnie systemy te pozwalają głównie na rozpoznawanie jednostek referencyjnych, choć w kontekście tych systemów mówi się częściej o *linkowaniu obiektów* (ang. *entity linking*). Trwają również intensywne prace nad zastosowaniem ich w problemie ekstrakcji relacji semantycznych. System tego rodzaju opisany jest między innymi w pracy Exnera i Nuguesa *Entity Extraction: From Unstructured Text to DBpedia RDF Triples* [39]. Charakterystyczną cechą tego systemu jest to, że na etapie tworzenia klasyfikatora relacji semantycznych, dane zgromadzone w DBpedii wykorzystywane są jako przykłady zarodkowe.

4.1.6. Aktualne tendencje

Niewątpliwą zaletą *Snowball* i jemu podobnych systemów ekstrakcji informacji opierających się na zastosowaniu danych zarodkowych jest to, że pozwalają istotnie ograniczyć nakład pracy ręcznej niezbędnej aby zaadaptować je do nowych dziedzin. Tym niemniej jeśli chcemy zbudować prawdziwie skalowalny system, który przetwarzałby informacje dostępne w Internecie do postaci ustrukturyzowanej, konieczność dostarczenia zaledwie kilku przykładów dla każdej ekstrahowanej relacji jest nadal istotną przeszkodą w realizacji tego celu. Z tego powodu ostatnio coraz większym zainteresowaniem cieszą się tzw. *otwarte* systemy ekstrakcji informacji. Przykładem systemu tego rodzaju jest *TextRunner* [10]. Ogólny schemat jego działania jest następujący:

1. Korzystając z niewielkiego korpusu wyjściowego (nie posiadającego żadnych ręcznie określonych metadanych) system analizuje go wykorzystując parser syntaktyczny. W drzewie rozbioru syntak-

[5]Dokładnie 359 w momencie pisanie tych słów. Aktualna liczba dostępna jest pod adresem `http://dbpedia.org/Ontology`

tycznego identyfikuje gałęzie, które stanowią potencjalne wystąpienia relacji semantycznych. Korzystając z kilku heurystyk oznacza poszczególnego wyekstrahowane gałęzie jako wiarygodne oraz niewiarygodne. Na tej podstawie trenowany jest klasyfikator bayesowski.

2. Następnie system przetwarza główny korpus (składający się z milionów dokumentów) i identyfikuje w nim wyrażenia nominalne, stosując powierzchniową analizę syntaktyczną. Na tej podstawie ekstrahowane są krotki, które podlegają ocenie klasyfikatora bayesowkiego wytrenowanego w poprzednim punkcie.

3. W ostatniej fazie krotki są ujednolicane i obliczana jest liczba wystąpień identycznych krotek. Stosując model statystyczny opisany w [36] krotkom tym przypisywane jest prawdopodobieństwo bycia poprawnymi ekstrakcjami. Prawdopodobieństwo to jest obliczane na podstawie liczby wystąpień danej krotki oraz liczby wszystkich krotek wyekstrahowanych dla danej relacji.

Innym systemem tego rodzaju jest *ReVerb* [40]. System ten charakteryzuje się jeszcze większą prostotą i ogólnością niż *TextRunner*, a mimo to osiąga lepsze wyniki ekstrakcyjne. Dzieje sie tak, ponieważ autorzy wykorzystują w nim prostą modyfikację w stosunku do *TextRunnera*. Mianowicie zamiast najpierw identyfikować argumenty relacji, a w następnym kroku określać wystąpienie relacji, odwracają ten proces, rozpoczynając od sprawdzenia czy w określonym fragmencie tekstu występuje określona relacja. Jeśli występuje kilka nakładających się na siebie relacji, to są one łączone. Dopiero w kolejnym kroku system ustala argumenty relacji i jeśli spełniają one określone ograniczenia syntaktyczne, system uznaje, że ma do czynienia z wystąpieniem relacji semantycznej. Należy zwrócić uwagę, że system ten nie korzysta z parsera języka angielskiego, lecz opiera się jedynie na dopasowaniu wartości kategorii gramatycznych do wyrażenia regularnego.

Dzięki zastosowaniu ogólnego wyrażenia regularnego *ReVerb* osiąga bardzo wysokie pokrycia. Niemniej jednak precyzja wyników jest dość niska. Aby przezwyciężyć ten problem autorzy stosują dwie metody – po pierwsze ignorują wzorce ekstrakcyjne, które w korpusie posiadały mniej niż 20 dopasowań. Dzięki temu udaje im się wyeliminować zbyt specyficzne relacje semantyczne. Ponadto wykorzystują zbiór 500 przykładów uczących (niezależnych od ekstrahowanych relacji) i na jego podstawie trenują klasyfikator wykorzystujący regresję logarytmiczną. W rezultacie system jest w stanie osiągać 90% precyzję przy 30% pokryciu, co daje wynik znacząco lepszy od *TextRunnera*.

Chociaż otwarte systemy ekstrakcji informacji nie pasują do definicji zaproponowanej w punkcie 2.1, ponieważ nie wykorzystują żadnego schematu interpretacyjnego oraz nie dokonują ujednoznaczniania wyrażeń oraz tylko w ograniczonym zakresie rozpoznają wyrażenia synonimiczne [11, s. 33-34], [88, s. 1003] to niewątpliwie stanowią bardzo przydatne narzędzia, które przyczyniają się do urzeczywistnienia wizji całkowicie automatycznej ekstrakcji informacji. Posiadają również wiele zastosowań praktycznych, np. ułatwiają weryfikację faktów na podstawie informacji zawartych w dokumentach tekstowych, a także pozwalają na badanie opinii wyrażanych przez internautów. Co więcej informacje ekstrahowane przez te systemy mogą zostać zinterpretowane, jeśli połączy się ich działanie z tradycyjnymi systemami ekstrakcji [11, s. 32], co dodatkowo może przyczynić sie do poprawy jakości otrzymywanych wyników oraz pozwolić na stworzenie systemu w pełni interpretującego język naturalny.

4.1.7. Ekstrakcja relacji semantycznych

W dotychczas przedstawionej historii ekstrakcji informacji dla języka angielskiego nie koncentrowaliśmy się na relacjach semantycznych. W praktyce większość opisywanych systemów stworzona została albo po to by wypełniać szablony ekstrakcyjne, w których spośród relacji semantycznych najczęściej pojawia

się *meronimia*, albo po to by ekstrahować specyficzne relacje ontologiczne (jak np. *autor – książka*, *przedsiębiorstwo – główna siedziba*, itp.). Istnieje jednak kilka interesujących prac poświęconych ekstrahowaniu specyficznych relacji semantycznych, w szczególności hiponimii oraz meronimii.

Jedną z najlepiej znanych i najczęściej cytowanych prac na ten temat jest artykuł Hearst [52]. Autorka przedstawiła w nim metodę rozpoznawania hiponimii z wykorzystaniem prostych wzorców gramatyczno-tekstowych. Przykładowo wzorzec

$$NP_0, \text{ such as } \{NP_1, NP_2 \ldots \text{ (and | or) } NP_n \} , \tag{4.2}$$

który może być dopasowany do zdania

> The *bow lute*, such as *Bambara ndang* is plucked and has an individual curved neck for each string[6],

pozwala wyekstrahować następujący przykład hiponimii: (*Bambara ndang*, *bow lute*).

W swojej pracy Hearst wskazała 6 wzorców tego rodzaju, które zdolne były do rozpoznawania relacji hiponimii z wysoką precyzją [52, s. 541][7]:

- **such** NP **as** $\{NP, \} * \{$ **(and | or)** $\} NP$,
  np. *works by such authors as Herrick, Goldsmith, and Shakespeare*,
- $NP \{NP, \} * \{, \}$ **or other** NP,
  np. *Bruises, wounds, broken bones or other injuries ...*,
- $NP \{NP, \} * \{, \}$ **and other** NP,
  np. *templates, treasuries, and other important civic buildings*,
- NP **including** $\{NP, \} * \{$ **(and | or)** $\} NP$,
  np. *All common-law countries, including Canada, England, ...*,
- $NP \{, \}$ **especially** $\{NP, \} * \{$ **(and | or)** $\} NP$,
  np. *most European countries, especially France, England, and Spain.*

Metoda pozwalająca na znalezienie tych wzorców była podobna jak w przypadku algorytmów opisanych w punkcie 4.1.4, z tym zastrzeżeniem, że wzorce ekstrakcyjne byłyby tworzone ręcznie przez eksperymentatora na podstawie zdań, w których znaleziono wystąpienia danych zarodkowych. Hearst próbowała również użyć powyższej metody do ekstrahowania meronimii. Okazało się jednak, że uzyskane wzorce były wieloznaczne (dominowały w nich wzorce postaci *X of Y* oraz *X's Y*, wskazujące na pewien typ relacji posesywnej) i autorka nie rozwijała tej metody.

Kolejne ważne osiągnięcia w zakresie rozpoznawania meronimii były opisane w pracy Berlanda i Charniaka [13]. W pierwszej kolejności autorzy określili na zasadzie analogicznej do metody Hearst, 5 wzorców charakteryzujących meronimię w języku angielskim [13, s. 58]:

- $NN[-PL]_w$**'s** $NN[-PL]_p$,
  np. *building's basement*,
- $NN[-PL]_p$ **of** $\{$ **(the | a)** $\} [JJ|NN] * NN_w$,
  np. *basement of a building*,

[6] Lutnia łukowa, taka jak Bambara ndang jest instrumentem szarpanym i posiada osobny, wygięty zaczep dla każdej struny – *tłum. autora.*

[7] Pominięto wzorzec nr 1, który przedstawiony jest wcześniej.

- NN_P **in{ (the | a) }**$[JJ|NN] * NN_w$,
  np. *basement in a building*,
- $NN - PL_p$ **of** $NN - PL_w$,
  np. *basements of buildings*,
- $NN - PL_p$ **in** $NN - PL_w$,
  np. *basements in buildings*,

gdzie indeks:

- w – oznacza całość,
- p – oznacza część.

Ponieważ jakość wyników produkowanych przez poszczególne wzorce mocno się różniła, autorzy wykorzystali tylko pierwsze dwa, gdyż charakteryzowały się najwyższą precyzją. Aby ograniczyć liczbę błędów ekstrakcji, wyniki były w pierwszym rzędzie filtrowane ze względu na występowanie w rzeczownikach końcówek *ing*, *ness* oraz *ity*, charakterystycznych dla cech obiektów, a następnie sortowane względem miary asocjacji słów tworzących relację meronimii. Autorzy wykorzystali w tym wypadku miarę *sigdiff*, która oparta jest na różnicy prawdopodobieństwa wystąpienia określonego słowa oraz tego samego słowa pod warunkiem wystąpienia drugiego słowa. W ten sposób dla danego słowa wyjściowego stanowiącego całość (np. *car* lub *building*), tworzony był ranking słów odpowiadających częściom danego obiektu. Lista ta mogła być dalej wykorzystywane np. do uzupełnienia zawartości słownika takiego jak WordNet.

Najciekawsza z naszego punktu widzenia są jednak prace Girju i współpracowników [47, 46]. Zasadnicza różnica w stosunku do metody Hearst, a także Berlanda i Charniaka dotyczyła określenia ograniczeń semantycznych dla ekstrahowanej relacji. Girju podobnie jak Hearst, również zidentyfikowała gramatyczno-tekstowe wzorce relacji (3 w przypadku meronimii), ale dopiero w następnym kroku jej algorytm określał, czy dana para symboli językowych powiązana jest odpowiednią relacją, badając czy oba symbole spełniają ograniczenia semantyczne powiązane z danym wzorcem.

Bardzo istotnym elementem algorytmu opisanego w [47, s. 4-6] było automatyczne określanie ograniczeń semantycznych na podstawie ręcznie oznakowanego zbioru danych uczących, składającego się z prawie 35 tys. par symboli (w tym niemal 28 tys. zaczerpniętych z angielskiego WordNetu). Algorytm ten nazwany w późniejszej pracy *iterative semantic specialization* (ISS) [46] wygląda następująco. W pierwszej kolejności fragmenty tekstu pasujące do wzorców ekstrakcyjnych są ręcznie oznaczane ze względu na występowanie w nich meronimii – w wyniku czego powstają zbiory pozytywnych oraz negatywnych przykładów wystąpienia tej relacji. Następnie napisy występujące na pozycjach odpowiadających *całości* oraz *części* ujednoznaczniane są względem angielskiego WordNetu (ten etap pomijany był w przypadku przykładów pochodzących z korpusu SemCor, były ujednoznacznione względem WordNetu). Wykorzystując przechodniość hiponimii, specyficzne pojęcia występujące w przykładach odnalezionych w tekście zastępowane były najbardziej ogólnymi pojęciami[8].

Uogólnienie ograniczeń semantycznych mogło jednak prowadzić ponownie do problemu wieloznaczności – tzn. dla tych samych uogólnionych par ograniczeń występowały zarówno pozytywne jak i negatywne przykłady meronimii. Dlatego też ogólne ograniczenia posiadające niejednoznaczne przykłady tekstowe, były zastępowane swoimi specjalizacjami, tak długo, aż niejednoznaczność ta została wyeliminowana. Przykładowo, jeśli wśród przykładów pozytywnych wystąpiła para (*noga,pszczoła*), która uogólniana jest do pary (*entity#1,entity#1*) a wśród negatywnych para (*ul,pszczoła*), która uogólniana jest do tej samej

[8]W angielskim WrodNecie dla rzeczowników określono 11 takich pojęć [41, s. 29].

pary, to algorytm zastępował niejednoznaczne ograniczenie (*entity#1*,*entity#1*), bardziej specyficznymi (*thing#12*,*organism#1*) oraz (*object#1*,*organism#1*) pozbywając się tym samym wieloznaczności. Ponadto reguły mogły przyjmować bardziej skomplikowaną formę, w której ograniczenia argumentu określane były jako koniunkcja pozytywnie określonej specjalizacji pewnego ogólnego pojęcia oraz negatywnie określonej specjalizacji jednego lub więcej pojęć, będących specjalizacjami tego ogólnego pojęcia. W ten sposób możliwe było bardziej zwięzłe określanie ograniczeń dla pojęć posiadających wiele specjalizacji, spośród których tylko niektóre wykluczały zastosowanie określonej reguły.

Precyzja rozpoznawania relacji meronimii dla tego algorytmu wynosiła 83%, a pokrycie od 72% (jeśli wziąć pod uwagę wszystkie wystąpienia meronimii) do 98% (jeśli wziąć pod uwagę wyłącznie wystąpienia pasujące do zdefiniowanych wzorców) [47]. Liczba rozpoznanych relacji nie była zbyt duża – w korpusie zawierającym 10000 zdań algorytm rozpoznał 140 wystąpień meronimii. Podobne wyniki były również osiągane dla bardziej zaawansowanej wersji algorytmu opisanej w [46].

Podsumowanie

Historia ekstrakcji informacji z tekstów w języku angielskim pokazuje istotny trend – pierwsze systemy ekstrakcji wykorzystywały ręcznie definiowane reguły i ograniczone były do wybranej, wąskiej dziedziny wiedzy. Problem ten został dostrzeżony i w latach dziewięćdziesiątych zaczęto zwracać się ku systemem, które pozwalały na przyspieszenie tego procesu, np. poprzez oznakowanie odpowiedniego zbioru tekstów lub podział tekstów na istotne oraz nieistotne z punktu widzenia prowadzonych analiz. Koniec lat dziewięćdziesiątych – pojawienie się coraz wydajniejszych systemów komputerowych oraz zwiększenie ilości informacji elektronicznych, w szczególności dostępnych w Internecie, skutkowało budową skutecznych algorytmów ekstrakcji opartych o analizę statystyczną, działających niezależnie od dziedziny wiedzy. Zjawiskiem najistotniejszym dla ekstrakcji informacji w XXI wieku było zaś pojawienie się Wikipedii i wykorzystanie jej w różnych algorytmach z dziedziny przetwarzania języka naturalnego, np. w ujednoznacznianiu słów i wyrażeń. Dzięki temu ekstrakcja informacji stała się dziedziną wiedzy, w której metody statystyczne łączy się z powodzeniem z metodami opierającymi się na strukturalnych oraz quasi-strukturalnych źródłach wiedzy. Trend wykorzystywania Wikipedii w ekstrakcji informacji utrzymuje się również w najnowszych badaniach, w szczególności w systemach pozwalających na analizę informacji w wielu językach jednocześnie [5].

Istnieją jednak problemy z zakresu ekstrakcji relacji semantycznych, dla których nie udało się jeszcze opracować całkowicie automatycznych algorytmów, których wyniki można by bez interwencji człowieka wykorzystywać praktycznie. Na przykład ekstrakcja meronimii, ze względu na wysoką wieloznaczność wzorców formalnych sygnalizujących jej wystąpienie, wymaga istotnych nakładów materialnych niezbędnych do opracowania danych treningowych, pozwalających na określenie odpowiednich ograniczeń semantycznych. Co więcej, jak pokażemy w punkcie 4.2.2, dla języków z mniejszą liczbą dostępnych zasobów językowych oraz gotowych narzędzi, takich jak język polski, osiągnięcie opisywanych w literaturze wyników jest jeszcze trudniejsze.

4.2. Ekstrakcja informacji w języku polskim

Badania nad ekstrakcją informacji w języku polskim rozpoczęły się znacznie później niż dla języka angielskiego, bo dopiero na początku XXI wieku. Pierwszy szerzej znany artykuł na temat ekstrakcji informacji jest autorstwa Piskorskiego [114] i dotyczy zastosowania platformy SProUT [37] do rozpoznawania jednostek referencyjnych w polskich tekstach. Z tego powodu zaawansowanie prac badawczych nad

ekstrakcją informacji w języku polskim jest znacznie mniejsze. Ze względu na istotne różnice występujące pomiędzy językiem polskim i angielskim, zaadaptowanie dla języka polskiego znanych algorytmów ekstrakcji informacji wymaga istotnego nakładu pracy, a czasami prowadzi do znacznie gorszych rezultatów. Dlatego też ekstrakcja informacji w języku polskim traktowana jest w niniejszej pracy w pewnym stopniu jako odrębny problem badawczy, a wyniki uzyskiwane przez autora porównywane są przede wszystkim z wynikami uzyskiwanymi przez systemy dedykowane dla języka polskiego.

Wśród badań nad ekstrakcją informacji w języku polskim największym zainteresowaniem cieszy się rozpoznawanie jednostek referencyjnych. Istnieje szereg prac poświęconych temu zagadnieniu, powstałych w szczególności w ostatnich latach [114, 116, 1, 111, 48, 157, 77, 78]. Na temat pozostałych zadań definiowanych w ramach ekstrakcji informacji literatura przedmiotu jest znacznie uboższa: istnieje zaledwie kilka pozycji poświęconych wypełnianiu szablonów [93, 94, 76], ekstrakcji relacji semantycznych [56, 57, 108] (porównaj punkt 4.2.2) oraz pojedyncze publikacje na temat rozpoznawania wyrażeń współodnoszących się [101]. Prace korzystające z semi-strukturalnych źródeł wiedzy są sporadyczne i ograniczają się do określania kategorii semantycznej haseł encyklopedycznych [23, 24].

4.2.1. Rozpoznawanie jednostek referencyjnych

Wykorzystanie systemu SProUT

Prace Piskorskiego i współpracowników [114, 115, 113] opisują najprawdopodobniej pierwsze badania w dziedzinie ekstrakcji informacji w języku polskim. Najbardziej kompletne ich omówienie znajduje się w pozycji [113]. Autorzy przedstawiają zastosowanie systemu SProUT [37] (**S**hallow **Pr**ocessing with **U**nification and **T**yped Feature Structures) w zadaniu rozpoznawania jednostek referencyjnych w języku polskim oraz szereg związanych z tym problemów. System SProUT powstał jako odpowiedź na problemy ekstrakcji informacji występujące w językach innych niż język angielski. Jego pierwsza wersja dedykowana była językowi niemieckiemu, następnie został on zaadaptowany m.in. dla języków słowiańskich (litewskiego, czeskiego oraz polskiego).

Rozpoznawanie nazw w tych językach wymaga uzgodnienia wartości kategorii gramatycznych pomiędzy elementami nazwy, np. rodzaju oraz liczby rzeczownika i przymiotnika będących składnikami nazwy. SProUT oparty jest na gramatyce unifikującej, w warstwie implementacji posługuje się jednak automatami skończonymi, dzięki czemu działa dość efektywnie. Jest on systemem regułowym – lewa strona każdej reguły definiowana jest z wykorzystaniem wyrażenia regularnego odwołującego się do struktur o cechach typowanych (ang. *typed feature structures – TFS*), natomiast prawa strona określa strukturę TFS, która powstanie w wyniku dopasowania tego wyrażenia. Unifikacja cech wewnątrz dopasowania odbywa się z wykorzystaniem zmiennych. Dzięki nim możliwe jest również określenie cech w strukturze wynikowej. Lewa strona reguł może również zawierać odwołania do innych reguł, natomiast w prawej mogą występować wywołania pozwalające na przekształcenie elementów dopasowania, np. połączenie ich znakiem spacji. Dzięki tym cechom formalizm SProUT jest wysoce ekspresywny i pozwala na zwięzłe definiowanie reguł opisujących nazwy własne.

Zaadaptowanie systemu dla języka polskiego wymagało uzgodnienia wymagań formalizmu z wykorzystywanym przez Piskorskiego i współpracowników słownika fleksyjnego Morfeusz [163]. Klasy gramatyczne słów zostały wykorzystane jako typy struktur TFS, a wartości kategorii gramatycznych stały się cechami tych struktur. Fleksja nazw w języku polskim jest jednak bardzo złożonym zagadnieniem. Ze względu na dość swobodny szyk słów, który obejmuje również znaczną część nazw, definicje reguł nie mogą zbyt ściśle odpowiadać jedynie najczęstszym sposobom ich wyrażania, lecz muszą uwzględniać ich dużą zmienność. Co prawda zazwyczaj odmienna jest jedynie syntaktyczna głowa nazwy, która uzgadniana jest z zależ-

Tablica 4.1: Jakość rozpoznawania jednostek referencyjnych uzyskiwana w oparciu o platformę SProUT w wiadomościach prasowych z dziedziny finansów – [115].

Rodzaj wyrażenia	**Precyzja** [%]	**Pokrycie** [%]
Wyrażenie temporalne	81,3	85,9
Waluty	97,8	93,8
Wartości procentowe	100,0	100,0
Nazwiska	90,6	85,3
Nazwy geograficzne	88,4	43,4
Nazwy organizacji	87,9	56,6

nymi od niej przymiotnikami, ale w nazwach takich jak *Biblioteka Główna Wyższej Szkoły Handlowej* drugi człon (*Wyższej Szkoły Handlowej*) jest odmieniony i w tej formie występuje w nazwie. Dlatego też algorytm sprowadzający wszystkie formy fleksyjne do form podstawowych nie może być zastosowany w tym przypadku.

Autorzy zauważają również, że wykorzystywanie słowników nazw własnych (ang. *gazetteer*), popularnych w systemach dla języka angielskiego, jest znacznie utrudnione w języku polskim. O ile określona nazwa jest jednosegmentowa, można na jej podstawie wygenerować wszystkie jej warianty. Jeśli jednak jest ona wielosegmentowa, to ze względu na złożoną fleksję nazw oraz wskazany wcześniej swobodny szyk wyrazów, automatyczne generowanie wariantów prowadzi do olbrzymiego wzrostu liczby wyrażeń, które muszą być rozpoznane, nie gwarantując jednak, że zbiór ten jest kompletny. Z tego względu autorzy zaadaptowali SProUT tak by akceptował również tekst z formami słów sprowadzonymi do formy podstawowej. Ponadto utworzono wiele generycznych reguł, np. zaczynających się od słów *agencja*, *bank*, *komisja*, które pozwalały na rozpoznawanie nazw niewystępujących w słowniku.

Autorzy wskazali również, że złożona fleksja nazw własnych w języku polskim istotnie utrudnia określanie form podstawowych poszczególnych składników nazwy. Dzieje się tak m.in. dlatego, że słownik fleksyjny jako formę podstawową przymiotnika zwraca zawsze formę rodzaju męskiego, podczas gdy musi ona być uzgodniona z występującym w nazwie rzeczownikiem.

Platforma SProUT zaadaptowana dla języka polskiego była wykorzystana i testowana w kilku konfiguracjach. W pracy [115] opisano jej zastosowanie w domenie wiadomości finansowych zaczerpniętych z dziennika *Rzeczpospolita*. Ekstrahowane były następujące typy wyrażeń i nazw: wyrażenia odnoszące się do *czasu*, *walut* oraz *wartości procentowych* a także *nazwiska*, *nazwy organizacji* oraz *miejsc*. Jakość wyników zestawiona jest w tabeli 4.1. Rozpoznawanie wyrażeń odnoszących się do walut oraz wartości procentowych jest bardzo dobre (na poziomie bliskim 100%). Nieco gorsze wyniki system uzyskiwał dla nazw osób oraz wyrażeń temporalnych (precyzja w przedziale 80-90%, pokrycie na poziomie 85%). Najtrudniejszym zagadnieniem okazało się rozpoznawanie nazw geograficznych oraz nazw organizacji – pokrycie w pierwszej dziedzinie nie przekroczyło 45% a w drugiej 57%, przy precyzji na poziomie 88%. Należy zwrócić uwagę, że testy były przeprowadzane na dość niewielkim zbiorze artykułów, obejmującym 100 notatek prasowych.

Ten sam algorytm rozpoznawania jednostek referencyjnych został również użyty w systemie ekstrahującym informacje z raportów medycznych dotyczących zmian nowotworowych [93]. Praca ta nie podaje jednak szczegółowych wyników w zakresie rozpoznawania jednostek referencyjnych. Inne zastosowanie algorytmu omówione jest w pracy Abramowicza i współpracowników [1], gdzie został on zastosowany w systemie ekstrakcji informacji katastralnych. System ten kładł szczególny nacisk na rozpoznawanie

Tablica 4.2: Jakość rozpoznawania jednostek referencyjnych uzyskiwana w oparciu o platformę SProUT na potrzeby systemu informacji katastralnej – [1]. Zestawienie zawiera jedynie najlepsze wyniki dla częściowych dopasowań nazw.

Rodzaj wyrażenia	**Precyzja** [%]	**Pokrycie** [%]
Nazwiska osób	91,0	78,0
Nazwy geograficzne	82,0	72,0
Nazwy organizacji	92,0	52,0

nazw geograficznych. W tym celu stworzona została klasyfikacja nazw obejmująca niemal 30 pozycji, oznaczających jednostki *podziału administracyjnego*, *elementy składowe adresów*, a także różne *typy organizacji*. System ten był testowany na zbiorze 156 dokumentów zawierających ponad 4000 jednostek referencyjnych. Proces ewaluacji był również bardziej rozwinięty, niż we wcześniejszych pracach – w szczególności odrębnie oceniano dopasowania dokładne oraz pokrywające się. Najwyższe wartości rozpoznania jednostek referencyjnych dla dopasowania częściowego zestawione są w tabeli 4.2.

Wyniki te nie różnią się znacznie od wyników przedstawionych w tabeli 4.1. Najistotniejsza różnica dotyczy wzrostu pokrycia w dziedzinie nazw geograficznych z 43% do 72%. Zauważalny jest również wzrost precyzji w rozpoznawaniu nazw organizacji z 88% do 91%. Z drugiej jednak strony system rzadziej rozpoznaje nazwiska (spadek pokrycia z 85% do 78%) oraz nazwy organizacji (spadek pokrycia z 57% do 52%). Należy jednak pamiętać, że celem systemu było przede wszystkim poprawne rozpoznawanie nazw geograficznych.

Wykorzystanie parsera Spejd

Nieco inne podejście do problemu rozpoznawania jednostek referencyjnych przedstawione jest w pracach Gralińskiego i współpracowników [48, 49, 157]. Zrezygnowano z użycia SProUT na rzecz parsera Spejd [20], ze względu na prostszy formalizm drugiego narzędzia. W pracach tych omówione jest zastosowanie mechanizmów rozpoznawania jednostek referencyjnych w kontekście tłumaczenia maszynowego, anonimizacji dokumentów oraz systemu odpowiadającego na pytania użytkowników.

Podstawowa różnica występująca w formalizmie prezentowanym przez Gralińskiego i współpracowników dotyczy niewystępowania w regułach parsera Spejd zmiennych. Zrezygnowano więc z uzgadniania wartości kategorii gramatycznych występujących w nazwie. Prezentowany system również jest oparty o reguły, przy czym ich lewe strony składają się z 6 elementów: lewego kontekstu zewnętrznego, lewego dopasowania, środkowego dopasowania, prawego dopasowania, prawego kontekstu oraz dowolnego dopasowanie w obrębie analizowanego zdania. Każdy z elementów reguły reprezentowany jest za pomocą wzorca gramatyki Spejda. W zależności od zastosowania algorytmu, po stronie akcji może występować np. przetłumaczenie danej nazwy (w systemie tłumaczenia maszynowego), zastąpienie jej ciągiem znaków uniemożliwiającym identyfikację (w systemie anonimizacji), bądź też uzupełnienie informacją o rozpoznanej kategorii semantycznej (w przypadku systemu odpowiadającego na pytania).

Przykładowe reguły prezentowane w pracach Gralińskiego i współpracowników obejmują takie elementy jak np. występowanie wyrażenia *pani prezes* przed nazwiskiem osoby, czy skrótowca *SA* po nazwie spółki giełdowej, a także wzorca pozwalającego na rozpoznanie aktów prawnych, obejmującego elementy takie jak *artykuł*, *ustęp* i *punkt* odpowiedniej ustawy.

Ewaluacja tego podejścia w kontekście tłumaczenia maszynowego oraz systemu odpowiadającego na pytania wskazała, że zastosowanie oddzielnego modułu odpowiedzialnego za rozpoznawanie jednostek referencyjnych przyczynia się istotnie do poprawy ich działania. W kontekście systemu anonimizacji,

który zrealizowany był w formie makr edytora tekstowego, uzyskano zaś bardzo dobre wyniki, których pokrycie było bliskie 100%[9].

Kategoryzacja nazw jednosegmentowych

Interesujące podejście do problemu kategoryzacji nazw własnych przedstawione jest w pracy Pietrasa [111]. W przeciwieństwie do innych algorytmów prezentowanych w tym zestawieniu, w podejściu tym zasadniczo nie wykorzystuje się informacji zawartej w kontekście wystąpienia określonej nazwy. Opisywany system bezkontekstowej ekstrakcji jednosegmentowych nazw własnych (SBEN), jako podstawę kategoryzacji przyjmuje morfologię nazwy. Rozpoznawanie kategorii ograniczone jest do nazw jednosegmentowych należących do jednego z następujących typów: *nazwa osobowa*, *nazwa miejscowa*, *nazwa geograficzna* i *nazwa instytucji*. Algorytm ten działa dwuetapowo: w pierwszym etapie określany jest paradygmat fleksyjny wyrażenia, natomiast w drugim etapie stosując reguły statystyczne, określana jest kategoria semantyczna nazwy.

Proces rozpoznawania paradygmatu nazwy realizowany jest w oparciu o dane zgromadzone w *Słowniku Fleksyjnym Języka Polskiego* [112]. W etapie poprzedzającym rozpoznawanie nazw, na podstawie treści słownika, określono sufiksy wyrazów charakterystyczne dla określonego paradygmatu fleksyjnego. Ponieważ dla wielu form fleksyjnych sufiks określonej formy jest pusty (tzn. wyraz na określonej pozycji fleksyjnej jest tożsamy z formą podstawową), wzięto również pod uwagę ostatni znak nie podlegający odmianie. Na podstawie sufiksów oraz ostatniego znaku nie należącego do sufiksu, możliwe jest przypisanie analizowanemu wyrazowi zbioru paradygmatów odmiany. Zbiór ten najczęściej nie jest jednoelementowy, ze względu na współdzielenie końcówek fleksyjnych przez różne paradygmaty, zatem rozpoznanie to nie jest jednoznaczne. Ograniczenie zbioru może nastąpić, jeśli w źródłowym tekście występuje również inna forma analizowanego wyrazu. Zjawisko to nie jest jednak zbyt częste.

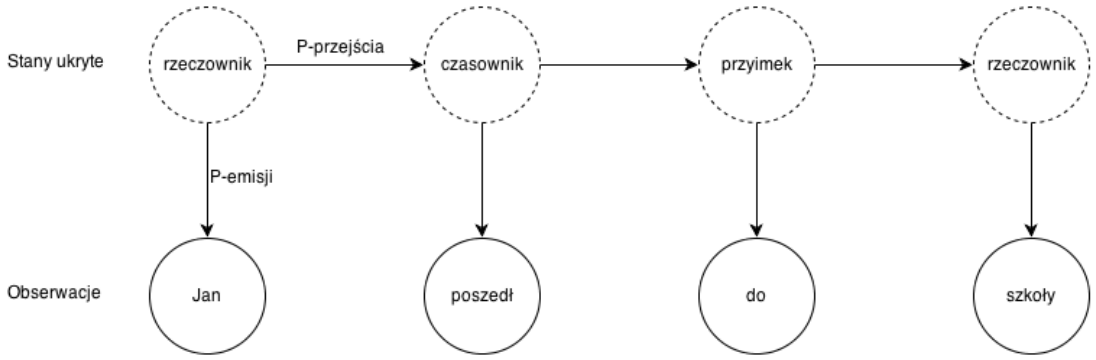
W drugim etapie kategoryzacji nazw następuje przypisanie kategorii semantycznej na podstawie dwóch typów reguł: strukturalno-fleksyjnych oraz strukturalnych. Reguły te mają podobną postać: lewą stronę stanowi sufiks nazwy, natomiast prawą kategoria semantyczna wraz z informacją o pewności określonej reguły. Ponadto reguły uzupełnione są o informacje statystyczne obejmujące selektywność oraz skuteczność reguły. Reguły strukturalno-fleksyjne posiadają dodatkowe ograniczenie określające konkretny paradygmat fleksyjny w ramach którego dana reguła obowiązuje. Reguły te są konstruowane w sposób automatyczny na podstawie zawartości *Bazy Fleksyjnej Nazw Własnych* [26].

Skuteczność algorytmu została zweryfikowana na podstawie wyników uzyskiwanych przez ludzi, w identycznym eksperymencie obejmującym określenie kategorii semantycznej dla ponad 100 nazw własnych. Na podstawie danych dostarczonych przez 67 wolontariuszy wykazano, że algorytm uzyskuje wyniki lepsze od przeciętnych wyników uzyskiwanych przez ludzi. W najbardziej restrykcyjnym wariancie uzyskiwał on precyzję na poziomie 67,8%, natomiast w najmniej restrykcyjnym 90,8%. Porównując ten algorytm z innymi algorytmami należy zwrócić uwagę, że nie wykorzystuje on żadnych informacji branych z kontekstu wystąpienia nazwy, a jedynie opiera się na analizie jej cech morfologicznych.

Metody indukcyjne

Zastosowanie metod indukcyjnych do rozpoznawania jednostek referencyjnych w języku polskim należy do bardzo niedawnych osiągnięć. Marcińczuk i współpracownicy w swoich badaniach z 2010 roku [77] przedstawiają zastosowanie ukrytych modeli Markowa (*Hidden Markov Models – HMM*) do rozwiązania tego problemu badawczego. HMM stosowane są w wielu zagadnieniach z dziedziny przetwarzania języka

[9]Precyzja reguł nie mogła być w pełni zbadana, ze względu na specyfikę przetwarzanych danych, do których autorzy nie mieli pełnego dostępu.



Rysunek 4.1: Przykładowy ukryty model Markowa służący do określania kategorii gramatycznej słów w zdaniu.

naturalnego, w szczególności od wielu lat stosuje się je w rozpoznawaniu mowy [58]. Działanie HMM opiera się na odzwierciedleniu pewnego zjawiska, dla którego domniemany model stanowy może być obserwowany wyłącznie pośrednio. Przykładowo w zadaniu polegającym na przypisaniu kategorii gramatycznych kolejnym słowom zdania zakłada się, że istnieje pewien ciąg stanów (*zjawisk ukrytych*) reprezentujących poszczególne kategorie gramatyczne na podstawie których emitowane są poszczególne słowa zdania (*zjawiska obserwowalne*). HMM określa prawdopodobieństwa przejścia pomiędzy poszczególnymi stanami oraz prawdopodobieństwo emisji określonego zjawiska obserwowalnego na podstawie zjawiska ukrytego. Przypisanie kategorii gramatycznych do słów określane jest na podstawie największego prawdopodobieństwa przejść przez stany ukryte oraz emisji obserwowanych słów. Parametry HMM mogą być określone na podstawie danych treningowych tak, aby maksymalizować prawdopodobieństwo obserwowanych danych, dlatego metodę tę zaliczama się do metod indukcyjnych. Przykładowy HMM przedstawiony jest na rysunku 4.1.

Modelem najczęściej wykorzystywanym w przetwarzaniu języka naturalnego jest łańcuch Markowa pierwszego rzędu, tj. taki HMM, w którym prawdopodobieństwo przejścia do stanu następnego określane jest wyłącznie na podstawie jednego stanu poprzedzającego. W kontekście rozpoznawania jednostek referencyjnych HMM stosuje się do określania kategorii semantycznej wyrażeń w zdaniu, na podstawie występujących w nim słów oraz ich cech.

Marcińczuk i współpracownicy w pracy [77] zastosowali HMM do rozpoznawania nazwisk osób oraz organizacji w dokumentach z dziedziny finansów oraz w raportach policyjnych. Z każdym z typów jednostki referencyjnej stowarzyszonych było 7 ukrytych stanów w modelu Markowa, ponadto wykorzystano 3 dodatkowe stany do oznaczenia początku i końca zdania oraz słowa, które nie było elementem jednostki referencyjnej. Prawdopodobieństwo emisji określonego słowa było określane na podstawie n-gramowego modelu znakowego. Ponieważ model taki jest podatny na problem niedoboru danych (ang. *data sparsity*), autorzy zastosowali wygładzanie metodą Wittena-Bella [159].

Podsumowanie wyników przeprowadzonych eksperymentów przedstawione jest w tabeli 4.3. Należy zwrócić uwagę, że zestawienie to zawiera jedynie maksymalne wartości uzyskiwane przez system dla różnych wariacji eksperymentu, toteż w praktyce określona para precyzja – pokrycie nie była osiągana przez system. Modele Markowa wykorzystane w eksperymencie uzyskane zostały na podstawie korpusu tekstów obejmującego raporty spółek giełdowych (CSER), a ich ewaluacja odbywała się zarówno na tym korpusie (w trybie 10-krotnej walidacji krzyżowej) oraz na korpusie zawierającym raporty policyjne. W stosunku do metod regułowych widać wyraźny spadek precyzji przy jednoczesnym wzroście pokrycia. Dla osób precyzja spadła z 91% do 87%, a dla organizacji z 92% do 77%, przy ewaluacji na tym samym

Tablica 4.3: Jakość wyników uzyskiwanych przez system oparty o HMM rozpoznający jednostki referencyjne w tekstach z dziedziny finansów (CSER) oraz w raportach policyjnych (CPR) – [77]. Zestawienie zawiera jedynie najlepsze wyniki uzyskiwane przez algorytm.

	Precyzja [%]		**Pokrycie** [%]	
Rodzaj wyrażenia	CSER	CPR	CSER	CPR
Nazwisko osoby	87,2	69,8	94,8	56,2
Nazwa organizacji	76,6	12,3	83,1	56,2

korpusie. Natomiast pokrycie dla osób wzrosło z 78-85% do ponad 94%, zaś dla organizacji z 52-56% do 83%. Należy jednak zwrócić uwagę, że wyniki te nie zostały utrzymane, jeśli ten sam model był wykorzystany w innej dziedzinie – spadek jakości uzyskiwanych wyników jest znaczny, w skrajnym przypadku tj. precyzji w rozpoznawaniu nazw organizacji sięgającej jedynie 12%, wyniku całkowicie nieprzydatnego z praktycznego punktu widzenia. Wyniki te wyraźnie wskazują, że prezentowana metoda indukcyjna jest mocno uzależniona od dostępnych danych treningowych, co istotnie ogranicza jej praktyczną stosowalność. Wynik ten nie jest jednak zaskakujący, jeśli odwołamy się do rozwoju metod ekstrakcji informacji w języku angielskim.

Rozwinięciem badań nad zastosowaniem modeli HMM do wykrywania nazw jest praca [78], w której zastosowano formalizm warunkowych pól losowych (ang. *Conditional Random Fields – CRF* [63]). Inspiracją dla tej pracy były wnioski wysunięte na podstawie analizy błędów popełnianych przez system oparty o HMM, w szczególności niemożność uwzględnienia prawego kontekstu przy dopasowywaniu nazwy. CRF, będący rozwinięciem HMM, nie posiada jednak tych istotnych ograniczeń. W szczególności modele zbudowane w oparciu o ten formalizm umożliwiają uwzględnienie prawego kontekstu, a także szerszej wiedzy językoznawczej na temat modelowanych zjawisk. W kontekście rozpoznawania nazw wiedza ta może być dostarczona np. w postaci listy imion i nazwisk osób oraz firm, a także słów, które pojawiają się przed oraz po tych wyrażeniach. Model CRF nie ogranicza się jednak tylko do tych danych, lecz jest wzbogacany na podstawie danych treningowych.

Marcińczuk i współpracownicy zidentyfikowali 34 cechy, które uwzględniane były w modelu. Cechy te podzielone zostały na pięć grup:

- *cechy ortograficzne*, takie jak np. forma odmieniona i forma podstawowa słowa, prefiks oraz sufiks słowa, czy wzorzec kapitalizacji liter,
- *binarne cechy ortograficzne*, takie jak np. występowanie wielkiej litery na początku słowa, występowanie wielkich liter na wszystkich pozycjach, itp.
- *cechy oparte o polski WordNet*, np. synonimy oraz hiperonimy danego słowa,
- *cechy morfologiczne*, takie jak kategoria gramatyczna, czy wartości kategorii gramatycznych,
- *cechy oparte o zbiory nazw własnych*, tzn. weryfikujące czy dane słowo występuje w określonym zbiorze nazw własnych.

Działanie algorytmu zostało przetestowane na danych z publikacji [77] oraz dodatkowym korpusie notatek finansowych (CEN). Ponadto korpus raportów giełdowych (CSER) został poddany pewnym modyfikacjom, związanym z wykrytymi w nim błędami. W tekstach wykrywane były imiona, nazwiska, nazwy państw, miast oraz ulic. Tablica 4.4 zawiera zestawienie wyników działania algorytmu dla poszczególnych korpusów, dla konfiguracji uzyskującej najlepsze wyniki miary F_1. Łatwo zauważyć, że precyzja

Tablica 4.4: Jakość wyników uzyskiwanych przez system oparty o CRF rozpoznający jednostki referencyjne w raportach spółek giełdowych (CSER), raportach policyjnych (CPR) oraz notatkach prasowych z dziedziny finansów (CEN) – [78]. Zestawienie zawiera jedynie najlepsze wyniki uzyskiwane przez algorytm.

	Precyzja [%]			**Pokrycie** [%]		
Rodzaj wyrażenia	CSER	CPR	CEN	CSER	CPR	CEN
Nazwa ulicy	96,7	100,0	71,4	95,1	50,0	16,1
Nazwisko	97,9	93,1	93,1	87,9	48,9	51,3
Imię	96,9	93,9	96,6	87,8	63,9	59,0
Nazwa państwa	89,7	100,0	91,2	82,7	81,5	70,9
Nazwa miasta	94,7	89,1	79,9	95,4	63,9	55,7

rozpoznawania nazw zarówno w korpusie o charakterze zgodnym z danymi treningowymi, jak i korpusie odmiennym jest bardzo wysoka i sięga w pierwszym przypadku powyżej 95% a w drugim osiąga wartość powyżej 90%. Istotnym mankamentem rozwiązania jest pokrycie uzyskiwanie dla tekstów o charakterze odmiennym niż dane uczące – waha się w zakresie od zaledwie 16% dla nazw ulic, do 81% dla nazw państw. W przypadku tekstów o podobnym charakterze osiąga wartości powyżej 80%, a w przypadku nazw ulic oraz nazw państw powyżej 95%. Uzyskane wyniki pokazują, że metoda oparta o CRF nadaje się najlepiej dla danych o charakterze jednolitym. Jej zastosowanie w różnych dziedzinach wiedzy prowadzi do istotnego pogorszenia wyników.

4.2.2. Ekstrakcja relacji semantycznych

Zagadnienie ekstrakcji relacji semantyczny dla języka polskiego nie doczekało się jeszcze tak bogatej literatury, jak to ma miejsce w przypadku języka angielskiego. Do najważniejszych prac w tym zakresie należy zaliczyć serię publikacji Piaseckiego i współpracowników [105, 107, 109], które zostały podsumowane w książce *A WordNet from the Ground Up* [108]. Celem tych badań było opracowanie algorytmów pozwalających na półautomatyczną rozbudowę polskiego WordNetu. Z tego względu opracowane algorytmy charakteryzują się dużą uniwersalnością – podobną do algorytmów opisanych w punkcie 4.1.4.

Drugim ważnym osiągnięciem w tym zakresie są publikacje Jaworskiego [56, 56], w których autor opisuje system ekstrakcji informacji z polskiego słownika bibliograficznego zdolny do rozpoznawania relacji semantycznych. W przeciwieństwie jednak do prac Piaseckiego i współpracowników, opisywany system jest zorientowany na wąską dziedzinę wiedzy, a jego adaptacja wymaga szczegółowej konstrukcji odpowiedniej gramatyki formalnej. W tym sensie system ten przypomina początkowe rozwiązania opracowane dla języka angielskiego.

Półautomatyczna rozbudowa polskiego WordNetu

Jako podstawę algorytmu zdolnego do rozpoznawania relacji semantycznych na potrzeby konstrukcji polskiego WordNetu, ze względu na uniwersalny charakter tego słownika, konieczne było opracowanie metody charakteryzującej się wysokim pokryciem, zdolnej do określenia semantycznego podobieństwa dwóch dowolnych symboli językowych.

Piasecki i współpracownicy opracowali w tym celu *miarę semantycznego powinowactwa symboli językowych* (ang. *Measure of Semantic Relatedness* – MSR) [105]. Ze względu na swobodny szyk charakterystyczny dla języka polskiego, miara ta nie była oparta wyłącznie na kolejności występowania słów, jak to

ma miejsce w języku angielskim. Co więcej, ze względu na brak dostatecznie uniwersalnego parsera języka polskiego, niemożliwe było również oparcie się na głębokiej analizie syntaktycznej. Z tego też powodu jako podstawę określania podobieństwa symboli językowych, Piasecki i współpracownicy opracowali metodę opartą o ograniczenia morfosyntaktyczne.

W celu określenia semantycznego powinowactwa symboli zgromadzono duży zbiór tekstów (w skład którego wchodził m.in. korpus IPI PAN [130]). Teksty w tym korpusie zostały oznakowane morfosyntaktycznie za pomocą tagger TaKIPI [110], a następnie wydobyto z nich pary symboli spełniające ograniczenia morfosyntaktyczne opisane w języku JOSKIPI [106]. Określono cztery rodzaje ograniczeń morfosyntaktycznych[10]:

1. `AdjC` – ograniczenie oparte o związek zgody rzeczownika z określonym przymiotnikiem lub imiesłowem przymiotnikowym,
2. `NcC` – ograniczenie oparte o współwystępowanie rzeczownika z innym rzeczownikiem,
3. `NmgC` – ograniczenie oparte o związek rządu z innym rzeczownikiem,
4. `VsbC` – ograniczenie oparte o związek zgody rzeczownika z określonym czasownikiem.

Przykładowo, na podstawie pierwszej reguły symbole *kot* oraz *chmura* byłyby do siebie podobne, ze względu na występowanie wyrażeń takich jak *czarny kot*, *czarnego kota*, *czarnemu kotu* oraz *czarna chmura*, *czarne chmury*, *czarnych chmur*, gdzie oba rzeczowniki łączą się za pomocą związku zgody z przymiotnikiem *czarny*. Oczywiście przykład ten ma charakter wyłącznie ilustracyjny, gdyż podobieństwo na podstawie tej i innych reguł określane jest z uwzględnieniem wszystkich symboli językowych, z którymi dany symbol wchodzi w odpowiednią relację.

Następnie na podstawie par symboli spełniających przedstawione ograniczenia tworzona była macierz koincydencji, określająca na ile istotna jest dana cecha w odniesieniu do innych cech. Macierz ta nie była jednak bezpośrednio wykorzystywana do określenia podobieństwa symboli, ponieważ tak określone cechy nie są zrównoważone. Przykładowo, jeśli jakieś dwa rzeczowniki są do siebie podobne, gdyż wielokrotnie współwystępują z przymiotnikiem *nowy*, to taka informacja nie powinna być tak samo cenna jak współwystępowanie z przymiotnikiem *dwukondygnacyjny*, gdyż ten drugi jest znacznie rzadszy. Z tego względu surowe wartości uzyskane na podstawie korpusu były zamieniane na rangi zgodnie z algorytmem opisanym w [19].

Tak skonstruowana miara powinowactwa semantycznego była najbardziej podobna do ogólnej relacji podobieństwa znaczeń. Jednakże relacjami podstawowymi dla WordNetu są synonimia i hiperonimia, dlatego też dalsze badania grupy Piaseckiego szły w kierunku opracowania algorytmów zdolnych do rozpoznawania właśnie tych relacji.

W tym celu przetestowano kilka opisanych w literaturze sposobów ekstrakcji relacji semantycznych. W pierwszym rzędzie zaadaptowano metodę opisaną przez Hearst (patrz p. 4.1.7) na potrzeby języka polskiego. Zmiany dotyczyły przede wszystkim sposobu konstrukcji wzorców zdolnych do ekstrakcji relacji hiperonimii – uwzględniały one opis nie tylko w terminach klas gramatycznych, jak to miało miejsce w pracy Hearst, ale również wartościach kategorii gramatycznych. Przykładowo, reguła zdolna rozpoznać relację hiperonimii w zdaniu *Pies jest ssakiem*, brała pod uwagę to, że *pies* występuje w mianowniku, a *ssak* w narzędniku oraz to, że oba rzeczowniki mają tę samą liczbę. Opracowano pięć reguł zaprojektowanych do rozpoznawania hiperonimii tego rodzaju [108, s. 103]:

1. `JestInst` – reguła łącząc podmiot z dopełnieniem w narzędniku np. ***Pies*** *jest* ***ssakiem***...,

[10] Ich nazwy pochodzą z pracy [108].

2. `NomToNom` – regułą łącząca dwa rzeczowniki za pomocą czasownika *to*, np. ***Pies** to **ssak**...*,
3. `IInne` – reguła wykorzystująca konstrukcję *i inne*, np. ***Pies** i inne **ssaki**...*,
4. `TakichJak` – reguła wykorzystująca konstrukcję *takich jak*, np. *Wśród **ssaków**, takich jak **psa**...*,
5. `WTym` – reguła wykorzystująca konstrukcję *w tym*, np. ***Ssaki** lądowe, w tym **pies**.*

Niestety okazało się, że pojedyncze reguły zastosowane do dużego korpusu tekstów zawierającego ponad 500 milionów segmentów osiągały precyzję w przedziale 10-30%. Dopiero dla par słów, które pasowały do więcej niż jednego wzorca, precyzja wzrastała do 41% (dla dwóch wzorców) oraz 74% (dla trzech wzorców), powodując jednak istotny spadek liczby znalezionych przykładów relacji hiperonimii [108, s. 105-108].

Ze względu na niezadowalające wyniki osiągnięte w oparciu o ręcznie konstruowane wzorce, dalsze badania zespołu szły w kierunku metod opartych o dane zarodkowe (porównaj p. 4.1.4). Naukowcy oparli się na systemie *Espresso* [102], lecz wprowadzili w nim modyfikacje specyficzne dla języka polskiego – tak opracowany system nazwali *Estratto* [62]. Wśród najważniejszych różnic występujących pomiędzy oboma systemami należy wymienić:

1. użycie innej miary służącej do określenia wiarygodności ekstrahowanych wzorców,
2. zastosowanie we wzorcach ekstrakcyjnych, obok kategorii gramatycznych również wartości kategorii gramatycznych, jako cech określających wzorzec,
3. weryfikację wiarygodności ekstrahowanych instancji relacji semantycznych w oparciu o korpus tekstowy, a nie wyszukiwarkę internetową.

Zespół Piaseckiego przetestował różne warianty algorytmów Espresso oraz Estratto. Najlepsza uzyskana precyzja wyników wynosiła między 39% a 59% (w zależności od przyjętej metody oceny). Ponieważ nie badano wprost pokrycia tych metod, można wskazać, że metoda posiadająca największą precyzję (względem ręcznej oceny) dawała ok. 2 razy mniej wyników, niż metoda gorsza od niej o 22 punkty procentowe. W ogólności jednak zmiany wprowadzone w algorytmie Estratto powodowały istotną poprawę jakości ekstrakcji relacji semantycznych. Naukowcy zidentyfikowali również dość istotny problem związany z oboma algorytmami, polegający na tym, że ich parametry, mające dość duży wpływ na jakość wyników, były mocno zależne od korpusu tekstów, poddawanego analizie. Dlatego też konieczne było każdorazowe zoptymalizowanie ich względem docelowego korpusu.

Jako ostatniej metody ekstrakcji relacji semantycznych na potrzeby polskiego WordNetu zespół Piaseckiego użył klasyfikatora opartego o uczenie maszynowe. Jako dane uczące wykorzystano pary rzeczowników połączone relacją hiperonimii w części polskiego WordNetu zbudowanej ręcznie. Wyszukując ich wystąpień w tekście można było określić wartości 17 cech [108, s. 134] służących do identyfikacji relacji hiperonimii. Chcąc znaleźć nowe instancje tej relacji, dla danego rzeczownika badano inne rzeczowniki, dla których miara powinowactwa semantycznego z wyjściowym rzeczownikiem była najwyższa. W ten sposób dokonywano odróżnienia w ramach niedookreślonej relacji podobieństwa semantycznego rzeczowników powiązanych oraz niepowiązanych relacją hiperonimii.

Wyniki uzyskane dla tej metody ekstrakcji informacji były dość dobre – dla najlepszego klasyfikatora uzyskiwano 80% poprawności zarówno w odniesieniu do precyzji jak i pokrycia [108, s. 139]. Jednakże wartości te były zbyt niskie, aby można było dokonywać całkowicie automatycznego rozbudowywania polskiego WordNetu. Ostatecznie zespół Piaseckiego opracował narzędzie WordNet Weaver, które łączyło w sobie wszystkie opisane wcześniej metody i w postaci graficznej przedstawiało proponowane

rozszerzenia istniejącej sieci relacji. To jednak użytkownik akceptował lub odrzucał wyniki dostarczane przez system. Według relacji językoznawców biorących udział w tym projekcie, program ten w istotny sposób przyczynił się zarówno do rozbudowy istniejącej sieci relacji oraz wykrycia istniejących błędów.

Ekstrakcja relacji ze słownika bibliograficznego

Inne ciekawe badanie w zakresie ekstrakcji informacji w języku polskim prowadzone były przez Jaworskiego [56, 57]. Badacz ten opracował system ekstrakcji informacji zbudowany z szeregu powiązanych modułów, w szczególności modułu opisującego gramatykę danego języka oraz modułu ontologicznego, służącego do interpretacji ekstrahowanych informacji. Istotnym założeniem przyświecającym twórcy systemu było ekstrahowanie informacji należących do określonej domeny. W związku z modularną budową systemu możliwe było łatwiejsze zaadaptowanie go do nowych dziedzin – autor stworzył dwie jego wersje: jedną ekstrahująca informacje ze słownika bibliograficznego oraz drugą ekstrahującą informacje ze zbioru dokumentów opisujących transakcje handlowe w języku sumeryjskim [56].

Ekstrakcja informacji w systemie Jaworskiego podzielona była na następujące etapy:

1. analizę leksykalną,
2. analizę syntaktyczną,
3. analizę semantyczną.

Analiza leksykalna odbywała się w oparciu o słownik regułowy – wyrażenia, które miały być rozpoznane przez system musiały posiadać ręcznie przypisaną do nich kategorię gramatyczną, a także, zaczerpniętą ze słownika Morfeusz [163] informację dotyczącą morfologii danego wyrażenia. Konstrukcja słownika leksykalnego polegała zatem na opisaniu wszystkich wyrażeń interesujących z punktu widzenia ekstrakcji informacji oraz ujednoznacznieniu opisu morfologicznego.

Analiza syntaktyczna odbywała się zaś w oparciu o ręcznie skonstruowane reguły gramatyczne oraz semantyczne. Każda reguła syntaktyczna określała kategorie gramatyczne oraz wartości kategorii gramatycznych, które muszą posiadać elementy danego wyrażenia złożonego, aby dana reguła mogła zostać zastosowana. Podobnie, reguły semantyczne określały wartości kategorii semantycznych składników wyrażenia niezbędne do tego, aby dana reguła semantyczna mogła być zastosowana oraz określały wartość wynikowego wyrażenia w *języku reprezentacji znaczenia*. Reguły semantyczne były zwykle proste i ograniczały się do określenia wartości kategorii semantycznych składników wyrażenia, natomiast reguły gramatyczne, poza określeniem kategorii gramatycznych, często zawierały szczegółowe wymagania dotyczące wartości oraz odpowiedniości wartości kategorii gramatycznych składników wyrażenia.

Aby mogła nastąpić ekstrakcja informacji, w obu zbiorach musiały występować reguły, które pozwalały na wyprowadzenie całego zdania z odpowiednich symboli początkowych. Parsowanie odbywało się z wykorzystaniem algorytmu Earleya [38], po wcześniejszym dostosowaniu odpowiednich zestawów reguł do wymagań gramatyki bezkontekstowej.

Istotnym składnikiem systemu był *język reprezentacji znaczenia* – niezależny od języka dokumentów język deskryptywny, zbudowany w oparciu o wykorzystywaną ontologię. Formułami atomowymi tego języka były predykaty, które odpowiadały kategoriom ontologicznym zdefiniowanym w ontologii. Pierwszym argumentem predykatów była zawsze stała reprezentująca obiekt, którego kategoria ontologiczna była określana przez predykat. Pozostałe argumenty odnosiły się do obiektów, bądź własności służących do opisu argumentu stojącego na pierwszej pozycji. Formuły atomowe można było łączyć za pomocą operatorów koniunkcji i alternatywy (wykorzystywanej do reprezentacji wieloznaczności występującej w dokumencie źródłowym). Należy zwrócić uwagę, że w języku tym nie występowały kwantyfikatory, negacja ani implikacja. Tym samym w języku tym można było opisywać jedynie pozytywne i „statyczne”

(tzn. nie będące regułami) fakty dotyczące rzeczywistości. Przykładowo wyrażenie

$$\texttt{Person}(p, j) \wedge \texttt{Job}(j) \tag{4.3}$$

oznacza, że stała p reprezentuje osobę powiązaną z obiektem j, który jest pracą (zatrudnieniem). Innymi słowy możemy powiedzieć, że p wykonuje, bądź wykonywał pracę j albo posiadał zatrudnienie j. Przekształcenie wyrażeń języka naturalnego w wyrażenia języka reprezentacji znaczenia, odbywało się poprzez transformacje (wyrażone w rachunku λ [25]) stowarzyszone z odpowiednimi regułami gramatycznymi.

System skonstruowany w oparciu o ponad 120 tys. reguł leksykalnych, 66 reguł syntaktycznych oraz 124 reguły semantyczne, został oceniony przez autora pod względem precyzji i pokrycia, które wyniosły odpowiednio 100% oraz 45% [57, s. 39]. Ten wynik jest bardzo dobry, niestety autor nie podał informacji dotyczących wykorzystanego zbioru testowego, zasad jego opracowania oraz metodologii oceny, dlatego należy podchodzić do niego z pewną ostrożnością.

Należy również zwrócić uwagę, że struktura oraz sposób konstrukcji systemu przypominają systemy dla języka angielskiego z początku lat dziewięćdziesiątych – były to systemy dostosowane do konkretnej domeny, a ich adaptacja do nowej dziedziny wymagała istotnych nakładów finansowych. Z tego powodu rozwiązanie to wymaga bardzo dużego nakładu pracy ręcznej, co widać w cytowanym artykule, gdzie budowa systemu wymagała konstrukcji ponad 120 tys. (*sic!*) reguł leksykalnych. Co więcej, chociaż system ten pozwalał reprezentować wieloznaczność strukturalną, nie posiadał on żadnego modułu odpowiedzialnego za ujednoznacznianie wyrażeń. Należy zatem założyć, że jego autor przyjął, że każde wyrażenie posiada zaledwie jedno znaczenie. Nie zmienia to jednak faktu, że opisywany system jest jednym z niewielu kompletnych rozwiązań pozwalających na ekstrakcję informacji z tekstów w języku polskim.

Podsumowanie

Przedstawiony przegląd literatury na temat badań nad ekstrakcją informacji w języku polskim pokazuje, że w niektórych obszarach są one daleko posunięte – w szczególności w zakresie obejmującym rozpoznawanie jednostek referencyjnych. Udostępnienie w ramach Narodowego Korpusu Języka Polskiego [141, 140] referencyjnego zbioru tekstów wraz z anotacjami jednostek referencyjnych pozwala na porównanie istniejących systemów oraz przyczynia się do postępu badań w tym zakresie.

Z drugiej jednak strony, inne problemy z zakresu ekstrakcji informacji nie doczekały się jeszcze takiego zainteresowania ze strony badaczy. Wynika to prawdopodobnie z niedostępności oraz niedoskonałości niektórych zasobów oraz narzędzi, specyficznych dla poszczególnych języków, takich jak słowniki semantyczne oraz parsery. Przekłada się to na znacznie mniejszą liczbę badań dotyczących np. ekstrakcji relacji. Natomiast te badania, które były dotychczas prowadzone w tym zakresie, skoncentrowane były bądź to na ulepszaniu zasobów leksykalnych, bądź też posiłkowały się zasobami cząstkowymi, zbudowanymi przez samych badaczy na potrzeby konkretnego, wąsko określonego zadania z obszaru ekstrakcji informacji (porównaj prace Piskorskiego [113], Mykowieckiej [93] oraz Jaworskiego [57]).

Cechą charakterystyczną badań prowadzonych w obrębie języka polskiego, na tle badań prowadzonych dla innych języków, w szczególności angielskiego, jest niewykorzystanie semi-strukturalnych źródeł wiedzy, których zastosowanie istotnie przyczynia się do ulepszenia algorytmów ekstrakcji informacji (porównaj np. [99]). Język polski nie doczekał się również wysoce uniwersalnych narzędzi w rodzaju *ReVerb* [40], głównie ze względu na niedojrzałość istniejących parserów języka polskiego.

Rozwój systemów ekstrakcji informacji, zarówno dla języka polskiego, jak i innych języków, nie osiągnął jeszcze swojego kresu. Jednakże niewielka ilość rozwiązań tego rodzaju dostępnych dla języka polskiego oraz względny brak uniwersalności sprawiają, że badania nad ekstrakcją informacji w języku polskim, w szczególności zaś rozpoznawaniem relacji semantycznych, są szczególnie interesujące.

5. Szkic algorytmu ekstrakcji relacji semantycznych

W niniejszym rozdziale omówione zostały: cel algorytmu ekstrakcji relacji semantycznych, ogólna struktura tego algorytmu, kilka algorytmów pomocniczych umożliwiających realizację tego zadania oraz zasoby wiedzy, niezbędne do działania głównego algorytmu – zgodnie z koncepcją od ogółu do szczegółu. W kolejnych rozdziałach te same zagadnienia omówione są bardziej szczegółowo, dzięki czemu możliwe jest zrozumienie pełnego kontekstu działania poszczególnych algorytmów.

5.1. Cel algorytmu

Celem algorytmu ekstrakcji relacji semantycznych jest rozpoznanie wystąpień zadanej relacji semantycznej oraz określenie jej argumentów. Przykładowo: jeśli poszukujemy wystąpień relacji *całość – część* i w analizowanym tekście pojawi się zdanie[1]

> Ponad 10 tys. antylop uciekło z wyjątkowo silnie zaśnieżonych *stepów* ***Mongolii*** i przedostało się w poszukiwaniu jedzenia do wschodniej Syberii[2],

to algorytm powinien stwierdzić, że relacja ta występuje pomiędzy wyrażeniami: *Mongolii* oraz *stepów*, oraz określić, że pierwsze wyrażenie odnosi się do *całości* a drugie do *części*, gdyż wymienione w tekście stepy leżą na terenie Mongolii, zatem są jej częścią.

Należy zwrócić uwagę, że celem algorytmu nie jest jedynie stwierdzenie występowanie określonej relacji pomiędzy określonymi symbolami językowymi, ale stwierdzenie tego dla każdego analizowanego zdania. Innymi słowy zadanie, które ma on rozwiązać nie polega wyłącznie na zbudowaniu słownika semantycznego, czy bazy wiedzy o zależnościach pomiędzy symbolami (porównaj p. 3.3), ale każdorazowe wykrycie relacji, które występują w analizowanych zdaniach. Próbując osiągnąć pierwszy cel, jakkolwiek ambitny, można zignorować sporadyczne występowania relacji pomiędzy określonymi pojęciami. W drugim jednak przypadku, każde wystąpienie relacji musi zostać rozpoznane. Dlatego też, gdy punktem odniesienia pierwszego sposobu ekstrakcji relacji może być wiedza o świecie, tak punktem odniesienia drugiego sposobu jest wiedza wyrażona w każdym zdaniu z osobna. Tak postawiony cel pozwala bowiem na skonstruowanie algorytmu dokonującego faktycznej interpretacji tekstu, a nie jedynie ograniczającego się do gromadzenia wiedzy wydobytej z wielu tekstów.

Należy zwrócić uwagę, że w literaturze przedmiotu termin *ekstrakcja relacji* może być rozumiany odmiennie. Piasecki i współpracownicy [108] opisując algorytm automatycznego rozbudowywania polskiego WordNetu, w rozdziale o tytule *Extracting Instances of Semantic Relations* (ekstrakcja instancji relacji semantycznych) opisują dwa algorytmy – jeden zbudowany w oparciu o ręcznie utworzone wzorce

[1] Przykład pochodzi z korpusu PAP.

[2] W przykładach dotyczących relacji *całość-część* przyjęto konwencję, zgodnie z którą *całość* pisana jest pismem pochyłym, pogrubionym, a *część* – pismem pochyłym.

morfosyntaktyczne, drugi zaś oparty o metody statystyczne (więcej szczegółów na temat tych algorytmów znajduje się w p. 4.2.2). O ile jednak pierwsza metoda pozwala na określenie relacji w konkretnym zdaniu – gdyż wystąpienie relacji jest utożsamiane z dopasowaniem wzorca – o tyle w drugim przypadku jej wystąpienie stwierdzane jest dla pary wyrażeń, o ile można potwierdzić je statystycznie na podstawie pewnej grupy zdań. Zatem w przekonaniu autorów w obu przypadkach mamy do czynienia z tym samym zadaniem. Niemniej jednak algorytm statystyczny, który znajdzie wystąpienia par wyrażeń: *płetwa* i *rekin* może uznać, że płetwa jest częścią rekina. I choć w kontekście budowania bazy wiedzy wynik ten należy uznać za słuszny, to następujące zdanie nie zawiera takiej informacji, pomimo bliskości wystąpienia wyrażeń *płetwa* i *rekin*[3]:

> Dziewczyna tylko w kostiumie kąpielowym i *płetwach*, w otoczeniu ***rekinów***, z których jeden groźnie otwiera paszczę – takie plakaty zapraszają na objazdowe widowisko.

W przytoczonym przykładzie *płetwy* można potraktować jako część lub wyposażenie *dziewczyny*, choć w ogólnej bazie wiedzy na temat świata umieszczenie takiej informacji nie byłoby uzasadnione. Podobnie – chociaż płetwy są częścią ciała rekina, relacja ta nie znajduje potwierdzenia w przytoczonym zdaniu.

Inaczej sprawę przedstawia Cimiano [28], w kontekście problemu jakim jest tworzenie ontologii na podstawie analizy tekstów w językach naturalnych. Autor rozróżnia dwa zadania: budowy ontologii (*ontology learning task*) oraz jej wypełnienia (*ontology population task*). Celem pierwszego zadania jest uchwycenie pojęć ogólnych, jakie mogą występować w ontologii, zależności jakie między nimi występują, schematów rozumowania z wykorzystaniem tych pojęć, itp. Celem drugiego zadania jest odnajdywanie instancji zdefiniowanych pojęć oraz konkretnych relacji, które występują pomiędzy instancjami pojęć. Opisując to drugie zadanie [28, s. 26], autor wprowadza termin *knowledge markup* (znakowanie wiedzy), który może być zastosowany jeśli spełnione są dwa warunki: (i) zachowane jest odniesienie do tekstu, w którym znaleziono instancje pojęć oraz (ii) zachowany jest kontekst, w którym przypisano określone pojęcie lub relację do ich instancji. Tym samym autor wyraźnie rozgranicza dwa możliwe zastosowania terminu *ekstrakcja relacji*: jedno, w którym analiza tekstu prowadzi do określenia relacji na poziomie ontologicznym, czego skutkiem jest określenie ograniczeń semantycznych dla różnych typów relacji[4] oraz drugie, w którym analiza ta prowadzi do uchwycenia konkretnych relacji, spełniających zadane ograniczenia semantyczne[5].

W tym kontekście zadanie stawiane przed opisywanym algorytmem odpowiada temu, co Cimiano nazywa *knowledge markup*, to znaczy każdorazowo wymaga od algorytmu pozostawienia odniesienia do tekstu, w którym relacja została rozpoznana. Różnica polega jednak na tym, że algorytm nie będzie różnicował czy rozpoznana relacja ma charakter ontologiczny czy też odnosi się do instancji jakiejś ogólnej relacji. Rozróżnienie to nie zawsze jest bowiem łatwe do określenia – niewątpliwie można przyjąć, że relacja występująca pomiędzy *Mongolią* a jej *stepami* jest instancją pewnej relacji ogólnej. Niemniej jednak trudno określić, czy relacja występując między pojęciami *państwo* i *step* jest specjalizacją czy może jednak instancją relacji *całość-część*. Jeśli przyjąć pierwsze rozstrzygnięcie, to należałoby się spodziewać, że sama posiada instancje, w których oba argumenty są instancjami tych pojęć. Z drugiej strony przyjęcie rozstrzygnięcia przeciwnego ma tę negatywną konsekwencję, że występują w niej pojęcia ogólne, które mogą być potraktowane raczej jako jej ograniczenia semantyczne, a nie konkretne argumenty. Niewątpliwie rozwiązanie tego problemu silnie zależy od kontekstu, w którym tak pozyskana wiedza miałaby być stosowana. Dlatego też nie będziemy rozstrzygać tego rodzaju spornych przypadków, zakładając je-

[3] Przykład pochodzi z korpusu Instytutu Podstaw Informatyki Polskiej Akademii Nauk, `http://korpus.pl`

[4] Np. dla relacji *reżyserowanie*, takimi ograniczeniami mogłyby być odpowiednio pojęcia *reżyser* oraz *film* lub *sztuka teatralna*.

[5] Np. dla tej samej relacji można by stwierdzić jej wystąpienie pomiędzy wyrażeniami *Andrzej Wajda* oraz *film Pan Tadeusz*

dynie pewną niewielką liczbę dosyć ogólnych relacji semantycznych, dla których poszukiwać będziemy **wystąpień** w tekstach.

Tak postawiony cel algorytmu ekstrakcji wymaga jeszcze doprecyzowania. W literaturze przedmiotu zazwyczaj odróżnia się sytuacje, w których algorytm ekstrakcji konstruowany jest dla konkretnej dziedziny wiedzy, np. danych na temat przebiegu choroby [93], bądź danych dotyczących biografii pisarzy [57] od sytuacji, w których algorytm ten ma działać dla dowolnej dziedziny wiedzy [92, s. 2-3]. Celem opisywanego algorytmu jest ekstrahowanie informacji **nieograniczonych dziedzinowo**. Należy przez to rozumieć algorytm, który nie jest dostosowany do konkretnej dziedziny wiedzy, co niesie za sobą dwie konsekwencje. Po pierwsze, ekstrahowane informacje nie będą obejmowały wiedzy specjalistycznej, to znaczy takiej, która zrozumiała jest wyłącznie dla ekspertów z określonej dziedziny wiedzy, np. szczegółowych danych na temat przebiegu choroby. Po drugie, ekstrahowane informacje będę obejmowały wszystko to co może zrozumieć zwykły użytkownik języka polskiego, posiadający podstawową wiedzę na temat świata (obejmującą np. podstawowe zasady fizyki, sposoby organizacji współczesnych państw, powszechnie znanych informacji na temat kultury popularnej, itp.), pozwalającą na zrozumienie informacji dostępnych w niespecjalistycznych książkach, dziennikach, czy popularnych portalach internetowych. W konsekwencji skuteczność algorytmu będzie testowana na danych obejmujących notatki Polskiej Agencji Prasowej (szczegółowy opis korpusu znajduje się w punkcie 6.1.2), gdyż z jednej strony, do ich zrozumienia nie jest wymagana wiedza specjalistyczna, a z drugiej strony, zawarte w nich informacje nie ograniczają się do wąskiej dziedziny wiedzy.

5.2. Struktura głównego algorytmu

Tak postawiony cel algorytmu ekstrakcji osiągany jest za pomocą automatycznej konstrukcji wzorców ekstrakcyjnych. Szablon ekstrakcyjny jest traktowany jako zbiór cech morfologicznych, syntaktycznych oraz semantycznych, które muszą zostać spełnione, aby można było uznać, że poszukiwana relacja występuje w analizowanym tekście. Dopasowanie wzorca ekstrakcyjnego interpretowane jest jako wystąpienie relacji semantycznej, przez co, zgodnie z postawionym celem, możliwe jest określenie wystąpienia poszukiwanej relacji dla każdego zdania z osobna, podobnie jak ma to miejsce w przypadku ręcznie konstruowanych wzorców morfosyntaktycznych opisywane przez Piaseckiego i współpracowników [108].

Algorytm konstrukcji wzorców ekstrakcyjnych jest następujący:

1. Wybierana jest relacja r, której wystąpienia mają być rozpoznawane.
2. Dla relacji r generowane są pary symboli: (σ_a, σ_b), połączone tą relacją.
3. Dla każdej pary symboli (σ_a, σ_b) w korpusie tekstów wyszukiwane są zdania z, w których współwystępują napisy powiązane z tymi symbolami za pomocą funkcji *strings* (patrz wzór 3.4).
4. Powstały zbiór zdań jest filtrowany i usuwane są zdania niespełniające kryteriów poprawności.
5. Zdania należące do wynikowego zbioru poddawana są ograniczonej analizie morfosyntaktycznej, na podstawie której tworzone są wzorce ekstrakcyjne posiadające wyłącznie cechy morfologiczne i syntaktyczne, tzw. formalne wzorce relacji: fp.
6. Identyczne wzorce są utożsamiane dzięki czemu powstają uogólnione wzorce formalne ufp.
7. Korpus tekstów, w którym ma być przeprowadzona ekstrakcja informacji, ujednoznaczniany jest względem słownika semantycznego $\boldsymbol{D}$.

8. Dla każdego unikalnego wzorca formalnego ufp w korpusie z punktu 7 wyszukiwane są zdania z' pasujące do tego wzorca, przy założeniu, że oba dopasowane argumenty zostały ujednoznacznione, a odpowiadające im symbole mają przypisane zbiory kategorii semantycznych ($\boldsymbol{SC_a}, \boldsymbol{SC_b}$).

9. Ze zbioru z' losowana jest próbka zdań, które są ręcznie oznaczana jako zawierające, bądź niezawierające wystąpienie relacji r. Na tej podstawie określane są ręczne ograniczenia semantyczne wzorca ekstrakcyjnego (mc_a, mc_b).

9'. Określenie ograniczeń semantycznych realizowane jest automatycznie, w oparciu o statystyczną analizę relacji wstępujących w ontologii lub semantycznej bazie wiedzy, w wyniku czego powstają ograniczenia (ac_a, ac_b).

10. Wynikowe wzorce ekstrakcyjne powstają w wyniku połączenia wzorców formalnych ufp z ograniczeniami semantycznymi (mc_a, mc_b) (wariant pół-automatyczny) bądź (ac_a, ac_b) (wariant automatyczny).

Uzyskane w ten sposób wzorce ekstrakcyjne wykorzystywane są do rozpoznawania wystąpień zadanej relacji r w tekstach języka polskiego.

Zasadnicza koncepcja algorytmu podobna jest do rozwiązań przedstawionych przez Girju [47, 46], w problemie ekstrakcji relacji *całość-część* z tekstów w języku angielskim oraz podejścia wykorzystanego przez Piaseckiego i współpracowników [108] zastosowanego w algorytmie Estratto. To co przede wszystkim różni prezentowane podejście, to zdolność algorytmu do rozpoznawania nazw własnych, dzięki czemu możliwe jest ekstrahowanie specyficznych informacji oraz możliwość automatycznego określenia ograniczeń semantycznych relacji (patrz p. 9' algorytmu). Ponadto w algorytmie zastosowano ujednoznacznianie wyrażeń względem słownika semantycznego zbudowanego na bazie Wikipedii oraz nowatorskie podejście wyboru przykładowych zdań, służących do określenia wzorców formalnych wykorzystywanych przez wzorce ekstrakcyjne. Najistotniejsza część tezy niniejszej rozprawy zostanie obroniona, jeśli wzorce ekstrakcyjne zbudowane w oparciu o punkt 9' będą dawały rezultaty co najmniej tak samo dobre jak te, zbudowane w oparciu o punkt 9.

5.3. Algorytmy pomocnicze

Zastosowanie głównego algorytmu ekstrakcji informacji w postaci przedstawionej w punkcie 5.2 mogłoby dać dość dobre wyniki, na co wskazują prace Girju [47], gdyby został on zastosowany do jednej, dobrze określonej dziedziny wiedzy. Ambicją algorytmu jest jednak ekstrahowanie danych nieograniczonych dziedzinowo, dlatego problemy związane z przetwarzaniem tekstów, omówione w punkcie 2.3, muszą zostać wzięte pod uwagę. W tym celu opracowano kilka algorytmów pomocniczych, które wspierają algorytm główny w realizacji postawionego przed nim zadania. Szczegółowe omówienie tych algorytmów znajduje się w rozdziale 7.

5.3.1. Wybór przykładowych zdań

Punkt 2 głównego algorytmu zakłada dostępność zbioru przykładowych par symboli, o których wiadomo, że pomiędzy nimi występuje zadana relacja semantyczna. Algorytmy ekstrakcji informacji posiłkujące się danymi zarodkowymi (patrz p. 4.1.4), zwykle wykorzystują bardzo niewielką liczbę takich przykładowych danych, aby w sposób iteracyjny odkrywać nowe wzorce ekstrakcyjne oraz nowe pary symboli połączone odpowiednią relacją. Tym niemniej wykonywanie wielu iteracji prowadzi zwykle do zjawiska

nazywanego dryfem semantycznym [69], polegającego na tym, że jakość ekstrahowanych wzorców oraz par symboli istotnie się pogarsza z każdą iteracją.

Chcąc uniknąć tego problemu można zastosować różne metody statystyczne oraz semantyczne [69], pozwalające na rozpoznawanie par wyrażeń, które wprowadzają szum w danych. W prezentowanym algorytmie zastosowano nieco inne podejście – podstawowy pomysł polega na automatycznym rozszerzeniu pierwotnej listy par symboli, zanim algorytm zacznie poszukiwać odpowiadające im wzorce ekstrakcyjne. Możliwe jest to dzięki wykorzystaniu rozbudowanej taksonomii pojęć zbudowanej w oparciu o ontologię (patrz p. 5.4.4) oraz słownika semantycznego zawierającego dużą liczbę nazw własnych (patrz p. 5.4.3). Szczegółowy opis algorytmu wyboru przykładów znajduje się w punkcie 7.1 oraz w publikacji [122].

5.3.2. Ujednoznacznianie sensu wyrażeń

W punkcie 7 algorytmu następuję ujednoznacznienie wyrażeń, które podlegają ekstrakcji względem słownika semantycznego. Ujednoznacznianie sensu jest zagadnieniem dość złożony i jego skuteczność w dużej mierze uzależniona jest od wykorzystywanego słownika. Jeden z popularnych algorytmów wykorzystywany do ujednoznaczniania sensu względem angielskiego WordNetu – algorytm Leska [68, 9] – zakłada, że każdy symbol językowy wyposażony jest w tradycyjną definicję. Założenie to spełnione jest dla angielskiego WordNetu, ale nie jest spełnione dla polskiego WordNetu [108], gdyż opisy symboli językowych w polskim WordNecie mają wyłącznie charakter relacyjny[6].

Chcąc rozwiązać problem ujednoznaczniania sensu w sposób zadowalający, konieczne było wykorzystanie innego zasobu oraz algorytmu posługującego się innym słownikiem. Autor oparł się na wynikach Milnego oraz Wittena [87, 86], którzy wykorzystali Wikipedię jako podstawowy zasób leksykalny. Opracowali oni również algorytm zdolny do ujednoznaczniania wyrażeń względem artykułów Wikipedii. Choć został on opracowany dla języka angielskiego, pozwalał na łatwą adaptację dla innych języków – w tym języka polskiego.

Algorytm opracowany przez Milnego i Wittena został jednak poddany istotnym udoskonaleniom. W szczególności autor wykorzystał inną miarę pokrewieństwa semantycznego oraz zastosował dodatkowe cechy służące do budowy klasyfikatora, co istotnie przyczyniło się do poprawy rezultatów ujednoznaczniania. Szczegółowy opis tego algorytmu znajduje się w punkcie 7.3 oraz w publikacji [123].

5.3.3. Określenie kategorii semantycznych symboli językowych

W punkcie 8 głównego algorytmu występuje założenie, że dla obu argumentów relacji określone są ich kategorie semantyczne. To założenie jest zgodne z założeniem z punktu 3.1.1, w którym przyjęto, że algorytm ekstrahuje relacje tylko pomiędzy symbolami językowymi, dla których dostępny jest opis w wybranym słowniku semantycznym. Koncepcja słownika semantycznego przedstawiona jest w punkcie 5.4.3. Ponieważ słownik ten jest konstruowany automatycznie, jednym z ważniejszych problemów, które muszą zostać rozwiązane jest automatyczne określenie kategorii semantycznych dla symboli zgromadzonych w słowniku.

Problem ten jest rozwiązywany za pomocą algorytmu klasyfikacji symboli językowych. Jest ona realizowana na podstawie kilku źródeł wiedzy, dzięki czemu możliwe jest uzyskanie wysokiego pokrycia, przy zachowaniu wysokiej precyzji wyników. Punktem odniesienia klasyfikacji jest ontologia Cyc (patrz p. 5.4.4), a elementami podlegającymi klasyfikacji są artykuły Wikipedii (patrz p. 5.4.3). W celu zaklasyfikowania artykułów, algorytm posiłkuje się informacjami o typie *infoboksów* występujących w artykułach (patrz

[6] W ostatnim czasie część symboli w polskim WordNecie również została zaopatrzona w tradycyjne definicje.

p. 4.1.5), pierwszymi zdaniami traktowanymi jak definicje opisywanych obiektów (porównaj [23, 45]), systemem kategorii Wikipedii (porównaj [149]) oraz bezpośrednim mapowaniem pomiędzy artykułami Wikipedii a ontologią Cyc (porównaj [138]). Klasyfikacje uzyskiwane tymi metodami są następnie uzgadniane z wykorzystaniem wewnętrznego mechanizmu wykrywania sprzeczności ontologii Cyc. Szczegółowy opis tego algorytmu przedstawiony jest w punkcie 7.2 oraz w publikacji [118].

5.3.4. Automatyczne określanie ograniczeń semantycznych relacji

Punkt 9 algorytmu zakłada, że wyrażenia dopasowane do wzorca formalnego zostaną przeanalizowane pod kątem występowania w nich zadanej relacji r oraz oznaczone jako zawierające, bądź niezawierające tę relację. W punkcie 9' zaproponowano alternatywny sposób określania ograniczeń semantycznych. W pierwszym rzędzie można pozyskać je z ontologii (porównaj p. 5.4.4) – kompletna ontologia powinna zawierać informacje o ograniczeniach semantycznych predykatów stosowanych do reprezentowania faktów. Zazwyczaj ontologie wykorzystują ograniczenia tego rodzaju, gdyż pozwalają one na stosunkowo łatwą kontrolę poprawności wprowadzanych danych, a także pozwalają na bardziej efektywne wnioskowanie na temat zgromadzonych faktów. Z drugiej jednak strony, ograniczenia te mogą być dość ogólne, ponieważ najczęściej nie są one projektowane pod kątem odróżniania poszczególnych relacji zdefiniowanych w danej ontologii.

Dlatego algorytm zakłada inny sposób pozyskania tych ograniczeń – korzystając z bazy wiedzy zawierającej znaczną ilość faktów można podjąć próbę automatycznego określenia szczegółowych ograniczeń semantycznych, dzięki czemu będą one bardziej precyzyjne, co powinno skutkować wyższą precyzją ekstrakcji. Co więcej – ponieważ baza wiedzy zawiera zwykle przykłady wielu różnych relacji, możliwe jest połączenie wzorców formalnych z różnymi ograniczeniami semantycznymi, a tym samym opracowanie wzorców ekstrakcyjnych dla wielu relacji jednocześnie. W tym celu wykorzystywana jest semantyczna baza wiedzy DBpedia. Jest ona opisana w punkcie 5.4.5, zaś algorytm automatycznego określania ograniczeń omówiony jest w punkcie 7.4.

5.4. Wykorzystywane źródła wiedzy

Opisany algorytm wymaga dostępności szeregu źródeł wiedzy, bez których jego realizacja byłaby niemożliwa. W niniejszym punkcie przedstawiona została jedynie ich krótka charakterystyka. Pełne ich omówienie znajduje się w rozdziale 6.

5.4.1. Korpusy tekstów

W pierwszym rzędzie zakładana jest dostępność *korpusu tekstów*, na podstawie którego tworzone są wzorce formalne pozwalające na ekstrakcję wybranej relacji. Odwołując się do terminologii wprowadzonej w rozdziale 3, korpus taki jest zbiorem *napisów* stanowiących odwołanie do odpowiednich *symboli językowych*. Jest on zatem najbardziej pierwotnym źródłem danych językowych. Z tego powodu korpus taki stanowi zarówno podstawę dla algorytmu tworzącego wzorce ekstrakcyjne oraz jest on zasobem pozwalającym ocenić jego skuteczność.

Przy konstrukcji algorytmu wykorzystywane są dwa korpusy: pierwszy, udostępniony przez Instytut Podstaw Informatyki Polskiej Akademii Nauk (w skrócie korpus IPI PAN), na podstawie którego budowane są wzorce ekstrakcyjne oraz drugi zgromadzony w Grupie Lingwistyki Komputerowej w Akademii

Górniczo-Hutniczej składających się z notatek Polskiej Agencji Prasowej (w skrócie korpus PAP), wykorzystywany do weryfikacji skuteczności algorytmu.

5.4.2. Słowniki fleksyjne

Analiza morfologiczna jest jednym z istotniejszych etapów algorytmu, gdyż cechy morfologiczne, obok cech syntaktycznych oraz semantycznych wchodzą w skład konstruowanych wzorców ekstrakcyjnych. Ze względu na fleksję języka polskiego, symbole językowe powiązane są z napisami wykorzystując pośrednictwo zbiorów form fleksyjnych, co pozwala na ich bardziej zwięzłą reprezentację (patrz p. 3.1.4). Dla języka polskiego analiza morfologiczna odbywa się z wykorzystaniem słownika fleksyjnego, czyli zbioru napisów uporządkowanych w zbiory form fleksyjnych, uzupełnione informacjami o formie podstawowej oraz cechach morfologicznych, zgodnie z definicją 3.2. Zastosowanie słownika fleksyjnego pozwala na sprowadzenie wyrażenia do jego formy podstawowej, określenie jego cech morfologicznych, a także generowanie wyrażeń posiadających określone własności morfologiczne.

Opisywany algorytm posiłkuje się w pierwszym rzędzie słownikiem fleksyjnym zbudowanym w Grupie Lingwistyki Komputerowej w Akademii Górniczo-Hutniczej [112]. Ze względu na charakterystykę tego słownika, który w obrębie kategorii gramatycznej rzeczownika zawiera głównie rzeczowniki pospolite, jest on uzupełniony wiedzą dostępną w słowniku Morfologik autorstwa Miłkowskiego [90].

5.4.3. Słownik semantyczny

Prezentowany algorytm ekstrakcji informacji zakłada, że w trakcie jego wykonania dostępne jest zbiór symboli językowych, których opis zawiera informację o kategorii semantycznej każdego symbolu (patrz p. 5.3.3). Najczęściej spotyka się dwa rodzaje zasobów, które zawierają opisy symboli językowych występujących w języku polskim i potencjalnie mogłyby zostać wykorzystane w prezentowanym algorytmie: jeden z polskich WordNetów [156, 108] albo otwarta encyklopedia Wikipedia[7]. Różnica pomiędzy tymi zasobami dotyczy zarówno sposobów ich konstruowania, zakresu dostępnych danych oraz sposobu organizacji wiedzy. Oba polskie WordNety konstruowane są w sposób pół-automatyczny przez ekspertów, co z jednej strony zapewnia wysokiej jakości dane, z drugiej jednak powoduje, że proces ten jest powolny i drogi. Dane w WordNetach zorganizowane są wokół teoretycznego pojęcia synsetu oraz powiązane są za pomocą relacji semantycznych [41] (porównaj p. 3.3). Ich organizacja powoduje zatem, że bezpośrednio można uzyskać informacje dotyczące kategorii semantycznej analizowanego wyrażenia.

Z drugiej strony Wikipedia jest projektem, którego pierwszorzędnym celem jest bezpłatne dostarczenie wiedzy encyklopedycznej jak największej grupie ludzi. Podstawowym celem jej konstrukcji nigdy nie było utworzenie zasobu wykorzystywanego do przetwarzania języka naturalnego. Niemniej jednak powstało wiele projektów [7, 149, 16, 23][8], których celem jest przekształcenie wiedzy dostępnej w Wikipedii w tym celu.

Skuteczna ekstrakcja informacji z tekstów o tematyce ogólnej, wymaga aby jak największa ilość wyrażeń, w szczególności nazw własnych, mogła zostać rozpoznana. Niespełnienie tego kryterium spowoduje, że wiele informacji może zostać całkowicie zignorowanych przez algorytm, przez co jego zastosowanie w praktycznych aplikacjach będzie stało pod znakiem zapytania. Dlatego też prezentowany algorytm wykorzystuje Wikipedię jako podstawowe źródło nazw własnych oraz wyrażeń wielosegmentowych.

[7] http://pl.wikipedia.org

[8] Bardziej kompletna lista projektów naukowych wykorzystujących Wikipedię znajduje się na stronie http://www.mkbergman.com/sweetpedia/.

Tablica 5.1: Liczba pojęć, relacji i asercji w ontologiach Cyc i SUMO.

Ontologia	Wersja	Pojęcia	Relacje	Asercje
OpenCyc	2.0	150 tys.	20 tys.	1,5 mln
ResearchCyc	1.1	542 tys.	24 tys.	3,4 mln
SUMO	1.52	29 tys.	0,9 tys.	158 tys.

5.4.4. Ontologia

Kolejnym ważnym zasobem wykorzystywanym w algorytmie jest ontologia, która traktowana jest jako źródło wiedzy na temat kategoryzacji pojęć, a także informacji na temat relacji semantycznych. Wiedza taksonomiczna jest wykorzystywana do rozszerzenia zbioru symboli stanowiących przykłady relacji, a także na etapie ekstrakcji relacji, kiedy dla danego symbolu o znanej kategorii semantycznej, trzeba określić, czy spełnia on ograniczenia semantyczne określone dla wzorca ekstrakcyjnego.

Ze względu na charakterystykę algorytmu, którego celem jest ekstrakcja informacji nieograniczonych dziedzinowo, wybór wykorzystywanej ontologii został zawężony wyłącznie do tych, które zawierają szeroki zasób wiedzy – tzw. ontologii ogólnych. Liczba ontologii tego rodzaju nie jest duża, gdyż z jednej strony przy ich konstrukcji wymagana jest dogłębna wiedza na temat formalizacji wiedzy, a z drugiej strony, koszt wytworzenia systemu zawierającego nietrywialne fakty jest znaczny. Spośród znanych systemów tego rodzaju rozważane były dwie ontologie: Cyc [66, 67] oraz SUMO [98, 97]. Obie ontologie dostępne są w wersjach darmowych (pierwsza posiada wersję OpenCyc, która może być wykorzystywana zarówno do badań naukowych jak i aplikacji komercyjnych, druga zaś od początku udostępniana jest na licencji GNU GLP). Ponadto pierwsza ontologia uważana jest za największą obecnie dostępną ontologię.

O wyborze ontologii Cyc zadecydowało kilka czynników: obszerność i klarowność dostępnej dokumentacji, zakres reprezentowanej wiedzy, dostępność dodatkowych narzędzi oraz model licencyjny pozwalający na wykorzystanie jej w aplikacjach, które nie są otwartoźródłowe. Bardzo istotnym czynnikiem decydującym o wyborze tej ontologii była dostępność zaawansowanego silnika inferencyjnego, który pozwala na efektywne wykorzystywanie wiedzy zgromadzonej w ontologii. Również proste porównanie statystyk tych ontologii przedstawione w tabeli 5.1 prowadzi do wniosku, że Cyc jest znacznie większą ontologią.

5.4.5. Semantyczna baza wiedzy

Ostatnim zasobem wiedzy wykorzystywanym w algorytmie jest baza wiedzy, zawierająca dużą liczbę faktów, opisanych z wykorzystaniem relacji semantycznych. Ze względu na rozwój technologii Semantic Web, a w szczególności inicjatywę Linked Data[9], której celem jest opublikowanie oraz powiązanie ze sobą wielu semantycznych zbiorów danych, możliwe jest łatwe znalezienie baz wiedzy, które mogą posłużyć do automatycznego określenia ograniczeń semantycznych ekstrahowanych relacji.

W prezentowanym algorytmie wykorzystywana jest DBpedia [7], gdyż jest to baza wiedzy bardzo podobna do Wikipedii (w istocie dane w DBpedii są ekstrahowane z Wikipedii), zatem część algorytmów opracowanych na potrzeby analizy Wikipedii działa również dla tej bazy wiedzy. Co więcej, w ramach opracowywania algorytmu klasyfikującego artykuły Wikipedii [118], opracowane zostało mapowanie pomiędzy ontologią Cyc a ontologią wykorzystywaną w DBpedii. Dzięki temu możliwe było łatwe przekształcenie danych z DBpedii na dane zgodne z ontologią Cyc i w konsekwencji automatyczne określenie ograniczeń semantycznych relacji z wykorzystaniem pojęć zdefiniowanych w Cyc.

[9] http://linkeddata.org

6. Zasoby wykorzystywane przez algorytm

6.1. Korpusy tekstów

6.1.1. Korpus Instytutu Podstaw Informatyki PAN

Przez korpus Instytutu Podstaw Informatyki Polskiej Akademii Nauk (w skrócie korpus IPI PAN) rozumiany jest tutaj korpus udostępniany na stronie `http://korpus.pl` opublikowany w marcu 2006 roku i zawierający ponad 250 milionów segmentów tekstowych. Zasób ten jest wykorzystywany do budowania wzorców formalnych relacji semantycznej na podstawie par symboli językowych, o których wiadomo, że zadana relacja występuje pomiędzy nimi (patrz p. 3 algorytmu).

Korpus ten nie jest zbalansowany, a dostępne statystyki dotyczą jego 30 milionowego podkorpusu (tabela 6.1). Wiadomo jedynie, że znaczną jego część stanowią artykuły prasowe, stenogramy posiedzeń parlamentarnych oraz akty prawne [128, s. 4].

W korpusie tym występuje ponad 250 milionów segmentów tekstowych (dokładna liczba segmentów podana jest w tabeli 6.2), co w momencie jego opublikowania czyniło go największym, w pełni otwartym korpusem języka polskiego. Obecnie istnieje znacznie większy Narodowy Korpus Języka Polskiego [129] zawierający ponad 1,8 miliarda segmentów, jednak jego użycie w badaniach naukowych jest utrudnione, gdyż nie ma możliwości pobrania go na komputer lokalny (ze względu na jego rozmiar oraz ograniczenia licencyjne), a odpowiedni zdalny interfejs programistyczny nie jest dostępny. Możliwe jest jedynie pobranie znacznie mniejszego korpusu zawierającego około milion segmentów.

Segment tekstowy w korpusie IPI PAN rozumiany jest w przybliżeniu jako ciąg liter lub innych znaków tworzący wyraz bądź inny napis. Należy zwrócić uwagę, że pojęcie to obejmuje znaki przestankowe oraz że tradycyjnie rozumiane wyrazy mogą składać się z dwóch lub większej liczby segmentów. Wynika to z faktu iż niektóre wyrazy konstruowane są z innych wyrazów, np. *polsko-niemiecki* oraz występującego w ograniczonym zakresie w języku polskim zjawiska aglutynacji. Przykładowo aglutynant czasownika *być* zazwyczaj jest częścią czasownika, ale może być również częścią innych części mowy, np. zdanie „Ale nas wszystkich zaskoczył**eś**!” może być przekształcone w zdanie „Ale**ś** nas wszystkich zaskoczył!”. Dlatego

Tablica 6.1: Skład 30-milionowego podkorpusu korpusu IPI PAN.

proza współczesna	9,7%
proza dawna	10,6%
teksty książkowe niebeletrystyczne	10%
prasa	49,3%
stenogramy sejmowe i senackie	15,5%
ustawy	4,9%

Tablica 6.2: Statystyki korpusu IPI PAN

Liczba segmentów	254524624
Liczba unikalnych segmentów	1396832
Liczba leksemów	774803
Liczba unikalnych tagów	1282

w obu zdaniach aglutynant traktowany jest jako osobny segment, przez co słowa *zaskoczyłeś* oraz *Aleś* składają się z dwóch segmentów.

Istotną cechą tego korpusu jest to, że został on wyposażony w znakowanie morfosyntaktyczne z użyciem formalizmu opisanego w pracy *System znaczników morfosyntaktycznych w korpusie IPI PAN* autorstwa Wolińskiego [162]. Interpretacja form wyrazowych odbywa się z wykorzystaniem analizatora morfologicznego Morfeusz [163]. Ponieważ w języku polskim, podobnie jak w innych językach, znaczna część form tekstowych jest morfologicznie wieloznaczna (porównaj p. 2.3.2), konieczne jest ich ujednoznacznienie. Ze względu na jego rozmiar, ujednoznacznianie korpusu IPI PAN było zrealizowane automatycznie, z użyciem algorytmu opracowanego przez Dębowskiego [35]. Poprawność tego algorytmu jego autor oceniał na 90,4%.

Korpus IPI PAN dystrybuowany jest w postaci binarnej, a dostęp do jego zawartości realizowany jest za pomocą serwera korpusowego Poliqarp [55]. Główną zaletą tego narzędzia jest możliwość przeszukiwania korpusów z wykorzystaniem zaawansowanego języka zapytań, pozwalającego wyszukiwać fragmenty tekstu nie tylko na podstawie słów kluczowych, ale również form podstawowych wyrazów oraz ich cech morfologicznych. Przykładowo, aby wyszukać wszystkie fragmenty tekstu, w których występuje wyraz *pies* w dowolnej z jego form fleksyjnych, konstruujemy zapytanie `[base=pies]`. Zapytanie to możemy dodatkowo ujednoznacznić zawężając wyszukiwanie do segmentów, o których wiadomo, że są rzeczownikami (`pos=subst`) rodzaju żywotnego (`gender=m2`): `[base=pies & pos=subst & gender=m2]`. Składnia języka pozwala również na wyszukiwanie wielu segmentów w obrębie jednego zdania oraz segmentów występujących w zadanej odległości. Szczegółowy opis języka zapytań wraz z przykładami dostępny jest w pracy „Korpus IPI PAN" pod redakcją Przepiórkowskiego [130].

6.1.2. Korpus notatek Polskiej Agencji Prasowej

Korpus notatek Polskiej Agencji Prasowej (w skrócie korpus PAP) jest zbiorem krótkich artykułów obejmujących szeroki zakres zjawisk opisywanych przez dziennikarzy PAP, które były gromadzone w Grupie Lingwistyki Komputerowej Akademii Górniczo-Hutniczej. Jest on wykorzystywany w zadaniach związanych z algorytmem ekstrakcji relacji – do wyodrębnienia przykładów zdań, które pasują do określonego wzorca formalnego relacji (porównaj p. 8 algorytmu) oraz do weryfikacji jakości wzorców ekstrakcyjnych.

Tabela 6.3 przedstawia szczegółowe statystyki dotyczące korpusu PAP. Istotną cechą tego korpusu jest dosyć duża liczba segmentów (około 18) przypadających na pojedyncze zdanie i wynika on z charakteru danych – notatek prasowych, które są pisane tak, aby w zwięzłej formie przekazać znaczną ilość informacji.

Do podziału tekstu na zdania użyte zostały reguły podziału dla języka polskiego autorstwa Miłkowskiego i Lipskiego [91] wyrażone w standardzie SRX [126]. Następnie zdania te zostały podzielone na segmenty z wykorzystaniem wyrażeń regularnych przedstawionych w tabeli 6.4. Ponieważ reguły użyte do podziału zdań na segmenty nie uwzględniają w ogóle analizy morfologicznej, ich definicja różni się od definicji segmentu używanej w korpusie IPI PAN: w korpusie PAP wyrazy mogą się składać wyłącznie

Tablica 6.3: Statystyki korpusu PAP

Liczba notatek	51572
Liczba zdań	196249
Liczba segmentów	3595398
Liczba unikalnych segmentów	165422
Średnia liczba zdań w notatce	3,8
Średnia liczba segmentów w zdaniu	18,3

Tablica 6.4: Wyrażenia regularne użyte do podziału tekstów na segmenty.

słowo	`\p{Alpha}\p{Word}*`
liczba	`\p{Digit}+(?:[:., _/-]\p{Digit}+)*`
znak interpunkcyjny	`\p{Punct}`
znak graficzny	`\p{Graph}`
inny znak	`[^\p{Word}\p{Graph}]+`

z pojedynczych segmentów. Należy zwrócić uwagę, że w przeciwieństwie do korpusu IPI PAN, segmenty korpusu PAP nie zostały oznakowane morfosyntaktycznie.

6.2. Słowniki fleksyjne

6.2.1. Słownik CLP

Przedstawiony algorytm ekstrakcji relacji semantycznych zakłada, że elementami składowymi wzorców ekstrakcyjnych będą cechy morfologiczne. Algorytm wykorzystuje jako podstawowe źródło informacji na temat morfologii słów języka polskiego słownik fleksyjny stworzony w Grupie Lingwistyki Komputerowej Akademii Górniczo-Hutniczej o nazwie CLP [112, 44].

Podstawowym sposobem organizacji danych w tym słowniku są tradycyjne kategorie gramatyczne: rzeczownik, czasownik, przymiotnik, liczebnik, zaimek, przysłówek oraz kategoria obejmująca wszystkie pozostałe wyrazy nieodmienne. Poszczególne kategorie wprowadzają dalsze podziały, np. rzeczowniki zostały podzielone według ich rodzaju gramatycznego, a liczebniki na wielorodzajowe, dwurodzajowe, jednorodzajowe, bezrodzajowe, nieokreślone oraz nieodmienne. Każda z tych klas fleksyjnych może być dalej podzielona ze względu na sposób odmiany wyrazów. Ponieważ próba nadania nazwy tego rodzaju klasom fleksyjnym byłaby mało efektywna, w słowniku poza nazwami funkcjonują tzw. *etykiety fleksyjne* – na każdym poziomie podziału wyróżnione klasy etykietowane są za pomocą dużych liter alfabetu łacińskiego, począwszy od litery A. Ponieważ klasyfikacja wyrazów ma strukturę ściśle hierarchiczną, uporządkowany ciąg liter reprezentujących poszczególne poziomy hierarchii umożliwia jednoznaczne zidentyfikowanie dowolnej klasy fleksyjnej, a w powiązaniu z formą hasłową wyrazu, umożliwia jednoznaczne zidentyfikowanie sposobu jego odmiany (porównaj wzór 3.3). Przykładowo, para `(robić,BBCA)` pozwala jednoznacznie zidentyfikować pełny wzorzec odmiany czasownika *robić*.

Drugim istotnym elementem organizacji słownika jest powiązanie etykiet fleksyjnych z wektorami form wyrazowych. Mechanizm ten przyporządkowuje poszczególnym formom identyfikatory liczbowe (pozycje w wektorze odmiany) w ramach określonej kategorii fleksyjnej. Dla każdej pozycji w wektorze odmiany ustalone są również wartości wszystkich kategorii gramatycznych, dzięki czemu dla określonej formy

Tablica 6.5: Wektor odmiany rzeczownika `(adorator,AAAAAA)`.

		Wartości kategorii gramatycznych	
Pozycja	**Forma**	**Przypadek**	**Liczba**
1	adorator	mianownik	pojedyncza
2	adoratora	dopełniacz	pojedyncza
3	adoratorowi	celownik	pojedyncza
4	adoratora	biernik	pojedyncza
5	adoratorem	narzędnik	pojedyncza
6	adoratorze	miejscownik	pojedyncza
7	adoratorze	wołacz	pojedyncza
8	adoratorowie	mianownik	mnoga
9	adoratorów	dopełniacz	mnoga
10	adoratorom	celownik	mnoga
11	adoratorów	biernik	mnoga
12	adoratorami	narzędnik	mnoga
13	adoratorach	miejscownik	mnoga
14	adoratorowie	wołacz	mnoga

można ustalić wartości jej kategorii gramatycznych (porównaj wzór 3.2). Przykładowo, rzeczownik posiada 14 pozycji w wektorze odmiany: pozycja pierwsza zajmowana jest przez formę mianownika w liczbie pojedynczej, druga przez formę dopełniacza w liczbie pojedynczej, trzecia przez formę celownika w liczbie pojedynczej, itd. W tabeli 6.5 podany jest przykładowy wektor odmiany wraz z interpretacją poszczególnych pozycji dla leksemu `(adorator,AAAAAA)`.

Taka organizacja informacji pozwala na wykonanie dwóch podstawowych operacji: dla zadanej formy wyrazowej (napisu) określenie jej etykiety fleksyjnej lub etykiet fleksyjnych w przypadku formy homonimicznej oraz jej pozycji fleksyjnej lub pozycji fleksyjnych w przypadku formy homonimicznej, bądź formy wewnętrznie homonimicznej oraz dla leksemu o znanej formie podstawowej oraz etykiecie fleksyjnej określenie jego form wyrazowych dla poprawnych kombinacji wartości kategorii gramatycznych. Operacje te są realizowane za pomocą interfejsu w języku C opisanego szczegółowo przez Gajęckiego w pracy [44]. Należy jednak zwrócić uwagę, że interfejs ten realizuje niskopoziomowy dostęp do słownika – programista operuje bezpośrednio liczbowymi identyfikatorami leksemów oraz pozycji fleksyjnych. Dlatego istnieje również bardziej wysokopoziomowy interfejs dla języka Icon, a autor niniejszej pracy opracował obiektowy interfejs dla języka Ruby.

Słownik CLP w wersji wykorzystywanej przez autora (2.1) zawiera informacje o 138331 leksemach. Szczegółowa statystyka słownika przedstawiona jest w tabeli 6.6. Według jego autorów [112, s. 64-67] słownik pokrywa 84% form tekstowych występujących w korpusie notatek PAP. Istotnym mankamentem słownika w kontekście ekstrakcji informacji jest to, że nie zawiera on nazw własnych (z wyjątkiem imion oraz niewielkiej liczby nazw geograficznych). Przykładowo w słowniku nie występuje leksem rzeczownikowy *Polska*, chociaż występuje powiązany z nim relacją derywacji leksem przymiotnikowy *polski*. Z tego względu autor algorytmu uzupełnił słownik CLP danymi dostępnymi w słowniku Morfologik.

Tablica 6.6: Liczba leksemów poszczególnych klas gramatycznych w słowniku CLP.

Klasa gramatyczna	**Liczba leksemów**
rzeczownik	74046
przymiotnik	38066
czasownik	20067
przysłówek	5068
leksem nieodmienny	734
zaimek	182
liczebnik	168
razem	138331

Tablica 6.7: Przykładowe wpisy znajdujące się w słowniku Morfologik.

Forma tekstowa	**Forma podstawowa**	**Opis morfologiczny**
Gliwicami	Gliwice	subst:pltant:inst:n
gliwickiej	gliwicki	adj:sg:dat.gen.loc:f:pos:aff
chromatograficznie	chromatograficznie	adv:pos:aff
dosiec	dosiec	verb:inf:perf

6.2.2. Morfologik

Morfologik jest wolnodostępnym słownikiem, który powstał poprzez wyekstrahowanie reguł tworzonych przez użytkowników programu `ispell` dla języka polskiego [90]. Z tego względu dane w nim zawarte z jednej strony w lepszym stopniu odzwierciedlają zasób słów, wykorzystywany we współczesnej polszczyźnie, ale z drugiej dane te są niższej jakości, niż te zgromadzone w słowniku CLP. Dlatego słownik Morfologik wykorzystywany jest wyłącznie w sytuacji, w której określona forma tekstowa nie została rozpoznana przez słownik CLP. Prowadzi to do pewnych problemów (np. leksem rzeczownikowy *Polska* nie występuje w słowniku CLP, ale forma *Polska* jest rozpoznawana jako przynależąca do przymiotnika *polski*) – przeciwne rozwiązanie prowadziłoby jednak do jeszcze większej liczby problemów, gdyż dostępne dane morfologiczne dublowałyby się, zwiększając wieloznaczność rozpoznania poszczególnych form.

Morfologik jest dystrybuowany w postaci pliku tekstowego zawierającego informacje o poszczególnych formach oraz skompilowanego automatu skończonego. Słownik w postaci pliku tekstowego zawiera trójki postaci: `(forma tekstowa, forma podstawowa, znaczniki morfosyntaktyczne)`. Tabela 6.7 zawiera przykładowe wpisy znajdujące się w słowniku, które nie występują w słowniku CLP. Zestaw znaczników użytych do opisu wartości kategorii gramatycznych w dużej mierze odpowiada formalizmowi stosowanemu w korpusie IPI PAN [162]. Różnice pomiędzy tymi formalizmami są opisane w pracy [132]. Istotnym mankamentem tego sposobu organizacji informacji jest niejednoznaczność, która pojawia się dla niektórych form tekstowych. Przykładowo forma tekstowa *rząd* przynależy do dwóch leksemów rzeczownikowych rodzaju męskiego nieżywotnego posiadających różne paradygmaty odmiany (pierwszy z nich posiada w bierniku liczby pojedynczej formę *rządy* a drugi *rzędy*). Niemniej jednak w słowniku Morfologik leksemów tych nie da się odróżnić, przez co bez dodatkowej analizy nie sposób jest przyporządkować im właściwe formy tekstowe.

Chcąc wykorzystać słownik Morfologik w algorytmie ekstrakcji relacji, konieczne było uzgodnienie sposobu organizacji informacji ze sposobem organizacji informacji w słowniku CLP. Ze względu na dobrą

Tablica 6.8: Liczba leksemów zaimportowanych ze słownika Morfologik do słownika CLP.

Klasa gramatyczna	**Liczba leksemów**
rzeczownik	168082
przymiotnik	69587
czasownik	21277
przysłówek	10063
razem	269009

znajomość słownika CLP, autor postanowił przekształcić dane Morfologika dostępne w postaci tekstowej, do postaci wykorzystywanej w tym pierwszym słowniku. Proces ten polegał na rozpoznaniu etykiety fleksyjnej, którą można by przyporządkować leksemom opisanym w słowniku Morfologik. Sposób opisu form fleksyjnych w słowniku Morfologik prowadzi do niejednoznaczności wskazanej w poprzednim paragrafie, część występujących w nim paradygmatów odmiany nie występuje w słowniku CLP oraz część danych w tym słowniku jest błędna (np. rzeczowniki opisane jako przymiotniki), dlatego też dość duża część danych została pominięta. Tabela 6.8 zawiera informacje o liczbie leksemów, które zostały zaimportowane ze słownika Morfologik do słownika CLP.

6.3. Słownik semantyczny

Elementem niezwykle istotnym z punktu widzenia uniwersalności konstruowanego algorytmu jest dostępność słownika semantycznego definiującego jak największa liczbę symboli językowych. W kontekście ekstrakcji relacji słownik ten powinien posiadać następujące cech: zawierać informacje dotyczące nazw własnych i wyrażeń wielosegmentowych, określać kategorie semantyczne występujących w nim symboli oraz pozwalać identyfikować właściwy sens wyrażeń synonimicznych. Prace nad słownikiem tego rodzaju prowadzone są w Katedrze Lingwistyki Komputerowej Uniwersytetu Jagiellońskiego [119], ale słownik ten jest daleki od kompletności. Dlatego autor algorytmu jako podstawowe źródło wiedzy zbliżone do słownika semantycznego wykorzystał polską Wikipedię.

Istotną barierą stojącą na przeszkodzie w bezpośrednim wykorzystaniu Wikipedii w algorytmach ekstrakcji informacji jest to, że dane w niej zawarte są słabo ustrukturyzowane. Znaczy to, że nie da się jej wykorzystać tak łatwo jak autentycznego słownika semantycznego, w którym relacje występujące pomiędzy wyrażeniami językowymi byłyby nazwane i zidentyfikowane bezpośrednio. Konieczne jest zastosowanie mniej lub bardziej zaawansowanych algorytmów, pozwalających przekształcić je do postaci przydatnej w automatycznej ekstrakcji informacji.

Dostępne są dwa szeroko stosowane narzędzia pozwalające na wydobycie z Wikipedii ustrukturyzowanych danych: Wikiepdia Miner [86] oraz moduł ekstrakcji informacji stosowany przez twórców DBpedii [15]. Autor zdecydował się na wykorzystanie w tym celu Wikipedia Minera. Moduł ekstrakcyjny DBpedii wykorzystywany jest przy konstrukcji słownika semantycznego jedynie do wydobycia informacji o obecności infoboksów w treści artykułów. Pozwala to na dokładniejsze określenie kategorii semantycznej symboli językowych, zgodnie z algorytmem opisanym w punkcie 7.2. DBpedia wykorzystywana jest również jako źródło wiedzy pozwalające automatycznie określić ograniczenia semantyczne, co zostało opisane w punkcie 6.5.

6.3.1. Wikipedia Miner

Wikipedia Miner [86] to projekt, którego celem jest przekształcenie pół-strukturalnych danych Wikipedii w dane ustrukturyzowane o postaci zbliżonej do słownika semantycznego. Dzięki temu możliwe jest np. określenia pokrewieństwa semantycznego dwóch symboli językowych [160], czy zbudowanie algorytmu linkowania do artykułów Wikipedii [87], będącego w istocie algorytmem ujednoznaczniania sensu wyrażeń. Mechanizm przekształcania danych Wikipedia Minera koncentrują się na jej strukturze hipertekstowej. Jest on zdolny do wydobywania następujących informacji:

- etykiety artykułów,
- powiązania z artykułami w innych językach,
- przekierowania pomiędzy różnymi tytułami artykułu,
- hiperłącza prowadzące do zasobów wewnątrz Wikipedii,
- kategorie Wikipedii przypisane artykułom.

Wikipedia Miner kładzie szczególny nacisk na wykorzystanie danych tekstowych oraz relacji pomiędzy artykułami Wikipedii, jakie można wydostać na podstawie jej struktury hipertekstowej. Bardzo istotnym składnikiem zestawu ekstrahowanych informacji są *nazwy odnośników* prowadzących do innych artykułów. Ponieważ można założyć, że Wikipedyści tworząc tego rodzaju powiązania, starali się przekazywać rzetelne informacji, powiązanie informacji o nazwie odnośnika z artykułem, do którego on prowadzi, daje dostęp do ujednoznacznionego sensu, kryjącego się pod nazwą odnośnika. Sama w sobie informacja ta pozwala istotnie rozszerzyć zestaw wyrażeń, za pomocą których można odnieść się do analizowanego artykułu. Co więcej uzupełniając tę informację o liczbę odwołań za pomocą określonej nazwy (również ekstrahowaną przez Wikipedia Minera) można uzyskać bardzo cenne informacje statystyczne. Pozwalają one z dużym prawdopodobieństwem określić, który sens określonego homonimicznego wyrażenia jest najczęstszy. Co więcej Wikipedia Miner oblicza również częstość z jaką określone wyrażenie jest wykorzystywane jako odnośnik. Dzięki temu możliwe jest automatyczne określenie, które wyrażenia niosą istotną informację (np. nazwy własne), a które, choć sporadycznie stają się odnośnikami, nie są zbyt istotne (np. spójniki, partykuły i przyimki).

Również struktura wewnętrznych odnośników jest bardzo ważnym składnikiem słownika semantycznego, który można zbudować na podstawie Wikipedii. Obliczając częstość z jaką dwa artykuły posiadają odnośniki prowadzące do tych samych artykułów, można określić ich pokrewieństwo semantyczne.

Autorzy wykorzystują te informacje na dwa sposoby: po pierwsze, na bazie struktury odnośników przychodzących i wychodzących definiują miarę semantycznego pokrewieństwa pomiędzy artykułami [160], po drugie, na bazie nazw odnośników budują algorytm pozwalający na ujednoznacznianie sensów wyrażeń oraz algorytm automatycznego uzupełnianie dowolnego tekstu odnośnikami do odpowiednich artykułów Wikipedii [87]. Algorytmy te mają wysoką skuteczność – miara pokrewieństwa semantycznego w 78% zgodna jest z ocenami ludzi, algorytm ujednoznaczniania sensu osiąga skuteczność mierzoną miarą F_1 na poziomie 97,1% a algorytm uzupełniający tekst o odnośniki do Wikipedii osiąga precyzję na poziomie 74,1%.

6.3.2. Konstrukcja słownika semantycznego

W wyniku działania algorytmów ekstrakcyjnych Wikipedia Minera, dla każdego hasła w Wikipedii ekstrahowany jest zbiór informacji przedstawiony w tabeli 6.9. Każdy artykuł Wikipedii traktowany jest jako osobny symbol językowy, a zestaw danych przedstawiony w tabeli traktowany jest jako jego opis.

Tablica 6.9: Informacje ekstrahowane przez Wikipedia Minera. Przykład przedstawia informacje dla artykułu *Gdańsk* z polskiej edycji Wikipedii.

Cecha	**Przykład**
Forma hasłowa	`Gdańsk`
Przekierowanie	`Danzig`
Nazwa w odnośniku	`Gdańsku`
Tłumaczenie (na niemiecki)	`Danzig`
Kategoria	`Miasta wojewódzkie`
Odnośnik wychodzący	**`Gdynia`**
Odnośnik przychodzący	**`Toruń`**

Powiązanie symbolu z napisami (porównaj p. 3.1.4) realizowane jest w oparciu o nazwy odnośników prowadzących do danego artykułu. W ten sposób możliwe jest zarówno powiązanie różnych form fleksyjnych oraz różnych wariantów pisowni odpowiadających pojedynczemu symbolowi językowemu. Przykłady powiązań pomiędzy symbolem a napisami przedstawione są w tabeli 6.10. Należy zwrócić uwagę na dwa fakty: po pierwsze napisy służące do odnoszenia się do danego symbolu językowego wyposażone są w informację ilościową, co pozwala określić, które spośród nich są bardziej, a które mniej prawdopodobne. Ma to bardzo duże znaczenie dla algorytmu rozstrzygania wieloznaczności. Z drugiej jednak strony, w przeciwieństwie do słownika semantycznego konstruowanego ręcznie, dane w słowniku skonstruowanym na bazie Wikipedii nie są ani kompletne, ani w pełni poprawne. Przykładowo wśród odnośników prowadzących do hasła *Polska* znajduje się np. *Polak*, który odnosi się do innego pojęcia (*obywatel Polski*) niż opisywany symbol językowy. Jest to cena, którą trzeba zapłacić za korzystanie z zasobu tego rodzaju.

Ponadto dane statystyczne określone na podstawie Wikipedii odzwierciedlają jej encyklopedyczny charakter. W tabeli 6.11 przedstawione są różne symbole językowe powiązane z napisem `zamek`. Opierając się na danych Wikipedii można by uznać, że w znaczeniu *budowli* wyraz ten występuje kilkakrotnie częściej niż w znaczeniu *elementu borni* i kilkasetkrotnie częściej niż w znaczeniu *urządzenia*. Te wyniki są niezgodne z wiedzą zawartą w tradycyjnych słownikach (gdzie częstość występowania odzwierciedlona jest w kolejności homonimicznych haseł), zgodnie z którą `zamek` w znaczeniu *urządzenia* jest najczęstszym sensem.

Istotnym uzupełnieniem tych danych statystycznych jest informacja o częstości z jaką dane wyrażenie używane jest w odnośnikach wewnątrz Wikipedii. Informacja ta pozwala określić czy wyrażenie to niesie informację na tyle istotną, aby twórcy Wikipedii uznali, że powinno posiadać odnośnik. Dzięki temu możliwe jest oszacowanie istotności danego wyrażenia. Dane na temat częstości użycia wybranych wyrażeń przedstawione są w tabeli 6.12. Widać wyraźnie, że wyrażenia będące nazwami własnymi (zgromadzone w górnej części tabeli) mają znacznie wyższe prawdopodobieństw wystąpienia jako odnośnik niż wyrażenia takie jak rzeczowniki pospolite, przymiotniki, czy partykuły.

Chociaż słownik semantyczny, który można zbudować automatycznie na bazie Wikipedii posiada znacznie niższą jakość, niż słownik semantyczny konstruowany przez językoznawców, posiada on kilka

Tablica 6.10: Odnośniki prowadzące w polskiej Wikipedii do hasła *Polska*. s_i – nazwa odnośnika, c_{link} – liczba wystąpień. W wynikach pominięto odnośniki dla których $c_{link} < 100$.

s_i	$c_{link}(s_i)$
Polsce	74196
polski	11564
Polski	5528
Polska	3271
polska	2234
Polskę	763
Polską	647
polskiego	605
polskiej	510
polsko	458
polskich	432
polskie	295
polskim	246
polską	130
Polak	108
RP	101
Polaków	100
Polacy	100

Tablica 6.11: Prawdopodobieństwa sensów wyrażenia zamek ustalone na podstawie Wikipedii. σ_i – symbol językowy (tytuł artykułu Wikipedii), $c_{link}(s, \sigma_i)$ – liczba odnośników o treści zamek prowadzących do symbolu σ_i, $P_{sense}(s, \sigma_i)$ – prawdopodobieństwo sensu σ_i. W wynikach pominięto sensy, dla których $c_{link} < 2$.

σ_i	$c_{link}(s, \sigma_i)$	$P_{sense}(s, \sigma_i)$
Zamek	425	0.698
Zamek (broń)	60	0.099
Zamek w Bydgoszczy	28	0.046
Zamek w Bolkowie	11	0.018
Zamek w Szydłowcu	5	0.008
Zamek Świny	4	0.007
Zamek Kapituły Warmińskiej w Olsztynie	4	0.007
Zamek (urządzenie)	4	0.007
Zamek Królewski w Poznaniu	3	0.005
Zamek w Malborku	2	0.003
Zamek w Rzeszowie	2	0.003
Zamek w Suchej Beskidzkiej	2	0.003
Zamek w Kowalu	2	0.003
Zamek Królewski na Wawelu	2	0.003
Zamek w Edynburgu	2	0.003

Tablica 6.12: Statystyki przykładowych wewnętrznych odnośników występujących w Wikipedii. s_i – napis, $c_{link}(s_i)$ – liczba wystąpień jako odnośnik, $c_{total}(s_i)$ – liczba wszystkich wystąpień, $P_{link}(s_i)$ – prawdopodobieństwo występowania jako odnośnik.

s_i	$c_{link}(s_i)$	$c_{total}(s_i)$	$P_{link}(s_i)$
Jerzy Buzek	108	236	0.458
Kraków	2756	29305	0.094
Polska	3720	32164	0.116
gleby brązowe	2	10	0.200
gliwickich	4	29	0.138
internetowy	4	1281	0.003
literatury	291	10472	0.028
małego	2	3083	0.001
miastem	121	8122	0.015
nie	3	714190	0.000
sascy	5	46	0.109
ulica	149	5776	0.026
zakopiański	4	51	0.078
zamek	609	8787	0.069

cech, które czynią go przydatnym z punktu widzenia ekstrakcji informacji. W pierwszym rzędzie zawiera on szereg nazw własnych, niezwykle istotnych z punktu widzenia ekstrakcji informacji, których raczej nie sposób spotkać w tradycyjnym słowniku semantycznym. Ponadto słownik taki jest wyposażony w szereg danych statystycznych, które pozwalają na stworzenie dość skutecznych algorytmów ujednoznaczniania sensu – istotnie przewyższających swoją skutecznością algorytmu bazujące jedynie na cechach strukturalnych dostępnych np. w WordNecie. Ma to szczególne znaczenie dla algorytmu ekstrakcji relacji semantycznych, od którego oczekujemy, że będzie zdolny do interpretacji informacji na poziomie pojedynczych zdań. Dzięki temu możliwe jest bowiem ekstrahowanie informacji o konkretnych obiektach, a nie jedynie bliżej nieokreślonych bytach, o których wiemy jedynie, że posiadają określoną nazwę, jak to ma miejsce w tradycyjnych algorytmach ekstrakcji informacji. Możliwe jest bowiem uzupełnienie tak ekstrahowanych danych odnośnikiem do Wikipedii, pod którym użytkownik może znaleźć więcej informacji na temat danego obiektu i skonfrontować je z informacjami dostarczonymi przez algorytm ekstrakcji informacji.

Pewne niedogodności związane z automatyczną konstrukcją takiego słownika (np. brak informacji o kategorii semantycznej), mogą być również rozwiązane algorytmicznie (patrz p. 7.2). Zaś błędne dane (np. forma `Polak` w haśle **`Polska`**) choć stanowią pewną wadę, nie przekreślają całkowicie przydatności tego narzędzia, gdyż ze względu na semantyczną bliskość tych wyrażeń, użytkownik nie powinien mieć trudności ze skorygowaniem tak uzyskanych informacji. Wszystkie te cechy łącznie zaważyły na wykorzystaniu Wikipedii jako źródła danych dla słownika semantycznego.

6.4. Ontologia

Kolejnym źródłem wiedzy wykorzystywanym w algorytmie ekstrakcji informacji jest ontologia Cyc [67], która wykorzystywana jest do pozyskiwania par symboli połączonych relacją, klasyfikacji symboli w słowniku semantycznym, automatycznego określania ograniczeń semantycznych oraz weryfikacji speł-

niania ograniczeń we wzorcach ekstrakcyjnych. Ontologia ta występuje w trzech wersjach: otwartej – *OpenCyc*, badawczej – *ResearchCyc* oraz komercyjnej – *Cyc*. Wyniki prezentowane w niniejszej pracy uzyskane zostały na podstawie *ResearchCyc* w wersji 4.0.

Ktoś kto pierwszy raz ma do czynienia z ontologią Cyc może odnieść wrażenie, że zawarta w niej wiedza jest bardzo chaotyczna, a informacje nie posiadają przejrzystej struktury. Wrażenie to jest po części spowodowane tym, że ilość informacji jaka znajduje się w tej ontologii jest ogromna i przeglądając je w przypadkowy sposób, trudno rozpoznać zasady ich organizacji. Niemniej jednak dłuższe korzystanie z tego zasobu pozwala zrozumieć jak trudno jest w prosty sposób zorganizować tak duży zbiór informacji. Poniżej przedstawione zostały podstawowe wiadomości niezbędne do zrozumienia sposobu organizacji danych w Cyc oraz sposobu ich wykorzystania w algorytmie ekstrakcji informacji. Więcej informacji na temat samej ontologii oraz możliwości jej zastosowania w języku polskim można znaleźć w pracy [124].

6.4.1. Organizacja pojęć

Najważniejszym i najbardziej podstawowym sposobem organizacji pojęć w Cyc jest relacja *generalizacji* (`#$genls`), która pozwala łączyć pojęcia bardziej specyficzne z pojęciami bardziej ogólnymi (patrz p. 3.2.5). Klasycznym przykładem tej relacji jest para: `#$Dog` (pol. *pies*) – `#$Animal` (pol. *zwierzę*). Pies posiada wszystkie cechy zwierzęcia, dlatego zwierzę jest generalizacją psa. Z drugiej strony pies, w stosunku do innych zwierząt, posiada pewne cechy specyficzne (np. szczeka) dlatego powiemy, że pies jest pewnym szczególnym zwierzęciem. Opierając się wyłącznie na relacji `#$genls` można przekształcić Cyc w sieć definicyjną (patrz p. 3.3.1).

Relacja generalizacji w Cyc jest ścisła, tzn. jest to relacja ściśle przechodnia. Możemy powiedzieć, że relacja ta ma charakter bardziej ontologiczny niż semantyczny. Jednakże wbrew ogólnym wyobrażeniom na temat ontologii, pojęcia w Cyc nie tworzą hierarchii (gdzie każde pojęcie może posiadać co najwyżej jedną generalizację), lecz heterarchię (gdzie każde pojęcie może posiadać wiele generalizacji) zwaną również polihierarchią.

Drugą ważną relacją, wiążącą się z relacją generalizacji jest relacja *typ-okaz* (`#$isa`), to znaczy relacja, która wiąże egzemplarz określonego pojęcia z tym pojęciem. Tutaj również klasycznym przykładem jest *pies*, którego widzimy za oknem i pojęcie `#$Dog`, pod które podpada ten *pies*.

Te dwie relacje są kluczowe dla zrozumienia sposobu organizacji informacji w Cyc – dzięki nim można zrozumieć podstawowy podział realizowany w tej ontologii na pojęcia, które w nomenklaturze Cyc nazywane są *kolekcjami* (`#$Collection`) oraz egzemplarze, które w nomenklaturze Cyc nazywane są *indywiduami* (`#$Individual`). Wszystkie obiekty opisywane w Cyc zaliczane są do jednego ze zbiorów – kolekcji bądź indywiduów, dlatego w praktyce dla każdego opisywanego obiektu można szybko zorientować się jaki jest jego status ontologiczny. W szczególności, w algorytmie ekstrakcji informacji interesować nas będą obiekty należące do tego pierwszego zbioru.

W tym miejscu należy zwrócić uwagę na fakt, że podział ten obejmuje również inne istotne elementy Cyc: *predykaty*, *funkcje* oraz *mikroteorie* – wszystkie one należą do zbioru indywiduów. Dlatego podział na kolekcje i indywidua ma fundamentalne znaczenie dla tej ontologii.

Wiedza na temat organizacji pojęć jest niezbędna do realizacji opisywanego w niniejszej pracy algorytmu ekstrakcji relacji. W pierwszym rzędzie, pozwala ona na przeprowadzanie wnioskowań na temat ograniczeń semantycznych argumentów ekstrahowanych relacji. Pozwala ona na stwierdzenie, np. że jeśli przyjmiemy, że częścią *obiektu geograficznego* może być inny *obiekt geograficzny*, to jeśli zidentyfikujemy w tekście jakieś *miasto* i *państwo*, będziemy mogli wywnioskować, że obiekty te spełniają wskazane ograniczenia semantyczne.

Dalej, na etapie wyszukiwania przykładów uczących wiedza ta pozwala zastąpić ogólne pojęcia (np. *zwierzę*) pojęciami bardziej specyficznymi (np. *pies*, *kot*, *kangur*, *słoń*, *ryjówka*, itd.) co umożliwia znalezienie znacznie większej liczby przykładowych zdań zawierających wystąpienia danej relacji, bez dodatkowej pracy po stronie osoby określającej przykładowe pary symboli językowych, co zostało opisane w punkcie 7.1.

6.4.2. Predykaty

Poza predykatami pozwalającymi opisywać strukturę pojęciową Cyc (`#$genls`, `#$isa`), w ontologii tej istnieje bardzo duża liczba innych predykatów, które w założeniu mają służyć do wyrażania wszelkiej wiedzy umożliwiającej opis otaczającego nas świata. Co więcej, znaczna część tych dodatkowych predykatów jest faktycznie wykorzystywana. Łączna liczba predykatów dostępnych w ResearchCyc przekracza 20 tys., dlatego można przypuszczać, że ontologia ta definiuje model pojęciowy, odzwierciedlający znaczną część wiedzy jaką posługują się ludzie.

W kontekście ekstrakcji informacji zbiór predykatów w Cyc może być eksploatowany na dwa sposoby. Po pierwsze, może być wykorzystany do opisu ekstrahowanych informacji, poprzez utożsamienie określonej relacji semantycznej zidentyfikowanej w tekście, z określonym predykatem Cyc, a w konsekwencji pozwala przekształcać wiedzę z tekstu w wiedzę w formalizmie Cyc. Po drugie, ze względu na występowanie w tej ontologii fragmentarycznych opisów faktograficznych, a także na konieczność każdorazowego określenia ograniczeń semantycznych występujących predykatów, w połączeniu ze zbiorem asercji oraz opisem samych predykatów, może być wykorzystywany w celu pozyskania par symboli językowych, których wystąpienia poszukiwane są w tekstach języka polskiego (patrz p. 7.1). Ograniczenia semantyczne predykatów mogą być również wykorzystywane do automatycznego określenia ograniczeń semantycznych ekstrahowanych relacji (patrz p. 7.4).

Dotychczasowe badania pokazują [124], że w pierwszym kontekście zawartość ontologii spełnia pokładane w niej nadzieje. Pozwala ona wyrażać relacje (zarówno semantyczne jak i ontologiczne) występujące w typowych zadaniach z zakresu ekstrakcji informacji, np. powiązania pomiędzy *osobami* a *instytucjami*, w których zajmują stanowiska, semantyczną relację *całość-część*, relację pomiędzy *osobami* i ich *wytworami*, itp.

W odniesieniu do drugiego zastosowania, tj. do możliwości wykorzystania wiedzy z Cyc w celu pozyskania przykładowych par symboli połączonych relacjami, sytuacja nie jest tak jednoznaczna. W szczególności ograniczenia semantyczne, które nakładane są na argumentu relacji są często zbyt ogólne, aby można na ich podstawie wygenerować wystarczająco specyficzne pary symboli.

Lepsze wyniki daje wykorzystanie fragmentarycznej wiedzy faktograficznej występującej w Cyc, co wymaga jednak dużego wysiłku poznawczego, w celu zidentyfikowanie tych grup faktów, które z jednej strony są na tyle bogate, a z drugiej na tyle ogólne, aby dało się je wykorzystać jako przykłady do nauki określonej relacji semantycznej. Należy zwrócić uwagę, że wiedza szczególnie przydatna w tym kontekście, czyli wiedza dotycząca relacji pomiędzy pojęciami ogólnymi, wyrażana jest w Cyc na dwa sposoby:

1. poprzez predykaty służące do wiązania ze sobą kolekcji, np. `#$internalPowerSourceTypes`, która pozwala określić jakie typy silników występują w różnych urządzeniach; przykładowo predykat ten pozwala wyrazić fakt, że *samochód elektryczny* wyposażony jest w *silnik elektryczny*,

2. poprzez meta-predykat `#$relationAllExists`, który pozwala stwierdzić, że dla określonego predykatu służącego do opisywania wiedzy na poziomie instancji, istnieją pewne ograniczenia występujące na poziomie kolekcji; np. predykat `#$anatomicalParts` wykorzystany jest do powiązania konkret-

Tablica 6.13: Przykłady predykatów bezpośrednio wiążących pojęcia Cyc.

Nazwa predykatu	Opis	Liczba asercji
#$symmetricPhysicalPartTypes	symetryczne fizyczne części obiektu	142
#$typeIngredientTypes	składniki potraw	199
#$geographicalSubRegionTypes	obszary geograficzne	22
#$agentTypeSellsProductType	produkty sprzedawane przez przedsiębiorstwa	1567
#$typeIntendedBehaviorCapable	przewidywane zastosowanie artefaktów	608
#$duties	obowiązki pracowników określonych zawodów	428
#$agentTypeCreatesArtifactType	artefakty tworzone przez ludzi	104

Tablica 6.14: Przykłady predykatów wiążących pojęcia Cyc z pomocą meta-predykatu #$relationAllExists.

Nazwa predykatu	Opis	Liczba asercji
#$anatomicalParts	części ciała	84
#$hasRooms	typy pomieszczeń w budynkach	25
#$performedBy	działania wykonywane przez podmioty	248
#$deviceUsed	narzędzia wykorzystywane w pracy	576
#$objectActedOn	przedmioty poddawane działaniom	3294

nego organizmu z jego narządami, natomiast w połączeniu z predykatem #$relationAllExists, można np. określić, że wszystkie *skorpiony* wyposażone są w *kolec jadowy*.

W tabelach 6.13 oraz 6.14 przedstawiono przykłady ciekawszych predykatów tego rodzaju wraz z liczbą asercji, które występują w ResearchCyc. Natomiast w punkcie 7.1 przedstawiona została metoda pozyskiwania przykładów dla algorytmu automatycznej ekstrakcji relacji.

6.5. Semantyczna baza wiedzy

DBpedia [7, 15] jest jednym z najciekawszych projektów związanych z Semantic Web. Jego celem jest wyekstrahowanie z Wikipedii półstrukturalnych danych oraz przekształcenie ich w bazę wiedzy udostępnianą w formacie RDF [65]. Ponieważ wszystkie uzyskane w ten sposób dane zostały udostępnione pod indywidualnymi adresami URL, DBpedia szybko stała się centralnym elementem *Linked Data* [14] – projektu, mającego na celu udostępnienie ustrukturyzowanych i powiązanych ze sobą danych w formatach pozwalających na ich automatyczne przetwarzanie. W chwili obecnej *Linked Data* obejmuje około 1000 powiązanych ze sobą zbiorów danych obejmujących wiele dziedzin wiedzy (w zbiorach tych dostępne są dane bibliograficzne, biologiczne, medyczne, geograficzne, rządowe oraz związane z sektorem mediów) oraz miliardy faktów.

Moduł ekstrakcji informacji w DBpedii wydobywa z Wikipedii dane na podstawie struktury hiperłączy oraz tak zwanych infoboksów (patrz rys. 6.1) – występujących w niektórych artykułach tabelek obejmujących podstawowe informacje na temat opisywanego obiektu. Mogą to być np. *data urodzenia* i *śmierci* osoby albo *stolica* i *liczba obywateli* określonego kraju.

Każdy infoboks ujmowany jest w podwójne nawiasy klamrowe i zaczyna się od wskazania typu (na rysunku 6.1 jest to Państwo infobox). Po określeniu typu następuje lista wpisów o strukturze *klucz – war-*

```
{{Państwo infobox
 |nazwa_oryginalna       = Rzeczpospolita Polska
 |nazwa_polska           =
 |flaga_obraz            = Flag of Poland.svg
 |godło_obraz            = Herb Polski.svg
...
 |głowa_państwa          = [[Bronisław Komorowski]]
 |głowa_państwa_opis     = [[Prezydent Rzeczypospolitej Polskiej|prezydent RP]]
 |szef_rządu             = [[Donald Tusk]]
 |szef_rządu_opis        = [[Premierzy Polski|prezes Rady Ministrów]]
...
 |powierzchnia           = 312 679 <ref>...</ref>
 |powierzchnia_wód~      =
 |powierzchnia_miejsce   = 70
 |ludność                = 38 485 779<ref>...</ref>
 |gęstość                = 123
 |gęstość_miejsce        = 90
 |ludność_rok            = 2014
...
 |pkb_rok                = 2013
 |pkb_osoba              = 13 394{{r|IMF}} [[Dolar amerykański|USD]]
 |pkb_ppp                = 817,4 mld{{r|IMF}} [[Dolar amerykański|USD]]
...
}}
```

Rysunek 6.1: Przykład *infoboksu* występującego w haśle *Polska* w polskiej Wikipedii.

tość. Wpisy znajdują się w kolejnych liniach tekstu, a klucz od wartości oddzielony jest znakiem równości (np. `nazwa_oryginalna` i `Rzeczpospolita Polska` stanowią jedną z par przedstawionych na rysunku 6.1). Ważną cechą infoboksów jest to, że wartości mogą mieć prostą strukturę napisu, bądź strukturę złożoną. W najprostszym przypadku wartości złożone mogą odnosić się do innych artykułów w Wikipedii (wtedy ujęte są w podwójne nawiasy kwadratowe), mogą być również uzupełnione źródłem z którego pochodzi dana informacja (znacznik `<ref>`), wartości liczbowe mogą posiadać jednostkę, a czasami wartością jest kolejny, zagnieżdżony infoboks.

W pierwotnej wersji DBpedii dane z infoboksów ekstrahowane były w sposób całkowicie automatyczny, tzn. każda para *klucz – wartość* skutkowała utworzeniem faktu w formalizmie RDF. Szybko jednak okazało się, że dane w tej surowej postaci są mało przydatne, gdyż posiadając np. informację o powierzchni porównywanych krajów nie możemy mieć pewności, że jest ona wyrażona w tych samych jednostkach. Co więcej – ze względu na sposób w jaki tworzona jest Wikipedia, tzn. możliwość dodawania informacji przez każdego, kto ma dostęp do Internetu – w nazewnictwie kluczy wykorzystywanych w infoboksach panuje duża niespójność. Bardzo popularne cechy, takie jak np. *data urodzenia* opisywane są za pomocą kilku różnych kluczy.

Ten brak spójności ekstrahowanych danych zdecydował o tym, że obok surowych danych, w obecnej wersji DBpedii ekstrahowane są również dane przetworzone. Założeniem jest ekstrahowanie spójnych danych, które zgodne są z odgórnie przyjętym schematem (ontologią DBpedii) oraz uzupełnionych o jed-

nostki. Za pomocą szablonów mapujących[1] wolontariusze określają jak typy infoboksów mapują się na klasy ontologii DBpedii oraz jak poszczególne klucze mapują się na atrybuty tej ontologii. Dodatkowo możliwe jest wprowadzanie przekształceń pozwalających na określanie jednostek, itp. Ze względu na międzynarodowy charakter tego przedsięwzięcia, do wspólnego schematu mapowane są infoboksy z różnych wersji językowych Wikipedii. Dzięki temu dane zgromadzone w ulepszonej wersji DBpedii są wyższej jakości i zawierają mniej błędów. Ceną jaką trzeba za to zapłacić, jest jednak znacznie mniejsza liczba ekstrahowanych danych.

Dane DBpedii dostępne są jako indywidualne strony internetowe[2] oraz jako zbiory danych do pobrania[3]. W algorytmie ekstrakcji dane te wykorzystywane są na dwa sposoby. Po pierwsze, informacje o typie infoboksów pozwalają na określenie semantycznej kategorii symboli językowych tworzonych na podstawie artykułów Wikipedii (porównaj p. 7.2). Po drugie, możliwe jest również wykorzystanie informacji o połączeniach *klucz – wartość*, w celu automatycznego określenia ograniczeń relacji semantycznych, co stanowi istotną innowację w zakresie algorytmów ekstrakcji relacji semantycznych.

6.6. Integracja źródeł wiedzy

Bardzo istotnym problemem, który musiał zostać rozwiązany zanim można było przystąpić do opracowania algorytmu ekstrakcji informacji, było zintegrowanie wiedzy dostępnej w prezentowanych wcześniej zasobach. Ze względu na ich wielkość, która w przypadku słowników fleksyjnych obejmuje miliony form tekstowych, w przypadku słownika semantycznego obejmuje setki tysięcy pojęć, a w przypadku ontologii dziesiątki tysięcy pojęć i miliony asercji, integracja ta nie mogła być realizowana w sposób ręczny. Konieczne było opracowanie dodatkowych algorytmów integracji wiedzy, mając na uwadze fakt, że pełna integracja oraz przetestowanie otrzymanego zasobu nie jest możliwe.

Podstawowy problem w integracji dotyczył wykorzystywanych języków programowania oraz sposobów reprezentacji informacji, a także sposobów identyfikacji opisywanych obiektów. Biorąc pod uwagę fakt, że praktycznie każde źródło wiedzy udostępniało interfejs napisany w innym języku programowania i każde źródło stosowało inny sposób przechowywania danych, autor zdecydował się na wykorzystanie wysokopoziomowego, obiektowego języka Ruby [42] oraz autorskiej, obiektowej bazy danych Ruby Object Database (w skrócie ROD) [125] w celu integracji dostępu do danych i ujednolicenia ich reprezentacji.

Pierwszy etap integracji polegał na przekształceniu biblioteki CLP w obiektową bazę danych. Dzięki temu zabiegowi dostęp do informacji o leksemach, ich formach tekstowych, a także ich opisach morfologicznych uległ istotnemu uproszczeniu. Następnie dane słownika Morfologik, dla których w słowniku CLP występowały odpowiednie etykiety fleksyjne, zostały zaimportowane do tej samej bazy danych. Dzięki temu powstał spójny słownik fleksyjny zawierający wyrażenia pospolite oraz dużą liczbę nazw własnych.

Dane dostępne w korpusie IPI PAN nie mogą być przekształcone do innej postaci, dlatego jedynym usprawnieniem było zaimplementowanie klienta serwera Poliqarp w języku Ruby[4]. Wyniki uzyskane na podstawie korpusu były przechowywane w relacyjnej bazie danych, wyposażonej w obiektowy interfejs dla języka Ruby – *ActiveRecord*[5]. Z kolei korpus notatek PAP był dostępny wyłącznie w formie tekstowej, dlatego korzystając z zestawu reguł podziału tekstu na zdania oraz reguł podziału zdania na słowa stworzono własny korpus, przekształcony do postaci obiektowej bazy danych. Istotną cechą tak otrzymanego korpusu było to, że do identyfikacji segmentów tekstu korzysta on z tych samych identyfikatorów co biblio-

[1] `http://mappings.dbpedia.org/index.php/Main_Page`
[2] Np. `http://dbpedia.org/page/Poland`
[3] `http://wiki.dbpedia.org/Downloads39`
[4] `https://github.com/apohllo/poliqarpr`
[5] Dobre omówienie biblioteki ActiveRecord znajduje się na stronie `http://guides.rubyonrails.org/`.

teka CLP przekształcona do postaci obiektowej. Można zatem powiedzieć, że oba zasoby zostały w pełni zintegrowane.

Dane wyekstrahowane z Wikipedii za pomocą różnych dostępnych narzędzi (Wikipedia Miner, DBpedia) można było stosunkowo łatwo powiązać, ponieważ każde z nich posługuje się adresem URL odpowiedniego artykułu w Wikipedii jako podstawowym mechanizmem identyfikacji informacji. Należy jednak zwrócić uwagę, że Wikipedia nieustannie ewoluuje, dlatego też niektóre artykuły pojawiają się, a inne zostają usunięte. Aby w pełni zintegrować dostępne dane konieczne jest aby były one wyekstrahowane z tej samej wersji Wikipedii. Ten warunek można spełnić jeśli pobierze się zrzut Wikipedii dostępny na stronie `http://dumps.wikimedia.org/` i na nim uruchomi skrypty ekstrahujące dane.

Dane wyekstrahowane z Wikipedii również zostały umieszczone w obiektowej bazie ROD, zatem z technicznego punktu widzenia mogły być zintegrowane ze słownikiem fleksyjnym. Niestety pełna integracja tych źródeł wiedzy wymagałaby określenia dla wszystkich wieloznacznych form występujących w nazwach pojęć ich paradygmatu fleksyjnego. Zadanie to samo w sobie stanowi dosyć spore wyzwanie, dlatego autor nie podjął się jego rozwiązania. Niemniej jednak wykorzystując algorytm Wikipedia Minera (patrz p. 7.3) możliwe jest uzyskanie dosyć dobrych wyników rozpoznania również odmienionych form wyrazowych w tekście, bez potrzeby ustalenia formy podstawowej dla form odmienionych. Rozwiązanie to bazuje na fakcie, że popularne artykuły w Wikipedii często posiadają wiele odnośników z innych stron, w których treści nazwy tych pojęć występuje w formie odmienionej. Posiadając informację o odnośnikach można zatem częściowo zrekonstruować paradygmat odmiany danego pojęcia, z pominięciem słownika fleksyjnego.

Ostatnim, istotnym zadaniem było powiązanie danych dostępnych w polskiej Wikipedii z pojęciami ontologii Cyc. W sensie logicznym powiązanie to odbyło się poprzez przypisanie artykułom polskiej Wikipedii pojęć ontologii Cyc jako ich kategorii semantycznych (patrz p. 7.2). W ten sposób można było wykorzystać ontologię jako podstawowy zasób taksonomiczny, przydatny w szczególności do weryfikacji ograniczeń semantycznych relacji. W przeciwieństwie do słownika semantycznego, wiedza z ontologii Cyc nie została zapisana w bazie ROD, lecz, podobnie jak korups IPI PAN, dostępna była poprzez wbudowany serwer. W tym celu opracowana została biblioteka o nazwie `Cycr`[6], pozwalająca komunikować się z serwerem z poziomu języka Ruby, umożliwiając wykorzystanie algorytmów zaimplementowanych w Cyc.

6.7. Opis symbolu językowego

W wyniku integracji źródeł wiedzy możliwe było utworzenie słownika zawierającego ustrukturyzowany opis wielu symboli występujących w języku polskim. Słownik ten zawiera w szczególności dane z polskiej Wikipedii, kategorie semantyczne wzięte z taksonomii Cyc, a także fakty zaczerpnięte z DBpedii. Przykładowy symbol przedstawiony jest na rysunku 6.2.

Forma hasłowa (wyróżniona pogrubieniem) symbolu jest tożsama z kanonicznym tytułem artykułu w polskiej Wikipedii. Formy tekstowe pochodzą z nazw odnośników prowadzących do tego artykułu wewnątrz polskiej Wikipedii. Kategorie semantyczne to przypisane przez algorytm klasyfikacji semantycznej pojęcia ontologii Cyc. W przytoczonym przykładzie tylko pierwsza kategoria przypisana jest *explicite*, pozostałe określone są poprzez relację generalizacji (`genls`). Powiązania semantyczne określone są na podstawie innych artykułów Wikipedii zgodnie z miarą pokrewieństwa semantycznego. Relacje natomiast określone są na podstawie informacji umieszczonych w infoboksie w artykule na temat Polski. Ostatni element stanowią tłumaczenia symbolu na inne języki, określone na podstawie odnośników po-

[6] `http://github.com/apohllo/rod`

Polska

formy tekstowe

– Polsce
– polski
– Polski
– Polska
– polska
– Polskę
– ...

kategorie semantyczne

– #$Country
– #$LegalAgent
– #$GeopoliticalEntityOrRegion
– #$GeopoliticalEntity
– #$SpatialThing
– #$CulturalThing
– ...

powiązania semantyczne

– **Podział administracyjny Polski 1975-1998**
– **Województwo mazowieckie**
– **Województwo wielkopolskie**
– **Województwo łódzkie**
– **Warszawa**
– ...

relacje

– **Złoty** [currency]
– **Język polski** [language]
– **Bronisław Komorowski** [leader]
– ...

tłumaczenia

– **Poland** [ang.]
– **Polsko** [cz.]
– **Polen** [duń.]
– **Pologne** [fra.]
– ...

Rysunek 6.2: Przykładowy opis symbolu językowego w słowniku semantycznym powstałym w wyniku integracji źródeł wiedzy.

między różnymi wersjami Wikipedii. Jak widać informacje te są bardzo bogate. Należy jednak zdawać sobie sprawę, że nie każdy symbol językowy posiada tak rozbudowany opis. W pierwszym rzędzie, wynika to z faktu, że algorytm określający semantyczną kategorię nie ma stuprocentowego pokrycia. Ponadto, wiele artykułów nie posiada w swojej treści infoboksu, a zatem nie zawierają opisu relacyjnego. Nie zmienia to jednak faktu, że tak skonstruowany słownik semantyczny jest narzędziem bardzo użytecznym przy ekstrakcji informacji.

7. Algorytmy pomocnicze

7.1. Algorytm wyboru zdań zawierających relacje semantyczne

7.1.1. Metody pozyskiwania przykładowych zdań

Celem 2 oraz 3 punktu głównego algorytmu tworzenia wzorców ekstrakcyjnych (porównaj p. 5.2) jest znalezienie przykładowych zdań, w których występują wyrażenia połączone zadaną relacją semantyczną. Przykłady tego rodzaju mogą być pozyskane na kilka sposobów:

1. przeglądanie korpusu tekstów i ręczne znakowanie zdań zawierających wystąpień par wyrażeń połączonych zadaną relacją,
2. wybór kilku par symboli językowych (danych zarodkowych), wyszukanie ich wystąpień w korpusie tekstów i ręczne oznakowanie zdań zawierających wystąpienia zadanej relacji,
3. wybór kilku par symboli językowych (danych zarodkowych), wyszukanie ich wystąpień w korpusie i automatyczne określenie ich przydatności w konstrukcji wzorców ekstrakcyjnych.

Pierwsze podejście promowane jest w jednym z popularniejszych narzędzi wykorzystywanych do budowy algorytmów ekstrakcji informacji, tj. General Architecture for Text Engeneering (GATE) [31]. Narzędzie to ułatwia przeglądanie zbioru tekstów i znakowanie występowania różnych zjawisk językowych, w tym występowań relacji semantycznych. Tak zbudowany korpus jest następnie analizowany w celu automatycznej konstrukcji wzorców pozwalających na rozpoznawanie określonej relacji. Podejście to jest jednak dość kosztowne, w szczególności jeśli relacja, którą chcemy rozpoznawać w tekście, występuje dość rzadko. Wtedy konieczne jest przeglądnięcie dużej ilości tekstów, a efektywność całego procesu jest dość niska.

Dla języka polskiego nie istnieje żaden korpus, w którym wprost byłyby oznakowane wystąpienia relacji semantycznych. Największy korpus zawierający bogate znakowanie, tj. Narodowy Korpus Języka Polskiego (NKJP) [127] (a dokładniej jego jednomilionowy podkorpus) zawiera dane przydatne z punktu widzenia ekstrakcji informacji jedynie w odniesieniu do ujednoznacznienia sensu wybranych pojęć oraz określenia kategorii semantycznych występujących w nim jednostek referencyjnych. Chociaż dane pozyskane przez szczegółową anotację korpusu byłyby niezwykle cenne, ponieważ pozwoliłyby nie tylko wytrenować odpowiednie modele zdolne do rozpoznawania relacji, ale również pozwoliłyby precyzyjnie określić miarę pokrycia opracowanej metody, nie zastosowano tego podejścia ze względu na jego koszto- i czasochłonność.

Problem ten jest dobrze znany w literaturze z zakresu uczenia maszynowego i nosi nazwę *knowledge acquisition bottleneck* [95, s. 197-201][1]. Dlatego też wiele metod, w szczególności w zakresie ekstrakcji informacji, stara się uniknąć ręcznego znakowania korpusów. Lepszym rozwiązaniem jest podejście

[1] W cytowanej pozycji omawiany w kontekście problemu ujednoznaczniania słów.

przedstawione w punkcie 2 – zamiast przeglądać cały korpus, wyszukiwane są w nim zdania, w których potencjalnie występuje zadana relacja semantyczna. W ten sposób można pozyskać wiele przykładów wystąpienia relacji, gdyż zadanie, które musi wykonać ręcznie osoba przeglądająca korpus, jest znacznie prostsze i polega jedynie na stwierdzeniu czy dana relacja występuje w znalezionym zdaniu.

To podejście stosowane jest m.in. przez Girju w problemie ekstrakcji meronimii dla języka angielskiego [47]. Tzn. po odnalezieniu zdań zawierających pary wyrażeń, o których wiadomo, że zadana relacja występuje pomiędzy nimi, zdania te są ręcznie oznaczane jako zawierające, bądź niezawierające daną relację semantyczną. Ten zbiór stosowany jest następnie (łącznie z przykładami negatywnymi) do wytrenowania klasyfikatorów zdolnych do określenia ograniczeń semantycznych relacji meronimii.

Metoda opisana w punkcie 3 znana jest w literaturze przedmiotu pod nazwą *active learning* [95, s. 199]. Polega ona na ręcznym wyborze kilku charakterystycznych par symboli i wytrenowaniu klasyfikatora zdolnego do rozpoznawania danego zjawiska językowego (np. wystąpienia relacji, bądź ujednoznacznienia sensu). Tak wytrenowany klasyfikator jest następnie używany do znajdowania w korpusie kolejnych przykładów określonego zjawiska. Przykłady, które klasyfikator oznaczył jako poprawne z dużym stopniem pewności, włączane są do początkowego zbioru przykładów uczących. W kolejnej iteracji na podstawie nowego zbioru przykładów trenowany jest nowy klasyfikator, który rozpoznaje zjawisko w dużym korpusie tekstów.

Podejście to jest bardzo popularne w ekstrakcji informacji (patrz p. 4.1.4 oraz p. 4.2.2), gdyż (przynajmniej teoretycznie) pozwala na całkowite pominięcie zaangażowania człowieka w proces konstrukcji wzorców zdolnych do ekstrakcji informacji. Niemniej jednak jakość uzyskanych w ten sposób wzorców ekstrakcyjnych rzadko pozwala na ich praktyczne zastosowanie. Zarówno dla języka angielskiego jak i polskiego ich precyzja oscyluje w granicach 80-85%. Ponadto można je zastosować tylko wtedy, gdy uzyskany wzorzec jest semantycznie wysoce jednoznaczny. Jak zauważa jednak Girju [47], najbardziej popularny wzorzec pozwalający na rozpoznawanie relacji meronimii, czyli związek pomiędzy rzeczownikami, w których argument po lewej stronie występuje w dopełniaczu saksońskim (odpowiadający polskiemu związkowi rządu), to znaczy zakończony jest sekwencją *'s*, nie pozwala na jej jednoznaczne rozpoznanie. Tym samym uniemożliwia zastosowanie tego podejścia dla tej relacji. Dlatego w jej algorytmie przykłady występowania relacji oznaczane są ręcznie.

7.1.2. Koncepcja algorytmu wyszukiwania zdań

Algorytm wyboru par symboli służących do odnajdowania przykładów zdań zawierających wybraną relację, wykorzystywany w niniejszej pracy, najbliższy jest drugiemu podejściu zaprezentowanemu w punkcie 7.1.1. Podstawą prezentowanego algorytmu jest wykorzystanie relacji taksonomicznych, w tym wypadku relacji generalizacji występującej w ontologii Cyc. Przyjmując, że określona relacja występuje pomiędzy symbolami σ_a i σ_b, algorytm automatycznie generuje dodatkowe pary symboli, opierając się na wiedzy o pojęciach będących ich specjalizacjami (pojęciami bardziej specyficznymi, hiponimami).

Podstawową cechą relacji generalizacji (patrz p. 3.3.1) jest dziedziczenie opisu przez specjalizacje danego symbolu. Oznacza to, że jeśli symbol σ_a jest połączony jakąś relacją z symbolem σ_b, to wszystkie specjalizacje σ_a powinny łączyć się za pomocą tej relacji z symbolem σ_b. Podobnie wszystkie specjalizacje symbolu σ_b powinny łączyć się za pomocą tej relacji z symbolem σ_a. Przykładowo jeśli przyjmiemy, że **Naczelny** i **Ramię** połączone są relacją *całość-część*, to będziemy oczekiwali, że wszystkie specjalizacje pojęcia **Naczelny** łączą się za pomocą tej relacji z pojęciem **Ramię**. Tym samym będziemy poszukiwali przykładów występowania par symboli takich jak:

– **Ramię**, **Małpa**

- **Ramię**, **Orangutan**
- **Ramię**, **Osoba**
- **Ramię**, **Matka**
- **Ramię**, **Strażak**

Ponieważ **Ramię** nie posiada specjalizacji, na tej podstawie nie można wygenerować analogicznych przykładów, w których **Ramię** byłoby zastąpione innym symbolem językowym. Pomimo tego na podstawie jednej pary symboli możliwe jest wygenerowanie dziesiątek, a nawet setek różnych par wyrażeń.

Możliwość wygenerowania dużej liczby przykładów, zakłada dostępność wyjściowego zbioru par pojęć połączonych określoną relacją. W prezentowanym podejściu korzysta się z wiedzy zgromadzonej w ontologii Cyc, która zawiera szereg informacji (nie tylko taksonomicznych) na temat powiązań pomiędzy pojęciami. Można by wykorzystać również inne źródło wiedzy zawierającej przykłady relacji semantycznych, np. WordNet albo DBpedię. Najistotniejszą cechą algorytmu jest jednak możliwość automatycznego rozszerzenia bezpośrednio dostępnych przykładów relacji z wykorzystaniem relacji taksonomicznych. Takie podejście pozwala określić częstość występowania wzorców relacji (a zatem zmierzyć ich poprawność), bez wielu iteracji. Tym samym możliwe jest pominięcie ręcznego określania poprawności przykładów odnalezionych w tekście oraz uniknięcie zjawiska dryfu semantycznego [69].

Ogólna struktura algorytmu znajdowania przykładów jest następująca:

- pozyskanie zbioru asercji, w których występują pary symboli połączone zadaną relacją,
- przetłumaczenie symboli ontologii Cyc na język polski,
- wyszukiwanie przykładów w korpusie,
- weryfikacja uzyskanych przykładów występowania relacji.

Najbardziej szczegółowe omówienie tego algorytmu znajduje się w pracy [122]. W punktach 7.1.3-7.1.6 omówiono najważniejsze elementy tego algorytmu, a jego zastosowanie w algorytmie konstrukcji wzorców ekstrakcyjnych znajduje się w punkcie 8.2.

7.1.3. Pozyskiwanie asercji zawierających relację

Istnieje kilka sposobów znalezienia asercji, które zawierają przykłady występowania symboli Cyc połączonych zadaną relacją semantyczną, bądź ontologiczną. Pierwszym, wydawać by się mogło, najbardziej oczywistym sposobem pozyskania takich par, jest znalezienie asercji, w których na pierwszym miejscu występuje predykat reprezentujący relację, tzn. asercji stwierdzających jej wystąpienie. Okazuje się jednak, że tego rodzaju asercje najczęściej stwierdzają występowanie określonej relacji pomiędzy dwoma indywiduami i w praktyce w ontologii Cyc występują one dość rzadko. Dzieje się tak dlatego, że asercje takie wykorzystywane są do opisywania *konkretnych* obiektów. Zatem jeśli mamy predykat `#$anatomicalParts` służący do reprezentacji relacji łączącej określony *organizm* z określoną *częścią jego ciała*, znajdziemy przykłady, w których np. konkretny człowiek (*Bill Clinton*) połączony jest z częścią swojego ciała (np. *nogą*). Jak można się domyślać, ze względu na ogólny charakter tej ontologii, asercji tego rodzaju jest bardzo niewiele, a czasami w ogóle nie występują (w szczególności w wersji OpenCyc), ponieważ ontologia nie zawiera tak szczegółowych opisów indywiduów.

Innym możliwym sposobem pozyskania wyjściowych par symboli jest wykorzystanie informacji o ograniczeniach semantycznych relacji. Przykładowo z predykatem `#$anatomicalParts` stowarzyszone są symbole `#$Organism-Whole` oraz `#$OrganismPart`. Możliwe byłoby zatem wzięcie specjalizacji tych pojęć w celu

odnalezienia przykładów występowania odpowiedniej relacji w tekście. Niestety mało prawdopodobne jest występowanie par symboli takich jak np.:

- **Organizm**, **Ręka**,
- **Organizm**, **Noga**,
- **Organizm**, **Głowa**,
- **Pies**, **Część ciała**,
- **Człowiek**, **Część ciała**,
- **Strażak**, **Część ciała**,

ze względu na ogólny charakter pojęć naukowych takich jak **Organizm** i **Część ciała**, które występują częściej w tekstach encyklopedycznych i naukowych, niż np. w notatkach prasowych.

Najlepszym źródłem informacji na temat połączeń pomiędzy pojęciami okazuje się rodzina predykatów zawierających prefiksy `#$relationAll` oraz `#$relationExist`, w szczególności predykat `#$relationAllExists`. Predykat ten stosowany jest do wyrażania wiedzy na temat możliwych powiązań pomiędzy pojęciami, gdyż asercja postaci `(#$relationAllExists RELATION X Y)` oznacza, że *każdy obiekt typu* `X` *łączy się za pomocą relacji* `RELATION` *z co najmniej jednym obiektem typu* `Y`. Na przykład asercja `(#$relationAllExists #$anatomicalParts #$Scorpion #$Stinger)` oznacza, że każdy `#$Scorpion` (pol. *skorpion*) wyposażony jest (domyślnie) w `#$Stinger` (pol. *kolec jadowy*). Mankamentem tego rozwiązania jest fakt, że nie wszystkie predykaty w Cyc posiadają informacje na temat tego rodzaju połączeń oraz to, że asercje tego rodzaju występują tylko w *ResearchCyc*.

7.1.4. Tłumaczenie pojęć na język polski

Ontologia Cyc pretenduje do zbioru wiedzy, który jest niezależny od języków naturalnych. W tym miejscu nie będziemy podejmować problemu, czy faktycznie możliwe jest zbudowanie takiego zasobu. Nie ulega jednak wątpliwości, że ontologia ta posiada jedynie mapowanie na symbole języka angielskiego. Aby zatem możliwe było wykorzystanie asercji z punktu 7.1.3 do wyszukiwania przykładów w polskich tekstach, konieczne jest aby występujące w nich pojęcia posiadały swoje polskie odpowiedniki.

Zagadnienie tłumaczenia Cyc na język polski podejmowane było przez autora w jego pracy magisterskiej [124] oraz w szeregu prac dotyczących możliwości automatycznego oraz pół-automatycznego przetłumaczenia tej ontologii [117, 120, 121]. Konkluzją tych badań było stwierdzenie, że uzyskanie automatycznego tłumaczenia o zadowalającej jakości jest bardzo trudne, dlatego skoncentrowano się na metodach pół-automatycznych. W tym celu opracowane zostało narzędzie opisane w pracy [121], które dla zadanego pojęcia Cyc proponuje szereg tłumaczeń. Rola tłumacza ogranicza się zatem w wielu wypadkach do wyboru jednego spośród nich. Korzystając z tego narzędzia Perliński pod kierownictwem autora przetłumaczył ponad 15 tys. pojęć ontologii Cyc[2].

W celu zwiększenia pokrycia, poza pojęciami, które występują bezpośrednio w asercjach reprezentujących daną relację, konieczne jest również przetłumaczenie pojęć, które stanowią ich specjalizacje. Dzięki temu możliwe jest wygenerowanie dodatkowych par pojęć połączonych zadaną relacją.

[2]Wyniki tłumaczenia dostępne są pod adresem `http://github.com/apohllo/polish-cyc`

7.1.5. Wyszukiwanie zdań w korpusie

W pewnym uproszczeniu wybór przykładowych zdań przebiega następująco: w korpusie tekstów poszukuje się wystąpień każdego z członów pary z osobna, a następnie weryfikuje się, czy drugi człon również występuje w znalezionym fragmencie tekstu. Zrezygnowano z bardziej oczywistego sposobu wyszukiwania par pojęć, polegającego na tworzeniu zapytań zawierających oba pojęcia. Wynika to przede wszystkim z ograniczeń serwera `Poliqarp`, który wymaga ścisłego określenia kolejności pojęć. Biorąc również pod uwagę fakt, że nazwy pojęć mogą być wielosegmentowe, konieczność generowania wielu zapytań do serwera skutkowałaby bardzo długim czasem przetwarzania.

Istotnym *novum* w stosunku do typowych algorytmów bazujących na parach przykładowych pojęć połączonych zadaną relacją jest to, że zarówno pierwszy jak i drugi argument relacji nie musi występować bezpośrednio w znalezionym fragmencie tekstu. W zależności od konfiguracji eksperymentu, w odniesieniu do argumentu wyszukiwanego w korpusie można żądać, aby w poszukiwanym fragmencie tekstu występował on bezpośrednio lub poszukiwać fragmentów, w których występuje jedna z jego specjalizacji. Natomiast w stosunku do drugiego argumentu można wymagać aby:

1. argument ten występował bezpośrednio w znalezionym fragmencie tekstu,
2. w tekście występował ten argument bądź którakolwiek z jego specjalizacji.

Wracając do wcześniejszego przykładu – dla pary pojęć `#$Primate` – `#$Arm`, w fragmentach tekstów zwróconych dla pojęcia **`Naczelny`** można poszukiwać pojęcia **`Ramię`** i *vice-versa* – w fragmentach tekstów zwróconych dla pojęcia **`Ramię`** poszukiwać pojęcia **`Naczelny`**. Jest to typowy schemat działania, realizujący pierwszy wariant algorytmu.

W drugim wariancie algorytmu, we fragmentach tekstu zwróconych dla pojęcia **`Ramię`** można poszukiwać dowolnego przedstawiciela naczelnych – np. pojęcia **`Człowiek`** i jego dalszych specjalizacji, np. pojęcia **`Matka`**. Dzięki temu na podstawie pary `#$Primate` – `#$Arm` możliwe jest odnalezienie fragmentu tekstu: „płakał rzewnie kryjąc twarz w *ramionach* ***matki***".

7.1.6. Weryfikacja odnalezionych przykładów

Ostatnim istotnym etapem algorytmu wyboru przykładów jest określenie, czy odnaleziony przykład jest pozytywny, czy negatywny. Możliwe są tutaj dwie metody działania – ręczna weryfikacja uzyskanych przykładów (porównaj [47]) oraz weryfikacja automatyczna (na wzór algorytmów opierających się na przykładach zarodkowych).

Aby ułatwić ręczną weryfikację uzyskanych przykładów, stworzona została aplikacja pozwalająca na wykonanie tego zadania niewykwalifikowaniu użytkownicy języka. Zasadniczą jej cechą jest to, że weryfikacja odbywa się z wykorzystaniem pytań formułowanych w języku polskim. Zakładając, że w tekście znaleziono zdanie: „***Drzewa*** są stare, a na *gałęziach* nie było aż takiej ilości owoców" program generuje następujący opis:

– Gałąź jest częścią drzewa.

– Drzewo jest rośliną drzewiastą.

– Gałąź jest zewnętrzną częścią organizmu.

a pod opisem pojawia się pytanie: „Czy opis odpowiada tekstowi?" oraz odpowiedzi: *tak*, *nie*, *nie wiem*. Taka konstrukcja zadania sprawia, że po pierwsze: nie ma wątpliwości, że znaleziony przykład zawiera

napisy odnoszące się do właściwych symboli językowych (tzn. wyrazy nie zostały użyte w innym znaczeniu, niż wynikałoby to z założeń algorytmu), a po drugie sprawia, że odpowiedź użytkownika jest bardziej naturalna.

W celu automatycznego określenia poprawności przykładów konieczne jest wcześniejsze wyodrębnienie z nich wzorców formalnych (patrz p. 8.5). Następnie licząc częstość występowania identycznych wzorców formalnych można odrzucić te wzorce (a z nimi przykłady zdań), które występują sporadycznie (np. raz) albo wzorce, które pojawiały się tylko dla jednej pary pojęć. W ten sposób część unikalnych, pozytywnych przykładów może zostać utracona, ale jakość przykładów jest znacznie wyższa (porównaj p. 9.5).

7.1.7. Skuteczność algorytmy wyboru zdań

Aby w przybliżeniu ocenić skuteczność algorytmu wyboru przykładowych zdań, przeprowadzono eksperymenty na bazie predykatu `anatomicalParts` (szczegółowy opis wykorzystanych pojęć znajduje się w punktach 9.1 oraz 9.2). Algorytm był uruchomiony w czterech wariantach:

1. zapytanie do korpusu na podstawie jednego argumentu relacji, wyszukiwanie w wynikach bezpośredniego wystąpienia drugiego argumentu (`direct-direct`),
2. zapytanie do korpusu na podstawie jednego argumentu relacji, wyszukiwanie w wynikach dowolnej specjalizacji drugiego argumentu (`direct-child`),
3. zapytanie do korpusu na podstawie losowej specjalizacji jednego argumentu, wyszukiwanie w wynikach bezpośredniego wystąpienia drugiego argumentu (`child-direct`),
4. zapytanie do korpusu na podstawie losowej specjalizacji jednego argumentu, wyszukiwanie w wynikach dowolnej specjalizacji drugiego argumentu (`child-child`).

Tak uzyskane wyniki zostały przefiltrowane aby usunąć z nich następujące niepożądane zjawiska:

– powtarzające się przykłady,

– argumenty rozdzielone znakami przestankowymi,

– co najmniej jeden z argumentów nie jest rzeczownikiem.

Następnie z każdego zbioru ręcznie oceniono 100 losowo wybranych zdań.

Tablica 7.1: Wyniki dla różnych wariantów algorytmu wyboru przykładowych zdań. Oznaczenia: c_{total} – liczba przykładów przed filtrowaniem, $c_{filtered}$ – liczba przykładów po filtracji, $q_{direct-direct}$ – wielokrotność wielkości zbioru zawierającego proste dopasowania argumentów (`direct-direct`), Pr – procent przykładów ocenionych jako poprawne (w próbie zawierającej 100 przykładów), d_{avg} – średnia odległość między argumentami (w słowach).

Wariant algorytmu	c_{total}	$c_{filtered}$	$q_{direct-direct}$	Pr[%]	d_{avg}
direct-direct	695	294	1,00	86	1,82
direct-child	6090	2276	7,74	79	1,97
child-direct	310	145	0,49	77	1,89
child-child	3123	877	2,98	66	2,17

Wyniki tego eksperymentu przedstawione są w tabeli 7.1. Dla podstawowego wariantu algorytmu (`direct-direct`) liczba znalezionych przykładów pozostałych po odfiltrowaniu nie jest duża i wynosi niespełna 300. Dla wariantu najlepszego (`direct-child`) jest to prawie 2300 przykładowych zdań – niemal 8 razy więcej. Pozostałe warianty dają znacznie mniej przykładów: `child-direct` jedynie 145, a `child-child` niespełna 900. Jakość otrzymanych wyników przemawia za najprostszym algorytmem, dla którego 86% zdań zawiera faktyczne wystąpienie relacji odpowiadającej predykatowi `#$anatomicalParts`. Warto zwrócić jednak uwagę, że wynik ten nie jest tak wysoki jak można by się spodziewać. Drugi w kolejności jest wariant `direct-child`, w którym 79% wyników jest poprawnych. Pozostałe warianty mają jeszcze niższy udział poprawnych zdań wśród całości wyników (odpowiednio 77% i 66%). Wynika to najprawdopodobniej z większej odległości pomiędzy argumentami w tych zbiorach danych.

Wyniki te pozwalają wysnuć następujące wnioski:

– użycie pierwszego wariantu algorytmu daje znacznie mniej przykładów uczących niż użycie drugiego wariantu,

– wykorzystanie wariantu trzeciego i czwartego nie jest uzasadnione, chyba, że zostaną one potraktowane jako uzupełnienie wariantu pierwszego, bądź drugiego,

– użycie drugiego wariantu algorytmu wydaje się być najlepszym wyborem, gdyż znacznie większa liczba przykładowych zdań, okupiona jest niewielkim spadkiem ich jakości.

Podsumowując – zaprezentowany algorytm wyboru przykładowych zdań (w wariancie drugim, tj. `direct-child`) w istotny sposób przyczynia się do pozyskania dużej liczby przykładów, bez istotnego spadku ich jakości. Może być ona dodatkowo podniesiona, poprzez zastosowanie statystycznej analizy wzorców formalnych uzyskanych na bazie przykładów, a bezwzględnie większa liczba przykładów pozwala uzyskać bardziej pewne i zróżnicowane wzorce formalne.

7.2. Semantyczna klasyfikacja symboli językowych

Zagadnienie semantycznej klasyfikacji symboli językowych również dotyczy wykrywania relacji semantycznej – relacji hiponimii. Algorytm klasyfikacji semantycznej nie jest jednak traktowany na równi z algorytmem głównym, gdyż klasyfikacja ta odbywa się w kontekście specyficznego zasobu, tj. Wikipedii. Ponieważ artykuły Wikipedii posiadają strukturę znacznie bogatszą niż zwykłe teksty, rozpoznawanie tej relacji jest istotnie uproszczone. W szczególności punktem odniesienia nie są pojedyncze napisy, jak to ma miejsce w algorytmie głównym, lecz całe artykuły, które posiadają wiele informacji ułatwiających klasyfikację.

7.2.1. Metody semantycznej klasyfikacji symboli

Istnieje wiele badań poświęconych zagadnieniu klasyfikacji artykułów Wikipedii względem wybranych taksonomii, bądź ontologii. Wśród najpopularniejszych sposobów klasyfikacji można wymienić metody oparte na:

– systemie kategorii Wikipedii,

– infoboksach,

– pierwszych zdaniach artykułów traktowanych jak definicje,

– bezpośrednim mapowaniu artykułów do schematów klasyfikacyjnych.

Pierwsze podejście zostało po raz pierwszy zastosowane przez Suchanka i współpracowników w projekcie YAGO [149, 148, 147]. Uznaje się, że jako pierwszy wskazał on na możliwość wykorzystania kategorii Wikipedii w celu klasyfikacji jej artykułów. Zwrócił on bowiem uwagę na fakt, że kategorie, których syntaktyczna głowa posiada liczbę mnogą, np. *People from Berlin*, zazwyczaj wskazują semantyczną kategorię obiektów, które opisywane są w artykułach należących do danej kategorii.

Drugie podejście eksploatowane jest w DBpedii [7, 15, 103]– każdy infoboks (patrz p. 6.5) posiada swój typ, który również zazwyczaj wskazuje na semantyczną kategorię odpowiedniego obiektu. Np. `Państwo_infobox` w polskiej Wikipedii wykorzystywane jest do opisywania państw. Dzięki temu można przypisać odpowiednią kategorię semantyczną artykułowi zawierającemu ten infoboks.

Kolejne podejście czerpie z prac zainicjowanych już w latach osiemdziesiątych ubiegłego wieku (m.in. [4]), kiedy zaczęto ekstrahować relacje (w szczególności hiponimię) z elektronicznych słowników, wykorzystując do tego proste wzorce określające np. kategorie gramatyczne dopasowywanych słów. Traktując pierwsze zdanie artykułów Wikipedii jak definicję słownikową, można zastosować dość proste metody dające dużą dokładność. Podejście to stosowane jest np. w projekcie Típalo [45], którego celem jest skategoryzowanie zasobów DBpedii z wykorzystaniem angielskiego WordNetu. Również Chrząszcz wykorzystuje tę metodę do określenia kategorii semantycznej wyrażeń wielosegmentowych, wyekstrahowanych z polskiej Wikipedii [23, 24].

Podejście oparte na bezpośrednim mapowaniu artykułów polega na powiązaniu artykułów Wikipedii z pojęciami zdefiniowanymi w ontologii, bądź bazie wiedzy i przypisanie kategorii semantycznych na podstawie tego mapowania. Jest ono wykorzystywane np. w badaniach Legg i współpracowników [80, 81, 138], których celem było znalezienie odpowiedniości pomiędzy pojęciami ontologii Cyc oraz artykułami Wikipedii. Na podstawie takiego mapowania można określić kategorię semantyczną artykułów posiadających swoje odpowiedniki w Cyc, gdyż symbole tej ontologii zawsze włączone są w jej strukturę taksonomiczną.

Każde z tych podejść ma jednak swoje ograniczenia. Metoda oparto o kategorie Wikipedii, przynajmniej w wariancie zastosowanym w YAGO, powoduje, że uzyskane kategorie semantyczne są bardzo specyficzny, a w konsekwencji jest ich bardzo dużo (kilkaset tysięcy). Często jednak nie są to prawdziwe kategorie semantyczne, lecz kombinacje kilku kategorii semantycznych, bądź cech (np. *Amerykańscy pisarze żydowskiego pochodzenia*). Ponieważ kategorie takie nie są sprowadzone do postaci kanonicznej, pomimo tego, że pewne informacje są w nich wyrażone *implicite*, nie są nigdzie reprezentowane bezpośrednio, a w konsekwencji nie można ich wykorzystać. Np. kategorie przypisane artykułowi dotyczącemu *Gertrudy Stein*, wyraźnie wskazują, że była ona Żydówką, ale kategoria semantyczna reprezentująca tę grupę etniczną nie pojawia się w YAGO.

Najważniejszą wadą infoboksów wykorzystywanych przez DBpedię jest to, że znaczna część artykułów jest ich pozbawiona, a zatem nie można na tej podstawie przypisać im żadnej kategorii semantycznej. Ponadto wiele różnych infoboksów wskazuje tę samą kategorię semantyczną (np. `Niemiecki_władca_infobox`, `Polski_władca_infobox`) a nazwy infoboksów w każdej wersji językowej Wikipedii są różne. Dlatego konieczne jest ręczne mapowanie infoboksów na wybraną taksonomię.

Podejście wykorzystujące definicje musi z kolei uporać się z dużą różnorodnością nazw kategorii semantycznych, które pojawiają się w pierwszych zdaniach artykułów. Często trudno jest określić czy przymiotnik stojący przed rzeczownikiem stanowi część nazwy kategorii semantycznej i powinien być zachowany, czy też nie. Konieczne jest również ujednoznacznianie nazw kategorii względem pojęć wybranego schematu taksonomicznego.

Mankamentem ostatniego podejście polegającego na bezpośrednim powiązaniu z wybraną ontologią jest zwykle bardzo niska liczba artykułów, które posiadają swoje odpowiedniki, sięgające co najwyżej

kilkudziesięciu tysięcy (spośród kilku milionów artykułów). W konsekwencji tylko niewielki procent artykułów otrzymuje klasyfikację.

7.2.2. Koncepcja algorytmu klasyfikującego

Przyjmując Cyc jako podstawową ontologię, względem której integrowane są pozostałe źródła wiedzy, nie można pominąć faktu, że posiada ona mapowanie tylko na symbole języka angielskiego. Dlatego też określenie kategorii semantycznych artykułów w *polskiej* Wikipedii na podstawie Cyc, musi oprzeć się na tłumaczeniu z angielskiego na polski. Możliwe byłoby wykorzystanie tłumaczenia opisanego w punkcie 7.1.4, lecz ze względu na jego niekompletność, liczba artykułów, które otrzymałyby klasyfikację byłaby dość niska.

Dlatego też zdecydowano się na wykorzystanie innego podejścia – zasadnicza część procesu klasyfikacji przeprowadzona jest w angielskiej Wikipedii. Klasyfikacja artykułów w polskiej Wikipedii odbywa się dzięki istnieniu powiązań pomiędzy wieloma hasłami występującymi w tych edycjach Wikipedii. Przypisanie kategorii semantycznej w polskiej Wikipedii odbywa się w pierwszej kolejności na podstawie istniejącego powiązania z artykułem angielskim. Jeśli takie powiązanie nie istnieje wykorzystuje się dodatkowe mechanizmy omówione w punkcie 7.2.8.

Podstawowe zadanie algorytmu polega zatem na klasyfikacji haseł angielskiej Wikipedii do taksonomii ontologii Cyc. Biorąc pod uwagę ograniczenia poszczególnych podejść stosowanych do klasyfikacji artykułów, zdecydowano, że zamiast koncentrować się na wykorzystaniu jednego typu informacji, wykorzystane zostaną **wszystkie** wcześniej opisane metody klasyfikowania artykułów. Klasyfikacja konkretnego artykułu uzyskana na podstawie różnych metod może być jednak niespójna, jeśli któraś z metod da niepoprawny wynik. W celu usunięcia tego rodzaju błędnych klasyfikacji wykorzystywany jest mechanizm wykrywania niespójności, zaimplementowany w ontologii Cyc. Omówienie adaptacji przedstawionych metod na potrzeby klasyfikacji, względem Cyc znajduje się w punktach 7.2.3-7.2.8, a także w publikacji [118].

7.2.3. Mapowanie kategorii Wikipedii

Mapowanie kategorii Wikipedii na pojęcia Cyc realizowane jest podobnie jak w YAGO, tzn. kategorie, których syntaktyczna głowa jest rzeczownikiem w liczbie mnogiej, mapowane są na kolekcje występujące w Cyc. Jeśli kategoria zostanie zmapowana, przyjmuje się, że wybrana kolekcja z Cyc określa kategorię semantyczną artykułów należących do tej kategorii. Mapowanie odbywa się z wykorzystaniem wywołania API Cyc `denotation-mapper`, które dla zadanego napisu zwraca zbiór symboli, do których może się on odnosić. Do wywołania przekazywana jest syntaktyczna głowa kategorii, wraz z poprzedzającymi ją modyfikatorami. Jeśli żaden symbol nie zostanie znaleziony, usuwany jest pierwszy modyfikator i ponownie wyszukuje się korespondujące symbole. Procedurę tę powtarza się do momentu, w którym dla zadanej nazwy zbiór odpowiadających jej kandydatów – symboli Cyc – przestanie być pusty. Ponieważ może być on wieloelementowy, konieczne jest ujednoznacznienie nazwy względem symboli Cyc. Odbywa się ono ręcznie.

Przykładowo, w kategorii *Norwegian black metal musical groups* (pol. *norweskie grupy blackmetalowe*), syntaktyczną głowę stanowi rzeczownik *groups*. W pierwszym wywołaniu do Cyc przekazuje się całą nazwę, następnie *black metal musical groups* itd. Cyc zwraca odpowiedź obejmującą całą nazwę wyłącznie dla *musical groups* – jest to symbol `#$MusicPerformanceOrganization`. Ponieważ jest to jedyna możliwa interpretacja, mapowanie pomiędzy kategorią a symbolem Cyc jest ustanawiane automatycznie. Nato-

miast dla kategorii *Pantheon Books books* (pol. *książki (wydane przez) Pantheon Books*) algorytm określa *Books* (a nie *books*) jako głowę syntaktyczną i wyszukuje mapowań dla wyrażeń *Pantheon Books* oraz *Books*. Dla tego drugiego wyrażenia zwracany jest wieloelementowy zbiór symboli Cyc, zawierający m.in. `#$MakingAReservation`, `#$BookCopy` oraz `#$Book-CW`. Ostatni wskazany symbol jest poprawnym mapowaniem, gdyż odnosi się do książki, jako bytu niematerialnego (tzn. do jej treści). Wybór tego mapowania dokonywany jest ręcznie.

Ponieważ zastosowanie zaproponowanego podejścia do mapowania kategorii Wikipedii wymagałoby bardzo dużego nakładu pracy (w najnowszej wersji angielskiej Wikipedii jest ponad milion kategorii), jest ono stosowane wyłącznie do najbardziej ogólnych kategorii – kategorie bardziej specyficzne otrzymują swoje mapowanie dzięki wykorzystaniu wewnętrznej hierarchii kategorii Wikipedii. Mapowanie propagowane jest w dół hierarchii, do wszystkich kategorii, których nazwa zawiera nazwę wcześniej zmapowanej kategorii nadrzędnej. W ten sposób na podstawie pół-automatycznego mapowania 2,5 tysiąca kategorii, możliwe było przypisanie korespondujących symboli Cyc kilkudziesięciu tysiącom kategorii, a w konsekwencji sklasyfikowanie ponad 2 milionów artykułów Wikipedii.

7.2.4. Mapowanie infoboksów na pojęcia Cyc

Mapowanie pomiędzy infoboksami a klasami DBpedii realizowane jest w ramach projektu *DBpedia mapping*[3]. Ponadto ontologia Cyc posiada mapowania na klasy ontologii DBpedii[4]. Dlatego mapowanie infoboksów na pojęcia Cyc w dużej mierze odbyło się automatycznie, dzięki przechodniości relacji mapowania. Np. jeśli `Infobox_book` był zmapowany na klasę **`Book`** w ontologii DBpedii, a ta klasa była zmapowana na symbol `#$Book-CW`, to ten infoboks został zmapowany na symbol `#$Book-CW`. Okazało się jednak, że w wersji Cyc udostępnionej w Internecie brakuje całkiem sporej liczby mapowań – zostały one uzupełnione przez autora i dzięki temu możliwe było zwiększenie pokrycia mapowania: *infoboksy – pojęcia Cyc*, a w konsekwencji zwiększenie pokrycia kategoriami semantycznymi artykułów w Wikipedii.

7.2.5. Wykrywanie kategorii semantycznej w pierwszym zdaniu

Określanie kategorii semantycznej na podstawie pierwszego zdania inspirowane było pracami z zakresu wykrywania relacji hiponimii w definicjach słownikowych [4], pracami Chrząszcza [23, 24], których celem było określenie kategorii semantycznej w polskiej Wikipedii oraz pracami Sarjanta [138], wykorzystywanymi w bezpośrednim mapowaniu artykułów Wikipedii na pojęcia Cyc. Założeniem tego sposobu określania kategorii semantycznej była obserwacja, że zwykle pierwsze zdania ciągłego tekstu Wikipedii (tzn. pomijając treść infoboksów, itp.) podobne jest do definicji słownikowej, tzn. zazwyczaj zawiera czasownik *być* w odpowiedniej formie osobowej, bądź półpauzę, po których następuje wskazanie kategorii semantycznej opisywanego obiektu. Przykładowo pierwsze zdanie artykuły **`Polska`** zaczyna się od słów

> Polska, Rzeczpospolita Polska (RP) – **państwo unitarne** w Europie Środkowej położone między Morzem Bałtyckim na północy a...[5],

gdzie nazwa kategorii semantycznej (`państwo unitarne`) występuje bezpośrednio po półpauzie.

Określenie samego miejsca wystąpienia kategorii semantycznej nie oznacza jednak automatycznie jej przypisania – konieczne jest bowiem znalezienie w Cyc symbolu odpowiadającego tej kategorii. Dlatego przypisanie to odbywa się w trzech krokach: w pierwszym, określane jest prawdopodobne położenie na-

[3] `http://mappings.dbpedia.org/`
[4] Są one dostępne w internetowym serwisie OpenCyc: `http://sw.opencyc.org`
[5] `http://pl.wikipedia.org/wiki/Polska`, dostęp: 25.11.2014, wytłuszczenie autora.

zwy kategorii semantycznej, w drugim, nazwa ta jest ujednoznaczniana względem artykułów Wikipedii, a trzecim, artykuł ten jest mapowany na odpowiedni symbol Cyc.

Określenie położenia odbywa się na podstawie informacji dostarczonej przez stanfordzki tagger języka angielskiego [153]. Potencjalne miejsce wystąpienia kategorii jest lokalizowane tuż za czasownikami *to be* oraz *to refere* i obejmuje ciąg przymiotników i rzeczowników występujących bezpośrednio po tym czasowniku (z opcjonalnym przyimkiem `of`).

Tak określone wyrażenie jest następnie ujednoznaczniane względem artykułów Wikipedii. Zazwyczaj do ujednoznacznienia kategorii semantycznej wykorzystuje się odnośniki występujące w artykule [138, 45]. Powoduje to jednak niepoprawne rozstrzygnięcia w wielu przypadkach, gdyż autorzy artykułów niejednokrotnie tworzą odnośniki dla osobnych wyrażeń, stanowiących składniki nazwy kategorii semantycznej. Przykładowo w angielskiej Wikipedii wyrazy `living` oraz `system` występujące w nazwie kategorii semantycznej `living system` są połączone z artykułami **`Life`** oraz **`System`**, zamiast z artykułem **`Living systems`**.

Dlatego w drugim kroku ujednoznacznianie kategorii semantycznej było realizowane z użyciem algorytmu opisanego w punkcie 7.3. Jako kontekst ujednoznaczniania brane były wszystkie artykuły, do których istniały odnośniki w analizowanym artykule. Po ujednoznacznieniu kategorii semantycznej względem Wikipedii, następowało jej mapowania na odpowiadający jej symbol Cyc. Również tutaj mogła wystąpić wieloznaczność, dlatego stosowany był szereg heurystyk pozwalających na ustalenie najbardziej prawdopodobnego mapowania (szczegóły tych heurystyk omówione są w pracy [118, s. 6-7]).

Przykładowo hasło **`Brazil`** posiada następującą definicję w Wikipedii[6]:

> Brazil, officially the Federative Republic of Brazil, is the **largest country** in both South America and the Latin American region.

Wyrażenie `largest country` jest ujednoznaczniane względem artykułów Wikipedii i wybierany jest artykuł `http://en.wikipedia.org/wiki/Country`. Ten artykuł mapowany jest następnie na symbol Cyc za pomocą wywołania `denotation-mapper`. Zwraca ono następujących kandydatów:

- `#$Country`
- `(#$ResourceWithURIFn "http://dbpedia.org/ontology/country")`
- `#$GeopoliticalEntity`
- `#$MusicalComposition-Country`
- `#$CountryMusic`
- `#$RuralArea`

Dzięki użyciu heurystyk, wybierany jest pierwszy symbol (tj. `#$Country`) i jako efekt końcowy metoda ta przypisuje hasłu **`Brasil`** właśnie tę kategorię semantyczną.

7.2.6. Wykorzystanie bezpośredniego mapowania Wikipedii i Cyc

W metodzie opartej o bezpośrednie mapowanie pomiędzy artykułami Wikipedii i symbolami ontologii Cyc bezpośrednio wykorzystano wyniki prac Sarjanta i współpracowników [138]. Wszystkie artykuły, które posiadają mapowanie na symbole Cyc otrzymują odpowiadające im kategorie semantyczne. Wybór tych kategorii uzależniony jest od kategorii ontologicznej symbolu, na który zmapowany jest dany

[6] `http://en.wikipedia.org/wiki/Brazil`, dostęp: 7.09.2014

artykuł: dla kolekcji – są to kolekcje stanowiące bezpośrednie jej uogólnienie (z wykorzystaniem wywołania `min-genls` w API Cyc), a dla indywiduum – są to kolekcje do których ono bezpośrednio należy (korzystając z wywołania `min-isa`).

7.2.7. Usunięcie niespójności klasyfikacji

Podstawowym celem wykorzystania różnych metod klasyfikacji artykułów Wikipedii jest zwiększenie pokrycia. Oczekuje się, że jeśli jedna metoda nie będzie potrafiła sklasyfikować artykułu, to uda się to z wykorzystaniem innej metody. Ubocznym efektem tego podejścia jest możliwość przypisania wielu, czasem niespójnych, kategorii semantycznych do pojedynczego artykułu. Dlatego istotnym etapem algorytmu klasyfikacji jest usunięcie niespójności, które mogły powstać w wyniku klasyfikacji.

Ontologia Cyc posiada kilka mechanizmów pozwalających na weryfikowanie spójności. Najważniejszym z nich, w kontekście przypisywania wielu kategorii semantycznych, jest mechanizm weryfikujący, czy jedno indywiduum może należeć jednocześnie do dwóch kolekcji. Wywołanie `any-disjoin-collection-pair?` zwróci *prawdę*, jeśli którejkolwiek para kolekcji przekazanych jako jego argument nie może posiadać wspólnych instancji. Można je zatem wykorzystać bezpośrednio do zweryfikowania, czy przypisany zbiór kategorii semantycznych jest spójny.

W sytuacji, w której okaże się, że występują niespójności w semantycznej klasyfikacji artykułu, stosowany jest szereg heurystyk, uzależnionych od tego, jaką metodą określono kategorie przypisane do danego artykułu. Najbardziej niezawodną metodą klasyfikacji jest metoda oparta o infoboksy, następna w kolejności jest metoda oparta o definicje, dalej o bezpośrednie mapowania, a na końcu metoda oparta o kategorie. W każdej sytuacji następuje porównanie wyników metody głównej z metodą opartą o kategorie. Tzn. porównuje się wyniki w następującej kolejności:

– infoboksy względem kategorii,

– definicje względem kategorii,

– bezpośrednie mapowanie względem kategorii.

Spośród kategorii semantycznych określonych na bazie kategorii Wikipedii wybiera się tylko te, które spójne są z daną metodą. W ten sposób można określić wiarygodność mapowań kategorii i na końcu, jeśli dany artykuł posiada jedynie klasyfikację w oparciu o te kategorie, przypisać te, które zostały wcześniej pozytywnie zweryfikowane. Dzięki temu możliwe jest osiągnięcie wysokiej precyzji klasyfikacji.

7.2.8. Przeniesienie klasyfikacji do polskiej Wikipedii

Dotychczas omówione metody klasyfikacji odnosiły się do angielskiej wersji Wikipedii. Zastosowanie tych metod w polskiej Wikipedii jest ograniczone, gdyż pomimo tego, że istnieje tłumaczenie pewnej części symboli Cyc na język polski (porównaj p. 7.1.4), jest ono niewystarczające, aby zrealizować zadanie klasyfikacji zarówno w oparciu o mapowanie kategorii, jak i w oparciu o pierwsze zdania artykułów, traktowanych jako ich definicje.

Przeniesienie klasyfikacji uzyskanej dla angielskiej Wikipedii na polską Wikipedię realizowane jest w oparciu o odnośniki występujące pomiędzy różnymi wersjami językowymi tego samego artykułu, a także odnośniki pomiędzy różnymi wersjami językowymi tej samej kategorii. Wykorzystując powiązania pierwszego rodzaju można przenieść kategoryzację uzyskaną dla angielskiego artykułu bezpośrednio na jego polski odpowiednik. Co prawda zdarzają się tutaj błędy, gdyż nie wszystkie powiązania pomiędzy

Tablica 7.2: Rezultaty weryfikacji klasyfikacji artykułów angielskiej Wikipedii względem Cyc przeprowadzone dla różnych metod. Weryfikacja obejmowała 250 artykułów dla każdej metody i była realizowana przez dwie osoby niezależnie. *Pr* – precyzja klasyfikacji, *Rc* – pokrycie klasyfikacji, *Agr* – zgodność pomiędzy osobami dokonującymi weryfikacji.

Metoda(y)	***Pr***	***Rc***	***Agr***
Infoboksy + kategorie	**97,8**	77,2	**92,5**
Pierwsze zdania + kategorie	93,5	69,4	89,0
Bezpośrednie mapowanie + kategorie	94,0	76,4	86,1
Kategorie	81,9	**80,4**	90,5

różnymi wersjami językowymi są poprawne [99, s. 159], ale mimo to przeważająca większość artykułów uzyskuje w ten sposób poprawną klasyfikację. Np. angielski artykuł o tytule **Autism** jest sklasyfikowany jako `#$CommunicationDisorder` oraz jako `#$NeurologicalDisease`. W polskiej Wikipedii odpowiada mu artykuł **Autyzm dziecięcy**, który dzięki temu powiązaniu otrzymuje poprawną klasyfikację.

Aby zwiększyć pokrycie klasyfikacji w polskiej Wikipedii stosuje się dodatkowo kilka metod. W pierwszym rzędzie możliwe jest wykorzystanie powiązań pomiędzy różnymi wersjami językowymi kategorii Wikipedii. Jeśli jakaś angielska kategoria została zmapowana na symbol Cyc i dodatkowo posiada ona swój polski odpowiednik, to ten odpowiednik również może zostać zmapowany na ten sam symbol Cyc. Tak np. angielska kategoria **Neurological disorder** jest zmapowana na symbol `#$NeurologicalDisease` oraz posiada swój polski odpowiednik o nazwie **Choroby układu nerwowego**. W ten sposób kategoria ta również mapowana jest na symbol `#$NeurologicalDisease`, a artykuły należące do tej kategorii **w polskiej Wikipedii** otrzymują ten symbol jako swoją kategorię semantyczną (dotyczy to wyłącznie artykułów, które nie posiadały klasyfikacji).

Poza tym infoboksy występujące w polskiej Wikipedii zostały ręcznie zmapowane na odpowiadające im symbole Cyc[7]. Dzięki temu możliwe było rozszerzenie klasyfikacji artykułów, w ten sam sposób, w jaki infoboksy były wykorzystywane w angielskiej Wikipedii. Ostatnią metodą było ręczne przypisanie kategorii o nazwach zgodnych ze wzorcami `Urodzeni w ...` oraz `Zmarli w ...` do symbolu `#$Person`, ze względu na bardzo wysoką częstość występowania tej kategorii w Wikipedii i wysoką precyzję tak uzyskanego mapowania.

W wyniku uzupełniania klasyfikacji w polskiej Wikipedii mogło okazać się, że jeden artykuł uzyskał wiele niespójnych kategorii. W tej sytuacji stosowana była identyczna metoda rozstrzygania konfliktów klasyfikacyjnych jak w przypadku angielskiej Wikipedii (porównaj p. 7.2.7).

7.2.9. Skuteczność algorytmu klasyfikującego

Skuteczność klasyfikacji artykułów Wikipedii względem taksonomii Cyc została zweryfikowana w pierwszej kolejności dla angielskiej Wikipedii. W tym celu dla każdej metody wylosowano 250 pojęć i zweryfikowano ręcznie poprawność klasyfikacji. Weryfikacja ta była realizowana niezależnie przez dwie osoby. Wyniki uzyskane przez poszczególne metody przedstawione są w tabeli 7.2.

W wyniku weryfikacji ustalono, że metoda oparta o infoboksy oraz kategorie, ma najwyższą precyzję (blisko 100%) a metoda opierająca się wyłącznie na zweryfikowanych kategoriach ma najniższą precyzję (niewiele ponad 80%). Uzyskano również dość wysoką zgodność pomiędzy osobami dokonującymi weryfikacji rezultatów. Pokrycie poszczególnych metod było nieco gorsze, ale było ono mierzone nie w odniesieniu

[7]Wyniki tego mapowania można pobrać ze strony `https://github.com/apohllo/polish-cyc`, plik *Infobox.txt*.

Tablica 7.3: Liczba artykułów (w tysiącach) z angielskiej Wikipedii sklasyfikowanych poszczególnymi metodami. C_t – całkowita liczba artykułów sklasyfikowanych pierwszą metodą, C_c – liczba artykułów sklasyfikowanych obiema metodami, Δ – liczba artykułów, sklasyfikowanych daną parą metod (metodą), które trafiły do zbioru wynikowego. Całkowita liczba artykułów w tej edycji Wikipedii wynosiła 3,6 miliona.

Metoda(y)	C_t	C_c	Δ
Infoboksy + kategorie	2188	1712	**1471**
Pierwsze zdania + kategorie	406	247	**154**
Bezpośrednie mapowanie + kategorie	35	25	**3**
Kategorie	742	—	**593**
Całkowita liczba klasyfikacji			**2221**

do całej zawartości Wikipedii, lecz w odniesieniu do decyzji podejmowanych przez algorytm. Znaczy to, że osoby weryfikujące otrzymywały do oceny kategorie Cyc, które algorytm uznał za niepoprawne. Jak widać mechanizm rozstrzygania konfliktów działa czasami zbyt restrykcyjnie, gdyż tak mierzone pokrycie wahało się w przedziale 70-80%.

Aby zweryfikować wpływ poszczególnych metod na możliwość sklasyfikowania całej zawartości Wikipedii, obliczono ile artykułów łącznie może zostać skategoryzowane za pomocą każdej metody. Wyniki dotyczące pełnego pokrycia poszczególnych metod przedstawione są w tabeli 7.3. W wyniku wykorzystania kombinacji różnych metod, całkowita liczba artykułów, które mogą być sklasyfikowane, jest istotnie mniejsza od całkowitej liczby artykułów, które można sklasyfikować za pomocą jednej metody. Jest to cena, którą trzeba zapłacić, za lepszą jakość klasyfikacji. Połączenie wyników wszystkich metod dało ponad 2 miliony sklasyfikowanych artykułów, co stanowiło nieco ponad 60% wszystkich artykułów, występujących w tej edycji Wikipedii.

Zastosowanie metod automatycznego przeniesienia klasyfikacji angielskich artykułów na artykuły w języku polskim dało dość zaskakujący rezultat – w polskiej Wikipedii udało się sklasyfikować ponad 800 tys. artykułów, co stanowiło ponad **80%** liczby wszystkich artykułów w tej edycji. Ponadto dla każdego artykuły średnio przypisane zostały **dwie kategorie semantyczne**. Ograniczone badania w zakresie poprawności klasyfikacji (realizowana przez jedną osobę weryfikacja 100 losowo wybranych artykułów, obejmująca 209 klasyfikacji), pokazały również że precyzja tej klasyfikacji jest bardzo wysoka i wynosi ponad **95%**. Choć do wyników tych należy podejść z pewną rezerwą, gdyż zakres weryfikacji był istotnie ograniczony, uznano, że klasyfikacja ta jest na tyle kompletna i precyzyjna, aby można było wykorzystać ją w algorytmie ekstrakcji relacji w języku polskim.

7.3. Ujednoznacznianie sensu wyrażeń w tekście

7.3.1. Metody ujednoznaczniania sensu

Jednym z istotniejszych założeń algorytmu ekstrakcji relacji semantycznych jest to, że określenie występowania danej relacji uzależnione jest od spełnienia ograniczeń semantycznych wzorca służącego do jej wykrywania. Przypisanie kategorii semantycznych wyrażeniom w tekście zazwyczaj odbywa się z wykorzystaniem algorytmów rozpoznających jednostki referencyjne. Dla języka polskiego opracowano i przetestowano całkiem sporo algorytmów rozpoznających te wyrażenia (porównaj p. 4.2.1).

Drugie podejście, prezentowane np. przez Girju w algorytmie rozpoznawania relacji *całość-część* [47],

opiera się na ujednoznacznieniu wyrażeń względem słownika semantycznego, w tym przypadku WordNetu. Na tej podstawie możliwe jest określenie kategorii semantycznej, dzięki wykorzystaniu przechodniości relacji hiponimii (porównaj p. 3.2.5 oraz 3.3.1).

W prezentowanym algorytmie wykorzystuje się ujednoznacznianie wyrażeń względem słownika semantycznego zbudowanego w oparciu o Wikipedię (patrz p. 6.7) i na podstawie wyników algorytmu przedstawionego w punkcie 7.2, przypisywana jest im kategoria semantyczna.

Najważniejszą zaletą takiego podejścia jest fakt, że dzięki temu możliwe jest określenie kategorii semantycznej zarówno jednostek referencyjnych (które często odnoszą się do obiektów opisanych w Wikipedii) jak i rzeczowników pospolitych oraz innych podobnych wyrażeń nominalnych. Tej cechy nie spełnia ani podejście opierające się wyłącznie na WordNecie, gdyż nazwy własne stanowią jedynie niewielki odsetek wyrażeń opisanych w tym słowniku, ani algorytmy wykrywające jednostki referencyjne w tekście, gdyż nie potrafią one wykrywać wyrażeń nominalnych. Ponadto, wykorzystanie Wikipedii jako zasobu odniesienia, pozwala częściowo rozwiązać problem koreferencji, zarówno wewnątrz- jak i między-tekstowej (porównaj p. 2.4.2). A dzięki odnośnikom między różnymi wersjami językowymi artykułów (porównaj p. 6.3.1 oraz 6.7), możliwe jest również, w ograniczonym zakresie, rozpoznawanie koreferencji między tekstami napisanymi w różnych językach. Co prawda prezentowany algorytm nie jest oceniany pod tym względem, ale w przyszłości mógłby zostać zastosowany również w scenariuszu obejmującym wykrywanie relacji w różnych językach.

7.3.2. Koncepcja algorytmu ujednoznaczniania

Problem ujednoznaczniania sensu w kontekście ekstrakcji relacji może być zilustrowany następującym przykładem – analizując zdanie „Spotkanie w Krakowie odbyło się w *Zamku Królewskim*", należy określić jaka jest kategoria semantyczna wyróżnionego wyrażenia. W tym wypadku algorytm powinien określić, że jest to `#$Castle`, czyli symbol w Cyc odpowiadający *zamkowi – budowli*. Ponadto można oczekiwać, że algorytm ujednoznaczniania wskaże również konkretny *zamek królewski*, który opisany jest w polskiej Wikipedii (w nawiasie podana jest kategoria semantyczna z ontologii Cyc):

- **Zamek Królewski w Warszawie** (`#$Castle`),
- **Zamek Królewski na Wawelu** (`#$Castle`),
- **Zamek Królewski w Poznaniu** (`#$Castle`),
- ...

Oczekiwalibyśmy aby algorytm ujednoznaczniania wybrał w tym kontekście pozycję drugą, tj. **Zamek Królewski na Wawelu**, gdyż spotkanie odbywa się w Krakowie. Wybranie tego znaczenia nie jest jednak trywialne, bo wyraz `Wawel` nie pojawia się w przytoczonym zdaniu.

W prezentowanym przypadku rozstrzygnięcie tej wieloznaczności nie musi być konieczne dla ustalenia kategorii semantycznej wskazanego wyrażenia, ale w przypadku ogólnym poszczególne artykuły, do których odnosi się to wyrażenie, mogą posiadać odmienne kategorie semantyczne. Dlatego podstawowym celem algorytmu jest wskazanie konkretnego artykułu Wikipedii, który najlepiej odpowiada wyrażeniu rozpoznanemu w tekście.

Algorytm ujednoznaczniania wyrażeń względem Wikipedii jest zrealizowany w oparciu o koncepcję przedstawioną przez Milnego i Wittena w artykule [87]. Problem ujednoznaczniania sensu jest problemem badawczym, który niewątpliwie zasługuje na osobną pracę doktorską, dlatego zastosowano rozwiązanie

opisywane w literaturze. Autor wprowadził jednak w tym algorytmie szereg ulepszeń, które istotnie poprawiają jego działanie, zarówno dla języka angielskiego, jak i języka polskiego. Szczegóły tych ulepszeń przedstawione są w punktach 7.3.3-7.3.4 oraz w pracy [123].

Koncepcja algorytmu ujednoznaczniania opiera się na dwóch założeniach. Chcąc określić, który artykuł Wikipedii najlepiej pasuje do określonego kontekstu, wykorzystuje się *powiązania symbolu językowego z innymi symbolami* oraz *cechy wyrażenia* podlegającego ujednoznacznieniu. Obie własności określane są w oparciu o Wikipedię i przekształcane są w jeden wektor cech. Na podstawie tych cech trenowany jest klasyfikator, dla którego dane uczące również pochodzą z Wikipedii, a którego celem jest wskazanie symbolu najlepiej odpowiadającego wyrażeniu.

7.3.3. Miara semantycznego pokrewieństwa symboli językowych

Siła powiązania symbolu językowego z innymi symbolami określana jest na podstawie *miary semantycznego pokrewieństwa symboli językowych*, która w pewnym stopniu odpowiada semantycznej relacji pokrewieństwa znaczeń (porównaj p. 3.2.3). Jako podstawowy wyznacznik pokrewieństwa symboli zdefiniowanych w oparciu o artykuły Wikipedii, brane jest współwystępowanie odnośników do odpowiadających im artykułów w innych artykułach Wikipedii.

Przykładowo, jeśli chcemy zbadać podobieństwo semantyczne symboli **Kot** i **Pies**, to w polskiej Wikipedii można znaleźć następujące odnośniki do odpowiednich artykułów:

- artykuły, które posiadają odnośnik do **Pies** ale nie do **Kot**:
  - **Rasy psów**,
  - **Dingo**,
  - **Sputnik**,
  - **Szarik**,
- artykuły, które posiadają odnośnik do **Kot** ale nie do **Pies**:
  - **Siedem życzeń**,
  - **Kocimiętka właściwa**,
  - **Alf (serial telewizyjny)**,
  - **Kot Schrödingera**,
- artykuły, które posiadają odnośnik do **Kot** i **Pies**:
  - **Zwierzęta hodowlane**,
  - **Chów wsobny**,
  - **Wścieklizna**,
  - **Zooterapia**.

Biorąc pod uwagę wyżej określone zbiory, można zdefiniować szereg różnych miar pokrewieństw znaczenia. Milne i Witten przedstawili w artykule [160] wyniki eksperymentów z dwoma miarami semantycznego pokrewieństwa opartymi na:

- mierze $tf \cdot idf$ [75, s. 543-544],
- mierze Normalized Google Distance [27].

Tablica 7.4: 15 artykułów o najwyższej wartości miary pokrewieństwa semantycznego SR_J względem hasła **Warszawa**.

Artykuł	$\boldsymbol{SR_J}$
Kraków	0,5054
Uniwersytet Warszawski	0,4853
Łódź	0,4753
Poznań	0,4740
II wojna światowa	0,4707
Wrocław	0,4595
Powstanie warszawskie	0,4561
1945	0,4449
Lublin	0,4446
Lwów	0,4440
Gdańsk	0,4423
Polska	0,4409
Paryż	0,4327
2007	0,4302
2006	0,4285

Wyniki eksperymentów przedstawionych w artykule [160] w sposób istotny przeważały na korzyść drugiej miary, której definicję zawiera następujące równanie

$$SR_G(\sigma_a, \sigma_b) = max\left\{0, 1 - \frac{log(max(|\boldsymbol{A}|, |\boldsymbol{B}|)) - log(|\boldsymbol{A} \cap \boldsymbol{B}|)}{log(|\boldsymbol{W}|) - log(min(|\boldsymbol{A}|, |\boldsymbol{B}|))}\right\}, \tag{7.1}$$

gdzie:

- $SR(\sigma_a, \sigma_b)$ – miara semantycznego pokrewieństwa symboli językowych σ_a i σ_b zdefiniowanych w oparciu o artykuły a i b,
- $|\boldsymbol{A}|$ – moc zbioru artykułów posiadających odnośniki do artykułu a,
- $|\boldsymbol{A} \cap \boldsymbol{B}|$ – moc zbioru artykułów posiadających odnośniki jednocześnie do a i do b,
- $|\boldsymbol{W}|$ – moc zbioru wszystkich artykułów w Wikipedii.

Autor analizując wyniki otrzymane na podstawie miary SR_G w odniesieniu do polskiej Wikipedii zauważył, że daje ona dość zaskakujące rezultaty (porównaj [123, s. 243-244]). Dlatego przeprowadził eksperymenty z jeszcze jedną miarą – tym razem opartą o miarę Jaccarda [54], zdefiniowaną w równaniu 7.2. Okazało się, że uzyskiwane wyniki są istotnie lepsze, co zaważyło na wykorzystaniu tej miary. Przykładowa lista pojęć najbardziej spokrewnionych z hasłem **Warszawa**, uzyskana na podstawie tej miary, przedstawiona jest w tabeli 7.4.

$$SR_J(\sigma_a, \sigma_b) = \begin{cases} \frac{1}{1 - log\left(\frac{|\boldsymbol{A} \cap \boldsymbol{B}|}{|\boldsymbol{A} \cup \boldsymbol{B}|}\right)} & |\boldsymbol{A} \cap \boldsymbol{B}| > 0 \\ 0 & |\boldsymbol{A} \cap \boldsymbol{B}| = 0 \wedge a \neq b \\ 1 & |\boldsymbol{A} \cap \boldsymbol{B}| = 0 \wedge a = b \end{cases} \tag{7.2}$$

7.3.4. Algorytm ujednoznaczniania

Algorytm ujednoznaczniania wyrażeń względem Wikipedii ma następującą strukturę [87]:

1. Rozpoznanie *wyrażeń jednoznacznych*.
2. Określenie *wagi* wyrażeń jednoznacznych na podstawie:
   - pokrewieństwa semantycznego z pozostałymi wyrażeniami jednoznacznymi,
   - statystycznej częstości wykorzystania tych wyrażeń do tworzenia odnośników do innych artykułów w Wikipedii.
3. Ujednoznacznienie sensu *wyrażeń wieloznacznych* na podstawie cech symboli, z wykorzystaniem algorytmu uczenia maszynowego C4.5.

Wyrażenia jednoznaczne

Krok pierwszy algorytmu przebiega następująco – w tekście wyszukiwane są wyrażenia – zarówno jendo- jak i wielosegmentowe – dla których w Wikipedii zarejestrowano tylko jedno znaczenie. Określenie jednoznaczności wyrażeń odbywa się na podstawie nazw wewnętrznych odnośników Wikipedii. Jeśli określona nazwa zawsze prowadzi do tego samego artykułu, to uznawana jest ona za jednoznaczną. Przykładowo wyrażenie `Skarżysku Kamiennej` jest jednoznaczne, gdyż w polskiej Wikipedii zawsze odnosi się do artykułu **`Skarżysko Kamienna`**.

Ponadto jeśli w danym fragmencie tekstu można rozpoznać kilka nazw odnośników, które na siebie nachodzą, to pierwszeństwo ma najdłuższy odnośnik występujący najbardziej na lewo – tym sposobem wyrażenia dłuższe preferowane są względem wyrażeń krótszych. Przykładowo jeśli ujednoznaczniany jest to samo wyrażenie, tj. `Skarżysku Kamiennej`, to algorytm pominie wyraz `Skarżysku`, która również wykorzystywana jest jako odnośnik w polskiej Wikipedii, gdyż jest on w całości zawarty w dłuższym wyrażeniu `Skarżysku Kamiennej`.

Określenie wagi wyrażeń

Bezpośrednie wykorzystanie symboli odpowiadających jednoznacznym wyrażeniom do ujednoznacznienia wyrażeń wieloznacznych, posiada jedną istotną wadę – niektóre rozpoznane pojęcia mogą być zupełnie nieistotne w kontekście głównego tematu poruszanego w określonym fragmencie tekstu. Dlatego przypisanie tym pojęciom zróżnicowanych wag powinno przyczynić się do uzyskania lepszych wyników.

Określenie wag pojęć odbywa się na podstawie dwóch cech:

1. *Średniego pokrewieństwa semantycznego* z pozostałymi pojęciami, określonego z wykorzystaniem miary SR_J, tj.
$$\overline{SR}(\sigma_i) = \frac{1}{n-1} \sum_{j=1, i \neq j}^{n} SR_J(\sigma_i, \sigma_j) \,, \tag{7.3}$$
gdzie σ_i to pojęcie odpowiadające wyrażeniu o numerze i[8], a n to liczba jednoznacznych wyrażeń w analizowany tekście.
2. Miary *prawdopodobieństwa odnoszenia się* (*link probability*) – częstości z jaką określone wyrażenie jest wykorzystywane w treści Wikipedii jako wewnętrzny odnośnik, tj.
$$P_{link}(s_i) = \frac{c_{link}(s_i)}{c_{total}(s_i)} \,, \tag{7.4}$$

[8]Ponieważ wyrażenia te są jednoznaczne, odpowiadające im symbole mają przypisane takie same indeksy.

gdzie s_i to wyrażenie o numerze i, $c_{link}(s_i)$ to liczba wystąpień wyrażenia s_i jako wewnętrzny odnośnik w Wikipedii, a $c_{total}(s_i)$ to liczba wszystkich wystąpień wyrażenia s_i w całej treści Wikipedii. Przykładowe wartości tej miary podane są w tabeli 6.12.

Waga każdego pojęcia ustalana jest jako średnia arytmetyczna tych dwóch cech, tzn.

$$W(\sigma_i) = \frac{\overline{SR}(\sigma_i) + P_{link}(s_i)}{2} \tag{7.5}$$

W ten sposób promowane są pojęcia istotne w danym fragmencie tekstu (posiadające wysoką średnią miarę pokrewieństwa semantycznego z pozostałymi pojęciami) oraz pojęcia, które są często wykorzystywane jako odnośniki w treści Wikipedii.

W stosunku do oryginalnego algorytmu, autor wprowadził jedną zmianę, polegającą na tym, że nie tylko wyrażenia jednoznaczne są uwzględniane w tym kroku. Często bowiem zdarza się, w szczególności dla krótkich tekstów, że występuje bardzo mało wyrażeń jednoznacznych. Dlatego uwzględniane są również najbardziej prawdopodobne, dominujące znaczenia wyrażeń *wieloznacznych*, dla których prawdopodobieństwo odnoszenia się do nich wyrażenia występującego w tekście przekracza 0,7 (wartość ta została ustalona empirycznie). W ten sposób zbiór pojęć względem których określane są cechy ujednoznaczniające jest zazwyczaj większy, niż gdyby były to jedynie pojęcia odpowiadające jednoznacznym wyrażeniom.

Cechy ujednoznaczniające

Po określeniu wag wyrażeń jednoznacznych, algorytm przystępuje do ujednoznaczniania wyrażeń wieloznacznych. Wybór ten nie opiera się jednak wyłącznie na podstawie miary semantycznego pokrewieństwa z jednoznacznymi artykułami – realizowany jest na podstawie kilku cech, a prawdopodobieństwo trafności wyboru określane jest z wykorzystaniem drzewa decyzyjnego.

W algorytmie Milnego i Wittena wykorzystywane są następujące cechy:

– *Średnia ważona pokrewieństwa semantycznego* (ang. *relatedness*) symbolu z symbolami odpowiadającymi jednoznacznym wyrażeniem[9],

$$\overline{SR}_w(\sigma_i) = \frac{1}{n}\sum_{j=1}^{n} SR_J(\sigma_i, \sigma_j) * W(\sigma_j) \ , \tag{7.6}$$

– *Prawdopodobieństwo sensu* (ang. *sense probability*), czyli częstość z jaką wyrażenie s_i odnosi się w Wikipedii do symbolu σ_j, tj.

$$P_{sense}(s_i, \sigma_j) = \frac{c_{link}(s_i, \sigma_j)}{c_{link}(s_i)} \ , \tag{7.7}$$

gdzie $c_{link}(s_i, \sigma_j)$ to liczba wystąpień wyrażenia s_i jako odnośnika do artykułu, na podstawie którego został zdefiniowany symbol σ_j. Przykładowe wartości tej miary podane są w tabeli 6.11.

– *„Jakość" kontekstu* (ang. *goodness*) danego wyrażenia, określona jako suma wag symboli odpowiadających jednoznacznym wyrażeniom, tj.

$$G(\boldsymbol{V}) = \sum_{i\ =1}^{n} W(\sigma_i) \ , \tag{7.8}$$

[9]Zakładamy, że symbole odpowiadające wyrażeniom jednoznacznym otrzymują indeksowanie od 1 do n, natomiast pozostałe symbole od $n+1$ do m, gdzie m to liczba wszystkich symboli, do których mogą odnosić się wyrażenia występujące w analizowanym tekście. $1 \leq n \ \leq m$.

gdzie $\boldsymbol{V}$ to zbiór wyrażeń jednoznacznych występujących w analizowanym tekście. Ta miara ma pomóc odróżnić konteksty, w których występuje wiele jednoznacznych wyrażeń, od kontekstów, w których jest ich niewiele.

Do cech stosowanych w pierwotnym algorytmie autor niniejszej pracy dodał następujące cechy:

– *pozycja* symbolu σ_j względem innych symboli, do których może odnosić się wyrażenie s_i, obliczona na podstawie miary $\overline{SR}_w(\sigma_j)$ (*relatedness position*), tj.

$$R_{SR}(s_i, \sigma_j) = \left|\left\{\sigma_k \ : P_{sense}(s_i, \sigma_k) > 0 \wedge \overline{SR}_w(\sigma_k) > \overline{SR}_w(\sigma_j)\right\}\right| \tag{7.9}$$

– *pozycja* symbolu σ_j obliczona względem innych symboli, do których może odnosić się wyrażenie s_i, obliczona na podstawie miary $P_{sense}(s_i, \sigma_j)$ (*sense position*), tj.

$$R_{sense}(\sigma_i) = |\{\sigma_k \ : P_{sense}(s_i, \sigma_k) > 0 \wedge P_{sense}(s_i, \sigma_k) > P_{sense}(s_i, \sigma_j)\}| \tag{7.10}$$

– *prawdopodobieństwo odnoszenia się* ujednoznacznianego wyrażenia, czyli miara $P_{link}(s_i)$.

Pierwsze dwie cechy zostały dodane dlatego, że cechy oparte wyłącznie o miarę średniego pokrewieństwa semantycznego oraz prawdopodobieństwo sensu dają w wyniku wartości rzeczywiste. Algorytm uczenia maszynowego nie jest w stanie utożsamić sytuacji, w których poprawny sens wyrażenia jest np. najbardziej prawdopodobny, lecz posiada inną wartość bezwzględną, wynikającą z odmiennych dystrybucji prawdopodobieństw sensów dla różnych wyrażeń. Rozwiązanie to ma pomóc przezwyciężyć ten problem. Dodanie miary prawdopodobieństwa odnoszenia się wyrażenia, pozwala zaś zróżnicować działanie algorytmu dla wyrażeń, które wykorzystywane są jako odnośnik w Wikipedii z odmienną częstością.

Na podstawie tych cech definiowany jest wektor cech $\hat{d}_{s_i,\sigma_j}$ służący do ujednoznaczniania sensu wyrażenia s_i

$$\hat{d}_{s_i,\sigma_j} = \left(\overline{SR}_w(\sigma_j), P_{sense}(s_i, \sigma_j), G(\boldsymbol{V}), R_{SR}(s_i, \sigma_j), R_{sense}(s_i, \sigma_j), P_{link}(s_i)\right) \ . \tag{7.11}$$

Indukcja drzewa decyzyjnego

Aby dokonać trafnego wyboru na podstawie cech określonych wcześniej, algorytm ujednoznaczniania posługuje się drzewem decyzyjnym. Drzewo to indukowane jest z wykorzystaniem algorytmu C4.5 [131]. Indukcja drzewa decyzyjnego jest algorytmem uczenia maszynowego ze wspomaganiem, dlatego wymaga przykładów uczących. Do wygenerowania przykładów wykorzystuje się ponownie odnośniki wewnątrz artykułów Wikipedii. Jeśli jakieś wyrażenie jest wieloznaczne, tzn. w Wikipedii pojawia się jako odnośnik do różnych artykułów, w każdym swoim wystąpieniu wskazuje ono dokładnie *jeden* sens, który jest właściwy w danym kontekście. Biorąc wszystkie pozostałe sensy tego wyrażenia w tym kontekście jako przykłady negatywne, można wygenerować bardzo dużą liczbę zarówno pozytywnych jak i negatywnych przykładów uczących.

Generowanie przykładów uczących odbywa się w następujący sposób: z artykułów zawierających ustaloną minimalną liczbę odnośników ekstrahowane są pary:

– treść odnośnika – *wyrażenie*, np. „`jądro systemu operacyjnego` charakteryzowało się...”,

– cel odnośnika – *artykuł Wikipedii*, np. **`Jądro systemu`**.

Dla pary (*wyrażenie*,*artykuł Wikipedii*) obliczany jest wektor cech zdefiniowany w równaniu 7.11. Para ta stanowi pozytywny przykład uczący. Negatywne przykłady uczące generowane są na podstawie wszystkich pozostałych artykułów, do których tworzone są odnośniki o tej samej treści.

Tablica 7.5: Przykładowe wektory cech ujednoznaczniających dla wyrażenia `Burowie` występującego w haśle **`Republika Południowej Afryki`** w polskiej Wikipedii. Ostatnia kolumna wskazuje czy przykład jest pozytywny (1), czy negatywny (0).

Hasło	$\overline{SR}_w(\sigma_j)$	$P_{sense}(s_i, \sigma_j)$	$G(\boldsymbol{V})$	$R_{SR}(s_i, \sigma_j)$	$R_{sense}(s_i, \sigma_j)$	$P_{link}(s_i)$	$Pos.$
`Burowie`	`0,316`	`0,926`	`83,936`	`0`	`0`	`0,181`	`1`
`Afrykanerzy`	`0,179`	`0,037`	`83,936`	`1`	`1`	`0,181`	`0`
`Burowo`	`0,002`	`0,037`	`83,936`	`2`	`1`	`0,181`	`0`

Przykładowo artykuł **`Republika Południowej Afryki`** w polskiej Wikipedii zawiera następujący fragment tekstu[10]

> Republika Południowej Afryki (RPA, afr. Republiek van Suid-Afrika, hol. Republiek Zuid-Afrika ang. Republic of South Africa) – państwo na południowym krańcu Afryki. Jego początki to dwie burskie republiki: Transwal i Orania. **Burowie** byli potomkami osadników holenderskich przybyłych tu w XVII wieku...

Wyrażenie `Burowie` w Wikipedii pojawia się jako odnośnik do następujących haseł:

- **`Burowie`**,
- **`Afrykanerzy`**,
- **`Burowo`**.

W haśle **`Republika Południowej Afryki`** pierwsze wymienione znaczenie jest tym, do którego prowadzi odnośnik, zatem stanowi ono (po przekształceniu na wektor cech zdefiniowany w równaniu 7.11) pozytywny przykład uczący dla algorytmu C4.5. Pozostałe hasła, po przekształceniu w wektory cech, stanowią zaś negatywne przykłady uczące. Wartości wektorów cech obliczone dla tych artykułów w tym kontekście przedstawione są w tabeli 7.5.

Ponieważ liczba artykułów w Wikipedii jest bardzo duża, pozyskanie setek tysięcy pozytywnych oraz negatywnych przykładów uczących nie stanowi większego problemu. W oryginalnym eksperymencie [87], autorzy posłużyli się 1 milionem przykładów uczących. W ten sposób można było wytrenować klasyfikator charakteryzujący się wysoką skutecznością.

Rozstrzyganie wieloznaczności polega na odtworzeniu ostatniej kolumny z tabeli 7.5. Ponieważ decyzja podejmowana przez drzewo decyzyjne zbudowane w oparciu o algorytm C4.5 zazwyczaj odbiega od skrajnych wartości $\{0, 1\}$, wybierane jest znaczenie, dla którego klasyfikator zwrócił najwyższą wartość *prawdopodobieństwa ujednoznacznienia* (P_{dg}). Ponadto można określić minimalną wartość prawdopodobieństwa, poniżej której decyzja będzie uznawana za niewiarygodną. W ten sposób algorytm może wstrzymywać się od podjęcia decyzji, jeśli kontekst ujednoznaczniania nie dostarcza wystarczających informacji do podjęcia trafnej decyzji.

7.3.5. Skuteczność algorytmu ujednoznaczniania

Algorytm ujednoznaczniania sensu wyrażeń względem Wikipedii został przetestowany na kilka sposobów. W pierwszej kolejności zbadano różnice jakie występują pomiędzy oryginalnym algorytmem Milnego i Wittena [87], a ulepszeniami zaproponowanymi w pracy [123]. Testy te realizowane były w oparciu

[10] `http://pl.wikipedia.org/wiki/Republika Południowej Afryki`, dostęp 11.09.2014

Tablica 7.6: Skuteczność różnych wariantów algorytmu ujednoznaczniania zmierzona dla przykładów wieloznacznych. Przykłady testowe i treningowe obejmowały artykuły zawierające od 5 do 100 odnośników. Zbiór testowy zawierał kilkaset tysięcy przykładów.

Metoda ujednoznaczniani	**Precyzja** [%]	**Pokrycie** [%]	**F1** [%]
Losowy sens	39,7	26,4	31,7
Losowy sens o $P > 0{,}5\%$	47,0	47,3	47,2
Najczęstszy sens	81,6	82,2	81,9
SR_G	82,5	83,5	83,0
SR_G + dodatkowe cechy	84,9	83,2	84,0
SR_J	85,4	89,8	87,6
SR_J + dodatkowe cechy	**90,4**	**93,0**	**91,7**

o wektory cech wyekstrahowane bezpośrednio z Wikipedii, tzn. w momencie ujednoznaczniania, algorytm dysponował pełną informacją kontekstową (wszystkie pojęcia w kontekście były już ujednoznacznione – wykorzystano w tym celu oryginale treści odnośników w Wikipedii). Innymi słowy, były to dane pozyskane w sposób identyczny jak dane użyte do treningu klasyfikatora C4.5.

W eksperymencie określono kilka wartości odniesienia opierających się na prostych heurystykach:

– wybór losowego sensu,

– wybór losowego sensu, którego prawdopodobieństwo wynosiło co najmniej 0,5% (według miary przedstawionej w równaniu 7.7),

– wybór najbardziej prawdopodobnego sensu.

Wyniki tego eksperymentu przedstawione w tabeli 7.6 wskazują, że algorytm Milnego i Wittena (oznaczony SR_G) dawał rezultaty niewiele lepsze, niż prosta heurystyka wyboru najbardziej prawdopodobnego sensu. Dodanie dodatkowych cech do wektora $\hat{d}_{s_i,\sigma_j}$, spowodowało poprawę miary F_1 o jeden punkt procentowy. Użycie miary SR_J (bez dodatkowych cech) spowodowało bardziej istotną poprawę – o 4,7 punktu procentowego, zaś użycie tej miary i dodatkowych cech – poprawę o 8,7 punktów procentowych. W ten sposób jakość algorytmu, mierzona za pomocą precyzji i pokrycia przekroczyła 90%.

Wyniki te są w istocie bardzo dobre, lecz nie pokazują one faktycznej skuteczności pełnego algorytmu, lecz jedynie jakość wytrenowanego klasyfikatora, który dysponuje bardzo dokładnymi informacjami kontekstowymi, zaczerpniętymi wprost z odnośników występujących w artykułach. Aby zbadać zachowanie algorytmu w bardziej realnym scenariuszu przeprowadzono 3 dodatkowe testy:

– oparte o wektory cech obliczone na bazie odnośników, z uwzględnieniem pojęć jednoznacznych,

– polegające na odtworzeniu odnośników Wikipedii, w sytuacji, w której wszystkie odnośniki zostały usunięte z tekstu,

– przeprowadzone na krótkich notatkach Polskiej Agencji Prasowej, w których decyzja o poprawności ujednoznacznienia podejmowana była przez człowieka.

Pierwszy test jest w zasadzie powtórzeniem scenariusza z pierwszego testu, z tą różnicą, że brano pod uwagę również odnośniki jednoznaczne (w pierwszym teście brano pod uwagę jedynie wyrażenia, które w tekście Wikipedii były używane jako odnośniki do wielu artykułów). Ponadto do treningu klasyfikatora

Tablica 7.7: Skuteczność algorytmu ujednoznaczniania wyrażeń względem artykułów polskiej Wikipedii. Przykłady treningowe obejmowały artykuły zawierające od 5 do 30 odnośników.

Zbiór testowy	**Rozmiar zbioru**	**Precyzja** [%]	**Pokrycie** [%]	**F1** [%]
Wektory cech z Wikipedii	281714	97,0	94,4	96,7
Odtworzenie odnośników	31092	94,1	90,3	92,2
Notatki PAP	500	79,6	–	–

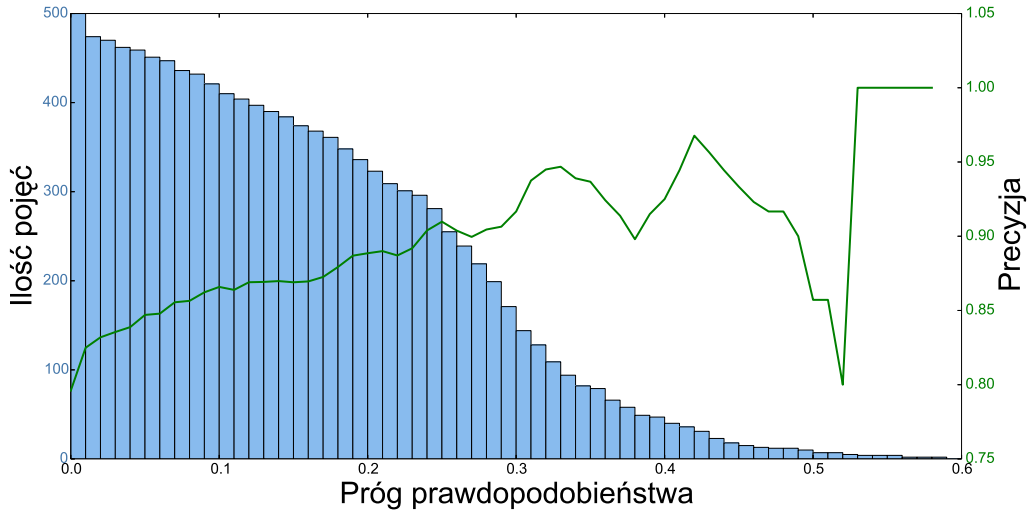
użyto wyłącznie artykułów zawierających od 5 do 30 odnośników (w pierwszej serii eksperymentów maksymalna liczba odnośników wynosiła 100). Powodem pierwszej zmiany jest fakt, że w realnym zadaniu algorytm ma do czynienia z wyrażeniami jednoznacznymi, dlatego ten scenariusz jest punktem odniesienia ewaluacji. Powodem drugiej zmiany był fakt, że algorytm ekstrakcji relacji jest testowany na notatkach Polskiej Agencji Prasowej, które są dość krótkie – przeciętnie zawierają jedynie 4 zdania (patrz tabela 6.3). W pierwszym scenariuszu założenie istnienia nawet 100 ujednoznacznionych pojęć było zdecydowanie zbyt optymistyczne.

Drugi test pokazuje faktyczne zachowanie algorytmu na surowym tekście, bez wykorzystywania żadnych informacji o ujednoznacznionych pojęciach. Z drugiej jednak strony punktem odniesienia jest tutaj nadal tekst Wikipedii, dlatego zadanie to jest zdecydowanie łatwiejsze niż to określone w ostatnim teście, gdyż ujednoznaczniany jest tekst encyklopedyczny – o takiej samej charakterystyce jak tekst, który został użyty do wytrenowania klasyfikatora. Test ten został przeprowadzony, ponieważ wykorzystanie oryginalnego tekstu Wikipedii pozwalało automatycznie zweryfikować jakość algorytmu na dużym zbiorze danych.

Ostatni test jest najbardziej miarodajny pod względem faktycznego zachowania algorytmu na docelowym korpusie tekstów. W tym teście każda wyrażenie, które algorytm uznał za ujednoznacznione, podlegało ręcznej ocenie ze względu na swoją poprawność. Niestety, z tego względu, zbiór testowy był znacznie mniejszy (obejmował tylko 500 ujednoznacznień). Dodatkowo nie określono pokrycia algorytmu, gdyż wymagałoby to znacznie większego nakładu pracy.

Wyniki tych testów przedstawione są w tabeli 7.7. Różnice pomiędzy skutecznością działania algorytmu w poszczególnych scenariuszach są dość istotne. Korzystając z pełnej informacji dostępnej w kontekście ujednoznacznianego wyrażenia można uzyskać bardzo wysoką precyzję i pokrycie, przekraczające 94%. Biorąc pod uwagę wielkość zbioru testowego (ponad 280 tys. ujednoznacznień) wynik ten jest wręcz znakomity. Skuteczność algorytmu istotnie pograsz się, kiedy konieczne jest ujednoznacznienie wielu pojęć jednocześni, tzn. w wariancie, w którym wszystkie odnośniki są usunięte. Niemniej nadal wyniki te są całkiem dobre, gdyż zarówno precyzja, jak i pokrycie przekraczają 90%. Najgorzej algorytm wypada w teście przeprowadzanym na notatkach PAP. Precyzja algorytmu spada poniżej 80%. Jednym z powodów tej sytuacji jest fakt, że wykorzystywane są wszystkie decyzje podejmowane przez algorytm, nawet te, w których decyzja podejmowana przez klasyfikator C4.5 była bardzo niepewna ($P_{dg} = 0$).

Aby ograniczyć ilość niepoprawnych rozpoznań zbadano wpływ miary P_{dg} na jakość wyników. Rysunek 7.1 przedstawia wykresy wpływu minimalnego progu tej miary na precyzję ujednoznaczniania oraz na ilość ujednoznacznionych pojęć. Można zauważyć, że podniesienie progu pozytywnie wpływa na precyzję ujednoznaczniania – przynajmniej w zakresie od 0 do 0.25. Próg ustalony powyżej wartości 0.25 powoduje istotny spadek liczby rozpoznanych pojęć oraz nieprzewidywalność precyzji ujednoznaczniania. Ustalając próg na 0.25 możliwe jest uzyskanie precyzji na poziomie 90% – zbliżonej do wartości uzyskanych dla tekstu Wikipedii. Oznacza to jednak redukcję ilości rozpoznanych pojęć o 50% w stosunku do pierwotnej wersji algorytmu. Podsumowując – ustalając minimalny próg w zakresie 0–0.25 można uzyskać liniową



Rysunek 7.1: Ilość ujednoznacznionych pojęć oraz precyzja ich ujednoznaczniania w notatkach PAP, w zależności od ustalonego progu minimalnego prawdopodobieństwa poprawności ujednoznacznienia P_{dg} określonego przez klasyfikator C4.5.

poprawę precyzji ujednoznaczniania, co skutkuje w przybliżeniu liniowym spadkiem liczby rozpoznanych pojęć, a określenie tej wartości powyżej 0.25 nie wydaje się być empirycznie uzasadnione (przynajmniej w kontekście tekstów z korpusu PAP).

7.4. Automatyczne określanie ograniczeń semantycznych

7.4.1. Metody określania ograniczeń semantycznych

Istotnym problemem, który musiał zostać rozwiązany w odniesieniu do prezentowanego algorytmu było określenie *ograniczeń semantycznych* argumentów ekstrahowanych relacji. Wzorce formalne, które pozyskiwane są ze zdań zawierających pary przykładowych wyrażeń połączonych wybraną relacją semantyczną nie są wystarczająco dokładne, aby tylko na ich podstawie można było stwierdzić jej występowanie. Rozwiązaniem tego problemu jest określenie ograniczeń semantycznych, które stanowią dodatkowy warunek konieczny dla stwierdzenia wystąpienia relacji.

Istnieje kilka sposobów określania ograniczeń semantycznych. Można np. oprzeć się na badaniach językoznawczych i poszukać ograniczeń semantycznych zdefiniowanych w słownikach. Niemniej jednak ograniczenia semantyczne definiowane są raczej dla argumentów czasowników, a nie dowolnych par wyrażeń, np. rzeczowników. Można byłoby również posłużyć się wiedzą zgromadzona w polskim WordNecie – wtedy jednak konieczne byłoby ujednoznacznianie kategorii semantycznych wyrażeń względem tego zasobu. Co więcej należałoby się ograniczyć do relacji zdefiniowanych w tym słowniku. To podejście stosowane jest w pewnym stopniu przez Girju [47] w algorytmie ekstrakcji relacji *całość-część* w języku angielskim.

Ponadto Girju zaproponowała, aby ograniczenia semantyczne były określane na podstawie przykładów występujących w tekście – indywidualnie dla każdego wzorca formalnego. Podejście to polega na wyszukaniu w tekście dopasowań wzorców formalnych i ręcznej ocenie tego, czy mamy do czynienia z zadaną relacją. Ten sposób wymaga jednak sporych nakładów czasowych i finansowych.

Inne rozwiązanie polega na wykorzystaniu istniejących ograniczeń semantycznych. Ontologia Cyc określa ograniczenia semantyczne dla wszystkich zdefiniowanych w niej predykatów. Podobnie ontologia DBpedii również zawiera, dla części występującej w niej predykatów, ograniczenia semantyczne. Należy jednak zwrócić uwagę, że ze względu na założoną uniwersalność tych ontologii, ograniczenia tego rodzaju są mało selektywne, co prowadzi do istotnego pogorszenia wyników ekstrakcji.

W prezentowanym algorytmie analizowane są trzy podejścia do problemu określania ograniczeń semantycznych. W pierwszy rzędzie ograniczenia określane są na podstawie przykładowych zdań, w których ręcznie stwierdzono występowanie, bądź niewystępowanie zadanej relacji. Ponadto wykorzystywane są ograniczenia semantyczne zdefiniowane w ontologii Cyc. Jednak najbardziej innowacyjne podejście polega na wykorzystaniu semantycznej bazy wiedzy, w celu automatycznego wykrycia ograniczeń semantycznych. Pomysł ten opisany jest w punktach 7.4.2-7.4.6.

7.4.2. Koncepcja algorytmu automatycznego określania ograniczeń semantycznych

Pomysł automatycznego określania ograniczeń semantycznych polega na wykorzystaniu wiedzy zgromadzonej w semantycznych bazach wiedzy. Przez semantyczną bazę wiedzy rozumiemy tutaj taką bazę wiedzy, w której opisywane obiekty mają przyporządkowaną kategorię semantyczną, najlepiej należącą do dobrze zdefiniowanej ontologii. Posługując się dużą bazą wiedzy, można automatycznie wykryć ograniczenia semantyczne wybranych relacji, poprzez analizę kategorii semantycznych przypisanych obiektom połączonym tymi relacjami. Jeśli określano relacja semantyczna ma odpowiednio wiele instancji, analiza statystyczna kategorii semantycznych jej argumentów może pozwolić na automatyczne wykrycie, ograniczeń semantycznych.

7.4.3. DBpedia jako źródło ograniczeń semantycznych

Pomysł wykorzystania DBpedii jako źródła ograniczeń semantycznych nie jest zupełnie nowy. W 2013 roku Paulheim i Bizer opublikowali artykuł [103], w którym wykorzystali podobne podejście do określania kategorii semantycznej artykułów, które pozbawione były infoboksu. Analizując statystycznie kategorie semantyczne przypisane artykułom połączonym relacjami wyekstrahowanymi z infoboksów, autorzy mogli przypisać kategorie semantyczne tym artykułom, które pozbawione były infoboksów, ale występowały jako odnośnik w infoboksach innych artykułów.

Przykładowo, w angielskiej Wikipedii artykuł **`University of Grenoble`** pozbawiony jest infoboksu, występuje on jednak jako odnośnik w infoboksie artykuły **`Richard von Weizsäcker`** – konkretnie jako argument predykatu `dbpedia-owl:almaMater` łączącego osoby z uczelniami, które ukończyły. Analizując statystycznie kategorie semantyczne pierwszego i drugiego argumentu tego predykatu, można z wysokim prawdopodobieństwem określić, że pierwszy argument musi być typu `dbpedi-owl:Person`, natomiast drugi typu `dbpedia-owl:University`. Na tej podstawie można przypisać kategorię semantyczną `dbpedia-owl:University` artykułowi **`University of Grenoble`**.

Ta sama metoda może być jednak wykorzystana nie tylko do przypisania kategorii semantycznej jednemu z argumentów, ale również określenia zależności jakie występują pomiędzy różnymi typami obiektów. W przypadku predykatu `dbpedia-owl:almaMater` ograniczenia te są pojedyncze, ale istnieje wiele innych predykatów, takich np. jak `dbpedia-owl:part`, które mogą służyć do opisywania relacji z różnych dziedzin wiedzy. Przyjęcie tylko jednych ograniczeń semantycznych dla wszystkich wystąpień tej relacji, powodowałoby jednak ekstrakcje niskiej jakości. Dlatego znacznie lepszym pomysłem jest określenie naj-

bardziej prawdopodobnych *par* ograniczeń semantycznych. W ten sposób, w obrębie jednej relacji (lub odpowiadającego jej predykatu) można wyróżnić wiele różnych par ograniczeń semantycznych i przypisywać daną relację tylko w przypadku, gdy spełniona jest odpowiednia para ograniczeń.

7.4.4. Algorytm określania ograniczeń semantycznych

Działanie algorytmy automatycznego wykrywania ograniczeń semantycznych na bazie DBpedii jest stosunkowo proste. DBpedia wykorzystuje jako podstawowy mechanizm przechowywania wiedzy standard RDF (patrz p. 3.3.2), zatem wszystkie fakty są reprezentowane jako trójki:

- *podmiot*,
- *predykat*,
- *przedmiot*.

Ponadto jako identyfikatory zasobów DBpedii wykorzystywane są nazwy artykułów Wikipedii, dzięki czemu istnieje jednoznaczne mapowanie pomiędzy stronami Wikipedii a zasobami DBpedii.

W zasadzie można by wykorzystać wszystkie asercje znajdujące się, czy to w polskiej, czy też w angielskiej DBpedii, w celu odkrycia powiązań pomiędzy predykatami i ich ograniczeniami semantycznymi. DBpedia zawiera jednak bardzo dużo faktów dotyczących tylko i wyłącznie etykiet powiązanych z artykułami oraz stwierdzeń dotyczących ich kategorii semantycznej. Co więcej, istotną część faktów stanowią nieprzetworzone informacje wydobyte wprost z infoboksów. Dlatego znacznie lepiej jest ograniczyć się do faktów, w których występują predykaty zdefiniowane w ontologii DBpedii, tzn. predykaty należące do przestrzeni nazw `http://dbpedia.org/ontology`.

Ten zbiór asercji trzeba jednak przefiltrować – pozostawiane są tylko te asercje, w których zarówno podmiot jak i przedmiot posiadają kategorię semantyczną należącą do ontologii Cyc. W tym miejscu wykorzystywane są wyniki algorytmu automatycznej klasyfikacji artykułów angielskiej Wikipedii (patrz p. 7.2) oraz fakt, że istnieje jednoznaczne mapowanie pomiędzy Wikipedią a DBpedią (przynajmniej w obrębie DBpedii powstałej na bazie konkretnej wersji Wikipedii).

W tak otrzymanym zbiorze asercji, każdy podmiot i każde dopełnienie posiada kategorię semantyczną (w wielu przypadkach wiele kategorii) z ontologii Cyc, natomiast predykat pochodzi z ontologii DBpedii. W celu jak najlepszego odzwierciedlenia zależności ujmowanych przez odpowiednie relacje, wykorzystywane są nie tylko wszystkie kategorie przypisywane przez algorytm klasyfikacji artykułów, ale dodatkowo wykorzystuje się informację o *wielokrotności* określonej kategoryzacji. Przykładowo, artykuł **`Michael Jackson`** ma przypisane następujące kategorie w angielskiej Wikipedii:

- `20th-century American` **`singers`**
- `20th-century American male` **`actors`**
- `American disco` **`musicians`**
- `American soul` **`singers`**
- `African-American` **`choreographers`**
- `Drug-related` **`deaths`** `in California`
- `1958` **`births`**
- `2009` **`deaths`**

Tablica 7.8: Częstość kategorii semantycznych (symboli Cyc), do których zaklasyfikowany został artykuł **Michael Jackson** na podstawie angielskiej Wikipedii.

Kategoria semantyczna	**Częstość**
#$Singer	8
#$Musician	6
#$Actor	5
#$Person	4
#$Artist	3
#$BusinessPerson	3
#$Songwriter	2
#$Writer	2
#$Choreographer	1
#$Dancer-Performer	1
#$Philanthropist	1
#$Poet	1
#$Producer	1
#$Tenor-Singer	1
#$Victim-UnfortunatePerson	1
Suma	40

– **Burials** at Forest Lawn Memorial Park (Glendale)

– ...

Na podstawie tego zestawienia widać, że niektóre kategorie się powtarzają, np. kategoria semantyczna **Singer**. W wyniku kompletnej analizy kategorii, do których należy artykuł **Michael Jackson**, powstaje statystyka przedstawiona w tabeli 7.8.

Na podstawie jednego wystąpienia krotki $t = (r, a_1, a_2)$ w DBpedii, gdzie r to predykat z DBpedii, a_1 to jej pierwszy argument, a a_2 to jej drugi argument, tworzony jest iloczyn kartezjański kategorii semantycznych przypisanych do pierwszego argumentu ($\boldsymbol{SC_1}$) oraz drugiego argumentu ($\boldsymbol{SC_2}$). Na podstawie każdej pary kategorii $(sc_i, sc_j), sc_i \in \boldsymbol{SC_1}, sc_j \in \boldsymbol{SC_2}$, określona para ograniczeń semantycznych, odpowiadających kategoriom sc_i i sc_j, otrzymuje część składową *wsparcia* G określoną następująco

$$G(r, sc_i, sc_j, t) = \frac{freq(sc_i, \boldsymbol{SC_1})}{\sum_{sc_k \in \boldsymbol{SC_1}} freq(sc_k, \boldsymbol{SC_1})} * \frac{freq(sc_j, \boldsymbol{SC_2})}{\sum_{sc_k \in \boldsymbol{SC_2}} freq(sc_k, \boldsymbol{SC_2})}, \tag{7.12}$$

gdzie $freq(sc_i, \boldsymbol{SC_j})$ – to częstość (liczba wystąpień) kategorii semantycznej sc_i w zbiorze kategorii semantycznych $\boldsymbol{SC_j}$ określonego pojęcia. Przykładowo w DBpedii występuje następująca krotka

```
(associatedBand, Michael Jackson, The Jackson 5).
```

Michael Jackson posiada kategorie semantyczne przedstawione w tabeli 7.8, natomiast **The Jackson 5** w tabeli 7.9. Na podstawie pierwszej pary kategorii tj. `#$Singer` oraz `#$MusicPerformanceOrganization`, predykat `associatedBand` otrzymuje wsparcie wynoszące $\frac{8}{40} * \frac{8}{12} = \frac{1}{5} * \frac{2}{3} = \frac{2}{15}$, dla pary kategorii `#$Singer` i `#$Group` otrzymuje wparcie wynoszące $\frac{1}{5} * \frac{1}{6} = \frac{1}{30}$, itd. Tabela 7.10 zawiera zestawienie wartości dla wszystkich par ograniczeń semantycznych, określonych na podstawie tej jednej krotki.

Tablica 7.9: Częstość kategorii semantycznych, do których zaklasyfikowany został artykuł **The Jackson 5** na podstawie angielskiej Wikipedii.

Kategoria semantyczna	**Częstość**
#$MusicPerformanceOrganization	8
#$Group	2
#$Band-MusicGroup	1
#$Quintet-MusicalPerformanceGroup	1
Suma	12

Tablica 7.10: Wartości wsparcia przypisane wszystkim parom ograniczeń semantycznych, określonych na podstawie krotki (associatedBand, Michael Jackson, The Jackson 5).

	#$Music-Performance-Organization	#$Group	#$Band-MusicGroup	#$Quintet-Musical-PerformanceGroup
#$Singer	$\frac{2}{15}$	$\frac{1}{30}$	$\frac{1}{60}$	$\frac{1}{60}$
#$Musician	$\frac{1}{10}$	$\frac{1}{40}$	$\frac{1}{80}$	$\frac{1}{80}$
#$Actor	$\frac{1}{12}$	$\frac{1}{48}$	$\frac{1}{96}$	$\frac{1}{96}$
#$Person	$\frac{1}{15}$	$\frac{1}{60}$	$\frac{1}{120}$	$\frac{1}{120}$
#$Artist	$\frac{1}{20}$	$\frac{1}{80}$	$\frac{1}{160}$	$\frac{1}{160}$
#$BusinessPerson	$\frac{1}{20}$	$\frac{1}{80}$	$\frac{1}{160}$	$\frac{1}{160}$
#$Songwriter	$\frac{1}{30}$	$\frac{1}{120}$	$\frac{1}{240}$	$\frac{1}{240}$
#$Writer	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Choreographer	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Dancer-Performer	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Philanthropist	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Poet	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Producer	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Tenor-Singer	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$
#$Victim-Unfortunate-Person	$\frac{1}{60}$	$\frac{1}{240}$	$\frac{1}{480}$	$\frac{1}{480}$

Po określeniu wartości wsparcia dla poszczególnych relacji oraz par ograniczeń semantycznych na podstawie indywidualnych krotek, wartości dla konkretnych par są sumowane

$$G_T(r, sc_1, sc_2) = \sum_{t \in \boldsymbol{T}} G(r, sc_1, sc_2, t) \ , \tag{7.13}$$

gdzie:

– G_T – całkowite wsparcie dla ograniczeń semantycznych sc_1 oraz sc_2 określonych dla relacji r,

– $\boldsymbol{T}$ – zbiór wszystkich krotek.

W ten sposób powstaje miara, wskazująca jak często określona para ograniczeń semantycznych może być przypisana krotkom połączonym daną relacją.

7.4.5. Rozpoznawanie relacji na podstawie ograniczeń semantycznych

Miara G_T przedstawiona w punkcie 7.4.4 nie jest jednak pozbawiona wad. Gdyby chcieć użyć jej bezpośrednio do rozpoznawania relacji, która najlepiej pasuje do określonej pary pojęć, to relacje, które posiadają wiele krotek w DBpedii byłyby faworyzowane, niezależnie od tego, czy określona para ograniczeń semantycznych dobrze je charakteryzuje. W celu wyeliminowania tego zjawiska, konieczne jest zbadanie korelacji jaka występuje pomiędzy określoną parą ograniczeń semantycznych a relacją, do której są one przypisane.

Korelacja występująca pomiędzy ograniczeniami semantycznymi a relacjami, może być określona na wiele sposobów. W niniejszej pracy zastosowano proste rozwiązanie opierające się na regule Bayesa [75]. Jeśli określona para ograniczeń semantycznych pasuje do wielu relacji, wybierana jest relacja, dla której *prawdopodobieństwo warunkowe* wystąpienia tej relacji, pod warunkiem wystąpienia określonych ograniczeń semantycznych, jest najwyższe. Prawdopodobieństwo warunkowe definiowane jest następująco

$$P(x|y) = \frac{P(x \ \cap y)}{P(y)} \ , \tag{7.14}$$

gdzie:

– $P(x \ \cap y)$ – to prawdopodobieństwo współwystąpienia zdarzeń x i y,

– $P(y)$ – to prawdopodobieństwo wystąpienia zdarzenia y.

Dla skończonej liczby obserwacji, korzystając z metody największej wiarygodności (ang. *maximum likelihood method*), obliczane jest ono następująco

$$P(x|y) = \frac{c(x, y)}{c(y)} \ , \tag{7.15}$$

gdzie:

– $c(x, y)$ – liczba obserwacji, w których współwystępowały zdarzenia x i y,

– $c(y)$ – liczba wszystkich obserwacji, w których wystąpiło zdarzenie y.

Wykorzystanie prawdopodobieństwa warunkowego w algorytmie określającym ograniczenia semantyczne, polega na zastąpieniu wartości uzyskanej za pomocą miary G_T, wartością prawdopodobieństwa i wybraniu predykatu, który uzyskuje najwyższe prawdopodobieństwo warunkowe. W tym wypadku zdarzenie x, to wystąpienie określonego predykatu w krotce DBpedii, a zdarzenie y, to wystąpienie określonej

pary ograniczeń semantycznych. Biorąc pod uwagę fakt, że jedna krotka daje zazwyczaj wiele takich par, dla każdej pary ograniczeń sumowana jest tylko część równa wartości miary G_T.

W rezultacie prawdopodobieństwo wystąpienia określonego predykatu, pod warunkiem wystąpienia określonych ograniczeń semantycznych, określone jest następująco

$$P(r|sc_1, sc_2) = \frac{G_T(r, sc_1, sc_2)}{\sum_{r_i \in \boldsymbol{R}} G_T(r_i, sc_1, sc_2)} , \tag{7.16}$$

gdzie $\boldsymbol{R}$ to zbiór wszystkich predykatów. Dzięki temu, że prawdopodobieństwo warunkowe spełnia ogólne założenia prawdopodobieństwa (w szczególności wartość P należy do przedziału $[0, 1]$), możliwe jest porównywanie prawdopodobieństw uzyskiwanych dla różnych predykatów oraz różnych ograniczeń semantycznych i w konsekwencji wybór najbardziej prawdopodobnego predykatu dla zadanej pary pojęć. Osobnym problemem pozostaje utożsamienie predykatów zdefiniowanych w DBpedii, z odpowiadającymi im relacjami semantycznymi. Zagadnienie to omówione jest w punkcie 8.8.3.

7.4.6. Skuteczność algorytmu określania ograniczeń

Ocena skuteczności algorytmu wykrywania ograniczeń semantycznych nie może być dokonana bez zbadania wpływu tak określonych ograniczeń, na skuteczność algorytmu ekstrakcji relacji semantycznych. Dlatego też w tym miejscu nie przedstawiamy takiej oceny. Wyniki zastosowania wzorców ekstrakcyjnych, do ekstrakcji informacji na podstawie tak określonych ograniczeń, przedstawione są w punkcie 10.2.

8. Algorytm tworzenia wzorców ekstrakcyjnych

Wstęp

Podstawowym celem algorytmu ekstrakcji jest rozpoznawanie relacji semantycznych, zachodzących pomiędzy wyrażeniami w tekście. Tym niemniej, istota algorytmu dotyczy konstrukcji *wzorców ekstrakcyjnych*, pozwalających na rozpoznanie tych relacji. Dlatego też opis algorytmu w przeważającej mierze koncentruje się na sposobie konstrukcji wzorców ekstrakcyjnych.

Wzorce ekstrakcyjne zawierają dwa rodzaje cech używanych do rozpoznawania relacji: ograniczenia morfosyntaktyczne oraz ograniczenia semantyczne. Ograniczenia morfosyntaktyczne dotyczą cech morfologicznych wyrazów, takich jak ich kategoria gramatyczna, wartości przypadka, liczby czy rodzaju, cech pozycyjnych, związanych z kolejnością wyrażeń oraz wyrazów, które występują pomiędzy argumentami relacji. Cechy te określane są poprzez wyszukiwanie par wyrażeń połączonych zadaną relacją w tekście. Na podstawie odnalezionych zdań powstają *formalne wzorce ekstrakcyjne*, tzn. wzorce obejmujące cechy morfosyntaktyczne.

Ograniczenia semantyczne określają zaś semantycznych cechy wyrażeń. Ich określenie wymaga znacznie bardziej zaawansowanej analizy, ponieważ te cechy są zdefiniowane w słowniku semantycznym. W wyniku uzupełnienia wzorców formalnych ograniczeniami semantycznymi, powstają kompletne *wzorce ekstrakcyjne*, które są używane do wykrywania zadanej relacji.

Rozpoznanie relacji semantycznych wymaga spełnienia wszystkich ograniczeń morfosyntaktycznych oraz semantycznych charakterystycznych dla określonego wzorca. Dlatego też w trakcie rozpoznawania relacji, konieczne jest powtórzenie analizy morfologicznej oraz semantycznej dla wyrażeń podlegających ekstrakcji. Oznacza to m.in. ujednoznacznienie opisów morfologicznych oraz ujednoznacznianie sensu wyrażeń.

8.1. Wybór ekstrahowanej relacji

W pierwszym etapie algorytmu konieczne jest określenie relacji, dla której ma zostać skonstruowany wzorzec. Ponieważ założeniem algorytmu, w przeciwieństwie do otwartych systemów ekstrakcji relacji (patrz p. 4.1.6), jest to, że zadana relacja zdefiniowana jest w przyjętym schemacie odniesienia, jej wybór nie jest całkowicie dowolny.

National Institute of Standards and Technology (NIST) [96] definiuje zbiór szczególnie istotnych relacji semantycznych, wykorzystywanych w eksperymentach służących do ewaluacji systemów ekstrahujących informacje. Poza zdefiniowaniem głównych relacji, wskazane są również ich podtypy. Przykładowo dla relacji *całość-część* określono następujące podtypy: *wytwór-część wytworu*, *terytorium-część terytorium* oraz *organizacja-część organizacji*. Szczegółowe zestawienie typów relacji definiowanych przez NIST przedstawione jest w tabeli 8.1.

Tablica 8.1: Typy i podtypy relacji semantycznych zdefiniowane przez NIST w [96]. (Przedruk za zgodą NIST. Tłumaczenie autora.)

Typ	**Podtyp**
artefakt	użytkowanie, własność, wynalezienie, produkcja
afiliacja ogólna	obywatelstwo, rezydencja, wyznanie, grupa etniczna, przynależność do organizacji, lokalizacja
metonimia	*brak*
afiliacja w organizacji	zatrudnienie, ufundowanie, własność, studiowanie/bycie absolwentem, przynależność do zespołu sportowego, inwestowanie/posiadanie udziałów, członkostwo
całość-część	część artefaktu, obszar geograficzny, oddział organizacji
związek społeczny	ekonomiczny, społeczny, trwały osobowy
relacja fizyczna	lokalizacja, bliskość

W kontekście zasobów wykorzystywanych w niniejszym algorytmie, wybór relacji do ekstrakcji zależy częściowo od dostępności odpowiednich danych uczących w ontologii Cyc. Niewątpliwie łatwiej jest skonstruować wzorzec ekstrakcyjny, dla relacji wymienionych w tabelach 6.13 oraz 6.14. Ponieważ ontologia Cyc została częściowo zmapowana na ontologię DBpedii, możliwe jest również wykorzystanie danych znajdujących się w tej bazie wiedzy. Zakłada się zatem, że ekstrahowane relacje ograniczają się do tych, dla których w ontologii *ResearchCyc* dostępny jest odpowiednio duży zestaw przykładów uczących. Analiza tej ontologii wskazuje, że relacji tego rodzaju (wraz z podtypami) jest ponad 30.

8.2. Określenie zbioru symboli połączonych relacją

Po wybraniu relacji (dalej nazywać będziemy ją relacją R), która ma być ekstrahowana z tekstów, algorytm wymaga dostarczenia zbioru przykładowych par symboli językowych połączonych tą relacją. W tym miejscu wykorzystywany jest pod-algorytm opisany w punktach 7.1.3-7.1.4.

W pierwszej kolejności wyszukiwane są asercji w ontologii Cyc, w których występuje wybrana relacja. Następnie pojęcia połączone tą relacją tłumaczone są na język polski. Wyszukiwanie asercji odbywa się za pomocą wywołania Cyc `(gather-gaf-arg-index R 1 #$relationAllExists)`. Wywołanie to powoduje znalezienie wszystkich asercji, w których zadana relacja występuje jako pierwszy argument predykatu `#$relationAllExists`. Dwa kolejne argumenty tego predykatu to pojęcia Cyc, które połączone są wybraną relacją. Wszystkie pary pojęć otrzymane w wyniku tego wywołania, zapisywane są w pliku w formacie CSV, który stanowi zbiór par pojęć przykładowych dla relacji R: $\boldsymbol{C_R}$.

Pojęcia Cyc w tym pliku identyfikowane są za pomocą swoich wewnętrznych identyfikatorów (niezrozumiałych dla człowieka). Dodatkowo, opisane są one za pomocą angielskiej nazwy, dzięki czemu przeglądając plik można się zorientować, czy zgromadzone dane faktycznie odpowiadają naszym intuicjom. $\boldsymbol{C_R}$ stanowi zatem zbiór krotek postaci $c_R = (id_1, label_1^{en}, id_2, label_2^{en})$, gdzie id to identyfikator pojęcia, a $label^{en}$ to angielska etykieta pojęcia Cyc. Rysunek 8.1 przedstawia przykładowe wpisy dla predykatu `#$anatomicalParts`.

Ten sposób opisu pojęć wyklucza jednak ich bezpośrednie zastosowanie w algorytmie, którego celem jest ekstrakcja informacji w języku polskim. Konieczne jest przełożenie abstrakcyjnych identyfikatorów Cyc na wyrażenia języka polskiego. W tym celu wykorzystuje się plik `PolishLexicon_multiple.txt`

```
Mx4rIcwFloGUQdeMlsOWYLFB2w  #$HomoSapiens  Mx4rvViGVZwpEbGdrcN5Y29ycA  #$Hand
Mx4rvViXQpwpEbGdrcN5Y29ycA  #$Virus        Mx4rprvNBe_FQFG-HrpcawmC5g  #$Capsid
Mx4rIcwFloGUQdeMlsOWYLFB2w  #$HomoSapiens  Mx4rvViEApwpEbGdrcN5Y29ycA  #$Eyebrow
Mx4rIcwFloGUQdeMlsOWYLFB2w  #$HomoSapiens  Mx4rvVi_f5wpEbGdrcN5Y29ycA  #$Finger
Mx4rIcwFloGUQdeMlsOWYLFB2w  #$HomoSapiens  Mx4rvVjCHZwpEbGdrcN5Y29ycA  #$Toe
```

Rysunek 8.1: Przykładowe krotki w pliku $\boldsymbol{C_R}$, zawierającym pary pojęć ontologii Cyc dla predykatu `#$anatomicalParts`. Pierwsza oraz trzecia kolumna zawiera identyfikatory, a druga oraz ostatnia – angielskie nazwy pojęć.

```
Mx4rwH3pQJwpEbGdrcN5Y29ycA  Adwentyzm Dnia Siódmego  #$SeventhDayAdventistReligion
Mx4rwAgs6ZwpEbGdrcN5Y29ycA  Afrodyta                 #$Venus-TheGoddess
Mx4rvi4gFZwpEbGdrcN5Y29ycA  Afrodyta                 #$Aphrodite-TheGoddess
Mx4rvVjtJ5wpEbGdrcN5Y29ycA  Afryka                   #$ContinentOfAfrica
Mx4rveJ8JpwpEbGdrcN5Y29ycA  Afryka północna         #$NorthernAfrica
Mx4rvujpA5wpEbGdrcN5Y29ycA  Ajurweda                #$Ayurveda
```

Rysunek 8.2: Fragment pliku $\boldsymbol{M^{pl}}$ zawierającego tłumaczenia angielskich symboli Cyc na wyrażenia języka polskiego.

udostępniony pod adresem `https://github.com/apohllo/polish-cyc`, zawierający tłumaczenia identyfikatorów na wyrażenia języka polskiego, który dalej oznaczać będziemy jako $\boldsymbol{M^{pl}}$. Zawiera on krotki postaci $m = (id, label^{pl}, label^{en})$, gdzie $label^{pl}$ to łańcuch znaków stanowiący polskie tłumaczenie pojęcia o identyfikatorze id. Fragment tego pliku przedstawiony jest na rysunku 8.2. W pliku tym podstawowym sposobem identyfikacji symboli Cyc są ich wewnętrzne identyfikatory. Dzięki temu możliwe jest bezpośrednie połączenie go ze zbiorem $\boldsymbol{C_R}$. Ponadto, ten schemat identyfikacyjny jest niezależny od ewentualnych zmian angielskich nazw w ontologii Cyc.

Należy podkreślić, że w algorytmie nie dokonuje się tłumaczenia pliku $\boldsymbol{C_R}$ (patrz rysunek 8.1) na język polski – dzięki temu, jeśli tłumaczenie ontologii Cyc zostanie rozszerzone lub zaktualizowane, możliwe będzie jego wykorzystanie bez konieczności ponownego generowania odpowiedniego zbioru par pojęć.

8.3. Wyszukiwanie par symboli w korpusie

Wykorzystując algorytm opisany w punkcie 7.1.5, realizowane są 4 warianty wyszukiwania przykładowych zdań, zawierających wystąpienia relacji R. W pierwszym kroku, korzystając z serwera *Poliqarp*, w korpusie wyszukiwany jest argument relacji, bądź jedna z jego specjalizacji. W ten sposób tworzone są zbiory zdań $\boldsymbol{S_{R,k,direct}}$ oraz $\boldsymbol{S_{R,k,child}}$, gdzie $k \in \{1, 2\}$ oznacza numer argumentu relacji, na bazie którego utworzono zbiór. W pierwszej parze zbiorów występują zdania w których odnaleziono bezpośrednio wystąpienie (*direct*) jednego z argumentów relacji. W drugiej parze zbiorów występują zdania, w których odnaleziono wystąpienie jednej ze specjalizacji (*child*) jednego z argumentów relacji. Następnie, w każdym zbiorze wyszukiwane są odpowiednio wystąpienia drugiego[1] argumentu relacji, bądź jednej z jego specjalizacji. W wyniku wyszukiwania powstają zbiory zdań $\boldsymbol{S_{R,k,direct-direct}}$ i $\boldsymbol{S_{R,k,direct-child}}$ (po-

[1]W sensie *innego*, tzn. argumentu którego wystąpienie nie było jeszcze wyszukiwane. Nie należy mylić tego z sensem pozycyjnym, tzn. argumentu stojącego na drugiej pozycji w relacji.

wstałe na bazie zbiorów $\boldsymbol{S_{R,k,direct}}$) oraz $\boldsymbol{S_{R,k,child-direct}}$ i $\boldsymbol{S_{R,k,child-child}}$ (powstałe na bazie zbiorów $\boldsymbol{S_{R,k,child}}$).

Utworzenie zbiorów $\boldsymbol{S_{R,k,direct}}$ odbywa się z wykorzystaniem zbiorów $\boldsymbol{C_R}$ oraz $\boldsymbol{M^{pl}}$. Dla każdego identyfikatora id_k, $k \in \{1,2\}$ należącego do krotki t ze zbioru $\boldsymbol{C_R}$ w zbiorze $\boldsymbol{M^{pl}}$ wyszukiwana jest krotka $m = (id, label^{pl}, label^{en})$, taka, że $id_k = id$. W ten sposób znajdowane jest polskie tłumaczenie każdego z argumentów połączonych relacją. Dla każdego argumentu, do serwera *Poliqarp* wysyłane jest zapytanie, którego treść stanowi tłumaczenie $label^{pl}$. Jeśli jest ono wielosegmentowe, dopuszcza się, by między każdą parą segmentów występował jeden segment nieokreślony. Dzięki temu możliwe jest odnalezienie np. wyrażeń *pranie* ***brudnych*** *pieniędzy* na podstawie tłumaczenia *pranie pieniędzy*. Wyszukiwanie odbywa się z wykorzystaniem form podstawowych oraz form tekstowych segmentów, tzn. poprzez zapytanie `[base=segment | orth=segment]`.

Przykładowo, dla krotki `(Mx4rIcwFloGUQdeMlsOWYLFB2w, #$HomoSapiens, Mx4rvViGVZwpEbGdrcN5Y-29ycA, #$Hand)` na podstawie zbioru tłumaczeń $\boldsymbol{M^{pl}}$ generowane są zapytania `[base=człowiek | orth=człowiek]` oraz `[base=dłoń | orth=dłoń]`, które pozwalają na znalezienie w korpusie zdań, zawierających dowolne formy wyrazów *człowiek* oraz *dłoń*. Zdania te trafiają odpowiednio do zbiorów $\boldsymbol{S_{R,1,direct}}$ oraz $\boldsymbol{S_{R,2,direct}}$.

Utworzenie zbiorów $\boldsymbol{S_{R,k,child}}$ przebiega następująco. Dla każdego identyfikatora id_k, $k \in \{1,2\}$ należącego do krotki t ze zbioru $\boldsymbol{C_R}$ wywoływana jest funkcja Cyc `all-specs`. W ten sposób zwracane są wszystkie specjalizacje wyjściowych pojęć dostępne w tej ontologii. Tworzą one zbiory specjalizacji $\boldsymbol{SPEC_k^{en}}$, $k \in \{1,2\}$. Ze zbioru tego usuwany jest wyjściowy symbol id_k, ponieważ stanowi on swoją własną specjalizację. Korzystając ze zbioru $\boldsymbol{M^{pl}}$ analogicznie jak w przypadku bezpośrednich wystąpień argumentów, znajdowane są tłumaczenia pojęć Cyc na język polski. W wyniku tego powstaje zbiór $\boldsymbol{SPEC_k^{pl}}$, $k \in \{1,2\}$, który zawiera jedynie niepuste tłumaczenia pojęć Cyc. Z tego zbioru losowanych jest co najwyżej 10 tłumaczeń i na ich podstawie tworzone są zapytania do serwera *Poliqarp*, analogiczne jak w przypadku bezpośrednich wystąpień argumentów.

Przykładowo, dla pierwszego argumentu krotki `(Mx4rIcwFloGUQdeMlsOWYLFB2w, #$HomoSapiens, Mx4rvViGVZwpEbGdrcN5Y29ycA, #$Hand)` tworzony jest zbiór $\boldsymbol{SPEC_1^{en}}$ zawierający następujące pojęcia Cyc: `#$Examiner`, `#$Surfer`, `#$Sheik`, `#$Sunbather`, `#$Pedestrian`, ... W zbiorze $\boldsymbol{M^{pl}}$ występują następujące tłumaczenia tych pojęć: *egzaminator*, *egzaminatorka*, *surfer*, *szejk*, ..., które trafiają do zbioru $\boldsymbol{SPEC_1^{pl}}$. Z tego zbioru losowanych jest 10 tłumaczeń, na podstawie których tworzone są zapytania, np. `[base=egzaminatorka | orth=egzaminatorka]`, `[base=szejk | orth=szejk]`, itd. Wyniki tych zapytań trafiają do zbioru $\boldsymbol{S_{R,1,child}}$.

Znajdowanie wystąpienia drugiego argumentu również uzależnione jest od tego, czy poszukujemy jego bezpośrednich wystąpień, czy też poszukujemy wystąpień jednej z jego specjalizacji. W pierwszym wypadku w zdaniach s_k należących do zbiorów $\boldsymbol{S_{R,k,type}}$, gdzie $k \in \{1,2\}$, a $type \in \{direct, child\}$, wyszukuje się wystąpienia drugiego argumentu (tj. argumentu o numerze 2 dla zdań s_1 oraz argumentu 1 dla zdań s_2) w ten sposób, że argument ten jest tłumaczony na język polski w oparciu o zbiór $\boldsymbol{M^{pl}}$, na podstawie tłumaczenia tworzone jest wyrażenie regularne zawierające formy podstawowe wszystkich słów występujących w tłumaczeniu i wyrażenie to jest dopasowywane do form podstawowych słów występujących w zdaniu. Wyrażenie to zawiera również kotwice dopasowujące jego początek i koniec do granic słów (`\b`). Jeśli w zdaniu zostanie znalezione dopasowanie, to operację tę powtarza się od miejsca wystąpienia dopasowania, co pozwala znaleźć wielokrotne wystąpienia drugiego argumentu. Zdania, w których zostanie znalezione bezpośrednie wystąpienia drugiego argumentu trafiają do zbioru $\boldsymbol{S_{R,k,type-direct}}$.

Przykładowo, na podstawie pierwszego argumentu krotki `(Mx4rIcwFloGUQdeMlsOWYLFB2w, #$HomoSapiens, Mx4rvViGVZwpEbGdrcN5Y29ycA, #$Hand)` powstało zapytanie `[base=człowiek | orth=człowiek]`,

które zwróciło m.in. zdanie $s_1 =$ ***Ludzie*** *patrzyli na słońce, przysłaniając oczy* ***dłonią***. Zdanie to trafiło do zbioru $\boldsymbol{S_{R,1,direct}}$. Drugim argumentem krotki t jest symbol `#$Hand`, który został przetłumaczony jako *dłoń*. Na podstawie tego tłumaczenia utworzone zostało wyrażenie regularne `/\bdłoń\b/`, które zostało dopasowane do zlematyzowanej formy zdania s_1, tzn. `człowiek patrzeć na słońce, przysłaniać oko` **`dłoń`**. W ten sposób dopasowany został drugi argument relacji i zdanie s_1 trafiło do zbioru $\boldsymbol{S_{R,1,direct-direct}}$.

W drugim przypadku, tj. w sytuacji, w której dopasowywana jest dowolna specjalizacja danego symbolu, zmiana sposobu działania algorytmu sprowadza się do konstrukcji innego wyrażenia regularnego. Wyrażenie to zawiera wszystkie słowa będące tłumaczeniami specjalizacji drugiego dopasowywanego argumentu. Jego konstrukcja przebiega następująco: w pierwszej kolejności za pomocą wywołania Cyc `all-specs` znajdowane są wszystkie symbole Cyc, które stanowią specjalizację tego argumentu. Zatem dla argumentu o identyfikatorze id_k tworzony jest zbiór $\overline{\boldsymbol{SPEC}}_{\boldsymbol{k}}^{\boldsymbol{en}}$. Ponieważ wśród tych specjalizacji występuje również symbol o identyfikatorze id_k, jest on usuwany z tego zbioru. Następnie korzystając ze zbioru tłumaczeń $\boldsymbol{M^{pl}}$ tworzony jest zbiór $\overline{\boldsymbol{SPEC}}_{\boldsymbol{k}}^{\boldsymbol{pl}}$ zawierający wszystkie niepuste tłumaczenia symboli ze zbioru specjalizacji. Te tłumaczenia łączone są znakiem alternatywy `|` oraz otaczane ograniczeniami `\b` – w ten sposób powstaje wyjściowe wyrażenie regularne. Wyrażenie to jest następnie dopasowywane do zlematyzowanej postaci zdania i jeśli zostanie znalezione dopasowanie, trafia ono do zbioru $\boldsymbol{S_{R,i,type-child}}$, gdzie $i \in \{1, 2\}$, a $type \in \{direct, child\}$, a proces ten jest powtarzany od miejsca dopasowania.

Przykładowo, na podstawie bezpośredniego dopasowania drugiego argumentu krotki `(Mx4rIcwFloGUQdeMlsOWYLFB2w, #$HomoSapiens, Mx4rvViGVZwpEbGdrcN5Y29ycA, #$Hand)`, czyli symbolu `#$Hand`, znalezione zostało zdanie zawierające fragment *delikatniejszą, subtelną* ***dłonią*** *kobiety*, które trafia do zbioru $\boldsymbol{S_{R,2,direct}}$. Pierwszy symbol czyli `#$HomoSapiens` posiada wiele specjalizacji, m.in. `#$Examiner`, `#$Surfer`, `#$Sheik` oraz `#$Woman`. Po przetłumaczeniu tych symboli na język polski, tworzone jest złożone wyrażenie regularne `\b(egzaminator|egzaminatorka|surfer|szejk|kobieta|...)\b` i dopasowywane jest ono do zlematyzowanych postaci zdań w zbiorze $\boldsymbol{S_{R,2,direct}}$. Ponieważ przytoczony fragment w zlematyzowanej postaci `delikatny, subtelny dłoń` **`kobieta`** zawiera formę `kobieta` , zdanie to dopasowuje się do skonstruowanego wyrażenia regularnego i rozpoznany zostaje w nim pierwszy argument relacji (czyli `#$HomoSapiens`). W konsekwencji trafia ono do zbioru $\boldsymbol{S_{R,2,direct-child}}$.

8.4. Filtrowanie zdań zawierających argumenty relacji

Zbiory zdań zawierające wystąpienia obu argumentów relacji (czy to w postaci bezpośrednich dopasowań, czy też dopasowań jednej ze specjalizacji), poddawane są filtrowaniu. Ma ono na celu wyeliminowanie następujących błędnych przykładów:

- zdublowanych przykładów,
- przykładów, w których argumenty występują w dwóch różnych zdaniach,
- przykładów, w których co najmniej jeden z argumentów nie spełnia ograniczeń co do klasy gramatycznej.

Filtrowanie to ma na celu podniesienie jakości przykładów, dzięki czemu możliwe będzie zastosowanie automatycznej, statystycznej oceny wzorców uzyskanych na podstawie przykładowych zdań. W pierwszym rzędzie, chodzi o wyeliminowanie powtórzeń, które powodują, że ocena statystyczna staje się mniej wiarygodna. Krok ten może zostać przeprowadzony dopiero do zbudowaniu odpowiednich zbiorów, gdyż serwer *Poliqarp* nie posiada mechanizmu pozwalającego na utożsamienie identycznych przykładów. Nawet

użycie zapytania i indeksu wyniku, jego identyfikacja nie pozwala jednoznacznie stwierdzić, że dwa zdania są różne, gdyż mogły one zostać wygenerowane dla każdego z argumentów z osobna. Wyeliminowanie powtórzeń możliwe jest zatem jedynie poprzez zbadanie treści znalezionych przykładów.

Powtarzające się przykłady eliminowane są w ten sposób, że zdania, które mają trafić do wynikowego zbioru, porównywane są z wszystkimi zdaniami, które już się tam znajdują. Ponieważ dokładne porównywanie nie dałoby wszystkich zdublowanych wyników (każde dopasowanie posiada nieco inny kontekst, ze względu na fakt, że z korpusu tekstów pobierana jest stała liczba segmentów po prawej i lewej stronie dopasowania), po stronie, po której nie występuje drugi dopasowany argument, ze zdań usuwa się skrajne segmenty, w liczbie równej odległości (w segmentach) pomiędzy dopasowanymi argumentami plus liczba segmentów w pierwszym dopasowanym argumencie (czyli tym, dla którego zostało wysłane zapytanie do serwera Poliqarp). Jeśli zdanie, które ma trafić do wynikowego zbioru, zawiera dokładne dopasowanie jednego z tak utworzonych fragmentów tekstu, to jest ono odfiltrowywane.

W odniesieniu do drugiej heurystyki chodzi o wyeliminowanie sytuacji, w których jeden argumenty znajdują się w dwóch różnych zdaniach, również w przypadku, gdy są to zdania połączone np. relacją podrzędności. W tym celu stosuje się prostą heurystykę opartą o dopasowanie następującego wyrażenia regularnego: `[.?!;:,"()-]` do tekstu, który występuje pomiędzy argumentami. Jeśli dopasowanie się powiedzie, przykład jest eliminowany.

Ostatnia heurystyka uzależniona jest od rodzaju relacji, która ma być ekstrahowana z tekstu. Np. w odniesieniu do relacji *całość-część* będziemy oczekiwać, że oba jej człony są rzeczownikami. Jednakże dopasowanie na podstawie formy podstawowej, bądź formy tekstowej nie weryfikuje, czy to założenie jest spełnione. Jeśli zatem dopasowujemy napis `bez`, wśród wyników mogą pojawić się przykłady, w których ten wyraz występuje jako rzeczownik oraz jako przyimek. Heurystyka ta eliminuje zatem przykłady, w których wyraz ten występuje w tej drugiej roli.

8.5. Ekstrakcja wzorców formalnych

Zgromadziwszy duży zbiór przykładowych zdań, zawierających wystąpienia zadanej relacji semantycznej, można przystąpić do budowania wzorców ekstrakcyjnych. W pierwszej kolejności określane są cechy formalne, inaczej nazywane wzorcami formalnymi. Spełnienie ograniczeń morfosyntaktycznych nakładanych przez wzorzec formalny jest warunkiem koniecznym do tego, aby algorytm ekstrakcji relacji dalej przetwarzał określony fragment tekstu.

Określenie wzorca formalnego jest stosunkowo proste, gdyż wymaga jedynie ujednoznacznienia opisów morfosyntaktycznych argumentów relacji, określenie ich wzajemnego położenia oraz rejestracji wyrazów, które występują pomiędzy argumentami. Wzorzec formalny określa następujące cechy:

- *kolejność* argumentów – określa szyk argumentów w tekście względem ich szyku w relacji – `direction`,
- *część mowy* (dla obu argumentów) – `pos_left`, `pos_right`,
- kategoria gramatyczna *liczby* (dla obu argumentów) – `number_left`, `number_right`,
- kategoria gramatyczna *przypadka* (dla obu argumentów) – `case_left`, `case_right`,
- kategoria gramatyczna *rodzaju* (dla obu argumentów) – `gender_left`, `gender_right`,
- *wewnętrzny kontekst*, czyli napis występujący pomiędzy argumentami – `inner_context`.

W sytuacji, w której argument stanowi wyrażenie wielosegmentowe, wewnętrzny kontekst określany jest jako wyrażenie, które nie obejmuje żadnego segmentu argumentu relacji. W przypadku tego rodzaju wyrażeń, cechy morfosyntaktyczne określa się na podstawie syntaktycznej głowy wyrażenia, która ustalana jest heurystycznie jako pierwszy rzeczownik od lewej, np. dla wyrażenia `Polskie` **`Koleje`** `Państwowe` będzie to wyraz `Koleje`.

Wzorce formalne budowane są w postaci par: (*nazwa cechy*, *wartość cechy*). Dopuszcza się aby cecha miała wartość pustą. Budowa wzorców formalnych sprowadza się zatem do przekształcenia przykładowych zdań, w których oba argumenty zostały odpowiednio oznaczone, do postaci par: (*cecha*,*wartość*), w taki sposób, że każdemu zdaniu odpowiada jeden zestaw cech powiązanych z odpowiednimi wartościami. Obok tych informacji, rejestrowane są również napisy stanowiące bezpośrednie dopasowanie argumentów relacji. Nie wchodzą one w skład wzorca formalnego, ale wykorzystywane są w poźniejszym etapie, w celu określenia jego statystycznych własności.

Przykładowo, fragment zdania, w którym występują dwa argumenty relacji *całość-część*: „długie nici pajęczyn, czepiały się żółtych *liści* ***akacji*** stojących pod murem" zostałoby przekształcone we wzorzec formalny przedstawiony w tabeli 8.2.

Tablica 8.2: Cechy formalne wekstrahowane z przykładu:
„długie nici pajęczyn, czepiały się żółtych *liści* ***akacji*** stojących pod murem".

Nazwa cechy	**Wartość**	**Opis**
`arg_left`	`liści`	wartość lewego argumentu (nie wchodzi w skład wzorca)
`arg_right`	`akacji`	wartość prawego argumentu (nie wchodzi w skład wzorca)
`direction`	`right_left`	kolejność argumentów, w tym przypadku argument stanowiący *całość* występuje w tekście po prawej stronie, a argument stanowiący *część* po lewej
`pos_left`	`noun`	część mowy lewego argumentu
`number_left`	`plural`	wartość liczby lewego argumentu
`case_left`	`genitive`	wartość przypadka lewego argumentu
`gender_left`	`masculine_3`	wartość rodzaj lewego argumentu
`pos_right`	`noun`	część mowy prawego argumentu
`number_right`	`singular`	wartość liczby prawego argumentu
`case_right`	`genitive`	wartość przypadka prawego argumentu
`gender_right`	`feminine`	wartość rodzaju prawego argumentu
`inner_context`	`--` (brak)	napis występujący pomiędzy argumentami

8.6. Określenie statystycznych cech wzorców

W wyniku wyszukiwania przykładów uczących w korpusie tekstów dla różnych wariantów algorytmu selekcji (patrz p. 8.3) oraz po ich przefiltrowaniu (patrz p. 8.4), powstają zbiory zawierające zdania, w których występuje poszukiwana relacja. W wyniku przekształcenia tych przykładowych zdań w formalne wzorce relacji powstaje zbiór wzorców, spośród których duża część ma identyczną treść. Wzorce, których cechy formalne są identyczne są utożsamiane.

Przykładowo, jeśli w korpusie tekstów odnalezione zostały następujące zdania dla predykatu `#$anatomicalParts`:

– Kubek z czarnymi liniami papilarnymi zdjętymi z *palców* ***matki***, talerz pęknięty pośrodku ostrą błyskawicą, ...

– Długie nici pajęczyn czepiały się żółtych *liści* ***akacji*** stojących pod murem ...

– kędzierzawość *liści* ***brzoskwini***, torbiel śliw, rak bakteryjny ...

– Zwalczają także inną groźną chorobę – opadzinę *liści* ***porzeczki***,

z tych zdań zostanie wyekstrahowany uogólniony wzorzec formalny przedstawiony w tabeli 8.3. Wzorzec ten uzupełniany jest o wartości dwóch miar:

1. Bezwzględna liczba przykładowych zdań, w których występuje dany wzorzec – CT_P, gdzie P to identyfikator wzorca; w uogólnionym wzorcu występuje jako cecha `total`.
2. Liczba różnych par argumentów `(arg_left,arg_right)`, które pasują do wzorca – CD_P; w uogólnionym wzorcu występuje jako cecha `distinct`.

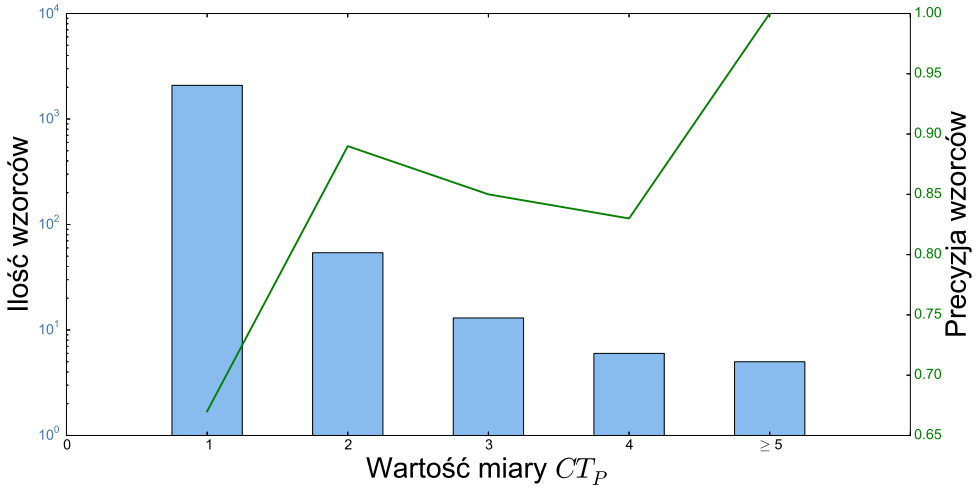
Miary te są wykorzystywane do automatycznego określenia wiarygodności wzorca. Choć wydawać by się mogło, że pierwsza miara powinna być wystarczająca do określenia jego poprawności – tzn. wzorzec, który występuje w dużej liczbie przykładowych zdań, powinien być wiarygodny, to wyniki eksperymentów przeprowadzone dla predykatu `#$anatomicalParts` pokazały, że miara ta jest problematyczna. W eksperymentach tych wykorzystano zbiór będący sumą wszystkich zbiorów odnalezionych zdań, tj.

$$S_R = \bigcup_{\substack{i \in \{1,2\} \\ type_1, type_2 \in \{direct, child\}}} S_{R,i,type_1-type_2} \ . \tag{8.1}$$

Na rysunku 8.3 przedstawiony jest wykres zależności precyzji wzorców (zdefiniowanej jako liczba przykładowych zdań zawierających faktyczne wystąpienie relacji *całość-część* do liczby wszystkich zdań podlegających ocenie) oraz liczby różnych wzorców, w zależności od wartości miary CT_P. Ostatnia kolumna zawiera wartości dla których $CT_P \geq 5$. Eksperyment przeprowadzono przy założeniu, że określony

Tablica 8.3: Uogólniony wzorzec formalny występujący w 10 zdaniach. Pierwszy argument odpowiada *całości*, a drugi argument *części*. Wzorzec uzupełniony jest o wartości miar CT_P (`total`) oraz CD_P (`distinct`).

Nazwa cechy	**Wartość**	**Kod wartości**
direction	right_left	rl
pos_left	noun	subst
number_left	plural	pl
case_left	genitive	gen
gender_left	masculine_3	m3
pos_right	noun	subst
number_right	singular	sg
case_right	gen	gen
gender_right	feminine	f
inner_context	--	
total	10	
distinct	4	



Rysunek 8.3: Ilość wzorców oraz ich precyzja w zależności od wartości miary CT_P dla wzorców o wartości miary $CD_P = 1$. Dla każdej wartości CT_P oceniono co najwyżej 100 przykładów pasujących do wzorców z danej grupy (mniej, jeśli łączna liczba przykładowych zdań pasujących do wzorców w danej grupie była mniejsza niż 100).

unikalny wzorzec posiada tylko jedną parę argumentów (`arg_left`, `arg_right`), tzn. $CD_P = 1$. Wyniki eksperymentu wskazują, że dla wartości $CT_P \geq 2$ następuje istotna poprawa precyzji – dla $CT_P = 1$ precyzja wynosi jedynie 67%, a dla $CT_P = 2$ 89%, ale wzrost wartości precyzji nie jest utrzymany dla wyższych wartości tej miary: precyzja dla $CT_P = 3$ wynosi 85%, a dla $CT_P = 4$ wynosi 83%. Dopiero dla wartości $CT_P \geq 5$ wynosi 100%. Liczba wzorców posiadających tę wartość jest jednak bardzo mała i wynosi 5.

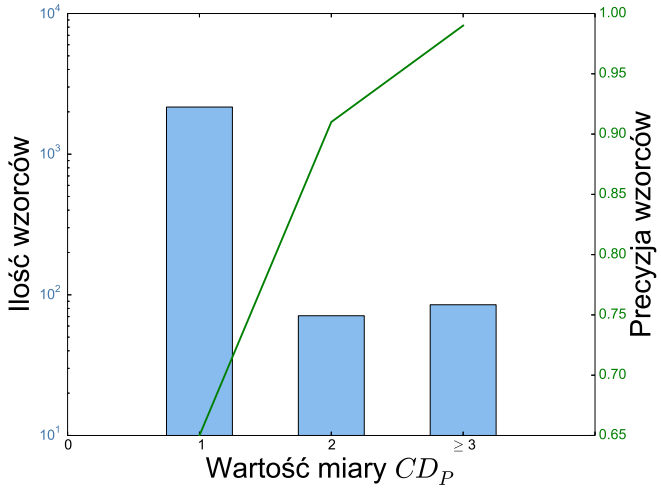
Przyczyną braku korelacji pomiędzy wartością miary CT_P, a precyzją wzorca, dla wzorców posiadających tylko jedną parę argumentów, jest fakt, że choć w tekście występują one wielokrotnie razem, to mogą nie być połączone zadaną relacją semantyczną. Przykładowo para: (`gwiazda`, `dłoń`), spowodowała znalezienie następujących przykładów:

– „Gwiazda w dłoni” – komedia romantyczna USA (1999) ...,

– „Gwiazda w dłoni” – komedia romantyczna USA (85) ...,

które prawdopodobnie pochodzą z programu telewizyjnego. Ponieważ opis nieznacznie się różni (w pierwszym przykładzie w opisie występuje rok produkcji filmu, a w drugim przykładzie jego długość), przykłady te nie zostały utożsamione.

Okazuje się, że miara CD_P jest znacznie lepszym wskaźnikiem jakości wzorców. Na rysunku 8.4 przedstawiony jest wykres zależności precyzji i liczby wzorców od wartości tej miary. Wynika z niego, że precyzja wzorca wprost zależy od liczby różnych par argumentów, na podstawie których zostały znalezione przykładowe zdania pasujące do tego wzorca. Dla pojedynczej pary wartość precyzji wynosi zaledwie 65%, dla dwóch różnych par 91%, a dla trzech 99%. Oczywiście im więcej różnych par argumentów, tym mniejsza liczba znalezionych wzorców, ale dla wartości $CD_P \geq 2$ znaleziono 156 różnych wzorców, a dla wartości $CD_P \geq 3$ znaleziono ich 85, 17 razy więcej niż dla wartości $CT_P \geq 5$, która dawała podobną precyzję.



Rysunek 8.4: Ilość wzorców oraz ich precyzja w zależności od wartości miary CD_P. Ostatnia kolumna reprezentuje wzorce o wartości $CD_P \geq 3$. Dla każdej wartości CD_P oceniono 100 przykładów pasujących do wzorców z danej grupy.

Przyczyną tak wysokiej skuteczności miary CD_P w stosunku do miary CT_P, w rozpoznawaniu wzorców wysokiej jakości jest fakt, że w przypadku tej pierwszej miary nie występuje problem omówiony na przykładzie pary (`gwiazda`, `dłoń`), tzn. nie zostaną znalezione zdania, które choć różnią się nieznacznie, mogą zawierać ten sam fragment tekstu, w którym poszukiwana relacja nie występuje. Widać to na przykładzie par (`palców`, `matki`), (`liści`, `akacji`), (`liści`, `brzoskwini`) oraz (`liści`, `porzeczki`). Choć wzorce są identyczne, to dopasowane zdania istotnie różnią się od siebie. Dlatego też tak otrzymany wzorzec jest bardziej wiarygodny.

W wyniku analizy statystycznej wzorców, w szczególności obliczywszy miarę CD_P, można automatycznie odfiltrować wzorce o niskiej wiarygodności. W zależności od przyjętych założeń, można dopuścić większą liczbę wzorców o nieco niższej wiarygodności (np. o $CD_P \geq 2$), albo mniejszą liczbę wzorców o wyższej wiarygodności (np. o $CD_P \geq 3$). Warto jednak pamiętać, aby analiza ta obejmowała wszystkie przykłady znalezione dla różnych wariantów algorytmu selekcji przykładowych zdań (patrz punkt 8.3).

8.7. Dopasowywanie wzorców formalnych do tekstu

Określenie formalnych ograniczeń wzorców ekstrakcyjnych pozwala na znalezienie potencjalnych wystąpień poszukiwanej relacji semantycznej w tekstach, z których ma być ona ekstrahowana. Aby dopasować wzorce formalne do tekstu, w pierwszej kolejności konieczne jest ujednoznacznienie wieloznacznych opisów morfosyntaktycznych wyrazów występujących w tekście. W tym celu wykorzystywany jest system *Concraft* autorstwa Waszczuka [158]. System ten na wejściu akceptuje tekst pozbawiony informacji na temat podziału na zdania, czy wyrazy, a na wyjściu produkuje tekst rozbity na zdania, składające się z segmentów wyposażonych w listę kompatybilnych form podstawowych oraz opisów morfosyntaktycznych (inaczej tagów), wraz z przypisanym do nich prawdopodobieństwem poprawności. Pozwala to na wybranie najbardziego opisu morfosyntaktycznego oraz rozwiązuje problem podziału tekstu na zdania.

Dane generowane przez *Concraft* są przekształcane w ten sposób, że poszczególne zdania otaczane są

znacznikami <s> oraz </s>, oznaczającymi odpowiednio początek i koniec zdania, a segmenty przekształcane są w trójki: (*forma tekstowa*, *forma podstawowa*, *opis morfosyntaktyczny*). Forma tekstowa zawiera również opcjonalny znak spacji, jeśli w tekście poprzedzał on tę formę. Trójki przekształcane są następnie w jednolitą listę napisów, zgodnie z ich kolejnością w tekście. Przekształcenie to pozwala na odtworzenie oryginalnej treści teksu (na podstawie form tekstowy), podziału na zdania (dzięki znacznikom <s> i </s>) oraz informacji o formie podstawowej i opisie morfosyntaktycznym. Przykładowo zdanie *Miał 59 lat.* zamieniane jest na ciąg napisów:[2]

```
<s>, Miał,mieć,praet:sg:m1:imperf, 59,59,num:pl:gen:m3:congr, lat,rok,subst:pl:gen:m3,
.,.,interp,</s>.
```

Dopasowanie wzorców formalnych nie odbywa się jednak bezpośrednio do tej postaci tekstu. Wykorzystując algorytm ujednoznaczniania sensu opisany w punkcie 7.3, określa się najpierw wystąpienia wyrażeń i nazw wielosegmentowych. Algorytm ujednoznaczniania sensu, podobnie jak algorytm ujednoznaczniania opisów morfosyntaktycznych, działa na surowym tekście. Dlatego na wejściu tego algorytmu pojawia się identyczna treść jak w przypadku tego drugiego algorytmu. Na wyjściu algorytmu ujednoznaczniania produkowane są krotki o postaci $(d_{start}, d_{end}, \sigma_a, P_{dg})$, gdzie:

- d_{start} – określona w znakach odległość początku rozpoznanego wyrażenia od początku tekstu,
- d_{end} – określona w znakach odległość końca rozpoznanego wyrażenia od początku tekstu.

Konieczne jest zatem uzgodnienie opisów wykorzystywanych przez oba algorytmy, gdyż jeden posługuje się segmentami tekstu, a drugi wskazuje początek i koniec wyrażenia (jedno lub wielosegmentowego) jako ilość znaków występujących od początku zdania. Zadanie to jest realizowane w ten sposób, że indeksy początku i końca wyrażeń produkowane przez algorytm ujednoznaczniania sensu zamieniane są na indeks zdania, w którym to wyrażenie wystąpiło oraz indeksy pierwszego i ostatniego segmentu wchodzącego w skład wyrażenia, w czego wyniku powstaje krotka $(h_s, h_{start}, h_{end}, \sigma_a, P_{dg})$, gdzie

- h_s – indeks zdania,
- h_{start} – indeks pierwszego segmentu,
- h_{end} – indeks ostatniego segmentu.

Dopasowanie wzorców formalnych realizowane jest osobno dla każdego zdania. W pierwszej kolejności tworzy się listę fragmentów zdania, które albo stanowią, albo nie stanowią składniki wyrażenia ujednoznacznionego względem Wikipedii. Jeśli pomiędzy dwoma ujednoznacznionymi wyrażeniami nie występuje żaden segment, to wstawia się tam fragment pusty. Podobnie, jeśli ujednoznacznione wyrażenie występuje na końcu, bądź na początku zdania. Wtedy pusty fragment wstawia się odpowiednio na początku, bądź na końcu tak przekształconego zdania. Dzięki temu, fragmenty odpowiadające ujednoznacznionym wyrażeniom zawsze występują na nieparzystych pozycja (zakładając, że pierwszy fragment ma indeks 0) i oddzielone są dokładnie jednym fragmentem nieujednoznacznionym.

Przykładowo zdanie

> Pochodzący z **Północnej Nadrenii-Westfalii prawnik** zasiada w **parlamencie** od 1996 roku,

[2]Poszczególne elementy oddzielone są przecinkiem. Wykorzystywane oznaczenia wyjaśnione są w tabeli D.2.

w którym ujednoznacznione wyrażenia są pogrubione, przekształcane jest na następującą listę fragmentów:[3]

`< Pochodzący z>< **Północnej Nadrenii-Westfalii**><><**prawnik**>< zasiada w>< **parlamencie**>< od 1996 roku.>`

Do tak utworzonej listy fragmentów dopasowywane są już bezpośrednio wzorce formalne. Algorytm przegląda ujednoznacznione fragmenty i pierwszy, najbardziej na lewo wysunięty rzeczownik ujednoznacznionego wyrażenia stojącego po lewej stronie, uznawany jest za jego syntaktyczną głowę. Jeśli opis morfosyntaktyczny zgadza się z ograniczeniami określonymi we wzorcu dla lewego argumentu, algorytm sprawdza prawy argumentu, czyli fragment oddalony o dwie pozycje w prawo. Dla tego fragmentu syntaktyczna głowa ustalana jest identycznie jak dla lewego argumentu. Jeśli ten argument również spełnia ograniczenia zdefiniowane we wzorcu, następuje zweryfikowanie wewnętrznego kontekstu wzorca – formy tekstowe w fragmencie oddalonym o jedną pozycję w prawo od lewego fragmentu, muszą być identyczne jak wartość pola `inner_context`. Jeśli to założenie jest spełnione, uznaje się, że wzorzec pasuje do tekstu.

Sprawdzenie dopasowania dla przykładu przedstawionego wyżej, rozpoczęłoby się od fragmentu `< Północnej Nadrenii-Westfalii>`. Segment `Nadrenii` zostały określony jako jego syntaktyczna głowa. Jeśli ograniczenia morfosyntaktyczne tego segmentu byłyby spełnione, sprawdzany byłby fragment `< prawnik>`, który składa się tylko z jednego segmentu. Następnie weryfikowany byłby wewnętrzny kontekst – wzorzec zostałby dopasowany, jeśli wartość pola `inner_context` byłaby pusta.

Poniżej przytoczone są przykłady zdań znalezionych w korpusie PAP na podstawie wzorca formalnego przedstawionego w tabeli 8.3:

– Oficjalna Agencja Prasowa OPEC podała, że koszyk siedmiu *gatunków* ***ropy*** kosztował w środę 26,31 dolarów za baryłkę (159 l) – o 26 centów więcej niż we wtorek (26,05 USD).

– Od 19 lutego Rafineria Gdańska SA podnosi ceny hurtowe wszystkich *gatunków* ***benzyny*** o 110 zł na tonie, cena oleju napędowego pozostaje bez zmian.

– Od najbliższej niedzieli 20 lutego, Polski Koncern Naftowy podnosi ceny hurtowe wszystkich *rodzajów* ***benzyny*** o 90 zł na tonie, czyli 9 gr na litrze.

– W 1991, podczas próby zamachu stanu i usunięcia Michaiła Gorbaczowa, przekonał dowódcę leningradzkiego okręgu wojskowego, by nie wykonał *rozkazów* ***Moskwy*** dotyczących skierowania na ulice wojsk i czołgów.

– Według autorów, jego założeniem jest wzbogacenie Wileńszczyzny, która obecnie jest jednym z najuboższych *regionów* ***Litwy***, a którą w 60 procentach zamieszkują Polacy.

Można zauważyć, że tylko ostatnie zdanie zawiera wystąpienie relacji *całość-część*. Bardzo podobny wynik został uzyskany po zweryfikowaniu zbioru 300 zdań losowo wybranych z korpusu PAP, które pasowały do wzorców formalnych otrzymanych na podstawie predykatu `#$anatomicalParts`, posiadających co najmniej 2 różne pary argumentów (tzn. $CD_P \geq 2$). Tylko 20% zawierało wystąpienie relacji *całość-część*. Dlatego też mechanizm dopasowywania wzorców ekstrakcyjnych, który opierałby się wyłącznie na wzorcach formalnych (w kształcie określonym w tabeli 8.2), nie jest w stanie poprawnie odróżnić różnych relacji semantycznych. Ich ekstrakcja wymaga zatem uzupełnienia wzorców formalnych ograniczeniami semantycznymi.

[3]Poszczególne fragmenty zaznaczone są za pomocą par znaków < oraz >. W opisie pominięto formy podstawowe wyrazów, ich opis morfosyntaktyczny, nazwy ujednoznacznionych artykułów Wikipedii oraz wartość prawdopodobieństwa ujednoznacznienia.

8.8. Określenie ograniczeń semantycznych

Biorąc pod uwagę wieloznaczność wzorców formalnych, określenie ograniczeń semantycznych jest konieczne do tego, aby zapewnić odpowiedni poziom precyzji algorytmu ekstrakcji relacji. Należy jednak wziąć pod uwagę, że również samo określenie ograniczeń semantycznych czasami nie wystarczy do tego, aby poprawnie rozpoznać wybraną relację semantyczną. Przyjmujemy jednak, że w prezentowanym algorytmie przypadki tego rodzaju są nierozwiązywalne, tzn. nie będziemy używać dodatkowych metod (np. wnioskowania wykraczającego poza relację hiperonimii), gdyż doprowadziłoby to do istotnej komplikacji algorytmu.

Ustalenie ograniczeń semantycznych realizowane jest na 3 sposoby:

1. na podstawie ręcznej oceny zdań zawierających dopasowania wzorców formalnych,

2. na podstawie ograniczeń semantycznych wyekstrahowanych z ontologii Cyc,

3. na podstawie ograniczeń semantycznych wyekstrahowanych z DBpedii.

Aby móc łatwiej porównać te trzy metody określania ograniczeń, algorytm rozpoznawania relacji semantycznych traktuje je identycznie. Taki sposób powoduje utratę części informacji (w szczególności nie są wykorzystywane negatywne decyzje określone w trakcie ręcznej oceny zdań), ale istotnie upraszcza implementację algorytmu.

8.8.1. Ręczna ocena zdań

Ręczna ocena zdań zawierających dopasowania wzorców formalnych, polega na tym, że użytkownik przegląda listę zdań zawierających dopasowania i decyduje, czy zadana relacja semantyczna występuje pomiędzy wyrażeniami dopasowanymi do wzorca formalnego. Przykładowo, jeśli wzorce formalne relacji *całość-część* dopasowane zostały do dwóch zdań:

1. Trzy najwyżej rozstawione tenisistki to *Szwajcarka* ***Martina Hingis*** (nr 1.), Amerykanka Lindsay Davenport (2.) i Rosjanka Anna Kurnikowa (3.).

2. Była to wówczas nowa *dziedzina* ***fizyki jądrowej***.

to dla pierwszego przykładu stwierdzi, że analizowana relacja w nim nie występuje, a dla drugiego, że relacja ta występuje.

W ten sposób powstają dwa zbiory – jeden, w którym znajdują się zdania zawierające wystąpienie relacji oraz drugi, w którym znajdują się zdania, które pasują do wzorców formalnych, lecz nie zawierają zadanej relacji semantycznej. Ograniczenia semantyczne określane są wyłącznie na podstawie pierwszego zbioru. Ponieważ konstrukcja takiego zbioru jest żmudna, zwykle nie zawiera on wystarczającej liczby przykładów, aby można było dokonać statystycznej analizy ograniczeń. Z tego względu przyjęto, że wszystkie pary kategorii semantycznych występujące w zdaniach należących do tego zbioru, definiują poprawne ograniczenia semantyczne dla zadanej relacji. W przeciwieństwie jednak do algorytmu opisanego w punkcie 7.4, dla obu wyrażeń połączonych relacją, dla wszystkich par kategorii semantycznych należących do iloczynu kartezjańskiego kategorii semantycznych tych wyrażeń, przypisuje się taką samą wartość prawdopodobieństwa warunkowego wynoszącą 1. Pary te stają się zatem ograniczeniami semantycznymi uzyskanymi na podstawie ręcznej oceny zdań.

Tablica 8.4: Losowo wybrane specjalizacje predykatu `#$parts` występującego w ontologii Cyc.

Nazwa predykatu	**Opis predykatu**
`#$hasRooms`	łączy budowlę z pokojami, które są jej częściami
`#$sandwichFillings`	łączy kanapkę z jej zawartością
`#$subNetwork`	łączy sieć komputerową z jej podsiecią
`#$subSystems`	łączy system z jednym z jego podsystemów
`#$essentialParts`	łączy obiekt z jego podstawowymi składnikami
`#$parliamentOf`	łączy jednostkę geopolityczną z jej parlamentem
`#$lastSubEvents`	łączy zdarzenie z jego końcem
`#$officialArmedForces`	łączy organizację z jej oficjalnymi siłami zbrojnymi
`#$groupMembers`	łączy grupę ludzi z jej członkami
`#$organizationKeyMembers`	łączy organizację z jej najważniejszymi członkami
`#$cellMemberInTerroristGroup`	łączy komórkę organizacji terrorystycznej z jej członkami
`#$trialOfCase`	łączy sprawę sądową z pojedynczą rozprawą sądową

8.8.2. Ekstrakcja ograniczeń semantycznych z Cyc

Ustalenie listy ograniczeń semantycznych na podstawie Cyc realizowane jest w następujących krokach:

1. wyekstrahowanie ograniczeń semantycznych z predykatów,
2. wybranie predykatów odpowiadających określonej relacji semantycznej,
3. odfiltrowanie zbyt ogólnych ograniczeń semantycznych.

Wyekstrahowanie ograniczeń semantycznych z predykatów odbywa się w sposób analogiczny, jak pozyskiwanie przykładów par pojęć połączonych predykatem `#$anatomicalParts`, tzn. z wykorzystaniem predykatu `#$relationAllExists`.

Pozyskanie ograniczeń na podstawie tego predykatu wygląda następująco: w pierwszej kolejności wszystkie asercje zawierające ten predykat są eksportowane do postaci trójek składających się z następujących elementów (R, a_1, a_2), gdzie R to predykat, a_1 to pierwszy argument, a a_2 to drugi argument. W następny kroku użytkownik musi określić, które predykaty odpowiadają relacji, dla której mają zostać określone ograniczenia semantyczne. Zadanie to jest uproszczone, ponieważ w Cyc poza hierarchią pojęć zdefiniowana jest również hierarchia predykatów. Dzięki temu możliwe jest wskazanie ogólnego predykatu odpowiadającego interesującej nas relacji, a jego specjalizacje mogą zostać uzyskane na podstawie jednego wywołania w API Cyc (`all-genl-predicates`).

Przykładowo, relacji *całość-część* odpowiada predykat `#$parts`, który posiada ponad 900 specjalizacji. Przykładowe specjalizacje tego predykatu przedstawione są w tabeli 8.4. Przykłady te pokazują, że w Cyc występują zarówno bardzo ogólne predykaty, takie jak `#$essentialParts` oraz bardzo specyficzne, jak `#$trialOfCase`. Widać również, że predykat `#$parts` jest bardzo ogólny, ponieważ obejmuje zarówno relacje fizyczne (np. `#$hasRooms`), przynależność do grupy (np. `#$groupMembers`) oraz relacje czasowe (np. `#$lastSubEvents`). Obecność tych relacji dość dobrze koresponduje z podtypami meronimii omówionymi w punkcie 3.2.6 oraz z typami relacji semantycznych zdefiniowanymi przez NIST, przedstawionymi w tabeli 8.1.

Korzystając z hierarchii predykatów można dość łatwo odfiltrować ograniczenia semantyczne, które nie pasują do relacji, którą zamierzamy ekstrahować. Tym niemniej proces filtrowania ograniczeń semantycz-

Tablica 8.5: Przykładowe ogólne pojęcia Cyc, które zostały wykluczone ze zbioru ograniczeń semantycznych.

Pojęcie	**Opis**
`#$Cavity`	wnęka, zagłębienie w jakimś obiekcie
`#$Base-Support`	podstawa obiektu fizycznego
`#$ExistingObjectType`	rodzaj obiektów o charakterze konkretnym
`#$GeneralPoint`	punkt odniesienia wykorzystywany przez wojskowych
`#$LineOfContact`	linia kontaktu pomiędzy dwoma wrogimi siłami zbrojnymi
`#$PurposefulAction`	celowe działanie podejmowane przez podmiot
`#$Translocation`	przemieszczenie się na pewnym dystansie
`#$MultiIndividualAgent`	grupa niezależnych podmiotów
`#$Analyst-PertinentConcept`	rodzaj pojęcia interesującego dla analityków
`#$FixedFunctionalSystem`	system funkcjonalny o stałej strukturze

nych musi obejmować jeden dodatkowy krok – znaczna liczba ograniczeń semantycznych, również tych pozyskanych na bazie predykatu `#$relationAllExists`, jest nadal zbyt ogólna, aby można je było wykorzystać bezpośrednio do ekstrakcji relacji.

Analizując najbardziej ogólne pojęcia występujące w Cyc, opracowana została lista pojęć, posiadających bardzo ubogą treść semantyczną. Następnie korzystając z wywołania Cyc `all-genls`, znalezione zostało domknięcie tego zbioru pojęć. Jeśli przy konstrukcji domknięcia okazało się, że któreś z wybranych pojęć posiadało generalizację, która posiadała specyficzną treść semantyczną, pojęcie to było usuwane z początkowej listy. W ten sposób została skompilowana lista 509 najbardziej ogólnych pojęć, które zostały wykluczone ze zbioru ograniczeń semantycznych (wystarczyło, aby tylko jedno pojęcie należało do tego zbioru, aby całe ograniczenie semantyczne zostało usunięte). Przykładowe pojęcia należące do tego zbioru przedstawione są w tabeli 8.5. W grupie tej uwidocznione zostały ogólne pojęcia, takie jak `#$ExistingObjectType`, którego treść semantyczna jest niezwykle uboga oraz bardziej specyficzne, jak `#$LineOfContact`, które przynależą do określonej dziedziny wiedzy (wiedza na temat działań zbrojnych), ale których charakter jest na tyle abstrakcyjny, że stają się nieprzydatne z punktu widzenia zdania jakim jest ekstrakcja relacji semantycznych.

Zbiór ograniczeń semantycznych, który pozostaje po odfiltrowaniu z niego zbyt ogólnych pojęć, wykorzystywany jest już bezpośrednio do ekstrakcji wybranej relacji semantycznej. W tabeli 8.6 przedstawione są przykładowe ograniczenia semantyczne relacji *całość-część* pozyskane na podstawie ontologii Cyc. Większość z przedstawionych ograniczeń jest dość specyficzna, z wyjątkiem predykatów `#$capitalCity`, łączących *państwa* z ich *stolicami* oraz `#$provinces`, łączącej *państwa* z ich *prowincjami*.

8.8.3. Ekstrakcja ograniczeń semantycznych z DBpedii

DBpedia jest drugim źródłem wiedzy, które wykorzystywane jest do określenia ograniczeń semantycznych ekstrahowanych relacji. Pierwszy kwestia, która musi być rozwiązana, jeśli chcemy wykorzystać ją do określenia ograniczeń semantycznych, to zweryfikowanie, czy relacja semantyczna, którą chcemy ekstrahować, występuje w ontologii DBpedii. W przeciwieństwie do ontologii Cyc, która zawiera ponad 26 tys. predykatów, ontologia DBpedii zawiera ok. 1 tys. predykatów, które nie są zbyt dobrze uporządkowane, ani opisane. Zatem znalezienie odpowiednich predykatów może być problematyczne. Ponadto, kiedy zdecydujemy się na wybór jednego predykatu, szybko może się okazać, że nie tylko on reprezentuje wybraną

Tablica 8.6: Przykładowe pary ograniczeń semantycznych dla relacji *całość-część*, pozyskane z wykorzystaniem predykatu `#$relationAllExists` z ontologii Cyc.

Predykat	**Całość**	**Część**
`#$linksOfCustomarySystem`	`#$RespiratoryTract`	`#$Trachea`
`#$provinces`	`#$IndependentCountry`	`#$Province`
`#$capitalCity`	`#$Country`	`#$CountryCapital`
`#$subordinateOrganizations`	`#$GeopoliticalEntity`	`#$PoliceOrganization`
`#$networkMember`	`#$RetailPharmacyNetwork`	`#$PharmaceuticalDispensing-Organization`
`#$keyGroupMembers`	`#$VeterinaryHospital`	`#$Veterinarian`
`#$familyHasMember`	`#$Family-Nuclear`	`#$HumanAdult`

Tablica 8.7: Lista predykatów występujących w DBpedii, odpowiadających relacji *całość-część*. Kierunek wskazuje, czy argumenty poszczególnych relacji odpowiadają przyjętemu założeniu, że jako pierwszy występuje obiekt stanowiący *całość*, a jako drugi obiekt reprezentujący *część*: `direct` oznacza, że preferowana kolejność jest spełniona przez predykat, a `inverse`, że kolejność ta jest odwrotna.

Predykat	**Kierunek**
`affiliation`	`inverse`
`album`	`inverse`
`board`	`inverse`
`athletics`	`direct`
`capital`	`direct`
`childOrganisation`	`direct`
`europeanParliamentGroup`	`inverse`
`keyPerson`	`direct`
`leader`	`direct`
`part`	`inverse`

przez nas relację. Ponieważ jednak hierarchia predykatów nie jest dobrze zdefiniowana, zidentyfikowanie wszystkich interesujących nas predykatów może okazać się czasochłonne. Co więcej – ponieważ predykaty te nie muszą tworzyć hierarchii, kolejność argumentów może być różna dla poszczególnych predykatów. Konieczne jest zatem określenie czy kolejność argumentów jest zgodna z przyjętymi założeniami.

Przykładowo, relacji *całość-część* odpowiadają predykaty przedstawione w tabeli 8.7 (kompletna lista znajduje się w dodatku E), gdzie określono również ich kierunek. Widać, że ontologia DBpedii zawiera bardzo zróżnicowany zbiór predykatów odpowiadających tej relacji. Obok bardzo ogólnego `#$part` występuje bardzo specyficzny predykat `europeanParliamentGroup`. Można również zauważyć, że około połowy predykatów wykorzystuje naturalną kolejność argumentów (`direct`, tzn. argument reprezentujący *całość* występuje jako pierwszy), a druga połowa wykorzystuje kolejność odwrotną (`inverse`).

Po ustaleniu zbioru interesujących nas predykatów, można przystąpić do uruchomienia algorytmu opisanego w punkcie 7.4. W wyniku uruchomienia tego algorytmu otrzymujemy listę krotek postaci (R, a_1, a_2, G_T). Na tym etapie można odfiltrować krotki, których wsparcie $G_T < n$. Oznacza to, że dana para kategorii semantycznych była zaobserwowana mniej niż n razy. Pozwala to usunąć ograniczenia,

Tablica 8.8: Wsparcie G_T dla predykatu `region`, określone na podstawie DBpedii: a_1 – pierwszy argument, a_2 – drugi argument.

a_1	a_2	G_T
#$Commune-State-Geopolitical	#$Region	24424,2
#$City	#$Region	3394,7
#$Person	#$CongressionalDistrict	852,1
#$Municipality	#$Region	787,2
#$River	#$County	728,7
#$PopulatedPlace	#$County	589,2
#$Town	#$County	515,7
#$Area	#$County	442,8
#$Person	#$State-#$Geopolitical	426,8
#$River	#$Region	396,6
...		
#$Prosecutor	#$County	5,0
#$Hospital	#$CountryCapital	5,0
#$Beach	#$County	5,0
#$School-#$AcademicOrganization	#$City	5,0
#$Banker	#$CongressionalDistrict	5,0
#$Person	#$Territory	5,0
#$Place	#$GeographicalRegion	5,0
#$Municipality	#$Territory	4,9
#$Municipality	#$AutonomousRegion	4,9
#$Sheriff	#$CongressionalDistrict	4,9
#$Person	#$Municipality	4,9
#$USStateSenator	#$City	4,8
...		

które są zbyt specyficzne. Co prawda nie powinny one wpłynąć negatywnie na precyzję algorytmu, ale w istotny sposób spowalniają proces dopasowywania relacji, gdyż konieczne jest rozpatrzenie znacznie większego zbioru ograniczeń semantycznych.

W następnym kroku, dla każdej unikalnej pary argumentów (a_1, a_2), określana jest lista krotek, w których argumenty te wystąpiły. Krotki te są sortowane malejąco względem wartości wsparcia. Następnie wartość wsparcia zastępowana jest wartością prawdopodobieństwa warunkowego, zgodnie ze wzorem 7.16. W wyniku tej operacji, para argumentów występujących w krotce o najwyższej wartości prawdopodobieństwa warunkowego, trafia to zbioru ograniczeń semantycznych relacji występującej w tej samej krotce. Jako wartość prawdopodobieństwa warunkowego tej pary argumentów, przyjmowana jest wartość prawdopodobieństwa występującego dla tej krotki.

Przykładowo, w tabeli 8.8 przedstawione są bezwzględne wartości wsparcia dla predykatu `region`. Przyjmując $G_T \geq 5$, wszystkie pary ograniczeń semantycznych znajdujące się poniżej podwójnej linii zostałyby usunięte. W tabeli 8.9 przedstawione zostały krotki zawierające identyczną parę argumentów `#$Hospital` oraz `#$City`, które występują w więcej niż jednej krotce. Najwyższą wartość posiada

Tablica 8.9: Wsparcie G_T oraz prawdopodobieństwo warunkowe $P(r|sc_1, sc_2)$ uzyskane przez predykaty DBpedii dla ograniczeń semantycznych `#$Hospital` i `#$City`.

R	G_T	$P(r\|sc_1, sc_2)$
location	222,9	0,49
region	202,4	0,44
locationCity	14,7	0,03
city	10,3	0,02
state	8,2	0,02

Tablica 8.10: Prawdopodobieństwo warunkowe wybranych par ograniczeń semantycznych, określonych dla predykatów odpowiadających relacji semantycznej *całość-część* (kolejność ograniczeń semantycznych została dostosowana do kolejności argumentów tej relacji).

Predykat	**Całość**	**Część**	$P(r\|sc_1, sc_2)$
affiliation	Organization	EducationalOrganization	0,9
album	LiveAlbum-CW	Song-CW	1,0
associatedBand	Band-MusicGroup	Person	1,0
capital	Country	City	1,0
employer	TelevisionNetwork	Person	1,0
isPartOfMilitaryConflict	War	Battle	1,0

krotka, w której predykatem jest `location` i tylko ona trafia do zbioru ograniczeń semantycznych, a przypisane jej prawdopodobieństwo warunkowe wynosi 0,49.

Ostatnim krokiem jest wybór tych ograniczeń semantycznych, które odpowiadają ekstrahowanej relacji. Zadanie to ogranicza się do wyboru tych spośród ograniczeń, które posiadają najwyższą wartość prawdopodobieństwa warunkowego, dla jednego z predykatów odpowiadających ekstrahowanej relacji. Przykładowo, w tabeli 8.10 pojawiły się ograniczenia, wraz z wartością prawdopodobieństwa warunkowego dla predykatów odpowiadających relacji *całość-część*.

8.9. Rozpoznawanie relacji semantycznych

Wynikiem algorytmu tworzącego wzorce ekstrakcyjne są dwa zbiory ograniczeń – wzorce formalne oraz ograniczenia semantyczne. Zbiory te są od siebie niezależne, to znaczy, że ograniczenia semantyczne, na podstawie których następuje ostateczne rozpoznanie relacji, stosowane są do wszystkich par wyrażeń pasujących do wzorców formalnych.

Dopasowywanie wzorców formalnych zawsze przeprowadzane jest w sposób ścisły – oba wyrażenia muszą spełnić wszystkie ograniczenia zdefiniowane we wzorcach formalnych, z zastrzeżeniem, że dopasowanie odbywa się na poziomie wyrażeń, a nie pojedynczych słów (patrz p. 8.7).

W odniesieniu do ograniczeń semantycznych sytuacja wygląda inaczej. Zasadniczym elementem prezentowanego algorytmu jest możliwość wykorzystania hierarchii pojęć zdefiniowanej w ontologii Cyc. Z tego względu dopasowywanie ograniczeń semantycznych może odbywać się na dwa sposoby: bezpośrednio lub z wykorzystaniem relacji generalizacji.

W pierwszym przypadku stwierdzenie wystąpienia określonej relacji semantycznej następuje wyłącznie wtedy, gdy wśród ograniczeń semantycznych zdefiniowanych dla tej relacji (bądź jednego z predykatów stowarzyszonych z tą relacją) znajduje się para, która odpowiada przynajmniej jednemu elementowi należącemu do iloczynu kartezjańskiego wszystkich kategorii semantycznych określonych dla symboli odpowiadających wyrażeniom dopasowanym do wzorca formalnego. Jeśli w wyniku tej operacji pojawia się wieloznaczność (tzn. istnieją co najmniej dwie pary ograniczeń semantycznych przypisanych do różnych relacji semantycznych), wybierana jest ta relacja, która ma wyższą wartość współczynnika K_{spec} zdefiniowanego następująco

$$K_{spec} = P(r|sc_1, sc_2) * |Genls(sc_1)| * |Genls(sc_2)| \,, \tag{8.2}$$

gdzie $Genls(sc)$ oznacza zbiór wszystkich generalizacji ograniczenia semantycznego sc. W ten sposób preferowane są relacje, które mają wyższą wartość prawdopodobieństwa warunkowego wykorzystywanych ograniczeń semantycznych oraz których ograniczenia semantyczne są bardziej specyficzne. Specyficzność odzwierciedlana jest bowiem przez liczbę wszystkich kategorii semantycznych, które stanowią generalizację tego ograniczenia. Choć istnieje niezerowe prawdopodobieństwo, że wartość tej miary będzie równa dla kilku relacji, to w praktyce zdarza się to bardzo rzadko, ze względu na rozbudowaną taksonomię ontologii Cyc.

W przypadku wykorzystania relacji generalizacji, przy dopasowywaniu ograniczeń semantycznych, zmieniane jest ograniczenie związane z bezpośrednim wystąpieniem jednego elementu (pary kategorii semantycznych), należących do iloczynu kartezjańskiego kategorii semantycznych, określonych dla obu symboli, odpowiadających wyrażeniom dopasowanym do wzorca formalnego. W tej wersji algorytmu oczekuje się, że istnieje para ograniczeń semantycznych, które stanowią generalizacje kategorii semantycznych, określonych dla wyrażeń dopasowanych do wzorca. Jeśli prowadzi to do wieloznaczności, to wykorzystywana jest miara K_{spec}.

Przykładowo, w zdaniu

> Według ostatecznych wyników wyborów, które odbyły się w sobotę i niedzielę, kierowany przez *Mugabe* ***ZANU-PF*** (Afrykański Narodowy Związek Zimbabwe – Front Patriotyczny) uzyskał 62 mandaty, opozycyjny Ruch na rzecz Zmian Demokratycznych (MDC) – 57, a jeden mandat zdobyła maleńka opozycyjna partia ZANU-DONGA,

wyrażenia `Mugabe` oraz `ZANU-PF` zostały dopasowane do wzorca formalnego relacji *całość-część*. Odpowiadające im symbole posiadają następujące kategorie semantyczne:

- **Robert Mugabe**: `#$RomanCatholic`, `#$Politician`, `#$Graduate`, `#$Person`, `#$Leader`,
- **Afrykański Narodowy Związek Zimbabwe - Front Patriotyczny**: `#$GuerillaForce`, `#$PoliticalParty`, `#$Organization`.

Relacji *całość-część* w DBpedii odpowiadają m.in. predykaty `leader, keyPerson` oraz `institution`. Dla predykatu `leader` wykryto w DBpedii m.in. ograniczenia semantyczne `#$PoliticalParty`, `#$Leader`, które pasują do kategorii semantycznych wskazanych wyrażeń. Ponieważ ograniczenia te są najbardziej specyficzne, a prawdopodobieństwo warunkowe tego predykatu, pod warunkiem wystąpienia tych ograniczeń semantycznych wynosi 1, jako pasujący do wskazanych wyrażeń wybierany jest predykat `leader` i w konsekwencji we wskazanym zdaniu rozpoznawana jest relacja semantyczna *całość* (**Afrykański Narodowy Związek Zimbabwe - Front Patriotyczny**) – *część* (**Robert Mugabe**).

9. Konstrukcja wzorców relacji *całość-część*

Wstęp

W niniejszym rozdziale szczegółowo opisujemy proces konstrukcji wzorców ekstrakcyjnych dla relacji *całość-część*. Wyniki ekstrakcji tej relacji z korpusu PAP przedstawione są natomiast w rozdziale 10. Relacja ta została wybrana z kilku powodów. Po pierwsze, jak pokazują badania Girju dla języka angielskiego [47], relacja ta wymaga znacznie bardziej zaawansowanych metod ekstrakcji, niż najczęściej ekstrahowana relacja, tj. *hiponimia*. Wykorzystanie jedynie wzorców formalnych do ekstrakcji tej relacji daje wyniki złej jakości, co zostało wykazane w literaturze obcojęzycznej oraz czego dowody dla języka polskiego przedstawione są w punkcie 10.1.1. Dlatego też, dla tej relacji konieczne jest określenie ograniczeń semantycznych. Dzięki temu możliwe będzie porównanie różnych metod określania ograniczeń oraz wykazanie tezy niniejszej pracy.

Należy również zwrócić uwagę, że ekstrakcja relacji *hiponimii* stanowiła sedno jednego z algorytmu pomocniczego, tj. algorytmu klasyfikacji artykułów Wikipedii (patrz p. 7.2). Co prawda, ekstrakcja ta realizowana była w kontekście Wikipedii, ale specyfika tej relacji polega właśnie na tym, że częściej występuje oraz jest łatwiej rozpoznawalna w tekście encyklopedycznym, niż w tekstach notatek prasowych.

Ponadto ekstrakcja relacji *hiponimii*, obok ekstrakcji relacji *synonimii*, była jednym z podstawowych problemów stawianych przy automatycznej konstrukcji polskiego WordNetu [108]. Próba powtórzenia tych eksperymentów przekraczałaby jednak zdecydowanie zakres niniejszej pracy. Dlatego, jako podstawowa relacja podlegająca analizie, wybrana została właśnie relacja *całość-część*. Biorąc pod uwagę te rozstrzygnięcia, w opisie dalszych eksperymentów przyjmiemy, że wszędzie tam gdzie pojawia się symbol R, symbolizuje on relację *całość-część*.

9.1. Pary pojęć dla relacji *całość-część*

Predykatem wykorzystywanym do budowania wzorca ekstrakcyjnego dla relacji *całość-część* jest `#$anatomicalParts`, który łączy *organizmy* z ich *częściami*. W najnowszej wersji *ResearchCyc* występują 94 unikalne asercje zawierające predykat `#$relationAllExists`, opisujące ten predykat. Pełna lista angielskich pojęć, stanowiąca zbiór $\boldsymbol{C_R}$ znajduje się w dodatku A, a jej tłumaczenie na język polski zostało przedstawione w dodatku B. Poniżej zaś znajduje się zestawienie przykładowych par argumentów, pogrupowanych ze względu na stopień ich ogólności.

- Związki specyficzne (semantyczne):
  - `#$Bear-Animal` (*niedźwiedź*) – `#$Claw` (*pazur*)
  - `#$Bee` (*pszczoła*) – `#$Stinger` (*żądło*)

  - `#$Bull-Cattle` (*byk*) – `#$Horn-AnimalBodyPart` (*róg*)
  - `#$Cactus` (*kaktus*) – `#$Thorn` (*kolec*)
  - `#$Cat` (*kot domowy*) – `#$SpinalColumn` (*kręgosłup*)
  - `#$Crab` (*krab*) – `#$Pincer` (*szczypce*)
  - `#$Elephant` (*słoń*) – `#$Trunk-TheAppendage` (*trąba*)
  - `#$HomoSapiens` (*człowiek rozumny*) – `#$Eyebrow` (*brew*)
- Związki pośrednie (semantyczne):
  - `#$Arthropod` (*stawonóg*) – `#$Leg` (*noga*)
  - `#$BirdOfPrey` (*ptak drapieżny*) – `#$Talon` (*szpon*)
  - `#$Bird` (*ptak*) – `#$Bill-Birds` (*dziób*)
  - `#$Bird` (*ptak*) – `(#$GroupFn #$Feather)` (*pióra*)
  - `#$Bird` (*ptak*) – `#$MobOfFeathers` (*upierzenie*)
  - `#$Fish` (*ryba*) – `#$Fin` (*płetwa*)
  - `#$Reptile` (*gad*) – `#$Tail-BodyPart` (*ogon*)
  - `#$Tree-ThePlant` (*drzewo*) – `#$TreeBranch` (*gałąź*)
- Związki abstrakcyjne (ontologiczne):
  - `#$Animal` (*zwierzę*) – `#$DigestiveSystem` (*system trawienny*)
  - `#$Animal` (*zwierzę*) – `#$VisualSystem` (*system wzrokowy*)
  - `#$FemaleAnimal` (*samica*) – `#$ReproductiveSystem-Female` (*żeński układ rozrodczy*)
  - `(#$FemaleFn #$Mammal)` (*samica ssaka*) – `#$MammaryGland` (*gruczoł sutkowy*)
  - `#$MaleAnimal` (*samiec*) – `#$ReproductiveSystem-Male` (*męski układ rozrodczy*)
  - `(#$MaleFn #$Mammal)` (*samiec ssaka*) – `#$Penis` (*penis*)
  - `#$Primate` (*ssak naczelny*) – `#$Hand` (*ręka*)
  - `#$Vertebrate` (*kręgowiec*) – `#$Appendage-AnimalBodyPart` (*członek ciała*)

W pierwszej grupie występują pary pojęć, w których pojęcie reprezentujące *organizm* jest dość specyficzne – tzn. zazwyczaj jest to nazwa gatunku (*Homo sapiens*, *kot domowy*) lub takson wyższego rzędu (*niedźwiedź*, *krab*). W tej grupie można zauważyć wyraźne związki semantyczne pomiędzy pojęciami, np. *słoń – trąba*, *krab – szczypce*, *kaktus – kolec*.

W drugiej grupie mamy do czynienia z bardziej ogólnymi pojęciami – w szczególności *organizmy* reprezentowane są przez taksony wyższego rzędu (np. gromady *ptaki* i *gady*), natomiast *cześci organizmu* są nadal dość specyficzne, dzięki temu związki te mają charakter semantyczny, np. *ryba – płetwa*, *drzewo – gałąź*, *stawonóg – noga*.

W ostatniej grupie występują dość ogólne pojęcia – zarówno po stronie *organizmów* (*zwierzę*, *samiec*, *samica*), jak i po stronie *części organizmu* (*system trawienny*, *żeński układ rozrodczy*, *członek ciała*). Przykłady te opisują ogólne zależności ontologiczne, które dalekie są od związków semantycznych. Ze względu na fakt, że te grupy organizmów oraz części organizmów zostały wyodrębnione naukowo i typowy użytkownik języka zapoznaje się z nimi dopiero w trakcie edukacji, nie stanowią one związków semantycznych. Mogą one jednak stanowić źródło przykładów zarodkowych, ale jedynie przy uwzględnieniu relacji hierarchicznych pomiędzy pojęciami.

Analiza wymienionych przykładów pokazuje również, że przedstawiony zbiór ma dwie wady:

– nie jest wyczerpujący,

– występują w nim pary redundantne.

Przykładem pokazującym, że zbiór ten nie jest wyczerpujący jest np. brak związku *ucho – ssak* (albo jakiegokolwiek innego związku, w którym występowałoby *ucho*). Natomiast redundancja może być zaobserwowana na dwóch parach: *osoba – ręka* oraz *ssak naczelny – ręka*. Nie zmienia to jednak faktu, że wymienione pary pojęć są poprawne i pozwalają na wygenerowanie znacznej liczby przykładów dla relacji *całość-część*.

9.2. Taksonomia ontologii Cyc

Bardzo istotną cechą algorytmy selekcji przykładów uczących jest wykorzystywanie taksonomii ontologicznej do automatycznego rozszerzenia zbioru przykładowych zdań, które zawierają wystąpienia argumentów relacji semantycznej. Korzystając z tłumaczenia opisanego w punkcie 7.1.4, możliwe było przetłumaczenie części taksonomii na język polski. W dodatku C pojawia się dość duży fragment taksonomii zakorzenionej w pojęciu `#$Bird`. Poniżej przedstawiony jest jej fragment:

- `#$Peacock` (*paw*)
- `#$Woodpecker` (*dzięcioł*)
- `#$Kingfisher` (*zimorodek*)
- `#$Stork` (*bocian*)
- `#$Crow` (*wrona*)
- `#$Cuckoo` (*kukułka*)
- `#$Parrot` (*papuga*)
  - `#$Parakeet` (*papużka*)
  - `#$Macaw` (*ara*)
  - `#$AfricanGreyParrot` (*papuga popielata*)
- `#$BirdOfPrey` (*ptak drapieżny*)
  - `#$Owl` (*sowa*)
  - `#$Condor` (*kondor*)
- `#$Poultry` (*drób*)
  - `#$Goose-Domestic` (*gęś domowa*)
  - `#$Chicken` (*kura*)
  - `#$Duck` (*kaczka*)
  - `#$Turkey-Bird` (*indyk*)

Dzięki wykorzystaniu taksonomii oraz tłumaczenia, dla przytoczonej wcześniej pary *ptak – upierzenie*, możliwe jest wygenerowanie dużej liczby par pojęć reprezentujących ten sam związek semantyczny.

Tablica 9.1: Liczność zbiorów przykładowych zdań znalezionych w korpusie IPI PAN, zawierających dopasowanie argumentów relacji, bądź ich specjalizacji. $\boldsymbol{S_{R,1,direct}}$ – zbiór zawierający bezpośrednie dopasowania pierwszego argumentu, $\boldsymbol{S_{R,2,direct}}$ – zbiór zawierający bezpośrednie dopasowania drugiego argumentu, $\boldsymbol{S_{R,1,child}}$ – zbiór zawierający dopasowania jednej z 10 losowo wybranych specjalizacji pierwszego argumentu, $\boldsymbol{S_{R,2,child}}$ – zbiór zawierający dopasowania 10 losowo wybranych specjalizacji drugiego argumentu.

Nazwa zbioru	**Rozmiar zbioru**
$\boldsymbol{S_{R,1,direct}}$	73416
$\boldsymbol{S_{R,2,direct}}$	100443
$\boldsymbol{S_{R,1,child}}$	446828
$\boldsymbol{S_{R,2,child}}$	27641
Suma	**648328**

9.3. Przykłady zdań zawierających relację *całość-część*

Na podstawie zbioru $\boldsymbol{C_R}$ zawierającego przykłady par pojęć relacji *całość-część*, taksonomii ontologii Cyc oraz tłumaczenia $\boldsymbol{M^{pl}}$, wygenerowano zapytania do korpusu IPI PAN. Serwer był skonfigurowany tak, by zwracać 5 tys. pierwszych wyników. Jeśli tłumaczenie pojęcia Cyc na język polski było wielosegmentowe, pomiędzy każdą parą segmentów mógł występować jeden dodatkowy segment. Dopasowanie otoczone było z każdej strony siedmioma segmentami tworzącymi kontekst, w którym w kolejnym etapie wyszukiwany był drugi argument. W ten sposób wygenerowano zbiory $\boldsymbol{S_{R,1,direct}}$ i $\boldsymbol{S_{R,2,direct}}$, zawierające bezpośrednie wystąpienia pierwszego oraz drugiego argumentu relacji oraz zbiory $\boldsymbol{S_{R,1,child}}$ i $\boldsymbol{S_{R,2,child}}$, zawierające wystąpienia losowo wybranych 10 specjalizacji argumentów. Tabela 9.1 zawiera zestawienie liczności poszczególnych zbiorów.

Największym zbiorem okazał się $\boldsymbol{S_{R,1,child}}$, a najmniejszym $\boldsymbol{S_{R,2,child}}$. Nie jest to zaskakujące, gdyż pierwszym argumentem były *organizmy*, dla których można było znaleźć wiele specjalizacji i w ten sposób wygenerować znaczną liczbę przykładowych zdań. Z drugiej strony, *części organizmu* nie mają tak rozbudowanej taksonomii i w konsekwencji zbiór znalezionych zdań, zawierających specjalizacje pojęć z tej grupy, jest mniejszy. Nieco zaskakująca jest natomiast relacja pomiędzy zbiorami $\boldsymbol{S_{R,1,direct}}$ oraz $\boldsymbol{S_{R,2,direct}}$ – oba zbiory wygenerowane były dla dokładnie takiej samej liczby pojęć, ale zbiór wygenerowany dla drugiego argumentu (*części ciała*), jest istotnie większy. Wynik ten można uzasadnić biorąc pod uwagę charakter danych w Cyc – występujące w nim fakty, zwykle wyrażane są na wysokim poziomie abstrakcji, z wykorzystaniem języka naukowego, dlatego też nazwy organizmów rzadziej występują w niespecjalistycznym korpusie IPI PAN.

W kolejnym etapie w zdaniach ze zbiorów $\boldsymbol{S_{R,i,type}}$, $i \in \{1,2\}$, $type \in \{direct, child\}$ wyszukiwano wystąpień drugiego argumentu (tj. argumentu, który nie występował w zapytaniu do serwera Poliqarp) – wykorzystując wariant z bezpośrednim wystąpieniem argumentu, bądź jego dowolnej specjalizacji. W ten sposób powstały zbiory zdań $\boldsymbol{S_{R,i,type_1-type_2}}$, $i \in \{1,2\}$, $type_1 \in \{direct, child\}$, $type_2 \in \{direct, child\}$ zawierające wystąpienia obu argumentów.

Zestawienie liczności tych zbiorów przedstawione jest w tabelach 9.2 oraz 9.3. W pierwszej kolejności można zauważyć dużą różnicę w wielkości pierwotnego zbioru zdań, obejmującego prawie 650 tys. przykładów, a wielkością wynikowego zbioru zawierającego nieco ponad 10 tys. przykładów. Wynikowy zbiór zdań stanowi zatem jedynie 1,5% początkowego zbioru. Ta różnica daje się łatwo wyjaśnić, gdyż pre-

Tablica 9.2: Liczność zbiorów, w których rozpoznano drugi argument relacji, powstałych na bazie zbiorów $\boldsymbol{S_{R,1,direct}}$, $\boldsymbol{S_{R,2,direct}}$, $\boldsymbol{S_{R,1,child}}$ i $\boldsymbol{S_{R,2,child}}$ (patrz tabela 9.1). Ostatni człon nazwy zbioru wskazuje, czy drugi argument występował bezpośrednio (*direct*), czy też znaleziono jedną z jego specjalizacji (*child*).

Nazwa zbioru	**Rozmiar zbioru**
$\boldsymbol{S_{R,1,direct-direct}}$	130
$\boldsymbol{S_{R,2,direct-direct}}$	565
$\boldsymbol{S_{R,1,direct-child}}$	16
$\boldsymbol{S_{R,2,direct-child}}$	6074
$\boldsymbol{S_{R,1,child-direct}}$	249
$\boldsymbol{S_{R,2,child-direct}}$	61
$\boldsymbol{S_{R,1,child-child}}$	37
$\boldsymbol{S_{R,2,child-child}}$	3086
Suma	**10218**

Tablica 9.3: Zbiorcze zestawienie liczności zbiorów z tabeli 9.2, ze względu na poszczególne cechy: numeru argumentu, na podstawie którego wygenerowano pierwotny zbiór (*argument*), sposobu dopasowania pierwszego argumentu ($type_1$) oraz sposobu dopasowania drugiego argumentu ($type_2$).

Nazwa cechy	**Wartość cechy**	**Rozmiar zbioru**	**Udział procentowy**
argument	1	432	4,2
argument	2	9786	95,8
$type_1$	*direct*	6785	66,4
$type_1$	*child*	3433	33,6
$type_2$	*direct*	1005	9,8
$type_2$	*child*	9213	90,2

dykat `#$anatomicalParts` reprezentuje relację encyklopedyczną, a korpus zawierał zróżnicowane teksty, z wielu dziedzin wiedzy (porównaj tabela 6.1).

Druga obserwacja dotyczy liczności poszczególnych zbiorów – można zauważyć, że największe zbiory ($\boldsymbol{S_{R,2,direct-child}}$ i $\boldsymbol{S_{R,2,child-child}}$) stanowią te przykłady, w których drugi dopasowywany argument stanowił specjalizację. Ta obserwacja zgodna jest z tezą, że rozszerzenie zbioru przykładowych zdań poprzez wykorzystanie relacji taksonomicznych, w istotny sposób wpływa na możliwość automatycznego zbudowania odpowiednio dużego zbioru.

Z drugiej jednak strony, warto zauważyć, że nie zawsze dopasowywanie specjalizacji drugiego argumentu daje tak liczne zbiory: $\boldsymbol{S_{R,1,direct-child}}$ oraz $\boldsymbol{S_{R,1,child-child}}$ są istotnie mniejsze od swoich odpowiedników, w których dopasowywano bezpośrednie wystąpienie argumentu. Wynika to z faktu, że w tych zbiorach w pierwszej kolejności dopasowywano *organizm*, a w drugiej *część organizmu*. Jak już wcześniej zauważono, taksonomia *organizmów* jest znacznie bardziej rozbudowana niż taksonomia *części organizmów*, w związku z czym zbiory, w których dopasowywano specjalizacje części ciała, są niewielkie.

Obserwacja ta jest szczególnie widoczna w wynikach zestawionych w tabeli 9.3. Przykłady, w których w pierwszej kolejności dopasowywano pierwszy argument (tj. *organizm*), stanowią jedynie 4,2% wszystkich przykładów. Aby zatem można było wygenerować duży zbiór przykładów na podstawie niewielkiego

zbioru par pojęć połączonych relacją, wskazane jest, aby dopasowywać argument posiadający rozbudowaną taksonomię jako drugi w kolejności. Można również zwiększyć zbiór specjalizacji wybieranych losowo w pierwszym kroku, ale będzie to prowadziło do znacznego wydłużenia całego procesu znajdowania przykładów wystąpień relacji.

Jeśli chodzi o rodzaj dopasowania w pierwszym kroku, to warto zwrócić uwagę na to, że lepiej sprawdza się bezpośrednie dopasowanie, które daje w przypadku wymienionej relacji prawie dwa razy więcej przykładów. Natomiast w drugim kroku znacznie lepiej wykorzystać dopasowanie jednej ze specjalizacji – wyniki dla analizowanej relacji pokazują, że można przez to uzyskać kilkukrotny wzrost liczby dopasowań.

Podsumowując wyniki dopasowania drugiego argumentu, można wyciągnąć następujące wnioski:

– wykorzystanie dopasowania specjalizacji argumentów istotnie zwiększa ilość pozyskanych zdań,

– argument, który posiada bardziej rozbudowaną taksonomię powinien być dopasowywany jako drugi,

– w pierwszym kroku warto dopasowywać bezpośrednie wystąpienia argumentów,

– w drugim korku warto dopasowywać specjalizacje argumentów.

Poniżej przedstawione są przykładowe zdania odnalezione w korpusie IPI PAN na podstawie par pojęć wymienionych w punkcie 9.1:

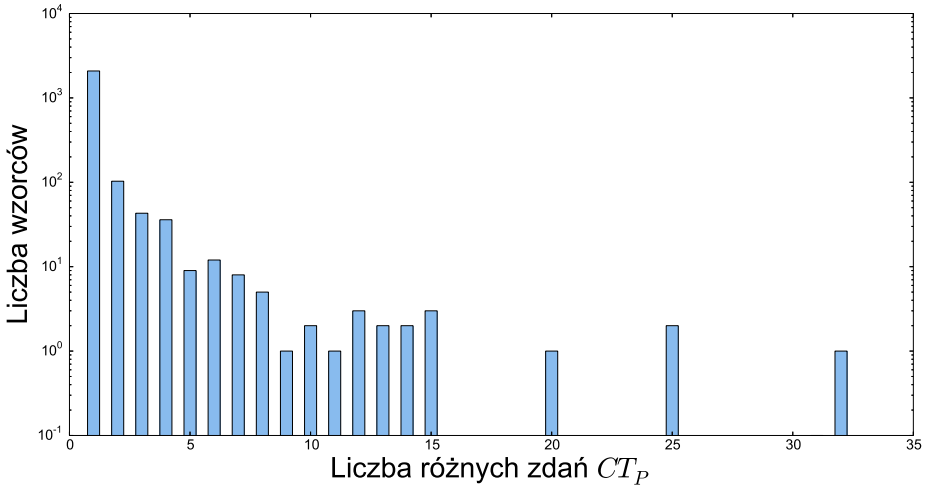
1. Myślał żałobnie o *liściu* ***buka***, który był niegdyś maleńki u łona matki-gałęzi ...
2. ..., że projekt robiony był w Krajowej Radzie w dużym stopniu *rękami* ***osób*** związanych z telewizją publiczną, ...
3. Biegają dzieci, przez *korony* ***drzew*** padają promienie słońca.
4. ... przy czym niewprawna *ręka* ***matki*** lub służącej usiała na ich głowach całe konstelacje gwiazd, ...
5. W „Akwarium" można dostrzec fragmenty *skóry* ***makreli*** i delikatne szkielety zbutwiałych liści; ...
6. Dla przykładu, kilogram *płetwy* ***rekina*** kosztuje średnio prawie 600 USD.
7. ... czepiały się żółtych *liści* ***akacji*** stojących pod murem.

W przytoczonych przykładach, z wyjątkiem przykładu drugiego, mamy do czynienia z relacją *organizm-część organizmu*. Drugi przykład również zawiera tę relację, lecz samo użycie jest metaforyczne, gdyż występuje w nim metonimia, trudno bowiem uznać, że jedynie *ręce*, członków Krajowej Rady Radiofonii i Telewizji, brały udział w tworzeniu projektu, o którym mowa w przytoczonym zdaniu. Sformułowanie to służy raczej do określenia osób, które brały udział w realizacji tego projektu. Poza tym jednym wyjątkiem, przytoczone przykłady są dobrymi reprezentantami relacji *całość-część*.

9.4. Filtrowanie przykładów

Obok tych pozytywnych przykładów, znaleziono również następujące fragmenty tekstów, zawierające wybrane pary pojęć:

1. ... uniosłem uspokajająco *dłonie*. – Po ekshumacji ***człowiek*** wyobraża sobie Bóg wie co ...
2. „Przestaję ściskać bochenki pięści i podaję ***ludziom*** kromki *dłoni*. Dzielę się sobą, ...



Rysunek 9.1: Wykres liczby wzorców formalnych względem liczby unikalnych zdań pasujących do wzorca – CT_P.

3. ...ściskać bochenki pięści i podaję ***ludziom*** kromki *dłoni*. Dzielę się sobą, mówiąc Ewa ...

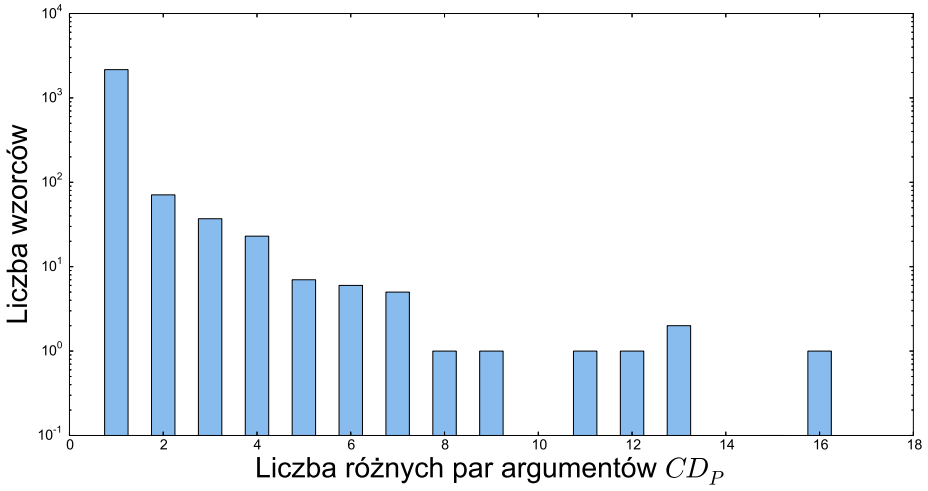
4. Pola są szare, drzewa ***bez*** *liści*, a twórcy chcą pokazać barwy.

Przykłady te wskazują wyraźnie na potrzebę odfiltrowania wyników. W pierwszym przykładzie oba argumenty występują w dwóch różnych zdaniach, dlatego poszukiwana relacja nie występuję pomiędzy tymi wyrażeniami. W drugim i trzecim przykładzie mamy do czynienia z dokładnie tym samym fragmentem tekstu, który jednak obejmuje nieco inny kontekst. Wyeliminowanie przykładów tego rodzaju pozwoli uzyskać bardziej wiarygodne częstości występowania poszczególnych wzorców. W ostatnim przykładzie natomiast mamy do czynienia z sytuacją, w której dopasowany wyraz jest wieloznaczny, zarówno syntaktycznie jak i semantycznie. W tym zdaniu występuje w roli przyimka, dlatego nie może być argumentem relacji semantycznej *całość-część*.

W celu wyeliminowania mało wiarygodnych przykładów zastosowano mechanizm filtrowania opisany w punkcie 8.4. Żeby jednak wyniki filtrowania były bardziej wiarygodne, w szczególności aby faktycznie uniknąć wszystkich powtarzających się przykładów, przed filtrowaniem przykłady należące do osobnych zbiorów trafiły do jednego wspólnego zbioru S_R. Co prawda w wyniku połączenia zbiorów, utracona została informacja o pochodzeniu poszczególnych przykładów, ale zbiór wynikowy zawierał więcej przykładów, niż każdy ze zbiorów źródłowych. Co więcej taki połączony zbiór lepiej nadawał się do przeprowadzenia analizy statystycznej, gdyż pojedynczy wzorzec formalny mógł zostać wygenerowany na podstawie bardziej zróżnicowanych przykładów. W wyniku filtrowania, z początkowej liczby 10218 przykładowych zdań, pozostały 3054 unikalne zdania.

9.5. Ekstrakcja wzorców formalnych

Zdania, które pozostały w zbiorze S_R po jego odfiltrowaniu, zostały użyte do wyekstrahowania wzorców formalnych zgodnie z opisem w punkcie 8.5. Powtarzające się wzorce zostały utożsamione, co pozwoliło określić wartości miar CT_P oraz CD_P. W wyniku ekstrakcji powstało 2319 unikalnych wzorców.



Rysunek 9.2: Wykres liczby wzorców formalnych względem liczby różnych par argumentów pasujących do wzorca – CD_P.

Wykresy 9.1 oraz 9.2 zawierają zestawienia ilościowe wzorców o identycznej wartości miar CT_P oraz CD_P. Dominującą grupę stanowią wzorce posiadające tylko jedno wystąpienie wśród przykładowych zdań oraz jedną parę wyrażeń, na bazie której został wygenerowany wzorzec. Biorąc pod uwagę niską jakość tak otrzymanych wzorców (patrz punkt 8.6), do dalszej analizy wybrane zostały wyłącznie wzorce o $CD_P \geq 2$. W ten sposób otrzymano 156 unikalnych wzorców wysokiej jakości.

Zestawienie wzorców o najwyższych wartościach miary CT_P przedstawione jest w tabeli 9.4. Wśród wzorców tych łatwo można zauważyć pewną prawidłowość – 1, 2 i 4 wzorzec posiadają pusty wewnętrzny kontekst, drugi argument, czyli *część organizmu* występuje po lewej stronie, a prawy argument ma prawie identyczne cechy formalne (różnica dotyczy wyłącznie rodzaju i w dwóch przypadkach jest to rodzaj męski osobowy, a w jednym przypadku rodzaj męski żywotny). Analizując je można wywnioskować, że wzorce te zostały wyekstrahowane ze zdań, w których argumenty relacji połączone są związkiem rządu.

Wzorzec 3 wyraźnie różni się od pozostałych – w wewnętrznym kontekście pojawia się wyrażenie *za*, kolejność argumentów jest odmienna (po lewej stronie występuje argument odnoszący się do pierwszego argumentu relacji – czyli *organizmu*), a drugi argument występuje w bierniku. Wzorzec ten wyekstrahowany został tylko z dwóch typów przykładów, w których dopasowany został fragment *byka za rogi* bądź *tygrysa za wąsy*. Duża liczba przykładów wynika z tego, że wyrażenie *złapać byka za rogi* jest skostniałą metaforą, często występującą w tekście.

Różnica pomiędzy przykładami 1, 2 i 4 a przykładem 3 pokazuje jeszcze raz ważność miary CD_P – wzorzec 3 ma najniższą wartość tej miary i nieco przypadkowo znalazł się w tym zestawieniu. Wiele kolejnych wzorców, które nie znalazły się w tym krótkim zestawieniu, podobnych było do wzorców z pierwszej grupy – w szczególności charakteryzowały się tym, że argumenty występowały w szyku `right_left`, a prawy argument występował w dopełniaczu.

Te wyniki wskazują, że rozpoznane wzorce formalne posiadają wspólne cechy, które w pewnym stopniu charakteryzują relację semantyczną *całość-część*. Niemniej jednak, przedstawione w punkcie 9.7 wyniki dopasowania wzorców formalnych do tekstu wskazują, że wzorce te są zbyt ogólne, aby w sposób precyzyjny można było na ich podstawie rozpoznawać tę relację.

Tablica 9.4: Najczęściej powtarzające się wzorce formalne o $CD_P \geq 2$, wyekstrahowane na podstawie predykatu `#$anatomicalParts`.

Nazwa cechy	**Wzorzec 1**	**Wzorzec 2**	**Wzorzec 3**	**Wzorzec 4**
direction	right_left	right_left	left_right	right_left
pos_left	noun	noun	noun	noun
number_left	singular	singular	plural	singular
case_left	genitive	genitive	accusative	nominative
gender_left	feminine	feminine	masculine_2	feminine
pos_right	noun	noun	noun	noun
number_right	singular	singular	plural	singular
case_right	genitive	genitive	accusative	genitive
gender_right	masculine_1	masculine_2	masculine_3	masculine_2
inner_context	--	--	za	--
arg_left	dłoni, nogi, tchawicy, . . .	płetwy, skóry, łapy, . . .	byka, tygrysa	trąba, płetwa, skóra, . . .
arg_right	człowieka, mężczyzny, premiera, . . .	rekina, wieloryba, bawołu, . . .	rogi, wąsy	słonia, rekina, węża, rysia, . . .
total	**32**	**25**	**25**	**20**
distinct	**13**	**16**	**2**	**12**

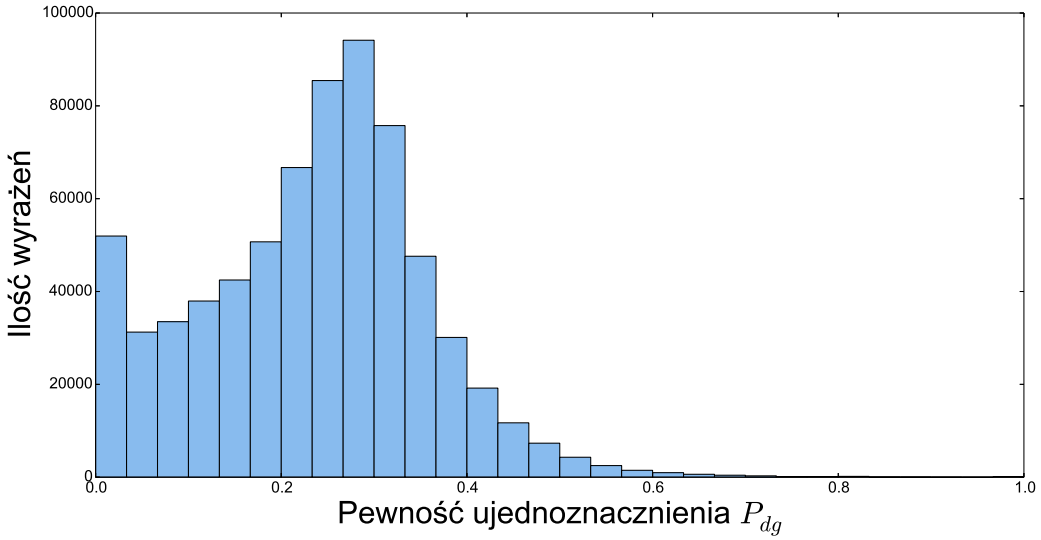
9.6. Ujednoznacznienie sensu wyrażeń w korpusie PAP

Ponieważ jeden z wariantów algorytmu określania ograniczeń semantycznych wykorzystuje zdania z korpusu PAP, w kolejnym etapie ujednoznaczniono sens wyrażeń występujących w tym korpusie, względem słownika semantycznego zbudowanego na bazie Wikipedii (patrz punkty 6.3 oraz 7.3).

Tabela 9.5 zawiera statystyki procesu ujednoznaczniania. Całkowita liczba ujednoznacznionych wyrażeń (prawie 700 tys.) stanowi niemal jedną piątą liczby wszystkich segmentów tekstu występujących w korpusie. Długość rozpoznanych wyrażeń wynosiła od 1 do 7 segmentów, przy czym średnia wartość wynosząca 1,34 wskazuje, że zdecydowana większość wyrażeń składała się z jednego segmentu. Pewność ujednoznaczniania P_{dg} obejmowała cały zakres – od 0 do 1 – ale wartość średnia wynosząca 0,23 wskazuje, że algorytm ujednoznaczniania musiał podejmować decyzje, zwykle na podstawie zbyt małej ilości informacji. Fakt ten koresponduje z oceną skuteczności algorytmu przedstawioną w punkcie 7.3.5. Chcąc zachować minimalny poziom poprawności ujednoznaczniania, konieczne jest zatem ustawienie progu pewności powyżej zera. Ustawienie wartości pewności nieco poniżej wartości średniej, co prawda doprowadzi do utraty dużej liczby rozpoznanych wyrażeń, ale jak widać na histogramie przedstawionym na rysunku 9.3, dominanta pewności przypada w okolicach wartości 0,3 a nie wartości 0, dlatego w kolejnych eksperymentach liczba dostępnych ujednoznacznionych wyrażeń powinna być stosunkowo duża.

Tablica 9.5: Statystyki ujednoznaczniania korpusu PAP.

Cecha	**Wartość**
Całkowita liczba segmentów	3595398
Liczba ujednoznacznionych wyrażeń	697085
Minimalna długość ujednoznacznionych wyrażeń	1
Maksymalna długość ujednoznacznionych wyrażeń	7
Średnia długość ujednoznacznionych wyrażeń	1,34
Minimalna pewność rozpoznania	0
Maksymalna pewność ropoznania	1
Średnia pewność rozpoznania	0,23



Rysunek 9.3: Wykres ilości wyrażeń rozpoznanych w korpusie PAP, w zależności od pewności ujednoznacznienia P_{dg}.

Tablica 9.6: 15 najczęściej rozpoznawanych pojęć w korpusie notatek PAP wraz z liczbą wystąpień, przy $P_{dg} \geq 2$.

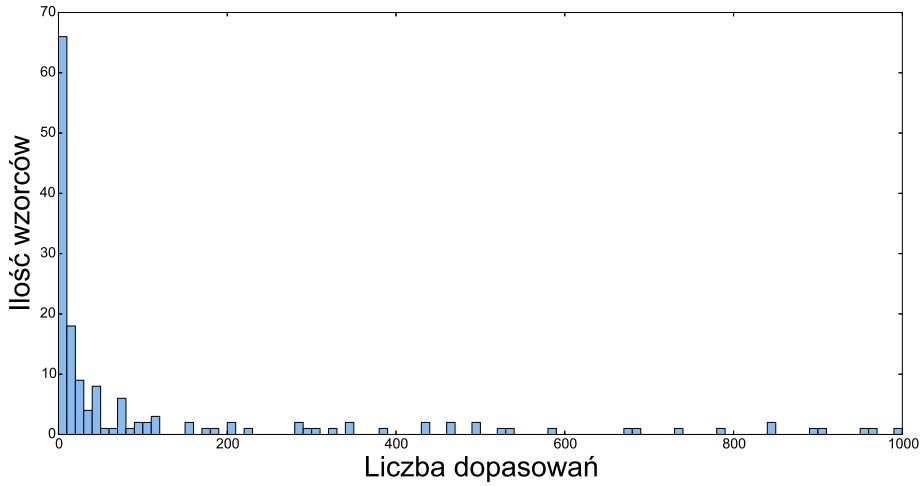
Pojęcie – tytuł artykułu Wikipedii	Liczba rozpoznań
Polska	12387
Stany Zjednoczone	6642
Unia Europejska	6199
Złoty	4641
Warszawa	4484
Rosja	4210
Niemcy	3293
Dolar amerykański	3238
Policja	3034
Akcja Wyborcza Solidarność	2926
Ustawa	2584
Polacy	2540
Sojusz Lewicy Demokratycznej	2391
Europa	2258
Rząd (prawo)	2168

W tabeli 9.6 przedstawiono 15 najczęściej rozpoznawanych symboli językowych występujących w notatkach, przy minimalnym progu pewności ustalonym na 0,2. Symbole te bardzo dobrze odzwierciedlają charakter korpusu PAP, który w dużej mierze składa się z informacji na temat bieżących wydarzeń politycznych i ekonomicznych, obejmujących zdarzenia krajowe oraz zagraniczne. Obok dominujących nazw własnych – w szczególności nazw państw takich jak **Polska**, **Stany Zjednoczone**, czy **Rosja** – występują rzeczowniki pospolite, takie jak **Ustawa** i **Rząd**, które wprost wiążą się z poruszaną tematyką. Ważnymi pojęciami, które pojawiają się w zestawieniu są również nazwy partii politycznych, które obecne były w sejmie w okresie z którego pochodzą notatki – sprawujące władzę **Akcja Wyborcza Solidarność** oraz główna partia opozycyjna **Sojusz Lewicy Demokratycznej**. Wyniki te świadczą, że algorytm ujednoznaczniania realizuje stawiane przed nim zadanie, przynajmniej w odniesieniu do popularnych nazw i pojęć często pojawiających się w notatkach Polskiej Agencji Prasowej.

9.7. Dopasowywanie wzorców formalnych do zdań

Uzyskane w punkcie 9.5 wzorce formalne zostały dopasowane do dokumentów znajdujących się w korpusie PAP za pomocą algorytmu opisanego w punkcie 8.7. Żeby umożliwić zbadanie wpływu miary pewności ujednoznacznienia na skuteczność całego algorytmu, nie ustalono minimalnej wartości tej miary. W wyniku dopasowania 156 wzorców do 51 tys. krótkich notatek prasowych uzyskano 20914 dopasowań, w których rozpoznano wystąpienie jednego z wyekstrahowanych wzorców. Liczba ta jest stosunkowo duża, biorąc pod uwagę fakt, że całkowita liczba zdań w korpusie PAP wynosiła prawie 200 tys., a dopasowania zostały ograniczone wyłącznie do tych wyrażeń, które zostały ujednoznacznione względem słownika semantycznego.

Na rysunku 9.4 przedstawiony jest histogram liczby dopasowań dla poszczególnych wzorców. Z jednej strony można zauważyć, że istnieje duża grupa wzorców posiadających bardzo niewielką liczę dopasowań –



Rysunek 9.4: Wykres przedstawiający ilość wzorców posiadających określoną liczbę dopasowań w korpusie PAP.

od 0 do 10. Najwięcej wzorców nie posiadało żadnego dopasowania, było ich aż 35. Z drugiej strony istnieje niewielka liczba wzorców, które posiadają od 100 do 1000 dopasowań. Istnieje również umiarkowanie duża grupa wzorców, które posiadają od 10 do 100 dopasowań.

Poniżej przedstawione zostały wyniki dopasowania trzech wzorców formalnych, z każdej z wymienionych grup: pierwszy z nich jest wzorcem dopasowanym najczęściej, drugi posiadał średnią frekwencję dopasowań, a trzeci został dopasowany tylko do jednego przykładu. Przykłady, w których występuje relacja *całość-część* zostały zaznaczone poprzez wytłuszczenie punktora[1]

1. Szablon: `rl`, `subst:sg:gen:f`, `--`, `subst:sg:gen:f`, $CT_P = 13$, $CD_P = 11$,
   ilość dopasowań: 1000.

   (a) Z udziałem szefów *dyplomacji* ***Polski*** i USA Bronisława Geremka i Madeleine Albright odbyła się w Departamencie Stanu uroczysta inauguracja Polsko-Amerykańskiej Fundacji Wolności.

   (b) Cena *baryłki* ***ropy*** Brent z Morza Północnego, z dostawą w marcu, wzrosła w czwartek o 15.15 na Międzynarodowej Giełdzie Paliw w Londynie do 27,19 dolarów.

   (c) Siódme zwycięstwo w 13. kolejce rozgrywek *grupy B* ***Euroligi***, odniosły koszykarki Polpharmy VBW Clima Gdynia.

   (d) Komitet Ekonomiczny Rady Ministrów przyjął dokument „Założenia *polityki energetycznej* ***Polski*** do 2020 roku”, po zweryfikowaniu przez resort gospodarki prognozy zapotrzebowania na energię.

   (e) Austriacka minister spraw zagranicznych odcięła się od wypowiedzi Joerga Haidera nt. poszerzenia UE i wezwała kraje członkowskie 15 do zakończenia *izolacji* ***Austrii*** na scenie europejskiej.

2. Szablon: `rl`, `subst:pl:gen:f`, `--`, `subst:sg:gen:m1`, $CT_P = 13$, $CD_P = 4$,
   ilość dopasowań: 85.

[1] W poniższych przykładach zastosowano skrócony zapis wzorców formalnych. Szczegółowy opis tego formatu znajduje się w dodatku D.

(a) Zdaniem pełnomocnika rządu ds. *negocjacji* ***Jana Kułakowskiego*** data zakończenia negocjacji o członkostwo Polski w Unii Europejskiej może być podana pod koniec roku na szczycie europejskim w Nicei.

(b) W Austriackim Instytucie Kultury w Warszawie otwarta została w sobotę wystawa *fotografii* ***Harry'ego Webera*** pt. „Wiedeń dzisiaj – współczesność żydowska w fotografii".

(c) Zabrzmią fragmenty chóralne m.in. z *oper* ***Moniuszki*** i Verdiego.

(d) Sześcioro sygnatariuszy Porozumień Sierpniowych z Gdańska zaapelowało w piątek do Kolegium Instytutu Pamięci Narodowej o wybór na szefa IPN wiceministra *spraw wewnętrznych i administracji* ***Bogdana Borusewicza***.

(e) Rządowy samochód ministra-koordynatora *specsłużb* ***Janusza Pałubickiego*** zderzył się w centrum Warszawy z innym samochodem – podała policja.

3. Szablon: `lr`, `subst:sg:acc:f`, `--`, `subst:pl:inst:f`, $CT_P = 4$, $CD_P = 2$,
ilość dopasowań: 1.

(a) Polakom zarzuca się niszczenie majątku (rozbijanie witryn) i czynną napaść na funkcjonariuszy publicznych (rzucanie w ***policję*** *kostkami* brukowymi).

Pierwszą obserwacją jaką można poczynić w odniesieniu do wyników dopasowania wzorców formalnych jest to, że większość spośród dopasowanych zdań nie zawiera relacji *całość-część*. Ten wynik wskazuje, że uzyskane wzorce formalne nie są wystarczające do tego by można na ich podstawie poprawnie rozpoznawać tę relację.

Druga obserwacja dotyczy charakteru relacji *całość-część*: pomimo tego, że jako dane wyjściowe użyte zostały wyłącznie pary pojęć połączonych predykatem `#$anatomicalParts`, to w przykładach pasujących do wzorca formalnego występują inne podtypy relacji *całość-część*. W pierwszym przykładzie (1.a) mamy do czynienia z organizacją (*Polska*) oraz jej funkcjonalną częścią (*dyplomacja*). W drugim przykładzie (1.b) mamy do czynienia z relacją wyodrębniającą pewną część (*baryłkę*) z abstrakcyjnej całości (*ropa*), która obejmuje „całość ropy we wszechświecie". Trzeci przykład (1.3) podobny jest do pierwszego, gdyż całość stanowi organizacja (*euroliga*) ale relacja ta ma charakter strukturalny (część stanowi *grupa B*). Widać zatem wyraźnie, że charakter rozpoznanej relacji nie jest jednolity, tzn. nie obejmuje wyłącznie typów obiektów, na podstawie których zbudowane zostały wzorce formalne. Ten wynik świadczy o uniwersalności prezentowanego podejścia.

Ostatnia istotna obserwacja dotyczy zdań, w których wzorzec został dopasowany, ale w których nie występuje poszukiwana relacja. W niektórych przypadkach mam do czynienia z inną relacją (np. *twórca-wytwór*, jak w przypadku *oper Moniuszki*, *działaniem-przdmiotem działania*, jak w przypadku *izolacji Austrii*, czy *relacją posesywną*, jak w przypadku *polityki energetycznej Polski*), zdarzają się jednak sytuacje, w których rozpoznanie wyrażenia wielosegmentowego nie było właściwe. Np. we fragmencie `na szefa IPN` **`wiceministra`** `spraw wewnętrznych i administracji` – *sprawy wewnętrzne i administracja* zostały wydzielone z nazwy stanowiska, co może prowadzić do błędnego rozpoznania relacji.

Ciekawą grupę stanowią również przykłady, które nie posiadały żadnego dopasowania. Wśród nich można znaleźć następujące wzorce:

1. `lr`, `subst:sg:acc:m2`, `za`, `subst:pl:acc:m3`, $CT_P = 25$, $CD_P = 2$,
2. `lr`, `subst:sg:gen:m2`, `za`, `subst:pl:acc:m3`, $CT_P = 6$, $CD_P = 2$,
3. `rl`, `subst:sg:acc:f`, `z`, `subst:sg:gen:m2`, $CT_P = 6$, $CD_P = 5$,

4. `lr`, `subst:sg:nom:m1`, `doznał wstrząśnienia`, `subst:sg:gen:m3`, $CT_P = 4$, $CD_P = 2$.

Ich charakterystyczną cechą jest to, że wewnętrzny kontekst dopasowania jest niepusty i pomimo posiadania wysokiej wartości współczynników CT_P (pierwszy wzorzec) oraz CD_P (trzeci wzorzec) nie zostały one ani razu dopasowane w korpusie PAP. Ten wynik pokazuje, że występowanie wewnętrznego kontekstu dopasowania być może powinno skutkować rozluźnieniem ograniczeń formalnych nakładanych na argumenty, aby w ogóle możliwe było wykorzystanie wzorców tego rodzaju.

Bardziej szczegółowa analiza ilościowa oraz jakościowa wyników uzyskanych na podstawie dopasowań wzorców formalnych przedstawiona jest w punktach 10.1.1 oraz 10.1.2.

9.8. Określenie ograniczeń semantycznych

Warunkiem wstępnym wykorzystania ograniczeń semantycznych do rozpoznawania relacji semantycznych jest to, aby oba wyrażenia dopasowane do wzorca formalnego miały przypisaną kategorię semantyczną. Algorytm określania kategorii semantycznych opisany został w punkcie 7.2. W punkcie 7.2.9 wskazano, że nie wszystkie artykuły występujące w polskiej Wikipedii otrzymały kategorie semantyczne – pokrycie algorytmu wyniosło około 80%. Z tego powodu nie wszystkie wyrażenia dopasowane do wzorców formalnych otrzymały kategorię semantyczną. Z niespełna 21 tys. dopasowań wzorca formalnego (patrz p. 9.7) 13267 (63%) posiadało przypisaną przynajmniej jedną kategorię semantyczną dla każdego z dopasowanych wyrażeń. Wynik ten nie jest zaskakujący jeśli weźmiemy pod uwagę fakt, że podobnie jak w przypadku ujednoznaczniania wyrażeń względem Wikipedii, konieczne jest określenie kategorii semantycznej dla obu wyrażeń i przy 80% pokryciu, teoretyczne prawdopodobieństwo[2] posiadania kategorii semantycznej przez oba wyrażenia wynosi dokładnie 64%.

Biorąc pod uwagę ten fakt, możemy przystąpić do przedstawienia rezultatów określania ograniczeń semantycznych relacji *całość-część*. W opisie algorytmu konstrukcji wzorców ekstrakcyjnych w punkcie 8.8 przedstawiono trzy metody pozwalające na określenie ograniczeń semantycznych:

1. ręczna ewaluacja zdań zawierających dopasowania wzorców formalnych,
2. ekstrakcja ograniczeń semantycznych z predykatów ontologii Cyc,
3. ekstrakcja ograniczeń semantycznych z wiedzy zgromadzonej w DBpedii.

Jednym z najważniejszych problemów postawionych w niniejszej pracy jest zweryfikowanie możliwości w pełni automatycznej konstrukcji wzorców ekstrakcyjnych. Oczywiście należy wziąć pod uwagę, że w prezentowanym algorytmie to użytkownik określa jaka relacja ma być ekstrahowana. Co więcej w podejściu opartym o DBpedię konieczne jest również ręczne określenie, które spośród predykatów zdefiniowanych w ontologii DBpedii odpowiadają ekstrahowanej relacji. Zatem mówiąc o automatyczności mamy tutaj na myśli przede wszystkim możliwość uniknięcia ręcznej oceny dużej ilości danych (zdań zawierających relację, wyekstrahowanych ograniczeń semantycznych, itp.), a nie całkowite zwolnienie użytkownika systemu z podejmowania jakichkolwiek działań.

Dlatego też określenie ograniczeń semantycznych realizowane jest na 3 sposoby. Pierwszy z nich reprezentuje podejście, w którym konieczne jest przeprowadzenie ręcznej oceny zdań, zatem nie spełnia wymogu automatyczności. Drugie podejście, oparte o ontologię Cyc, wymaga od użytkownika jedynie wskazania najbardziej ogólnego predykatu, który reprezentuje ekstrahowaną relację. Trzecie podejście

[2]Obliczone na podstawie rozkładu Bernoulliego.

oparte o DBpedię, wymaga pewnych działań ze strony użytkownika (określenie zbioru predykatów odpowiadających relacji oraz określenie kolejności występowania argumentów), ale ograniczone są one do niezbędnego minimum. Można również przypuszczać, że w trakcie rozwoju ontologii DBpedii większa część predykatów zostanie połączona w hierarchię, dzięki czemu możliwe będzie wskazanie jedynie najbardziej ogólnego predykatu, tak jak to ma miejsce w przypadku ontologii Cyc.

Jeśli okaże się, że podejście drugie lub trzecie dają wyniki lepsze od podejścia pierwszego, będzie to znaczyło, że teza niniejszej pracy jest obroniona. Oczywiście należy przyjąć pewne założenia wstępne w odniesieniu do pierwszej metody, gdyż w skrajnym przypadku, w drodze ręcznej oceny wszystkich dopasowań wzorca formalnego możliwe byłoby pozyskanie dokładnie tych samych ograniczeń semantycznych, które zostały określone w sposób automatyczny na bazie Cyc czy DBpedii. W takiej sytuacji metoda ręczna zawsze dawałaby wyniki co najmniej tak samo dobre jak pozostałe metody.

Przyjęto zatem, że ręcznej ocenie poddanych zastanie 10% wyników dopasowania wzorców formalnych relacji, zakładając, że oba argumenty posiadają przypisane kategorie semantyczne oraz że minimalny poziom pewności ujednoznaczniania $P_{dg} \geq 0,2$. Pierwsze założenie ma charakter praktyczny – konieczność ręcznej weryfikacji 10% wyników dopasowania wzorców formalnych w warunkach eksperymentalnych nie jest wysokim wymaganiem (sprowadza się do oceny kilkuset zdań), natomiast w praktycznym zastosowaniu algorytmu do ekstrakcji danych z notatek prasowych oznacza 10 krotną redukcję pracy[3].

Wymóg dotyczący kategorii semantycznych jest dość oczywisty – niemożliwe jest określenie ograniczeń semantycznych, jeśli chociaż jeden spośród argumentów nie posiada przypisanej kategorii semantycznej. Ostatnie wymaganie, dotyczące minimalnego progu pewności, wynika z analizy przeprowadzonej w punktach 7.3 oraz 9.6. Biorąc pod uwagę fakt, że skoro precyzja algorytmu ujednoznaczniania wyrażeń w notatkach PAP wynosi około 80%, a ujednoznaczniania dwóch argumentów około 60% (patrz tabela 10.1), znaczna część analizowanych przykładów zawierałaby niepoprawnie ujednoznacznione wyrażenie. Określając wartość parametru P_{dg} na 0,2 możliwe jest podniesienie poprawności ujednoznaczniania obu argumentów do wartości około 80%, co jest wynikiem znacznie lepszym. Ponadto końcowe eksperymenty przeprowadzane na notatkach PAP również zakładały ten sam minimalny poziom pewności ujednoznaczniania.

Przyjmując te założenia można w dość realistyczny sposób porównać wszystkie metody określania ograniczeń semantycznych i zweryfikować tezę o możliwości automatycznej ekstrakcji relacji semantycznych.

9.8.1. Ograniczenia semantyczne pozyskane z ręcznej oceny zdań

Przyjęcie założeń opisanych w poprzednim punkcie, w szczególności zaś wymogu co do minimalnego poziomu pewności dopasowania $P_{dg} \geq 0,2$ spowodowało, że spośród ponad 13 tys. zdań, w których występowały wyrażenia dopasowane do wzorców formalnych relacji *całość-część*, z których każde posiadało przynajmniej jedną kategorię semantyczną, pozostała grupa 6816 zdań. Z tej grupy zdań wylosowano 682 zdania (10%), w których dokonano ręcznej oceny występowania relacji *całość-część*. W wyniku ewaluacji zdań ustalono, że 116 z nich (17%) zawiera wystąpienie tej relacji. Zdania te zostały użyte do zbudowania listy ograniczeń semantycznych zawierającej 405 pozycji. Liczba par ograniczeń semantycznych jest większa od liczby zdań zawierających relację, ponieważ, zgodnie z opisem z punktu 8.8.1, jako ograniczenia

[3]Należy jednak wziąć pod uwagę, że oceniane byłyby wyłącznie te zdania, w których rozpoznano wystąpienie wzorca formalnego. Zatem w praktyce wymagałoby ocenienia bardzo niewielkiej części wszystkich zdań przetwarzanych przez algorytm.

Tablica 9.7: Przykładowe pary ograniczeń semantycznych dla relacji *całość-część* pozyskane na podstawie ręcznej oceny zdań z korpusu PAP, zawierających dopasowania formalnych wzorców relacji.

Całość	**Część**
#$IndependentCountry	#$IntelligenceAgency
#$City	#$CityCouncil
#$InternationalOrganization	#$Institution
#$DemocraticGeopoliticalEntity	#$GovernmentAgency
#$Coal	#$UnitOfMass
#$IndependentCountry	#$CapitalCityOfRegion
#$GeopoliticalEntity	#$Diplomat

semantyczne brane były wszystkie pary kategorii semantycznych należących do iloczynu kartezjańskiego kategorii semantycznych wyrażeń, dla których rozpoznano wystąpienie zadanej relacji.

Przykładowe ograniczenia semantyczne wyekstrahowane w ten sposób przedstawione są w tabeli 9.7. Charakterystyczną cechą ograniczeń uzyskanych na podstawie ręcznej oceny zdań jest wysoka częstość występowania pojęć związanych z organizacjami międzynarodowymi. Nie jest to wynik zaskakujący, jeśli weźmiemy pod uwagę fakt, że zdania podlegające ręcznej ocenie pochodzą z korpusu notatek PAP.

9.8.2. Ograniczenia semantyczne pozyskane z ontologii Cyc

Określanie ograniczeń semantycznych dla relacji *całość-część* na podstawie ontologii Cyc odbywa się zgodnie z procedurą opisaną w punkcie 8.8.2. W pierwszej kolejności eksportowane są wszystkie asercje zawierające na pierwszym miejscu predykat `#$relationAllExists`. W kroku następnym z asercji tych ekstrahowane są krotki (R, a_1, a_2). W ontologii ResearchCyc w wersji 4.występuje ponad 47 tys. takich asercji. W następnym kroku wybierany jest predykat odpowiadający relacji semantycznej *całość-część* – `#$parts`. Predykat ten posiada w tej wersji Cyc 965 predykatów, które stanowią jego specjalizacje. Dzięki temu możliwe jest automatyczne określenie krotek, wygenerowanych na podstawie predykatu `#$relationAllExists`, które stanowią ograniczenia semantyczne relacji *całość-część*. Zbiór uzyskanych w ten sposób ograniczeń zawiera 3679 par, natomiast po odfiltrowaniu zbyt ogólnych ograniczeń pozostają w nim 3543 pary ograniczeń semantycznych. Przykładowe pary ograniczeń semantycznych dla relacji *całość-część* uzyskane na postawie ontologii Cyc przedstawione są w tabeli 8.6.

9.8.3. Ograniczenia semantyczne pozyskane z DBpedii

Ograniczenia semantyczne pozyskiwane są z DBpedii zgodnie z algorytmem opisanym w puntach 7.4.4 oraz 8.8.3. W przeprowadzonych eksperymentach została wykorzystana angielska DBpedia w wersji 3.9 (utworzona na podstawie angielskiej Wikipedii z marca 2013 roku) dostępna pod adresem `http://wiki.dbpedia.org/Downloads39`.

W pierwszej kolejności ustalona została lista predykatów zdefiniowanych w obrębie przestrzeni nazw `http://dbpedia.org/ontology/`. Asercje utworzone z wykorzystaniem tych predykatów powstają na bazie ręcznego mapowania pomiędzy elementami infoboksów a ontologią DBpedii (patrz p. 7.4.4). Liczba tak uzyskanych predykatów wyniosła 1358. Następnie dla każdego predykatu z tej listy pobierane były występujące w DBpedii asercje zawierające ten predykat. Ze względu na to, że niektóre predykaty posia-

dają bardzo dużą liczbę asercji, ograniczono ich liczbę do pierwszy 300 tys., korzystając z klauzuli `LIMIT` języka SPARQL. Wynikiem tego procesu był zbiór ponad 10 milionów asercji.

W następnej kolejności dla każdego predykatu oraz pary kategorii semantycznych, obliczana jest wartość wsparcia G_T (patrz p. 7.4.4). Ponieważ w asercjach występują identyfikatory zasobów DBpedii (odpowiadające artykułom Wikipedii) wartość tego współczynnika jest obliczana tylko na podstawie tych asercji, w których oba argumenty otrzymały kategorię semantyczną określoną na podstawie algorytmu opisanego w punkcie 7.2. Otrzymany w ten sposób zbiór trójek (R, sc_1, sc_2) zawierał ponad 1,4 miliona elementów.

Aby poprawić jakość oraz szybkość przetwarzania tak uzyskanych ograniczeń semantycznych, w następnym kroku ograniczenia te były filtrowane na 2 sposoby. W pierwszej kolejności usunięto z tego zbioru ograniczenia, które były niezgodne z ręcznie zdefiniowanymi ograniczeniami argumentów poszczególnych predykatów zdefiniowanych w ontologii DBpedii. Ponieważ ograniczenia te wyrażone są w formie klas (kategorii semantycznych) ontologii DBpedii, a w otrzymanym zbiorze wykorzystane zostały kategorie semantyczne Cyc, w kroku tym wykorzystano mapowanie pomiędzy pojęciami występującymi w obu ontologiach (patrz p. 7.2.4). Weryfikacja spełniania ręcznych ograniczeń odbywała się z wykorzystaniem wywołania `genls?` w API Cyc – jeśli obie kategorie określone w sposób automatyczny generalizowały się do ograniczeń określonych ręcznie, to dana para ograniczeń była zachowywana. Podobnie postępowano, jeśli w ontologii DBpedii zdefiniowano ograniczenie tylko dla jednego argumentu – wtedy wystarczyło aby automatycznie określone ograniczenie generalizowało się do ograniczenia ręcznego, przyjmując, że nieokreślony argument jest typu `Thing`. Jednakże w sytuacji, gdy żaden z argumentów predykatu nie posiadał ograniczenia ręcznego, dany predykat oraz wszystkie jego ograniczenia semantyczne były usuwane, przyjmując, że jest on niezdefiniowany. W wyniku tej weryfikacji, z początkowego zbioru 1,4 mln. ograniczeń pozostało ich nieco ponad 660 tys.

Aby dodatkowo przyspieszyć działanie algorytmu, z otrzymanego zbioru usunięto ograniczenia, których współczynnik wsparcia G_T był mniejszy niż 5. W ten sposób zbiór ograniczeń został zmniejszony do 30782 pozycji, czyli został zredukowany ponad 20-krotnie. Dopiero na tym zbiorze przeprowadzona została operacja określenia prawdopodobieństwa warunkowego wystąpienia określonego predykatu pod warunkiem wystąpienia określonej pary ograniczeń semantycznych (patrz punkty 7.4.4 i 8.8.3) oraz wybór najbardziej prawdopodobnych ograniczeń. W efekcie końcowy zbiór zawierał 20901 pozycji.

Należy zwrócić uwagę, że przedstawiony dotychczas opis ekstrakcji ograniczeń semantycznych z DBpedii nie brał pod uwagę faktu, że jego celem była ekstrakcja relacji *całość-część*. Zatem przedstawiony wynik dotyczy wszystkich predykatów – zarówno tych reprezentujących jak i niereprezentujących tę relację. Takie podejście do problemu pozwoliło uniknąć żmudnego procesu przeglądania 1358 predykatów zdefiniowanych w DBpedii i pozwolił skoncentrować się jedynie na 400 predykatach, które faktycznie mogły być wykorzystane przez mechanizm ekstrakcji relacji. Wśród tych predykatów, w wyniku ręcznej weryfikacji zidentyfikowano 39 predykatów reprezentujących relację *całość-część*. Kompletna lista tych predykatów znajduje się w tabeli E.1.

Efektem końcowym określania ograniczeń semantycznych relacji *całość-część* na podstawie DBpedii był zbiór 1662 par ograniczeń semantycznych. Przykładowe ograniczenia uzyskane w ten sposób, wraz z wartością przypisanego im prawdopodobieństwa warunkowego, przedstawione są w tabeli 8.10.

10. Wyniki ekstrakcji relacji *całość-część*

10.1. Wyniki dopasowania wzorców formalnych

W celu weryfikacji tezy pracy głoszącej, że hybrydowy algorytm ekstrakcji relacji daje wyniki bardziej precyzyjne niż algorytm oparty wyłącznie o statystyczna analizę formalnych cech relacji, wyniki dopasowania wzorców formalnych poddane zostały analizie ilościowej oraz jakościowej. W pierwszej kolejności zbadano precyzję tak uzyskanych wyników oraz zidentyfikowano podstawowe źródła błędów ekstrakcji. Następnie przeprowadzono analizę jakościową, której podstawowym celem było zidentyfikowanie problemów natury semantycznej, mogących mieć istotny wpływ na skuteczność algorytmu hybrydowego.

10.1.1. Analiza ilościowa

Ilościowa analiza dopasowań polegała na porównaniu wyników algorytmu dopasowującego wzorce formalne z odpowiedziami ludzi, którzy zostali poproszeni o ocenienie, czy relacja *całość-część* występuje we fragmencie tekstu, który został dopasowany do wzorca formalnego. Gdyby okazało się, że wzorce formalne są wystarczająco skuteczne, zdecydowana większość dopasowań powinna zawierać wystąpienie relacji. Ponieważ jednak ocena algorytmu rozpoznawania relacji zależała od algorytmu ujednoznaczniania pojęć, istotnym elementem oceny przykładów było również stwierdzenie, czy pojęcia wykryte w tekście zostały poprawnie ujednoznacznione.

Zbiór przykładów poddany ewaluacji zawierał ponad 1000 elementów, a jego charakterystyka ilościowa przedstawiona jest w tabeli 10.1. Największą grupę przykładów stanowiły te, oznaczone jako zawierające niepoprawne dopasowanie argumentów relacji, tzn. przynajmniej jeden z argumentów nie został właściwie ujednoznaczniony. Tak duża liczba błędnych rozpoznań może budzić obawy o poprawność działania algorytmu rozstrzygania wieloznaczności. Wynika one jednak przede wszystkim z tego, że nie ustalono minimalnego progu pewności dopasowania P_{dg}. Dzięki temu pewna liczba poprawnych rozpoznań, które zostałyby odrzucona, ze względu na niski współczynnik prawdopodobieństwa ich poprawności, została uwzględniona zwiększając liczbę przykładów, poddanych analizie. Należy również zwrócić uwagę, że wartość procentowa liczby niepoprawnych dopasowań jest znacznie wyższa, niż przedstawiona w punkcie 7.3.5 (tutaj 40% wobec 20% we wcześniejszej ewaluacji).

Wynik ten ma dwie przyczyny: po pierwsze, wystarczyło aby jeden z argumentów relacji został niepoprawnie ujednoznaczniony, aby cały przykład był uznawany za niepoprawny. Przyjmując, że poprawność rozpoznań bez ustalonego minimalnego progu P_{dg} wynosi 80%, wystąpienie dwóch poprawnych rozpoznań z rzędu, zgodnie z rozkładem Bernoulliego wynosi $P = 0.8^2 = 0.64$. Otrzymany wynik wynoszący 58,5% jest zatem dość zgodny z przewidywaniami teoretycznymi. Po drugie zaś, ocena w tym scenariuszu dokonywana była przez dwie osoby niezależnie, a we wcześniejszej, wstępnej ewaluacji tylko przez jedną. Ten fakt miał również wpływ na obniżenie całościowego wyniku ewaluacji, gdyż wystarczyło, aby jedna osoba wskazała ujednoznacznienie jako błędne, aby wynik rozpoznania uznawany był za błędny.

Drugą istotną grupę stanowią przykłady, w których relacja *całość-część* nie występuje. Te wyniki pokazują wyraźnie, że opieranie się jedynie na wiedzy zawartej we wzorcach formalnych prowadzi wprost do wyników niskiej jakości, gdyż stosunek liczby przykładów zawierających relację, do liczby przykładów niezawierających relacji jest w przybliżeniu 1:4 (wynik ten jest zgodny z wynikiem przedstawionym wcześniej w punkcie 8.7).

Ostatnią interesującą grupą jest zbiór zawierający przykłady problematyczne. Część z tych przykładów została omówiona w punkcie 10.1.2. Zostały one oznaczone w ten sposób, ponieważ były to przykłady, w których odpowiedzi osób określających występowanie relacji różniły się miedzy sobą lub przynajmniej jedna z osób oznaczyły określony przykład jako problematyczny. Widać wyraźnie, że liczba przykładów tego rodzaju była niemal taka sama jak liczba przykładów zawierających wystąpienie relacji. Ten wynik wskazuje, że jednoznaczne rozstrzygnięcie, czy w tekście występuje relacja *całość-część* stanowi wyzwanie nie tylko dla komputera, ale również dla ludzi.

Tablica 10.1: Charakterystyka zdań dopasowanych do wzorców formalnych relacji *całość-część*.

Typ zdania	**Liczba**	**Udział %**
zawierające relacje	110	10,2
niezawierające relacji	416	38,6
niepoprawne dopasowanie	447	41,5
przykład problematyczny	105	9,7
w sumie	**1078**	**100,0**

10.1.2. Analiza jakościowa

Analiza jakościowa dopasowań wzorców formalnych polegała na skrupulatnym przejrzeniu zdań ze zbioru zawierającego dopasowania wzorców formalnych, w celu wykrycia zjawisk które należałoby uwzględnić w trakcie ulepszania algorytmu ekstrakcji oraz zjawisk, które nie spełniają przyjętych założeń jego działania. W szczególności chodzi tutaj o zdania, w których trudno było jednoznacznie stwierdzić występowanie poszukiwanej relacji. Przykłady te są o tyle istotne, że ilustrują trudność w konstrukcji precyzyjnego algorytmu ekstrakcji informacji.

Przykłady zawierające relację

Pierwszą grupę stanowią przykłady oznaczone jako zawierające relację *całość-część*. Analizując te przykłady można w szczególności określić kilka podtypów tej relacji, które pojawiły się w wynikach działania algorytmu. Pierwszym podtypem relacji jest relacja wiążąca *organizacje* z jej *częściami*. Przykładowo w zdaniu[1]

> Prezes Amerykańskiej Akademii Filmowej Robert Rehme zarzuca *redakcji* ***gazety*** Wall Street Journal, że przeprowadziła wśród członków Akademii prywatna ankietę, chcąc przedwcześnie ujawnić laureatów Oscarów,

redakcja jest częścią organizacji jaką jest *gazeta*.

Podobnie w zdaniach:

– Wrocławska *klinika* ***Akademii Medycznej*** jako pierwsza w Polsce zacznie niedługo stosować terapię genową,

[1]Wszystkie przykłady w tym rozdziale pochodzą z korpusu PAP.

– Odnieśliśmy ogromną satysfakcję – powiedział PAP Kaziemierz Kord po zakończeniu polskiego tournee Chóru i *Orkiestry* ***Filharmonii Narodowej*** w Warszawie,

– Bomba, zapakowana w niewielką torbę, była umieszczona pod samochodem zaparkowanym na placu przed bazarem – podało Ministerstwo Spraw Wewnętrznych Osetii Północnej, autonomicznej *republiki* ***Federacji Rosyjskiej***,

mamy do czynienia z tym samym podtypem relacji *całość-część*.

Drugim, często występującym podtypem był typ wiążący region z jego stolicą, np.

> W dniach 12-14 czerwca odbędzie się w Phenianie – *stolicy* ***Korei Północnej*** konferencja na szczycie obu państw koreańskich.

Podtyp ten jest w pewnym stopniu zbliżony do podtypu podorganizacji, niemniej jednak niesie w sobie dodatkową informację o wyróżnionej roli, jaką jedna z części pełni w stosunku do innych części. Ze względu na charakter korpusu podtyp ten występował najczęściej w odniesieniu do stolic krajów:

– Jest pierwszym obcokrajowcem wyróżnionym w ten sposób przez *stolicę* ***Dolnego Śląska***,

– „Prezydent byłby bardzo szczęśliwy, gdyby mógł gościć króla Jordanii w swej rezydencji w Jerozolimie, *stolicy* ***Izraela***, natomiast nie uda się do Tel Awiwu" – powiedział rzecznik szefa państwa,

– Izrael traktuje natomiast obie części miasta jako jednolitą *stolicę* ***państwa żydowskiego*** – czego nie uznaje wspólnota międzynarodowa,

– Natomiast główna prokurator haskiego trybunału do spraw zbrodni w byłej Jugosławii, Carla del Ponte, powiedziała na konferencji prasowej w Prisztinie, *stolicy* ***Kosowa***, że Kosztunica powinien przekazać Miloszevicia do Hagi,

– Premier Chin Zhu Rongji przybył – jako pierwszy przywódca chiński – do *stolicy* ***Unii Europejskiej*** i rozpoczął rozmowy, w których przedstawiciele UE oprócz handlu podnoszą kwestię praw człowieka.

Należy zwrócić uwagę, że o ile w literalnym znaczeniu, w każdym z tych przykładów mowa jest o stolicy określonego regionu, o tyle rozpoznanie relacji nie zawsze prowadzi do jednoznacznych rezultatów. W szczególności w przykładzie dotyczącym stolicy Izraela, gdzie mowa jest o rozbieżnych punktach wiedzenia Izraela oraz wspólnoty międzynarodowej. Dlatego też każdy algorytm, który nie analizuje w pełni treści zdania, w szczególności algorytm opisywany w niniejszej pracy, nie może produkować całkowicie poprawnych wyników. Zwykle zakłada się bowiem, że dane uzyskane na wyjściu algorytmu są *prawdziwe*, rzadko jednak uwzględnia się fakt, że pojęcie prawdy jest kontekstualne, a prawdziwość wyników algorytmu powinna być określana kontekstowo. Problem ten zaczęto zauważać np. w kontekście baz danych tworzonych w ramach Semantic Web, gdzie coraz częściej oczekuje się, że poza surowymi danymi, zostaną również udostępnione informacja na temat tego skąd dane pochodzą i w jakim kontekście są prawdziwe [22].

Kolejnym ważnym podtypem jest relacja wiążąca organizm z jego anatomicznymi częściami. Występowanie przykładów tej relacji nie powinno być zaskoczeniem, gdyż ona właśnie została użyta w celu wygenerowania wstępnego zbioru zdań służących do tworzenia wzorców formalnych. Oto przykładowe zdania odnalezione w korpusie PAP zawierające ten podtyp:

– Brytyjski chirurg James Shapiro z uniwersytetu stanowego w kanadyjskim stanie Alberta ogłosił, że udało mu się z powodzeniem przeszczepić *komórki* ***trzustki*** zdrowych dawców ośmiu pacjentom, u których rozpoznano chroniczny stan cukrzycy,

– Komórki *skóry* ***rybki*** produkują chroniący ją przed poparzeniem płyn, który może być zastosowany również do ochrony ludzi,

– Zmodyfikowana genetycznie *komórka* ***bakterii*** może służyć do przetwarzania informacji – donosi najnowszy „New Scientist".

Nawet w tych, dość oczywistych przykładach wystąpienia relacji, można natknąć się na pewne wątpliwości. W ostatnim przykładzie, w którym mowa o *komórkach bakterii*, należy zastanowić się, czy jeśli bakteria jest organizmem jednokomórkowym, to ta jedyna jej komórka jest jej częścią. Uogólniając to pytanie można zapytać: czy całość może być swoją częścią. Odpowiedź na to pytanie nie jest oczywista. Jeśli uznamy, że tak, to intuicyjna różnica, która występuje pomiędzy pojęciami *całość* i *część* zostanie zanegowana. Jeśli natomiast uznamy, że nie, będziemy mieli problem z określeniem relacji jaka występuje w wyrażeniach takich jak *komórka bakterii*, czy *ciało zwierzęcia*. Można również uznać, że to pierwsze wyrażenie jest pleonazmem.

Interesującym i posiadającym zróżnicowany zbiór przykładów jest podtyp relacji *całość-część* obejmujący relacje występujące pomiędzy fizycznymi obiektami. Przykładowe zdanie, w których rozpoznano ten typ relacji przedstawione są poniżej:

– Protestujący, niektórzy z dziećmi, przeszli przez główne *ulice* ***miasta***, śpiewając i wznosząc okrzyki,

– Interweniowała policja, ale bójka była kontynuowana w *tunelu* ***stadionu***,

– *Kopuła* ***bazyliki Św. Piotra*** i pozostałe części świątyni otrzymają, częściowo lub w całości, pokrycie z najszlachetniejszej chilijskiej miedzi,

– Dobiega końca renowacja *wieży* ***sanktuarium*** na Jasnej Górze w Częstochowie.

Ponieważ przykłady te mogą być potraktowane jako prototypowe wystąpienia tej relacji, żaden z nich nie nastręcza większych problemów interpretacyjnych. Warto jednak przyjrzeć się następującym przykładom:

– Niewielkie zmiany w dynamice ruchu wewnętrznych planet Układu Słonecznego mogły zwiększyć tempo bombardowań *powierzchni* ***Ziemi*** przez niewielkie planetoidy, co było przyczyną wyginięcia dinozaurów 65 mln lat temu,

– Na Marsie trwa obecnie największa burza pyłowa od 1997 roku, która obejmuje prawie połowę *powierzchni* ***planety*** – poinformowała NASA,

– Tymczasem dwaj żołnierze jugosłowiańscy zginęli, gdy ich samochód wjechał na minę w pobliżu miasta Preszewo w południowej Serbii, tuż za strefą buforową, która biegnie wzdłuż *granicy* ***Kosowa*** z resztą Jugosławii i styka się z granicą Macedonii,

– Władze NATO zdecydowały o redukcji strefy bezpieczeństwa rozciągającej się wzdłuż *granicy* ***Kosowa*** i Serbii właściwej oraz zgodziły się na powrót armii jugosłowiańskiej do kolejnego sektora strefy.

Przykłady te pokazują, że *stricte* fizyczna relacja *całość-część* również może rodzić problemy interpretacyjne. W szczególności przykłady z powierzchnią: planety Mars oraz Ziemi pokazują, że dokładnie to samo sformułowanie użyte w bardzo podobnym kontekście nie może być w całości interpretowane tak samo. W pierwszym przypadku, niewątpliwe mamy do czynienia z *powierzchnią* jako częścią *Ziemi*, gdyż bombardowania dotyczą Ziemi i jej szczególnej części, czyli jej powierzchni. W drugim przypadku natomiast, mowa jest o powierzchni planety jako pewnej mierze. O ile połowa powierzchni jest niewątpliwie

częścią powierzchni, to nie jest oczywiste, że powierzchnia w tym wypadku jest częścią planety. Chodzi tutaj raczej o miarę w sensie matematycznym, a nie część w sensie fizycznym.

Z innym zjawiskiem mamy do czynienia w drugiej parze przykładów. *Granica* jest skrajną częścią *państwa*. Ale w powyższych przykładach wyraźnie widać, że granica nie jest częścią tylko jednego obiektu. Ta cecha granicy odróżnia ją od innych części fizycznych, od których wymagamy, żeby przynależały tylko do jednego obiektu-całości. Widać również, że algorytm stosujący proste dopasowania wzorców nie jest w stanie rozpoznać złożonych, dwuargumentowych relacji semantycznych, które wymagają głębszej analizy syntaktycznej oraz semantycznej przetwarzanego zdania.

Kolejny podtyp relacji *całość-część* może być nazwany podtypem *pojemnik-zawartość*. W tym wypadku relacja występuje pomiędzy pewną całością, która ma charakter obiektu o kształcie fizycznego lub abstrakcyjnego pojemnika oraz częścią, która jednak stanowi niezbywalną część całości. W analizowanych przykładach ten typ relacji pojawia się w opisie akwenów:

– Podnoszenie się temperatury na Ziemi, spowodowane emisją gazów cieplarnianych do atmosfery ogrzewa *wody* ***oceanów***, a te przez tysiące lat zatrzymują w swych głębiach to ciepło – podaje najnowszy numer „Science",

– W Belgradzie 71-letni nauczyciel wychowania fizycznego Aleksandar Vasak po raz pięćdziesiąty w swym życiu powitał Nowy Rok skokiem do lodowatej *wody* ***rzeki Sawa***, ale oświadczył, że był to ostatni raz,

– W czasie wizyty w heroicznie odpierającym wielką *wodę* ***Wisły*** mieście, szef rządu obiecał też, że dzieci powodzian otrzymają wyprawki do szkoły.

Przykłady te są o tyle interesujące, że zawartość pojemnika jest jednocześnie jego częścią. Należy zwrócić uwagę, że istnieją również symetryczne względem nich przykłady, w których zawartość stanowi całość, np. *szklanka wody* jest pewną częścią całej wody we wszechświecie, podobnie jak *kieliszek wódki*, czy *kostka masła*. W tych przykładach „pojemnik" służy do odmierzania części, zatem jego zawartość jest częścią, ale nie pojemnika, lecz całości jaką jest odmierzana substancja.

Ostatnim ważnym typem przykładów występujących w korpusie, były te, w których całość i część miały charakter temporalny:

– Wstęp na wszystkie *koncerty* ***festiwalu*** jest wolny,

– Polskie Zakłady Zbożowe skupiły od rolników w pierwszym *tygodniu* ***kwietnia*** 32,1 proc. zboża mniej niż tydzień wcześniej,

– Zakłady mięsne skupiły na przełomie lutego i marca o 2,9 proc. mniej trzody chlewnej oraz o 16,9 proc. więcej bydła niż w trzecim *tygodniu* ***lutego***.

Relacja ta ma charakter temporalny, ponieważ w pierwszym przypadku jest to jedno ze zdarzeń, które stanowi część innego zdarzenia, a w pozostałych dwóch przypadkach mamy do czynienia z pewnym okresem i jego wydzieloną częścią. O ile przytoczone przykłady nie są problematyczne, o tyle inne przykłady zawierające relacje temporalne nastręczają znacznie więcej problemów. W szczególności jeśli określona relacja występuje pomiędzy zdarzeniem, a mniej lub bardziej określonym czasem jego wystąpienia.

Wyróżnione podtypu relacji *całość-część* korespondują z podtypami opisanymi w literaturze i przytoczonymi w punkcie 3.2.6. Najlepiej reprezentowane są *kompozycja* (relacje fizyczne pomiędzy obiektami materialnymi), *bycie członkiem* (w szczególności relacje w obrębie organizacji) oraz *materiał* (budowa ciała). Niektóre wyróżnione podtypy nie są jednak tak szeroko opisywane w literaturze – np. przykłady

dotyczące *powierzchni* i *granicy* trudno zaklasyfikować do podziału przedstawionego w punkcie 3.2.6. Podobnie jest z relacjami temporalnymi, na co zwróciliśmy już uwagę wcześniej.

W trakcie analizy jakościowej powyższe przykłady zostały zakwalifikowane jako zawierające relację *całość-część*. Niemniej jednak nawet w tych dosyć oczywistych przypadkach widać, że analiza semantyczne jest procesem bardzo złożonym i w wielu sytuacjach odpowiedź na pytanie o występowanie określonej relacji semantycznej, nie musi być oczywista dla człowieka. Co więcej, stwierdzenie jej występowania często uzależnione jest od właściwego zrozumienia pełnego kontekstu, w którym określone zdanie występuje.

Przykłady problematyczne

Drugą grupę analizowanych przykładów stanowiły przykłady, co do których osoby weryfikujące występowanie relacji nie zgadzały się w swej ocenie, bądź przynajmniej jedna z osób oznaczyła je jako przykład problematyczny. W części przykładów należących do tej grupy można zidentyfikować pewne bardziej ogólne zjawiska, które przyczyniają się do trudności w analizie semantycznej. Natomiast spora część przykładów problematycznych stanowi przypadki jednostkowe. Analizę rozpoczniemy od przykładów należących do pierwszej kategorii.

Jednym z ważniejszych i powszechnie występujących w językach naturalnych zjawisk wpływających na analizę semantyczną jest zjawisko metafory (porównaj p. 2.3.5). Może być ono zilustrowane następującym przykładem:

– Trzy kraje Europy Środkowo-Wschodniej – Czechy, Węgry i Bułgaria – wymienione zostały w raporcie na temat rasizmu w Europie opublikowanym przez jeden z *organów* ***Rady Europy***.

W swoim podstawowym znaczeniu *organ* rozumiany jest jako część organizmu. W przytoczonym przykładzie *organ* występuje jednak w znaczeniu metaforycznym. Jak wiadomo *Rada Europy* nie jest organizmem w sensie literalnym, dlatego też nie może posiadać organów. Niemniej jednak ze względu na przytoczone użycie metaforyczne, w powyższym zdaniu mamy do czynienia z relacją *całość-część*. Dzieje się tak dlatego, że wyróżnia się tylko jeden aspekt *organu* – bycie częścią całości, pomijając istotną dla niego przynależności do organizmu żywego. *Rada Europy* traktowana jest metaforycznie jako pewien *organizm*, który posiada swoje *części składowe-organy*.

Pewnym rozwiązaniem tego problemu jest rozbudowanie słownika semantycznego używanego do analizy semantycznej, w taki sposób, by zawierał niektóre użycia metaforyczne. Niestety rozwiązanie to jest z góry skazane na niepowodzenie, ponieważ zjawisko metafory nie posiada ograniczeń. Użytkownik języka może zawszy wybrać tylko pewien podzbiór cech, wiążących się z określonym znaczeniem i w ten sposób konstruować nowe użycia, nie spełniające ogólnych ograniczeń semantycznych, o ile podejrzewa, że jego rozmówca będzie w stanie rozpoznać to zjawisko. Dlatego też odpowiedni słownik musiałby zawierać nieskończenie wiele możliwych interpretacji znaczeń, co oczywiście byłoby całkowicie niepraktyczne. Choć metafory rządzą się pewnymi prawami, to praw tych nie da się ująć w ramy prostego algorytmu ekstrakcji relacji semantycznych.

Drugim, dosyć częstym zjawiskiem związanym z rozpoznawaniem relacji semantycznych, które można już było zaobserwować w zbiorze przykładów oznaczonych jako zawierających relację, jest zjawisko wieloaspektowej kategoryzacji argumentów relacji. Występuje ono w przykładzie należącym do pierwszej grupy – obejmującej zdania zawierające wystąpienie poszukiwanej relacji:

– Prezes Amerykańskiej Akademii Filmowej Robert Rehme zarzuca *redakcji* ***gazety*** „Wall Street Journal", że przeprowadziła wśród członków Akademii prywatna ankietę, chcąc przedwcześnie ujawnić laureatów Oscarów.

Otóż wyrażenie *gazeta* może być interpretowane trojako – jako *organizacja*, jako *produkt*, który tworzony jest przez tę organizację oraz jako *budynek*, w którym ta organizacja się znajduje. Zjawisko to nazywamy polisemią systematyczną. W drugiej interpretacji *redakcja* nie jest częścią *gazety* lecz jest zbiorem osób, które bezpośrednio odpowiadają za stworzenie *produktu-gazety*, w odróżnieniu od np. innych osób zatrudnionych w tej organizacji. Należy zwrócić uwagę, że większość algorytmów ekstrakcji informacji w ogólności oraz ekstrakcji relacji w szczególności, zakłada, że istnieje jedna wyróżniona kategoria semantyczna (bądź zbiór przecinających się kategorii), do których należą wyrażenia w analizowanych tekstach [92]. Powoduje to, że rozpoznanie relacji dla pojęć tego rodzaju, w pewnych kontekstach zawsze będzie dawać błędne wyniki. Jeśli uznamy, że *gazeta* jest wyłącznie *organizacją* to trudno będzie nam zinterpretować zdanie: „W ostatnim wydaniu „Wall Street Journal" można przeczytać niezwykle interesujący wywiad...", gdyż organizacja nie może mieć *wydań*. Podobnie uznanie, że pojęcie to odnosi się wyłącznie do *produktu*, uniemożliwi nam poprawne zinterpretowanie wcześniejszego zdania. Jeśli zatem algorytm ekstrakcji informacji polega na jednoaspektowej klasyfikacji pojęć, może mieć istotne problemy z właściwą analizą tego rodzaju przykładów.

Podobnie jak w przypadku metafory, tak i w tym, rozwiązaniem problemu mogłoby być odpowiednie rozbudowanie słownika semantycznego, względem którego ujednoznaczniane są wyrażenia. W słowniku tym można by wyróżnić jako odrębne sensy: „Wall Street Journal" jako *organizację* oraz jako *produkt*. To rozwiązanie również jednak jest dosyć problematyczne, ponieważ obejmowałoby bardzo wiele wyrażeń, które wykazują tego rodzaju dualizm (np. *organizacje* i ich *siedziby*, takie jak np. uniwersytet), przez co istotnie zwiększyłaby się wieloznaczność wyrażeń występujących w słowniku. Ponadto w niektórych kontekstach wyrażenia tego rodzaju ujawniają jednocześnie oba aspekty semantyczne (np. „W ostatnim wydaniu *Gazety Wyborczej* jej redakcja odcina się od spekulacji na temat katastrofy...", „Wczoraj na *Wydziale Filozoficznym UJ* spotkałem jego dziekana."). Wydaje się, że najlepszym rozwiązaniem tego problemu jest stworzenie wieloaspektowej kategoryzacji wyrażeń i wybór jednej z kategorii w trakcie semantycznej interpretacji zdania. To rozwiązanie jednak jest trudne w realizacji, ponieważ odbiega od przyjętego modelu, w którym kategoria semantyczna przypisywana jest do wyrażenia, przed etapem ekstrakcji relacji.

Kolejnym problemem, z którym muszą zmierzyć się twórcy algorytmów ekstrakcji informacji są nazwy wielosegmentowe. Co prawda, wykorzystywane zasoby semantyczne, w szczególności Wikipedia, zawierają bogaty zbiór wyrażeń tego rodzaju, ale nie zawierają i nigdy nie będą zawierać wszystkich wyrażeń, które mogą pojawić się w zdaniach wybranego języka naturalnego.

Wśród dopasowań wzorców formalnych można wskazać następujące zdania:

– 79 prac Stasysa Eidrigeviciusa, plakacisty, rysownika, grafika, malarza oglądać można od piątku w warszawskiej *Galerii* ***Grafiki*** i Plakatu,

– W specjalnym schronisku dla egzotycznych zwierząt w USA zakończył swój żywot afrykański słoń „Sonny", którego kolejne *ogrody* ***zoo*** pozbywały się z powodu trudnego charakteru i skłonności do ucieczek – podały media.

W obu przypadkach wyrażenia będące argumentami relacji są poprawnie dopasowane, jednak stanowią one część tej samej nazwy wielosegmentowej. O ile rozpoznanie odpowiednich relacji semantycznych wewnątrz nazw nie jest błędem, o tyle stanowi problem interpretacyjny. Np. *ogrody zoo* są pewny szczególnym typem *ogrodów*, a nie częścią *zoo*, które same w sobie jest ogrodem. Zdecydowanie lepszym rozwiązaniem byłoby, gdyby algorytm rozpoznał je jako wyrażenia wielosegmentowe i nie rozpoznawał relacji wewnątrz nich. Prezentowany algorytm nie posiada jednak modułu odpowiedzialnego za wykrywanie ogólnej struk-

tury nazw własnych, a jedynie potrafi rozpoznać nazwy występujące w Wikipedii. Z tego też względu prezentowane przykłady są dla niego problematyczne.

Wśród ciekawych, jednostkowych przykładów trudnych w interpretacji można wymienić następujące zdania:

1. W *lasach* ***Gór Świętokrzyskich*** rozpoczął się – nie notowany zazwyczaj w lipcu – obfity wysyp grzybów jadalnych: borowików, czerwonych kozaków, maślaków oraz kani,

2. Czasową ekspozycją *fauny* ***Puszczy Białowieskiej*** wznowiło działalność Muzeum Przyrodniczo-Leśne Białowieskiego Parku Narodowego w Białowieży (podlaskie),

3. Wojciech Brzozowski (MOS Era Warszawa) okazał się najlepszy we wszystkich rozegranych dotychczas trzech wyścigach żeglarskich mistrzostw obu Ameryk w Formule Windsurfing PanAM, które odbywają się u *wybrzeży* ***stolicy*** Portoryko w San Juan.

W przykładzie pierwszym intuicyjnie można przyjąć, że lasy są częścią *Gór Świętokrzyskich*. Niemniej jednak bardziej dokładna analiza semantyczna wskazuje, że lasy w sensie fizycznym porastają *Góry Świętokrzyskie*. Problem ten bierze się stąd, że zarówno lasy jak i góry mogą być traktowane jako obiekty fizyczne – i w tym wypadku obiekty te jedynie stykają się ze sobą – oraz jako pewne obszary geograficzne – w tym wypadku las faktycznie stanowiłby część *Gór Świętokrzyskich*. Gdyby w przytoczonym przykładzie pojawiła się nazwa własna – *Puszcza Świętokrzyska* – można by jednoznacznie zinterpretować ten przykład. Natomiast zdanie w prezentowanym kształcie pozostawia pewną swobodę interpretacyjną, która stanowi problem dla algorytmu ekstrakcji relacji.

W drugim przykładzie mamy podobną sytuację – *Puszcza Białowieska* jest lasem, zaś dla *fauny*, czyli zwierząt w niej żyjących, jest to miejsce zamieszkania. Odnosząc tę sytuację do zjawisk typowych dla człowieka, trudno zgodzić się ze stwierdzeniem, że człowiek jest częścią mieszkania, w którym mieszka. Z drugiej jednak strony, w języku funkcjonuje sformułowania *fauna i flora*, które odnosi się do zwierząt i roślin w sensie kolektywnym. Niewątpliwie flora jest częścią puszczy. Czy zatem fauna jest również jej częścią? Pytanie to może mieć różne odpowiedzi w zależności od tego co rozumiemy przez *puszczę*. Jeśli jest to wyłącznie zbiór *roślin*, to *fauna* nie będzie jej częścią. Jeśli natomiast potraktujemy ją jako *ekosystem*, to odpowiedź będzie przeciwna. Przykład ten pokazuje, że decyzja o interpretacji określonego zdania zależy w tym wypadku od dalszego zastosowania tak otrzymanej wiedzy. Ze względu na niejednoznaczność odpowiedzi, nie może być ona udzielona na etapie ekstrakcji.

W ostatnim przykładzie mamy natomiast do czynienia z wypowiedzią niepoprawną. Stolica *Portoryko – San Juan* – zlokalizowana jest na wybrzeżu i stanowi część lądu. Ale sformułowanie *wybrzeża stolicy* jest niepoprawne, gdyż wybrzeża mogą być częścią lądu, natomiast nie miasta. Z drugiej jednak strony osoba analizująca tego rodzaju przykład jest w stanie zinterpretować komunikat nadawcy i na tej podstawie przyjąć, że *wybrzeże* jest częścią *San Juan*, przez co rozumie się ten fragment lądu, na którym usytuowane jest *San Juan*. W tym wypadku człowiek wykazuje się dużą elastycznością rozumienia tekstu, której trudno oczekiwać od komputera.

Powyższe przykłady wskazują wyraźnie, że interpretacja wielu wystąpień relacji semantycznych nie jest jednoznaczna również dla ludzi. Zjawisko to świadczy o trudności z jaką musi zmierzyć się konstruktor algorytmów ekstrakcji informacji, zarówno na etapie ich tworzenia, jak i ewaluacji. Jeśli bowiem znaczna część przykładów, na bazie których oceniane są algorytmy, stanowi problem interpretacyjny dla ludzi, to trudno oczekiwać, że uda się skonstruować algorytm który zawsze podaje „właściwe" odpowiedzi. Te zjawiska wskazują, że proces ekstrakcji informacji powinien być realizowany i oceniany w kontekście określonego, ściśle zdefiniowanego zadania. Dzięki temu jego ocena może być bardziej obiektywna.

Przykłady niezawierające relacji

Ostatnią grupę przykładów poddanych analizie, stanowiły przykłady, w których nie występowała relacja *całość-część*. Są one tutaj przywołane dlatego, że również wskazują problemy z jakimi musi zmierzyć się twórca algorytmu. Pokazują również, że ocena algorytmu ekstrakcji informacji przez osoby bez treningu językoznawczego, może prowadzić do niewłaściwych rezultatów.

Pierwszą grupę stanowiły przykłady, w których zamiast relacji *całość-część* występowała relacja *rezultatywna*:

– Proteina wytwarzana przez gruczoły limfatyczne zabija *komórki* ***raka płuc***,

– Toksyna wydzielana przez bakterie spowalnia wzrost *komórek* ***raka jelita grubego*** dających przerzuty – donoszą uczeni z USA na łamach najnowszego numeru pisma „Proceedings of the National Academy of Sciences".

Oba powyższe przykłady mają strukturę podobną do tej występującej w zdaniu:

– Zmodyfikowana genetycznie *komórka* ***bakterii*** może służyć do przetwarzania informacji – donosi najnowszy „New Scientist",

jednak w przytoczonych przykładach *rak* nie jest częścią organizmu, a w konsekwencji *komórki* nie są częścią *raka*. *Rak* jest chorobą, która prowadzi do degradacji *komórek*. Wyrażenie *komórki raka płuc* oznacza komórki, które zostały zaatakowane i zmodyfikowane przez tę chorobę. Dlatego też relacja, która występuje pomiędzy *rakiem* a *komórkami* jest relacją rezultatywną.

Inny przykład, który może być uznany przez niewprawioną osobę, jako zawierający relację *całość-część* jest następujący:

– Minister środowiska wydał zgodę na odstrzał sanitarny kolejnych dziewięciu żubrów w *lasach* ***nadleśnictwa Brzegi Dolne (Podkarpacie)***.

W tym przykładzie ze względu na pokrewieństwo wyrażeń *las* oraz *nadleśnictwo* można odnieść wrażenie, że *las* jest częścią *nadleśnictwa*. Niemniej jednak, *nadleśnictwo* jest pewną strukturą organizacyjną funkcjonującą w obrębie państwa, natomiast *las* jest obiektem fizycznym. Dlatego przykład ten jest dość nietypowym przykładem relacji *posesywnej*. *Lasy* są administrowane przez *nadleśnictwo* oraz na gruncie prawa jest ono ich właścicielem. Dlatego też w tym przypadku nie mamy do czynienia z relacją *całość-część*.

Z ciekawymi przykładami relacji mamy do czynienia w następujących zdaniach:

– Brazylia udostępni bazę wyrzutni rakiet kosmicznych Alcantara w stanie Maranhao amerykańskim kompaniom w celu wysyłania stamtąd w kosmos cywilnych *satelitów* ***Ziemi***,

– Astronomowie europejscy poinformowali, że odkryli osiem nowych planet, które mogą znajdować się na *orbitach* ***gwiazd*** poza Układem Słonecznym.

Związek jaki łączy *satelity* z *ziemą* czy *orbity* z *gwiazdami* wykazuje pewne cechy relacji *całość-część*. W szczególności pierwszy argument w każdym przypadku jest związany z drugim argumentem za pomocą relacji przestrzennej, typowej dla relacji *całość-część*. Przemieszczenie się *ziemi* lub *gwiazdy* powoduje również przemieszczenie się powiązanych z nimi *satelitów* oraz *orbit*. Z drugiej jednak strony brak tutaj typowego dla relacji *całość-część* powiązania materialnego, czy też stykania się wyróżnionego elementu z innymi, niewyróżnionymi elementami, dlatego w zdaniach tych nie występuje wskazana relacja. Najbezpieczniej można przyjąć, że wymienione elementy znajdują się w swojej fizycznej bliskości.

Bardziej kontrowersyjna (metaforyczna) interpretacja mogłaby przypisać w tym przypadku relację *posesywną*. Przykłady te pokazują, że nie tylko kategoryzacja poszczególnych pojęć nastręcza problemów interpretacyjnych, ale również relacje semantyczne muszą być czasami interpretowane metaforycznie. Jest to szczególnie trudne zagadnienie z punktu widzenia konstruktora algorytmu ekstrakcji tych relacji.

Kolejny przykład, w którym analizowana relacja nie występuje, nie wymaga szczególnych zabiegów interpretacyjnych:

– Beznikotynowe tabletki Zyban, dostępne już w W. Brytanii na recepty państwowej służby zdrowia, wpływają na zmianę *równowagi* ***związków chemicznych*** w mózgu, zostały życzliwie przyjęte przez organizacje zwalczające palenie papierosów.

W tym zdaniu pomiędzy wyróżnionymi wyrażeniami nie występuje relacja *całość-część*, ponieważ *równowaga* jest cechą, a nie częścią *związków chemicznych*. Niemniej jednak przykład ten pokazuje, że do właściwego zinterpretowania występującej relacji, czasami konieczne jest zbudowanie skomplikowanego modelu – *równowaga związków chemicznych* nie jest cechą pojedynczego *związku chemicznego*, lecz grupy związków chemicznych traktowanych kolektywnie. Przykład ten pokazuje na ile trudna jest analiza semantyczna zdań – w wielu przypadkach wymaga bowiem określenia, czy dane pojęcie (np. *związek chemiczny*) występuje w sensie kolektywny (jak w przytoczonym przykładzie), czy dystrybutywnym. Autor nie spotkał się jednak z żadnym algorytmem ekstrakcji informacji, który brałby pod uwagę to rozróżnienie.

Ostatnią, dużą grupę przykładów negatywnych stanowiły zdania, w których wzorzec formalny dopasowany został do relacji *typ-okaz*:

– Do nich doszły nagrody za najlepszy scenariusz i najlepszy debiut kobiecy dla *aktorki* ***Audrey Tautou***,

– Legenda *aktorki* ***Marleny Dietrich*** (1901-1992) powróciła na ekrany kinowe w Niemczech wraz z wtorkową premierą w Berlinie filmu „Marlene" w reżyserii Josepha Vilsmaiera,

– Ekspozycja liczy przeszło sto obrazów olejnych, akwarel i rysunków autorstwa Juliusza Kossaka i jego syna Wojciecha, córek Wojciecha – *poetki* ***Marii Pawlikowskiej-Jasnorzewskiej*** i *pisarki* ***Magdaleny Samozwaniec*** oraz Jerzego, Karola i Leona Kossaków.

We wszystkich powyższych przykładach mamy do czynienia z tym samym zjawiskiem stylistycznym – przed nazwiskiem znanej osoby pojawia się nazwa zawodu, z którym powszechnie jest lub była utożsamiana. Przykłady te są o tyle istotne, że wymagają specjalnego traktowania, ponieważ wiele ograniczeń semantycznych dla relacji *całość-część* dopuszcza sytuację, w której oba jej człony należą do tej samej kategorii semantycznej (np. *organizacja-organizacja*), dlatego na etapie określania ograniczeń semantycznych (porównaj p. 9.8) konieczne jest specjalne potraktowanie tych przypadków. Różnica jaka występuje pomiędzy obiema sytuacjami jest następująca: w przypadku relacji *typ-okaz* mamy do czynienia z występowaniem pewnego pojęcia ogólnego, reprezentowanego przez rzeczownik pospolity, po którym występuje nazwa własna, będąca egzemplarzem tego pojęcia; w przypadku relacji *całość-część* najczęściej oba człony są albo pojęciami ogólnymi (np. *partia partii*) albo jeden z nich jest nazwą własną, ale nie stanowiącą egzemplarza pojęcia-drugiego argumentu. W ten sposób stosunkowo łatwo można odróżnić te dwa typy relacji. Wymagają one jednak modyfikacji algorytmu, która została szczegółowo omówiona w punkcie 9.8.

10.2. Wyniki dopasowania wzorców ekstrakcyjnych

W celu zweryfikowania tezy pracy, głoszącej, że możliwa jest automatyczna ekstrakcja relacji semantycznych, przeprowadzono szereg eksperymentów porównujących wyniki ekstrakcji relacji *całość-część* z wykorzystaniem ograniczeń semantycznych pozyskanych ręcznie, na podstawie Cyc oraz DBpedii.

10.2.1. Metoda oceny wyników

Podstawową miarą wykorzystywaną do weryfikacji skuteczności poszczególnych wariantów algorytmu jest klasyczna miara *precyzji* (ang. *precision*, oznaczana za pomocą skrótu Pr), zdefiniowana pierwotnie w dziedzinie wyszukiwania informacji [74, s. 142]. W kontekście problemu ekstrakcji relacji wartość tej miary określona została następująco

$$Pr(m_i) = \frac{tp_i}{tp_i + fp_i} , \tag{10.1}$$

gdzie:

- m_i – metoda o indeksie i,
- tp_i – liczba wystąpień w tekście par wyrażeń oznaczonych przez metodę i, jako połączone zadaną relacją semantyczną, które zostały ocenione jako poprawne,
- fp_i – liczba wystąpień w tekście par wyrażeń oznaczonych przez metodę i, jako połączone zadaną relacją semantyczną, które zostały ocenione jako niepoprawne.

Należy podkreślić, że ta miara uwzględnia wszystkie wystąpienia danej pary wyrażeń w tekście, nawet jeśli wyrażenia te się powtarzają, zgodnie z założeniami algorytmu opisanymi w punkcie 5.1. Oznacza to, że jeśli jakaś para wyrażeń odnoszących się do tych samych obiektów, w wielu różnych zdaniach została poprawnie, bądź niepoprawnie oznaczona jako zawierająca wybraną relację, każda z tych decyzji liczona jest osobno. Miara ta pokazuje zatem jak często algorytm poprawnie wskazuje wystąpienie wybranej relacji w indywidualnych zdaniach, jeśli rozpoznaje jej wystąpienie.

Drugą miarą szeroko wykorzystywaną w dziedzinie wyszukiwania informacji jest miara *pokrycia* (ang. *recall*). Wykorzystanie tej miary bezpośrednio do ewaluacji algorytmu ekstrakcji relacji jest stosunkowo trudne, ponieważ sprowadzałoby się do konieczności przeczytania wszystkich tekstów występujących w korpusie i oznaczenia wszystkich par wyrażeń połączonych zadaną relacją. Zadanie takie byłoby jednak bardzo czasochłonne. Co gorsza, w największym polskim korpusie zawierającym ręcznie oznakowane wystąpienia różnych zjawisk językowych, czyli Narodowym Korpusie Języka Polskiego [127], nie uwzględniono znakowania wystąpień jakichkolwiek relacji semantycznych. Dlatego też zrezygnowano z obliczenia bezwzględnej miary pokrycia uzyskiwanego przez różne warianty algorytmu.

Zamiast tego, wykorzystano pokrewną jej miarę *względnego pokrycia* [61, s. 131] (ang. *relative recall*, oznaczana za pomocą skrótu Rc_{rel}) – jest to miara pokazująca względne różnice występujące pomiędzy różnymi metodami. Nadaje się ona bardzo dobrze do weryfikacji postawionej tezy – wystarczy bowiem wykazać, że względne pokrycie metod automatycznych jest wyższe niż pokrycie metody ręcznego określania ograniczeń semantycznych. Miara ta jest zdefiniowana następująco

$$Rc_{rel}(m_i) = \frac{tp_i}{|\bigcup_j \boldsymbol{TP_j}|} , \tag{10.2}$$

gdzie:

– $\boldsymbol{TP_i}$ – zbiór poprawnych rezultatów rozpoznawania zadanej relacji semantycznej uzyskanych na podstawie metody i; $tp_i = |\boldsymbol{TP_i}|$,

– $\bigcup_i \boldsymbol{TP_i}$ – zbiór będący sumą teoriomnogościową zbiorów poprawnych rezultatów uzyskanych przez każdą metodę z osobna.

Względne pokrycie określane jest zatem jako pokrycie względem zbioru wszystkich poprawnych rezultatów, które udało się uzyskać za pomocą którejkolwiek z metod.

Uzupełnieniem tych miar jest miara F_1 pozwalająca porównać wyniki poszczególnych metod w jednym wymiarze. Miara ta definiowana jest następująco [74, s. 144]

$$F_1(m_i) = \frac{2 * Pr(m_i) * Rc_{rel}(m_i)}{Pr(m_i) + Rc_{rel}(m_i)} \,. \tag{10.3}$$

Dla każdej metody przeprowadzono 4 warianty eksperymentu – biorąc pod uwagę wszystkie kombinacje dwóch parametrów:

– użycia relacji *generalizacji* do wykrywania zgodności kategorii semantycznych wyrażeń z ograniczeniami semantycznymi zdefiniowanymi dla relacji,

  – oznacza, że relacja ta nie była użyta, zatem wyrażenia musiały posiadać kategorie dokładnie pasujące do ograniczeń semantycznych,

  \+ oznacza, że kategorie semantyczne wyrażeń mogły być również specjalizacjami ograniczeń semantycznych,

– wykluczenia wystąpienia relacji dla *identycznych kategorii* obu argumentów,

  \+ oznacza, że jeśli dwa wyrażenia, dla których sprawdzano wystąpienie, posiadały przynajmniej jedną parę identycznych kategorii semantycznych, to przyjmowano, że relacja nie występuje,

  – oznacza, że warunek ten nie był weryfikowany.

10.2.2. Wyniki ekstrakcji

Wyniki ekstrakcji relacji weryfikowane były przez autora niniejszej pracy. Aby uniknąć stronniczości w ich ocenie, wyniki uzyskane przez wszystkie metody zostały wprowadzone do jednego zbioru. Następnie były one weryfikowane w losowej kolejności, przez co autor nie wiedział, z wynikiem której spośród metod ma do czynienia.

Łączna liczba wyników, które zostały ocenione, jako zawierające wystąpienia relacji *całość-część* wynosiła 889. Zbiór ten wykorzystywany był jako punkt odniesienia do obliczania wartości precyzji oraz względnego pokrycia poszczególnych metod. Wyniki pierwszej serii eksperymentów porównującej skuteczność metod określania ograniczeń semantycznych przedstawione są w tabeli 10.2. Przyjęto w nich dwa założenia – pierwsze, że liczba różnych par wyrażeń, użytych do stworzenia danego wzorca formalnego wynosiła co najmniej 2 ($CD_P \geq 2$) oraz, że minimalny poziom dokładności rozstrzygania wieloznaczności wynosił 0,2 ($P_{dg} \geq 0,2$). Ponadto w wynikach dla ograniczeń określonych ręcznie, wykluczono zdania, które zostały użyte do konstrukcji tych ograniczeń.

Najważniejszym wynikiem tych eksperymentów jest potwierdzenie tezy postawionej w prezentowanej pracy, tzn. **możliwe jest skonstruowanie algorytmów automatycznej ekstrakcji relacji z tekstów w języku polskim, których wyniki byłyby lepsze od algorytmów opierających się na ręcznej weryfikacji zdań, przyjmując, że liczba zdań ewaluowanych w celu określenia ograniczeń semantycznych wynosi 10% całkowitej liczby zdań poddawanych analizie**. Najlepszy

Tablica 10.2: Wyniki dopasowania wzorców relacji *całość-część* wyposażonych w ograniczenia semantyczne, $CD_P \geq 2$, $P_{dg} \geq 0,2$.

Źródło ograniczeń	**Generalizacja**	**Identyczne kategorie**	$\boldsymbol{Pr}$ [%]	$\boldsymbol{Rc_{rel}}$ [%]	$\boldsymbol{F_1}$ [%]
Weryfikacja ręczna	–	–	52,5	51,5	52,0
	–	+	**89,0**	49,9	64,0
	+	–	49,0	61,5	54,5
	+	+	78,6	57,6	66,5
Cyc	–	–	74,2	10,0	17,6
	–	+	73,0	9,1	16,2
	+	–	41,0	45,6	43,2
	+	+	80,6	40,2	53,6
DBpedia	–	–	61,2	21,8	32,2
	–	+	78,9	18,6	30,1
	+	–	46,1	**85,9**	60,0
	+	+	68,9	77,7	**73,0**

wynik (w sensie miary F_1) uzyskany został dla ograniczeń pozyskanych z DBpedii, w wariancie wykorzystującym relację generalizacji oraz wykluczającym identyczne kategorie semantyczne argumentów.

Nie jest zaskoczeniem, że biorąc pod uwagę wyłącznie precyzję otrzymywanych wyników, ograniczenia określone ręcznie dają najlepsze rezultaty, sięgające 89%. Drugie w kolejności są ograniczenia otrzymywane na podstawie ontologii Cyc (wynik maksymalny to 80%), a trzecie ograniczenia pozyskane z DBpedii (wynik maksymalny to 78,9%).

Jeśli zaś chodzi o względne pokrycie, to wyniki DBpedii są najlepsze (maksymalnie 85,9%), a najgorzej wypadają ograniczenia pozyskane z Cyc (maksymalnie 45,6%). Ograniczenia określone ręcznie dają najlepszy wynik na poziomie 61,5%, co należy uznać za dość dobry rezultat, biorąc pod uwagę fakt, że zostały one określone na podstawie 10% zdań podlegających analizie.

Wpływ poszczególnych parametrów na jakość wyników był następujący. Zazwyczaj wykluczenie wyników, w których oba argumenty posiadały *identyczne kategorie semantyczne* prowadziło do poprawy precyzji (z wyjątkiem Cyc w wariancie bez generalizacji). Jest to wynik zgodny z analizą przeprowadzoną w punkcie 10.1.2 – częstym źródłem błędów dopasowania wzorców formalnych było występowania kategorii semantycznej przed nazwą własną, np. *jezioro Wiktorii*, *pan Andrzej*, itp. Dzięki wyeliminowaniu tych dopasowań możliwe było istotne poprawienie precyzji rozpoznań. Zazwyczaj nie prowadziło ono do istotnego spadku pokrycia, w wyniki czego miara F_1 poprawiała się dla tego wariantu.

Wpływ wykorzystania relacji *generalizacji* był łatwy do przewidzenia, przynajmniej w zakresie zwiększenia pokrycia metody. Zawsze użycie tego wariantu prowadziło do zwiększenia pokrycia, czasami o ponad 60 punktów procentowych, jak w przypadku ograniczeń pozyskanych na podstawie DBpedii. Wynik ten jest całkowicie zgodny z samą ideą ograniczeń semantycznych – są to kategorie semantyczne, które stanowią generalizacje kategorii semantycznych wszystkich obiektów, które mogą być połączone daną relacją. Co więcej – zazwyczaj duży wzrost pokrycia nie powodował nadmiernego spadku precyzji rozpoznań – w najgorszym przypadku (ograniczeń na bazie Cyc) doprowadził do spadku precyzji o 33 punkty procentowe.

Najciekawszy wynik uzyskany został, kiedy wzięto pod uwagę kombinacje tych parametrów. Okazuje się bowiem, że dla każdej metody wariant, w którym wykorzystywano relację generalizacji oraz wykluczano

wyrażenia z identycznymi kategoriami semantycznymi, dawał najlepsze rezultaty. W konsekwencji można przyjąć, że ten wariant działania algorytmu jest optymalny.

Tablica 10.3: Wyniki dopasowania wzorców relacji *całość-część* wyposażonych w ograniczenia semantyczne, $CD_P \geq 3$, $P_{dg} \geq 0.2$.

Źródło ograniczeń	**Generalizacja**	**Identyczne kategorie**	$\boldsymbol{Pr}$ [%]	$\boldsymbol{Rc_{rel}}$ [%]	$\boldsymbol{F_1}$ [%]
Weryfikacja ręczna	–	–	89,2	56,9	69,5
	–	+	**92,8**	55,1	69,2
	+	–	77,7	64,8	70,7
	+	+	84,3	61,8	71,3
Cyc	–	–	82,4	9,4	17,0
	–	+	81,9	8,7	15,7
	+	–	76,6	41,2	53,5
	+	+	87,6	37,5	52,5
DBpedia	–	–	81,0	20,6	33,5
	–	+	84,4	16,4	27,5
	+	–	70,6	**80,3**	**75,2**
	+	+	76,4	74,0	**75,2**

Aby zbadać wpływ jakości wzorców formalnych na uzyskiwane rezultaty, przeprowadzono serię eksperymentów w których współczynnik CD_P określony dla wzorców wynosił co najmniej 3. Wyniki tych eksperymentów są przedstawione w tabeli 10.3. Najistotniejszy ich rezultat to poprawienie najlepszego wyniku uzyskanego dla ograniczeń semantycznych określonych na podstawie DBpedii z 73 do 75,2%. Oznacza to, że możliwe jest uzyskanie lepszych wyników ekstrakcji, jeśli dysponuje się odpowiednio dużym korpusem, w którym poprawne wzorce występują dla większej liczby zróżnicowanych przykładów. Wynik ten jest też o tyle ciekawy, że został uzyskany dla dwóch wariantów eksperymentu – wykorzystującego relację generalizacji i wykluczenie identycznych kategorii oraz wykorzystującego relację generalizacji, ale nie wykluczającego identycznych kategorii.

Drugim ważnym rezultatem jest poprawienie najlepszego wyniku w zakresie precyzji – ograniczenia określone na podstawie ręcznej ewaluacji, uzyskały bardzo wysoką precyzję przekraczającą 90%, dokładnie 92,8%. Należy zwrócić uwagę, że ta poprawa nie została okupiona analogicznym spadkiem precyzji i w konsekwencji miara F_1 dla tego wariantu również wzrosła (z 64% do 69%). Można również zauważyć, że dla każdego wariantu algorytmu poprawiona została precyzja rezultatów. Możliwe jest zatem dość łatwe poprawienie tego parametru poprzez użycie lepszych wzorców formalnych.

Jedyny niepożądany wynik uzyskany został dla ograniczeń pozyskanych z Cyc. W tym wypadku miara F_1, nawet dla najlepszego wariantu, była niższa niż we wcześniejszych eksperymentach. Analiza miar precyzji i pokrycia pokazuje jedynie, że spadek pokrycia nie był rekompensowany odpowiednim wzrostem precyzji. Bez szczegółowej analizy przykładów trudno jest jednak wyrokować co było dokładną przyczyną tego zjawiska.

10.2.3. Analiza błędów

W celu lepszego zrozumienia wyników uzyskanych eksperymentów przeprowadzono analizę błędów generowanych przez poszczególne warianty algorytmu. Analiza ta była przeprowadzona w trakcie oceny wyników ekstrakcji relacji – w przypadku gdy określony wynik oznaczany był jako błędny, należało wskazać

przyczynę błędu. Aby możliwe było jej ustalenie, system prezentował kompletny zbiór informacji wykorzystywany do podjęcia decyzji o ekstrakcji relacji. Można było wskazać jedną z następujących przyczyn błędu:

1. niepoprawne tagowanie morfosyntaktyczne (`tagowanie`),
2. niepoprawny wzorzec formalny (`wzorzec`),
3. niepoprawne dopasowanie wzorca (`dopasowanie`),
4. niepoprawne ujednoznacznienie przynajmniej jednego z wyrażeń (`ujednoznacznianie`),
5. niepoprawna kategoria semantyczna przypisana do wyrażenia (`kategoria`),
6. niepoprawne ograniczenia semantyczne (`ograniczenia`),
7. niepoprawna kolejność argumentów (`kolejność`),
8. wystąpienie relacji *typ-okaz* pomiędzy pierwszym i drugim argumentem relacji (`typ-okaz`).

Niepoprawne tagowanie morfosyntaktczne mogło wystąpić, jeśli narzędzie używane do ujednoznaczniania morfosyntaktycznego (Concraft [158]) podjęło błędną decyzję. Zjawisko to występowało najczęściej jeśli kategorie gramatyczne były określane dla skrótowców, których wszystkie formy morfologiczne są identyczne.

Niepoprawny wzorzec formalny był wybierany jeśli ewidentnym źródłem błędu był wzorzec formalny. Biorąc pod uwagę fakt, że wzorce były wekstrahowane w sposób automatyczny na podstawie ich analizy statystycznej, zjawisko takie mogło wystąpić dość często. Dotyczyło to w szczególności wzorców zawierających niepusty wewnętrzny kontekst.

Niepoprawne dopasowanie wzorca było wybierane, jeśli przynajmniej jedno z dopasowanych wyrażeń nie było właściwie dopasowane do wzorca. Najczęściej miało to miejsce, jeśli algorytm ujednoznaczniania semantycznego niepoprawnie wyznaczył granicę wyrażenia wielosegmentowego, albo rozpoznał tylko fragment takiego wyrażenia.

Niepoprawne ujednoznacznienie wyrażenia oznaczało, że źródłem problemy był błąd wprowadzony przez algorytm ujednoznaczniania sensu wyrażeń względem Wikipedii. Biorąc pod uwagę zawodność tego algorytmu oraz fakt, że ma on problemy z krótkimi notatkami, problem ten występował stosunkowo często.

Niepoprawna kategoria semantyczna była wybierana, jeśli ujednoznacznienie względem Wikipedii było poprawne, ale źle została określona kategoria semantyczna przez algorytm klasyfikacji wyrażeń. Ten błąd był również wskazywany, jeśli rozpoznane wyrażenie posiadało inną kategorię semantyczną niż obiekt opisywany w Wikipedii. Przykładowo wyrażenie `Polacy` jest ujednoznaczniane jako **`Polska`**, a oba wyrażenia mają odmienną kategorię semantyczną.

Niepoprawne ograniczenia semantyczne były wybierane jako źródło błędu, jeśli wszystkie wcześniejsze założenia zostały spełnione, a mimo to relacja nie występowała pomiędzy wyrażeniami.

Niepoprawna kolejność argumentów był wybierany, jeśli pomiędzy wyrażeniami faktycznie występowała relacja *całość-część*, ale algorytm niepoprawnie wskazał wyrażenie odpowiadające *całości* oraz wyrażenie odpowiadające *części*.

Ostatni błąd, czyli sytuacja, w której *pomiędzy pierwszym, a drugim argumentem występowała relacj typ-okaz*, było wskazywane dla specyficznego przypadku występowania tej relacji, zamiast relacji *całość-część*, np. *jezioro Wikitorii*, itp.

Tablica 10.4: Błędy rozpoznawania relacji *całość-część* popełnione przez algorytm przy wykorzystaniu ograniczeń semantycznych określonych ręcznie.

Rodzaj błędu	**Liczba błędów**	**Udział [%]**
ujednoznacznianie	29	28,4
ograniczenia	25	24,5
kolejność	11	10,8
typ-okaz	11	10,8
wzorzec	10	9,8
kategoria	7	6,9
tagowanie	7	6,9
dopasowanie	2	2,0

Tablica 10.5: Błędy rozpoznawania relacji *całość-część* popełnione przez algorytm przy wykorzystaniu ograniczeń semantycznych określonych na podstawie Cyc.

Rodzaj błędu	**Liczba błędów**	**Udział [%]**
typ-okaz	22	25,6
wzorzec	21	24,4
ograniczenia	13	15,1
kategoria	9	10,5
kolejność	6	7,0
dopasowanie	6	7,0
tagowanie	5	5,8
ujednoznacznianie	4	4,7

Zestawienie błędów dla poszczególnych źródeł ograniczeń kategorii semantycznych obejmuje wyłącznie najlepsze rezultaty (w sensie miary F_1) uzyskane dla tych ograniczeń, przy założeniu, że $CD_P \geq 2$. W tabeli 10.4 przedstawione są wyniki dla ograniczeń określonych na podstawie ręcznej ewaluacji zdań. Najistotniejszym problemem w tym kontekście jest ujednoznacznianie wyrażeń względem słownika semantycznego (Wikipedii) – udział błędów tego rodzaju wynosił blisko 30 procent. Na drugim miejscu wskazane są błędne ograniczenia semantyczne, z udziałem wynoszącym blisko 25%. Udział indywidualny pozostałych typów błędów wynosił 11 i mniej procent przypadków. Wyniki te pokazują, że całkowicie poprawne określenie ograniczeń semantycznych wpłynęłoby na poprawę ostatecznego wyniku jedynie o prawie 2 punkty procentowe (7,2 pp * 0.25 = 1,8 pp). Z drugiej jednak strony, gdyby udało się rozwiązać wszystkie pozostałe problemy, to precyzja tego rozwiązania sięgałaby niemal 98%.

Dość odmienne wyniki daje analiza błędów uzyskanych dla ograniczeń semantycznych określonych na podstawie Cyc – są one przedstawione w tabeli 10.5. W tym przypadku największym źródłem błędów było niewłaściwe rozpoznanie relacji typ-okaz (25%). Biorąc pod uwagę fakt, że ten wariant algorytmu wykorzystywał heurystykę wykluczającą tę relację, jest to wynik nieco zaskakujący. Należy jednak wziąć pod uwagę, że mamy tutaj do czynienie z heurystyką, która jak widać na tym przykładzie, nie zawsze działa poprawnie. Być może lepszym rozwiązaniem byłoby przyjęcie, że kategorie semantyczne prawego argumentu nie mogą generalizować się do kategorii lewego argumentu. Drugim w kolejności problemem były błędne wzorce formalne relacji *całość-część* (24%). Ten wynik nie jest zaskakujący, bo najlepszy wynik dla ograniczeń określonych na bazie Cyc został uzyskany dla wzorców o $CD_P \geq 2$. Wzorce te

Tablica 10.6: Błędy rozpoznawania relacji *całość-część* popełnione przez algorytm przy wykorzystaniu ograniczeń semantycznych określonych na podstawie DBpedii.

Rodzaj błędu	**Liczba błędów**	**Udział [%]**
kolejność	40	19,7
ujednoznacznianie	37	18,2
tagowanie	34	16,7
ograniczenia	27	13,3
typ-okaz	23	11,3
kategoria	19	9,4
wzorzec	18	8,9
dopasowanie	5	2,5

charakteryzują się umiarkowanym stopniem poprawności, co zostało potwierdzone przez powyższy wynik. Błędne ograniczenia semantyczne znalazły się dopiero na trzecim miejscu z wynikiem nieco przekraczającym 15%. Gdyby były określone w pełni poprawnie, to precyzja poprawiłaby się o 4 punkty procentowe. Natomiast gdyby były one jedynym źródłem błędów, to precyzja sięgałaby w tym wypadku 96%.

Ostatnią analizowaną grupą były błędy uzyskane dla ograniczeń określonych na podstawie DBpedii – przedstawione w tabeli 10.6. W tym przypadku najczęstszym problemem była niepoprawna kolejność argumentów ustalonych dla relacji. W pierwszej kolejności można by przypuszczać, że kolejność ta, określona ręcznie i przedstawiona w tabeli E.1 dla niektórych była ustalona błędnie. Dokładna analiza przykładowych zdań pokazuje jednak, że problem miał inne podłoże. W korpusie PAP często występowały zdania postaci „zdaniem Washington Post przyjęcie do *NATO* ***Polski***, Czech i Węgier...", w którym *Polska* była rozpoznawany jako *całość*, a *NATO* jako *część*, podczas gdy poprawna ekstrakcja powinna odwrotnie określić argumenty. Taka kolejność wynikała jednak z postaci wzorców formalnych, które mogły być np. zbudowane na podstawie następującego zdania „w szpitalu opatrzono zranioną *nogę* ***matki***", gdzie *noga* jest oczywiście częścią osoby. Ograniczenia uzyskane z DBpedii były w tym wypadku jednak za mało selektywne, aby odrzucić taką ekstrakcję jako błędną. Problemy tego rodzaju były odpowiedzialne za ok. 20% wszystkich błędów.

Drugim w kolejności problemem było niewłaściwe ujednoznacznianie wyrażeń względem Wikipedii (18%), a trzecim błędne tagowanie (17%). Problemy z ograniczeniami semantycznymi pojawiły się dopiero na 4 miejscu i odpowiadały za 13% błędów. Zatem w tym przypadku całkowicie poprawne określenie ograniczeń semantycznych poprawiłoby ostateczny wynik o 3%. Gdyby jednak udało się rozwiązać wszystkie pozostałe problemy, to precyzja sięgnęłaby 97%. Należy jednak wziąć pod uwagę, że problem wskazany na pierwszym miejscu (niepoprawna kolejność argumentów) pośrednio związany jest z ograniczeniami semantycznymi.

Biorąc pod uwagę wyniki uzyskane dla poszczególnych metod można zaobserwować następujące prawidłowości. Moduły nie-semantyczne (tagowanie, wzorzec, dopasowanie) zwykle miały znacznie mniejszy udział w błędach algorytmu, niż moduły semantyczne. Co więcej poprawne określenie ograniczeń semantycznych nie jest warunkiem wystarczającym poprawności wyników. Bez wątpienia istotny wkład ma również ujednoznacznianie wyrażeń względem Wikipedii. Biorąc pod uwagę różnice w wynikach uzyskanych przy $CD_P \geq 2$ oraz $CD_P \geq 3$ można dojść do wniosku, że podniesienie tego parametru skutkuje znaczną redukcją błędów związanych z niepoprawnie określonym wzorcem oraz z dopasowaniem wyrażeń do wzorca. W tej sytuacji na pierwszy plan wychodzą problemy związane z analizą semantyczną. Istotnym

wnioskiem jest również to, że żadne ze źródeł błędów nie dominuje, co z jednej strony oznacza, że jakość poszczególnych modułów jest dość wysoka, ale z drugiej, że poprawienie uzyskiwanych wyników jest trudne, ze względu na konieczność jednoczesnej eliminacji wielu różnych przyczyn ich powstawania.

10.3. Ekstrakcja innych relacji semantycznych

Opisane dotychczas eksperymenty koncentrowały się na rozpoznawaniu pojedynczej – choć nie homogenicznej – relacji semantycznej jaką jest relacja *całość-część*. Istotną zaletą DBpedii, jako zasobu służącego do określania ograniczeń semantycznych, jest możliwość jednoczesnego odkrycia wielu ograniczeń dla różnych relacji semantycznych. W celu zweryfikowania tej możliwości przeprowadzono eksperymenty z użyciem dwóch predykatów: `owner` – łączącego właściciela z posiadanym przez niego dobrem oraz `location` – pozwalającej na określenie lokalizacji. Predykaty te reprezentują odpowiadające im relacje semantyczne – relację *posesywną* oraz relację *lokalizacji*. W odniesieniu do pierwszej spośród tych relacji wiadomo, że formalnie jest ona bardzo podobna do relacji *całość-część*. Można to zauważyć porównując następujące wyrażenia: `ręka matki` oraz `dom matki`. W obu wyrażeniach występuje ta sama relacja formalna – związek rządu.

W odniesieniu do relacji lokalizacji sytuacja jest nieco inna – obie relacje nie wykazują takiego podobieństwa formalnego. Niemniej jednak ekstrakcja tej relacji w oparciu o te same wzorce formalne powinna wzbogacić naszą wiedzę, na temat wpływu wzorców formalnych na poprawność całego procesu.

W celu zweryfikowania możliwości ekstrahowania innych relacji semantycznych w oparciu o ograniczenia semantyczne pozyskane z DBpedii przeprowadzono eksperyment, w którym powtórzono wszystkie etapy konstrukcji wzorca ekstrakcyjnego dla relacji *całość-część*, z wyjątkiem ostatniego, tj. określenia ograniczeń semantycznych. Ograniczenia te zostały określone na podstawie odpowiadających im predykatów z DBpedii – `owner` oraz `location`. Ponadto określono kolejność argumentów dla tych relacji, podobnie jak dla poszczególnych predykatów relacji *całość-część* (porównaj tabela E.1). W ten sposób skonstruowano wzorce ekstrakcyjne, których wzorce formalne odpowiadały relacji *całość-część*, a ograniczenia semantyczne relacji *posesywnej* oraz relacji *lokalizacji*.

Szablony te zostały wykorzystane do ekstrakcji informacji z tego samego zbioru tekstów, tj. z korpusu notatek PAP. Eksperyment przeprowadzono jedynie dla najlepszej kombinacji parametrów (tj. z wykorzystaniem relacji generalizacji oraz z wykluczeniem wyrażeń, których ograniczenia semantyczne były identyczne), dla wzorców formalnych o $CD_P \geq 2$. Tak uzyskane wyniki zostały poddane ręcznej ewaluacji przez autora. Dla relacji *posesywnej* uzyskano precyzję wynoszącą 79%, a dla relacji *lokalizacji* precyzję wynoszącą 54%.

Uzyskana precyzja ekstrakcji relacji *posesywnej* jest zatem nieco wyższa niż relacji *całość-część*, co należy uznać za wynik bardzo dobry. Ponieważ jednak nie przeprowadzono eksperymentów porównawczych w odniesieniu do pokrycia tej metody, a liczba odnalezionych instancji tej relacji była niewielka (47 par wyrażeń), wyniki te traktować należy jako orientacyjne.

W odniesieniu do relacji *lokalizacji* wynik jest dużo gorszy. Nie jest on jednak zaskakujący, ponieważ formalnie relacja ta jest znacznie mniej podobna do relacji *całość-część*, niż relacja posesywna. Wynik ten świadczy zatem na korzyść tezy głoszącej, że precyzyjna ekstrakcja relacji wymaga poprawnego określenia zarówno cech formalnych wzorca oraz jego cech semantycznych. Opierając się wyłącznie na cechach semantycznych nie można skutecznie ekstrahować relacji.

Wnioski

Szczegółowa analiza przykładów dopasowania wzorców formalnych dostarczyła wielu przykładów zdań, które stanowią problem dla algorytmu ekstrakcji relacji. Najtrudniejsze z punktu widzenia konstruktora algorytmu są zdania, w których mamy do czynienia z wypowiedziami metaforycznymi oraz częściowo wadliwymi. Algorytm wymaga określenia ścisłych kryteriów dopasowania wzorców, natomiast przytoczone przykłady pokazują, że analizowane zjawiska mają charakter rozmyty.

Druga grupa problemów obejmuje zjawiska, które dałoby się rozwiązać odpowiednio komplikując algorytm (np. zjawiska związane z wieloaspektową klasyfikacją pojęć), ale które trudno jest wdrożyć ze względów praktycznych, gdyż wymagałby albo istotnej rozbudowy słownika semantycznego, albo zastosowania zaawansowanych mechanizmów wnioskowania na etapie ekstrakcji relacji. Tylko jedno z tego rodzaju zjawisk (związane z relacją *typ-okaz*) można stosunkowo łatwo uwzględnić w konstruowanym algorytmie.

Trzecią grupę problemów stanowią zjawiska związane z problemami interpretacyjnymi. Widać wyraźnie, że stworzenie algorytmu ogólnego przeznaczenia nie jest łatwe, ze względu na trudność jednoznacznej interpretacji występujących zjawisk. Dostosowanie algorytmu do określonej dziedziny wiedzy, ograniczającej zbiór dopuszczalnych interpretacji, może pomóc rozwiązać ten problem w praktyce.

Wyniki ekstrakcji relacji semantycznych w oparciu o wzorce ekstrakcyjne, świadczą na korzyść tezy głoszącej, że korzystając z odpowiedniego bogatego źródła wiedzy możliwa jest ich automatyczna konstrukcja. Jeśli przyjmiemy, że w celu określenia ograniczeń semantycznych analizie ręcznej podlega 10% zdań zawierających dopasowanie wzorców formalnych, to wyniki automatycznej konstrukcji wzorców uzyskanych na bazie DBpedii są lepsze (w odniesieniu do miary F_1). Co prawda precyzja tych wzorców jest niższa średnio o ok. 8 punktów procentowych, ale jest ona rekompensowana przez znacznie wyższe względne pokrycie.

W tym względzie znacznie gorzej wypadają ograniczenia semantyczne określone automatycznie na podstawie ontologii Cyc, dla których wyniki zarówno pod względem precyzji, jak i pokrycia są niższe. Świadczą one, że wykorzystanie jedynie dość ogólnej wiedzy zawartej w tej ontologii jest niewystarczające do prawidłowego określenia ograniczeń semantycznych. Pokazują również, że skuteczna ekstrakcja relacji wymaga użycia bardzo zróżnicowanych zbiorów danych. Nie należy bowiem zapominać, że w metodzie opartej o ograniczenia pozyskane z DBpedii również wykorzystywana jest wiedza z ontologii Cyc.

Analiza błędów dla poszczególnych metod pokazuje, że nie istnieje jedno dominujące źródło tych błędów. W zależności od metody, większy udział może mieć błędne ujednoznacznianie wyrażeń (jak w przypadku ograniczeń określonych ręcznie), czy też niepoprawnie określona kolejność argumentów (jak w przypadku ograniczeń określonych na bazie Cyc). Wyraźnie widać jednak, że udział błędów związanych z analizą semantyczną jest większy i w tym kierunku powinny iść dalsze prace zmierzające do poprawy uzyskanych wyników. Nie zmienia to jednak faktu, że poprawność wzorców formalnych ma również istotny wpływ na uzyskanie zadowalających rezultatów, co można było zaobserwować poprzez zwiększenie wartości miary CD_P oraz przeprowadzając eksperymenty z ograniczeniami charakterystycznymi dla innych relacji semantycznych.

11. Podsumowanie

Teza rozprawy przedstawiona w punkcie 1.1 składa się z trzech części. Część pierwsza dotyczy precyzji wyników uzyskiwanych przez hybrydowy algorytm ekstrakcji informacji. Na rzecz tej tezy świadczy różnica w precyzji wyników ekstrakcji uzyskanych na podstawie wzorców formalnych oraz wyników uzyskanych na podstawie wzorców wyposażonych w ograniczenia semantyczne zdefiniowane z wykorzystaniem pojęć ontologii Cyc. Jak pokazane zostało w punkcie 10.1.1, wzorce formalne zbudowane na podstawie analizy statystycznej, dają wyniki ekstrakcji o precyzji w przedziale 20%-40%. Natomiast algorytm hybrydowy uzyskuje wyniki charakteryzujące się wyższą precyzją – najgorszy wariant bazujący w całości na ontologii Cyc ma precyzję wynoszącą 41%, natomiast wariant najlepszy, oparty o ograniczenia semantyczne wyekstrahowane z ręcznie ocenionych zdań, posiada precyzję wynoszącą 92%. Te wyniki pokazują, że **zastosowanie algorytmu hybrydowego istotnie przyczynia się do poprawy precyzji uzyskiwanych wyników**, co dowodzi słuszności pierwszej części tezy.

Druga część tezy dotyczy obszaru zastosowania algorytmu i zakłada, że nie ma być on ograniczony do pojedynczej dziedziny wiedzy. Ta część tezy potwierdzona została na kilka sposobów. Po pierwsze, na żadnym etapie konstrukcji wzorców ekstrakcyjnych nie była wykorzystywana wiedza dziedzinowa. Co prawda, jako przykładów użyto par pojęć z dziedziny anatomii, ale uzyskane wzorce okazały się skuteczne w ekstrakcji informacji w innych dziedzinach.

Po drugie, wynikowe wzorce ekstrakcyjne wykorzystywane były do analizy notatek PAP. Zakres tematów poruszanych w notatkach nie jest ograniczony do jednej dziedziny wiedzy, choć dominują informacje związane z polityką międzynarodową. W efekcie, hybrydowy algorytm ekstrakcji relacji semantycznych rozpoznał wystąpienia relacji *całość-część* w następujących zdaniach[1]:

- Prof. Edward Borowski, szef gdańskiego oddziału tej organizacji, szacuje, że ok. 30 procent mieszkańców trójmiejskiej aglomeracji stanowią osoby, które pochodzą ze *stolicy* ***Litwy*** i okolic,
- Zdaniem posłów koalicji „obowiązkiem Krajowej Rady Radiofonii i Telewizji oraz *Rady Nadzorczej* ***TVP*** jest przerwanie tych destruktywnych działań”,
- Koszykarze Portland Trail Blazers i Los Angeles Lakers zagrają w finale *Konferencji Zachodniej* ***ligi NBA***,
- *Posłanka* ***Unii*** dodała, że sama jest za jeszcze dalej idącym rozwiązaniem, które zakłada, że rady nadzorcze nie miałyby wpływu na skład zarządów mediów publicznych.

W każdym z nich mamy do czynienia z inną dziedziną wiedzy. W pierwszym wiedza dotyczy zależności geopolitycznych – *stolica* jest częścią *państwa*, w drugim wiedza dotyczy organizacji spółek handlowych – częścią *spółki* jest jej *rada nadzorcza*, w trzecim zdaniu rozpoznane są zależności w obszarze sportu – *liga NBA* podzielona jest na dwie *konferencje*, natomiast w ostatnim zdaniu rozpoznane zostały zależności

[1] Przykłady te pochodzą z korpusu PAP.

w organizacji politycznej – *posłanka* jest częścią *partii politycznej*. Widać zatem, że algorytm zdolny jest do ekstrakcja relacji w wielu dziedzinach wiedzy.

Po trzecie zaś – wszystkie źródła wiedzy wykorzystywane w algorytmie mają charakter uniwersalny. Dotyczy to słownika fleksyjnego, Wikipedii, ontologii Cyc oraz semantycznej bazy wiedzy jaką jest DBpedia. Wszystkie te fakty świadczą na rzecz tezy, że **hybrydowy algorytm ekstrakcji relacji semantycznych jest uniwersalny**, co dowodzi słuszności drugiej części tezy.

Ostatnia część tezy rozprawy dotyczy nakładu pracy ręcznej, jaka musi zostać wykonana, aby można było zastosować prezentowany algorytm do ekstrakcji nowych relacji semantycznych. W założeniu spełnienie warunku o mniejszym nakładzie pracy ręcznej, niezbędnej do wykorzystania algorytmu, ma umożliwić jego praktyczne zastosowanie. Ta część tezy została potwierdzona poprzez wyższą wartość miary F_1 uzyskaną przez wariant algorytmu oparty o DBpedię, w stosunku do wariantu opartego o ręczną ocenę zdań. Czas który trzeba poświęcić na zidentyfikowanie w DBpedii predykatów reprezentujących interesującą nas relację oraz określenie kolejności argumentów w tych predykatach jest znacznie krótszy, niż czas potrzebny na ręczną weryfikację zbioru zdań, który zostałby użyty w algorytmie o porównywalnej skuteczności. W konsekwencji **algorytm hybrydowy w wariancie wykorzystującym ograniczenia semantyczne określone na podstawie DBpedii, wymaga mniejszego nakładu pracy ręcznej, niż analogiczny algorytm oparty o zbiór danych oznaczonych ręcznie, oferując wyższą jakość uzyskiwanych wyników**, co dowodzi słuszności trzeciej części tezy.

Ponieważ wszystkie części tezy rozprawy zostały udowodnione należy uznać, że teza głosząca, że **możliwe jest skonstruowanie hybrydowego algorytmu ekstrakcji relacji semantycznych** została obroniona.

11.1. Najważniejsze osiągnięcia naukowe

Do najważniejszych osiągnięć naukowych autora przedstawionych w niniejszej rozprawie należą:

1. opracowanie oraz weryfikacja algorytmu automatycznego wyboru przykładowych zdań, zawierających wystąpienia zadanej relacji semantycznej – punkt 7.1,
2. określenie statystycznych cech wzorców formalnych, decydujących o ich poprawności – punkt 8.6,
3. analiza formalnych wzorców relacji semantycznych – przedstawiona w punktach 8.7 oraz 10.1,
4. porównanie różnych metod określania ograniczeń semantycznych – punkt 10.2,
5. analiza błędów ekstrakcji – punkt 10.2.3,
6. opracowanie kompletnego systemu pozwalającego na automatyczną ekstrakcję relacji semantycznych na bazie ontologii Cyc oraz DBpedii.

W niniejszej pracy zastosowano innowacyjny algorytm wyboru przykładów zdań, zawierających wystąpienia par symboli połączonych wybraną relacją. Algorytm ten różni się od podobnych algorytmów opisanych w literaturze następującymi cechami. Po pierwsze, lista par pojęć zarodkowych określana jest automatycznie na podstawie zawartości ontologii Cyc. Zwykle algorytmy wymagają, aby lista ta była określona przez człowieka (porównaj algorytmu Brina [18] oraz algorytm Snowball [2]). Istnieją jednak algorytmy działające na bazie źródła wiedzy takiego jak WordNet, które również unikają na tym etapie zaangażowania człowieka – np. algorytm Girju [47]. Dlatego drugą istotną różnicą prezentowanego algorytmu jest to, że nie ogranicza się on jedynie do par pojęć określonych *explicite*, lecz wykorzystuje mechanizm

wnioskowania na bazie relacji generalizacji, który istotnie rozszerza wyjściowy zbiór pojęć połączonych relacją. Wyniki przedstawione w tabeli 7.1 świadczą o tym, że choć mechanizm ten zmniejsza poprawność otrzymywanych przykładowych zdań, pozwala na istotne (nawet siedmiokrotne) zwiększenie wielkości zbioru przykładów.

Określenie statystycznych cech wzorców ekstrakcyjnych, decydujących o ich poprawności, jest drugim istotnym osiągnięciem. Wykorzystanie miary CD_P zamiast bardziej oczywistej miary CT_P, pozwala automatycznie odfiltrować przykłady zdań, tak by wynikowy zbiór zawierał niemal wyłącznie przykłady poprawne (dla $CD_P \geq 3$ precyzja wynosi 99%). Wynik ten jest w pewnym stopniu podobny do rezultatów uzyskanych przez Piaseckiego i współpracowników [108]. Wykorzystując dopasowania trzech różnych wzorców relacji uzyskali oni istotną poprawę precyzji otrzymywanych rezultatów. Różnica polega jednak na tym, że w ich podejściu metoda ta stosowana była do oceny wyników ekstrakcji na podstawie wzorców opracowanych ręcznie, a w prezentowanym podejściu to cechy wzorca wykorzystywane były do oceny przykładowych zdań. Ponadto Piasecki ekstrahował relację *hiponiomii*, a w prezentowanym algorytmie ekstrahowana była relacja *meronimii*. Co więcej, w ich przypadku wykorzystanie trzech różnych wzorców prowadziło do ponad dwustukrotnej redukcji liczby ekstrahowanych relacji, przy precyzji sięgającej 74% (porównaj [108, s. 107]), a w prezentowanym podejściu dla 3 różnych par wyrażeń precyzja sięgała 99% przy 30-krotnej redukcji liczby przykładów. W istocie wyniki te dotyczą jednak nieco innych zagadnień.

Jednym z kluczowych rezultatów przedstawionych w niniejszej rozprawie jest wykazanie, że ekstrakcja relacji wyłącznie na bazie wzorców formalnych, prowadzi do wyników niskiej jakości. Wyniki ten został po raz pierwszy przedstawiony w punkcie 8.7. Pomimo tego, że użyto wzorców uzyskanych na podstawie przykładów charakteryzujących się wysoką precyzją ($CD_P \geq 2$, $Pr = 91\%$), po dopasowaniu ich do korpusu testowego, precyzja uzyskanych wyników wynosiła jedynie 20%. Należy podkreślić, że wyniki te nie biorą pod uwagę żadnej analizy semantycznej, zatem jednym źródłem błędów była zbyt mała selektywność wzorców formalnych.

Podobne wyniki zostały przedstawione w punkcie 9.7, w szczególności w tabeli 10.1. W tym przypadku uwzględniona została również analiza semantyczna. Największym źródłem błędów okazało się niepoprawne ujednoznacznienie sensu wyrażeń (udział w całkowitej liczbie dopasowanych zdań 41%), ale wynik ten można łatwo poprawić, w szczególności jeśli weźmiemy pod uwagę, że dla tych eksperymentów nie ustalono minimalnego poziomu pewności ujednoznacznienia P_{dg}. Na drugim miejscu pod względem udziału są zdania niezawierające relacji – ponad 38%. Jeśli by pominąć wszystkie przykłady, w których wyrażenie nie zostało poprawnie ujednoznacznione oraz dodatkowo założyć, że wszystkie przypadki problematyczne zawierają relację, to i tak udział zdań niezawierających relacji w ogólnej liczbie wyników wyniósłby 66%, co oznacza, że ekstrakcja na podstawie samych wzorców formalnych mogłaby osiągnąć precyzję co najwyżej 34%.

Oba te wynik są bardzo ważne, ponieważ pokazują, że analiza cech formalnych nie jest wystarczająca, aby skutecznie ekstrahować relacje semantyczne, w szczególności relacje takie jak *całość-część*. Obserwacja ta jest zgodna z wcześniejszymi wynikami prac nad ekstrakcją tej relacji, opublikowanymi w kontekście języka angielskiego [52, 13, 47, 46], ale nikt wcześniej nie przeprowadził podobnych badań dla języka polskiego. Wyniki te uzasadniają podstawową tezę pracy, głoszącą, że ekstrakcja relacji semantycznych możliwa jest w oparciu o algorytm hybrydowy.

Najważniejsze są jednak wyniki przedstawione w punkcie 10.2. W pierwszej kolejności pokazują one, że możliwe jest automatyczne skonstruowanie hybrydowych wzorców ekstrakcyjnych, których skuteczność (mierzona miarą F_1) jest wyższa niż analogicznych wzorców, w których ograniczenia semantyczne określone są na podstawie ręcznie ocenianych dopasowań, a nakład pracy człowieka w odniesieniu do

pierwszej metody jest mniejszy, niż w przypadku metody drugiej. Jedynymi działaniami niezbędnymi do realizacji ekstrakcji na podstawie ograniczeń semantycznych określonych na bazie DBpedii było ustalenie, które spośród 159 predykatów odpowiadają ekstrahowanej relacji oraz określenie właściwej kolejności argumentów dla 39 predykatów odpowiadających relacji *całość-część*. Całkowity czas poświęcony na realizację tych zdań był istotnie krótszy, niż czas niezbędny do ręcznej ewaluacji 682 zdań, stanowiących 10% zdań potencjalnie zawierających relację, na podstawie których ustalono ograniczenia semantyczne w tym scenariuszu.

Nie należy jednak zapominać o tym, że wyniki określone na podstawie ręcznej oceny zdań cechowały się najwyższą precyzją, przekraczającą 90%. Ten wynik jest również bardzo dobry i koresponduje z wynikami ekstrakcji relacji dla języka angielskiego przedstawionymi przez Girju w pracy [46], w których autorka w systemie wymagającym ręcznej oceny przykładów, uzyskała precyzję nieco przekraczającą 80%. Oczywiście takie bezpośrednie porównanie wartości precyzji musi być opatrzone wieloma zastrzeżeniami, wynikającymi przede wszystkim z faktu, że oba algorytmy testowane były dla różnych języków oraz różnych zbiorów testowych. Warto również zauważyć, że precyzja metody opartej na DBpedii wynosi 76% (w wariancie dla najlepszego wyniku mierzonego miarą F_1), zatem, choć jest gorsza od wariantu „ręcznego", bliska jest wynikom uzyskanym przez Girju.

Wyniki te pokazują również, że wykorzystanie ontologii Cyc do określenia tych ograniczeń jest niewystarczające, gdyż ekstrakcje uzyskane wyłącznie na podstawie tej ontologii były gorsze, niż ekstrakcje uzyskane na podstawie ręcznej oceny zdań. Ważnym wynikiem jest również określenie optymalnych parametrów algorytmu, tzn. wariantu, w którym wykorzystywana była relacja generalizacji oraz heurystyka wykluczająca wystąpienie relacji pomiędzy wyrażeniami posiadającymi identyczne kategorie semantyczne.

Analiza błędów popełnianych przez różne warianty algorytmu, świadczy przede wszystkim na rzecz tezy, że powodzenie ekstrakcji relacji semantycznych zależy przede wszystkim od właściwej analizy semantycznej, gdyż w każdym wariancie błędy związane z semantyką miały największy udział. Ponadto okazuje się, że nie zidentyfikowano modułu, który byłby dominującym źródłem błędów. Poprawienie skuteczności algorytmu ekstrakcji wymaga ulepszenia każdej składowej – co z jednej strony pokazuje, że wszystkie moduły były należycie dopracowane, ale z drugiej, że uzyskanie znacząco lepszych wyników (przynajmniej w obrębie prezentowanego paradygmatu) nie będzie łatwe.

Ostatni – być może najważniejszy wynik – to fakt, że prezentowany system został skonstruowany niemal w całości od podstaw przez autora. Jedynie algorytm ujednoznaczniania morfosyntaktycznego realizowany był przez zewnętrzny program *Concraft* [158]. Wszystkie pozostałe moduły zostały zaimplementowane przez autora. Dzięki temu możliwe było uczynienie ontologii Cyc centralnym zasobem wykorzystywanym przez system ekstrakcji oraz wykazanie jej przydatności w tym zastosowaniu. Ostateczne wyniki ekstrakcji relacji pokazują jednak, że sama ontologia jest niewystarczająca, żeby skutecznie ekstrahować relacje semantyczne, ale dobrze sprawdza się jako referencyjny zasób semantyczny, pozwalający na integrację heterogenicznych źródeł wiedzy.

11.2. Dalsze kierunki badań

W pracy tej przedstawiono kompletny system ekstrakcji relacji semantycznych z tekstów w języku polskim. Pomimo jej obszerności nie poruszono wielu istotnych zagadnień związanych z ekstrakcją informacji w języku polskim.

Wśród najważniejszych zagadnień, które zostały pominięte w niniejszej pracy można wymienić:

– wykorzystanie większej liczby cech do konstrukcji wzorców ekstrakcyjnych,

- wykorzystanie parsera języka polskiego do ekstrakcji relacji semantycznych,
- wykorzystanie innych źródeł wiedzy, np. polskiego WordNetu, jako referencyjnych zasobów semantycznych,
- opracowanie algorytmu uogólniania ograniczeń semantycznych,
- zweryfikowanie działania algorytmu na większej liczbie relacji semantycznych,
- szersze wykorzystanie mechanizmów wnioskowania w trakcie ekstrakcji informacji,
- wykorzystanie bogatszej wiedzy na temat obiektów opisanych w Wikipedii/DBpedii, w celu lepszej identyfikacji relacji łączących je z innymi obiektami.

Jednym z kluczowych założeń algorytmu było wykorzystanie bardzo prostych wzorców ekstrakcyjnych, których opis ograniczał się do kolejności argumentów, cech morfosyntaktycznych oraz wyrazów występujących pomiędzy wyrażeniami. Pomimo tego, dokładne dopasowanie wzorców formalnych skutkowało dość niską precyzją rozpoznawania relacji. Zastosowanie większej liczby cech, np. szerszego kontekstu, mogłoby przyczynić się do podniesienia jakości ekstrakcji z użyciem samych wzorców formalnych. Autor zdecydował jednak, że ulepszenie to może zostać uzyskane poprzez nałożenie ograniczeń semantycznych. Niemniej jednak wykorzystanie bogatszych wzorców formalnych, w połączeniu z mniej restrykcyjnymi zasadami ich dopasowania, mogłoby skutkować uzyskaniem lepszych wyników, wyłącznie na podstawie tak ulepszonych wzorców formalnych.

Ponadto przyjęty schemat zastosowania wzorców formalnych, w którym wymaga się dokładnego ich dopasowania do linearnej struktury tekstu jest stosunkowo prosty. W analogicznych systemach tworzonych dla języka angielskiego, wzorce formalne często konstruowane są na podstawie drzewa rozbioru syntaktycznego zdania. Rezygnacja z tego rozwiązania w kontekście języka polskiego była po części podyktowana niedostępnością parsera, który charakteryzowałby się wysoką precyzją uzyskiwanych rezultatów, przy wykluczeniu wielu alternatywnych postaci drzewa rozbioru syntaktycznego. Badania nad tego rodzaju parserami jednak trwają i w przyszłości warto byłoby wykorzystać ten rodzaj informacji przy konstrukcji oraz dopasowywaniu wzorców formalnych.

Kolejnym ważnym elementem konstrukcji wzorców ekstrakcyjnych, który został pomięty, jest etap uogólniania ograniczeń semantycznych wykrytych w Cyc, czy DBpedii. Co prawda algorytm przy dopasowywaniu wzorców korzystał z relacji generalizacji, niemniej jednak bardziej precyzyjne określenie maksymalnego poziomu uogólnienia mogłoby przyczynić się do poprawy wyników ekstrakcji. Koncepcja ta jest szczególnie pociągająca w kontekście ograniczeń pozyskiwanych na podstawie DBpedii, gdyż ilość danych dostępnych w tej bazie wiedzy umożliwia zastosowanie zaawansowanych algorytmów uogólniania ograniczeń, np. klastrowania hierarchicznego.

Weryfikacja działania algorytmu koncentrowała się na relacji *całość-część*. W punkcie 10.3 pokazano wyniki ograniczonych eksperymentów z wykorzystaniem relacji *posesywnej* oraz relacji *lokalizacji*. Pełniejsza weryfikacja skuteczności algorytmu powinna obejmować znacznie szerszy zbiór relacji semantycznych. Podstawowy problem, jaki musi jednak zostać wcześniej rozwiązany w tym zakresie to dostępność odpowiedniego korpusu testowego, który zawierałby znakowanie obejmujące wystąpienia relacji semantycznych. Konstrukcja takiego korpusu jest jednak stosunkowo droga i wymaga dokonania wielu rozstrzygnięć natury teoretycznej. Ponadto, powinien zostać on opracowany w oderwaniu od konkretnej implementacji algorytmu ekstrakcji informacji, aby uniknąć stronniczości w podejmowaniu decyzji. Ponieważ dla języka polskiego nie istnieje taki korpus, weryfikacja skuteczności algorytmu została ograniczona do wymienionych relacji semantycznych.

W prezentowanym algorytmie wykorzystywano również tylko niewielki procent wiedzy oraz mechanizmów inferencji udostępnianych przez ontologię Cyc. Wiedza ta ograniczała się do relacji taksonomicznych oraz zależności pomiędzy argumentami relacji, wyrażonymi za pomocą predykatu `#$relationAllExists`. Cyc, w szczególności wersja Research, zawiera jednak znacznie więcej informacji wyrażonych w postaci reguł logicznych. Ich wykorzystanie mogłoby istotnie przyczynić się do poprawy jakości ekstrakcji, poprzez uwzględnienie dodatkowej wiedzy semantycznej, na temat pojęć podlegających analizie. Jest to jeden z najciekawszych obszarów badań nieuwzględnionych w prezentowanym algorytmie. Podejście takie stanowi jednak wyzwanie również w kontekście języka angielskiego.

Ostatni bardzo interesujący obszar badań, który został pominięty w tej pracy, to możliwość wykorzystania zbioru wiedzy zgromadzonej w zasobach takich jak Wikipedia oraz DBpedia. W szczególności ta ostatnia baza wiedzy zawiera olbrzymie zasoby pozwalające nie tylko automatycznie określić ograniczenia semantyczne relacji, co było zademonstrowane w niniejszej pracy, ale zawiera również informacje na temat rozpoznawanych indywiduów, np. osób, państw, instytucji. Wykorzystanie tych szczegółowych informacji mogłoby nie tylko przyczynić się do ulepszenia algorytmu ekstrakcji relacji, ale umożliwić interpretację faktów wyrażonych w tekście nie wprost, a nawet pozwolić na interpretację metafor, przy założeniu, że wiedza na temat opisywanego indywiduum jest wystarczająco bogata.

W swojej dalszej pracy autor zamierza rozwijać metody automatycznego odkrywania wiedzy w tekstach oraz innych zasobach wiedzy, z szczególnym uwzględnieniem ontologii Cyc, jako płaszczyzny integracyjnej. Wyniki przedstawione w niniejszej pracy świadczą, że wybór tej ontologii, jako punktu odniesienia jest dobrze uzasadniony również w kontekście języka polskiego.

Bibliografia

[1] Abramowicz W., Filipowska A., Piskorski J., Węcel K., Wieloch K. (2006). *Linguistic Suite for Polish Cadastral System.* [w:] Calzolari N., Gangemi A., Maegaard B., Mariani J., Odijk J., Tapias D. (red.), *Proceedings of the LREC* (s. 53–58).

[2] Agichtein E., Gravano L. (2000). *Snowball: Extracting relations from large plain-text collections.* [w:] Nürnberg P.J., Hicks D.L., Furuta R. (red.), *Proceedings of the fifth ACM conference on Digital libraries* (s. 85–94).

[3] Agirre E., Edmonds P.G. (2007). *Word sense disambiguation: Algorithms and applications.* Berlin, Heidelberg: Springer.

[4] Alshawi H. (1987). *Processing dictionary definitions with phrasal pattern hierarchies.* „Computational Linguistics" 13/3-4, s. 195–202.

[5] Aprosio A.P., Giuliano C., Lavelli A. (2013). *Automatic Expansion of DBpedia Exploiting Wikipedia Cross-Language Information.* [w:] Cimiano P., Corcho O., Presutti V., Hollink L., Rudolph S. (red.), *The Semantic Web: Semantics and Big Data* (s. 397–411). Berlin, Heidelberg: Springer.

[6] Arystoteles (1978). *Topiki; O dowodach sofistycznych.* Warszawa: PWN.

[7] Auer S., Bizer C., Kobilarov G., Lehmann J., Cyganiak R., Ives Z. (2007). *DBpedia: A Nucleus for a Web of Open Data.* [w:] Aberer K., Choi K.-S., Noy N., Allemang D., Lee K.-I., Nixon L., Golbeck J., Mika P., Maynard D., Mizoguchi R., Schreiber G., Cudré-Mauroux P. (red.), *The Semantic Web* (s. 722-735). Berlin, Heidelberg: Springer.

[8] Baker C.F., Fillmore C.J., Lowe J.B. (1998). *The Berkeley FrameNet project.* [w:] Boitet C., Whitelock P. (red.), *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics–Volume 1* (s. 86–90).

[9] Banerjee S., Pedersen T. (2002). *An adapted Lesk algorithm for word sense disambiguation using WordNet.* [w:] Gelbukh A. (red.), *Computational Linguistics and Intelligent Text Processing* (s. 136–145). Berlin, Heidelberg: Springer.

[10] Banko M., Cafarella M.J., Soderland S., Broadhead M., Etzioni O. (2007). *Open Information Extraction from the Web.* „Communications of the ACM" 51/12, s. 2670–2676.

[11] Banko M., Etzioni O., Center T. (2008). *The Tradeoffs Between Open and Traditional Relation Extraction..* [w:] McKeown K. (red.), *Proceedings of ACL-08: HLT* (s. 28–36).

[12] Banko M., Etzioni O. (2007). *Strategies for lifelong knowledge extraction from the Web.* [w:] Sleeman D., Barker K. (red.), *Proceedings of the 4th international conference on Knowledge capture* (s. 95–102).

[13] Berland M., Charniak E. (1999). *Finding parts in very large corpora.* [w:] Dale R., Church K. (red.), *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics* (s. 57–64).

[14] Bizer C., Heath T., Berners-Lee T. (2009). *Linked Data-The Story So Far.* „International Journal on Semantic Web and Information Systems" 5/3, s. 1–22.

[15] Bizer C., Lehmann J., Kobilarov G., Auer S., Becker C., Cyganiak R., Hellmann S. (2009). *DBpedia-A crystallization point for the Web of Data.* „Web Semantics: Science, Services and Agents on the World Wide Web" 7/3, s. 154–165.

[16] Bollacker K., Evans C., Paritosh P., Sturge T., Taylor J. (2008). *Freebase: a collaboratively created graph database for structuring human knowledge.* [w:] Lakshmanan L.V.S., Ng R.T., Shasha D. (red.), *Proceedings of the 2008 ACM SIGMOD international conference on Management of data* (s. 1247–1250).

[17] Brachman R.J., Levesque H.J. (2004). *Knowledge Representation and Reasoning.* Burlington, Massachusetts: Morgan Kaufmann.

[18] Brin S. (1999). *Extracting Patterns and Relations from the World Wide Web.* [w:] Atzeni P., Mendelzon A., Mecca G. (red.), *The World Wide Web and Databases* (s. 172–183). Berlin, Heidelberg: Springer.

[19] Broda B., Piasecki M., Szpakowicz S. (2009). *Rank-Based Transformation in Measuring Semantic Relatedness.* [w:] Gao Y., Japkowicz N. (red.), *Advances in Artificial Intelligence 22nd Canadian Conference on Artificial Intelligence* (s. 187–190).

[20] Buczyński A., Przepiórkowski A. (2009). *Spejd: A shallow processing and morphological disambiguation tool.* [w:] Vetulani Z., Uszkoreit H. (red.), *Human Language Technology. Challenges of the Information Society* (s. 131–141). Berlin, Heidelberg: Springer.

[21] Carlson A., Betteridge J., Kisiel B., Settles B., Hruschka Jr E.R., Mitchell T.M. (2010). *Toward an architecture for never-ending language learning.* [w:] Fox M., Poole D. (red.), *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)* (s. 1306-1313).

[22] Carroll J.J., Bizer C., Hayes P., Stickler P. (2005). *Named graphs, provenance and trust.* [w:] Elias A., Hagino T. (red.), *Proceedings of the 14th international conference on World Wide Web* (s. 613–622).

[23] Chrząszcz P. (2009). *Automatyczne rozpoznawanie i klasyfikacja nazw wielosegmentowych na podstawie analizy haseł encyklopedycznych.* Praca magisterska, Kraków: Akademia Górniczo-Hutnicza.

[24] Chrząszcz P. (2012). *Enrichment of Inflection Dictionaries: Automatic Extraction of Semantic Labels from Encyclopedic Definitions.* [w:] Sharp B., Zock M. (red.), *9th International Workshop on Natural Language Processing and Cognitive Science* (s. 106-119).

[25] Church A. (1985). *The calculi of lambda-conversion.* Princeton, New Jersey: Princeton University Press.

[26] Cieślikowa A. (2002). *Mały słownik odmiany nazw własnych.* Warszawa: Oficyna Wydawnicza RYTM.

[27] Cilibrasi R.L., Vitanyi P.M.B. (2007). *The Google similarity distance.* „Knowledge and Data Engineering, IEEE Transactions on" 19/3, s. 370–383.

[28] Cimiano P. (2006). *Ontology learning and population from text: algorithms, evaluation and applications.* Berlin, Heidelberg: Springer.

[29] Collins A.M., Quillian M.R. (1969). *Retrieval time from semantic memory.* „Journal of verbal learning and verbal behavior" 8/2, s. 240–247.

[30] Cowie J., Lehnert W. (1996). *Information extraction.* „Communications of the ACM" 39/1, s. 80–91.

[31] Cunningham D.H., Maynard D.D., Bontcheva D.K., Tablan M.V. (2002). *GATE: A framework and graphical development environment for robust NLP tools and applications.* [w:] Isabelle P. (red.), *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)* (s. 168–175).

[32] Day D., Aberdeen J., Hirschman L., Kozierok R., Robinson P., Vilain M. (1997). *Mixed-initiative development of language processing systems.* [w:] Grishman R. (red.), *Proceedings of the Fifth Conference on Applied Natural Language Processing* (s. 348–355).

[33] De Melo G., Suchanek F., Pease A. (2008). *Integrating YAGO into the Suggested Upper Merged Ontology.* [w:] Hatziligeroudis I., Lu C.-T. (red.), *Tools with Artificial Intelligence, 20th IEEE International Conference on* (s. 190–193).

[34] DeJong G. (1979). *Prediction and substantiation: A new approach to natural language processing.* „Cognitive Science" 3/3, s. 251–273.

[35] Dębowski Ł. (2004). *Trigram morphosyntactic tagger for Polish.* [w:] Klopotek M.A., Wierzchon S.T., Trojanowski K. (red.), *Proceedings of the International IIS: IIPWM'04 Conference* (s. 409–413).

[36] Downey D., Etzioni O., Soderland S. (2006). *A probabilistic model of redundancy in information extraction.* Raport techniczny, DTIC.

[37] Drozdzynski W., Krieger H.-U., Piskorski J., Schäfer U., Xu F. (2004). *Shallow Processing with Unification and Typed Feature Structures — Foundations and Applications.* „Künstliche Intelligenz" 1, s. 17–23.

[38] Earley J. (1970). *An efficient context-free parsing algorithm.* „Communications of the ACM" 13/2, s. 94–102.

[39] Exner P., Nugues P. (2012). *Entity Extraction: From Unstructured Text to DBpedia RDF Triples.* [w:] Rizzo G., Mendes P., Charton E., Hellmann S., Kalyanpur A. (red.), *Proceedings of the Web of Linked Entities Workshop in conjuction with the 11th International Semantic Web Conference* (s. 58–69).

[40] Fader A., Soderland S., Etzioni O. (2011). *Identifying relations for open information extraction.* [w:] Lapata M., Ng H.T. (red.), *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (s. 1535–1545).

[41] Fellbaum C. (1998). *WordNet: An Electronic Lexical Database.* Cambridge, MA: MIT Press.

[42] Flanagan D., Matsumoto Y. (2008). *The Ruby programming language.* Sebastopol, California: O'Reilly Media.

[43] Gabrilovich E., Markovitch S. (2007). *Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis.* [w:] Veloso M. (red.), *Proceedings of the 20th International Joint Conference on Artificial Intelligence* (s. 12).

[44] Gajęcki M. (2009). *Słownik fleksyjny jako biblioteka języka C* [w:] Lubaszewski W. (red.), *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu* (s. 107–134). Kraków: Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH.

[45] Gangemi A., Nuzzolese A.G., Presutti V., Draicchio F., Musetti A., Ciancarini P. (2012). *Automatic typing of DBpedia entities.* [w:] Cudré-Mauroux P., Heflin J., Sirin E., Tudorache T., Euzenat J., Hauswirth M., Parreira J.X., Hendler J., Schreiber G., Bernstein A., Blomqvist E. (red.), *The Semantic Web–ISWC 2012* (s. 65–81). Berlin, Heidelberg: Springer.

[46] Girju R., Badulescu A., Moldovan D. (2006). *Automatic discovery of part-whole relations.* „Computational Linguistics" 32/1, s. 83–135.

[47] Girju R., Badulescu A., Moldovan D. (2003). *Learning semantic constraints for the automatic discovery of part-whole relations.* [w:] Hearst M., Ostendorf M. (red.), *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1* (s. 1–8).

[48] Graliński F., Jassem K., Marcińczuk M. (2009). *An environment for named entity recognition and translation.* [w:] Farwell D., Fonollosa A.R., Mariño J., Márquez L. (red.), *Proceedings of the 13th Annual Conference of the European Association for Machine Translation* (s. 88–95).

[49] Graliński F., Jassem K., Marcińczuk M., Wawrzyniak P. (2009). *Named Entity Recognition in Machine Anonymization.* [w:] Kłopotek M.A., Przepiórkowski A., Wierzchoń S.T., Trojanowski K. (red.), *Recent Advances in Intelligent Information Systems* (s. 247–260). Warszawa: EXIT.

[50] Grishman R., Sundheim B. (1996). *Message understanding conference-6: A brief history.* [w:] Tsujii J. (red.), *Proceedings of COLING* (s. 466–471).

[51] Harabagiu S., Hickl A., Lacatusu F. (2006). *Negation, contrast and contradiction in text processing.* [w:] Cohn A. (red.), *AAAI* (s. 755–762).

[52] Hearst M.A. (1992). *Automatic acquisition of hyponyms from large text corpora.* [w:] Zampolli A. (red.), *Proceedings of the 14th conference on Computational linguistics-Volume 2* (s. 539–545).

[53] Hobbs J.R., Appelt D., Bear J., Israel D., Kameyama M., Stickel M., Tyson M. (1997). *FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text.* [w:] Roche E., Schabes Y. (red.), *Finite-State Language Processing* (s. 383–406). Cambridge, MA: MIT Press.

[54] Jaccard P. (1901). *Étude comparative de la distribution florale dans une portion des Alpes et des Jura.* „Bulletin del la Société Vaudoise des Sciences Naturelles" 37, s. 547–579.

[55] Janus D., Przepiórkowski A. (2007). *Poliqarp: An open source corpus indexer and search engine with syntactic extensions.* [w:] Ananiadou S. (red.), *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions* (s. 85–88).

[56] Jaworski W. (2008). *Ontology-Based Knowledge Discovery from Documents in Natural Language.* Praca doktorska, Warszawa: Uniwersytet Warszawski.

[57] Jaworski W. (2009). *Ontology-Based Content Extraction from Polish Biobibliographical Lexicon.* [w:] Kłopotek M.A., Przepiórkowski A., Wierzchoń S.T., Trojanowski K. (red.), *Recent Advances in Intelligent Information Systems* (s. 27–40).

[58] Jurafsky D., Martin J.H. (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition (second edition).* Upper Saddle River, New Jersey: Prentice Hall.

[59] Kripke S.A. (1972). *Naming and necessity.* Berlin, Heidelberg: Springer.

[60] Kripke S.A. (1963). *Semantical Considerations on Modal Logic.* „Acta Philosophica Fennica" 16, s. 83–94.

[61] Kumar B.T.S., Prakash J.N. (2009). *Precision and relative recall of search engines: A comparative study of Google and Yahoo.* „Singapore Journal of Library & Information Management" 38/1, s. 124–137.

[62] Kurc R., Piasecki M. (2008). *Automatic acquisition of Wordnet relations by the morpho-syntactic patterns extracted from the corpora in Polish.* [w:] Ganzha M., Paprzycki M., Pełech-Pilichowski T. (red.), *Computer Science and Information Technology, International Multiconference on* (s. 181–188).

[63] Lafferty J. (2001). *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* [w:] Brodley C.E., Danyluk A.P. (red.), *Proceedings of the Eighteenth International Conference on Machine Learning* (s. 282–289).

[64] Lakoff G. (1987). *Women, Fire, and Dangerous Things.* Chicago: University of Chicago Press.

[65] Lassila O., Swick R.R. (1998). *Resource description framework (RDF) model and syntax.* Dostępne <http://www.w3.org/1998/10/WD-rdf-syntax-19981008/>

[66] Lenat D.B., Guha R.V. (1990). *Building Large Knowledge-Based Systems.* Boston: Addison Wesley.

[67] Lenat D.B. (1995). *CYC: A Large-Scale Investment in Knowledge Infrastructure.* „Communications of the ACM" 38/11, s. 33–38.

[68] Lesk M. (1986). *Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone.* [w:] DeBuys V. (red.), *Proceedings of the 5th annual international conference on Systems documentation* (s. 24–26).

[69] Li Z., Li H., Wang H., Yang Y., Zhang X., Zhou X. (2014). *Overcoming Semantic Drift in Information Extraction.* [w:] Christophides V. (red.), *Processing of the 17th International Conference on Extending Database Technolog* (s. 169–180).

[70] Linné C.von (1762). *Species Plantarum: Exhibentes Plantas Rite Cognitas, Cum Differentiis Specificis, Nominibus Trivialibus, Synonymis Selectis, Locis Natalibus, Secundum Systema Sexuale Digestas.* Sztokholm: Lars Salvius.

[71] Lubaszewski W. (2009). *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu.* Kraków: Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH.

[72] Lyons J. (1968). *Introduction to Theoretical Linguistics.* Cambridge: Cambridge University Press.

[73] Lyons J. (1984). *Semantyka.* Warszawa: Państwowe Wydawnictwo Naukowe.

[74] Manning C.D., Raghavan P., Schütze H. (2008). *Introduction to information retrieval.* Cambridge: Cambridge University Press.

[75] Manning C.D., Schütze H. (1999). *Foundations of Statistical Natural Language Processing.* Cambridge, MA: MIT Press.

[76] Marciniak M., Mykowiecka A. (2007). *Automatic processing of diabetic patients' hospital documentation.* [w:] Piskorski J., Pouliquen B., Steinberger R., Tanev H. (red.), *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing: Information Extraction and Enabling Technologies* (s. 35–42).

[77] Marcińczuk M., Piasecki M. (2010). *Named Entity Recognition in the Domain of Polish Stock Exchange Reports.* [w:] Kłopotek M.A., Marciniak M., Mykowiecka A., Penczek W., Wierzchoń S.T. (red.), *Intelligent Information Systems* (s. 127–140).

[78] Marcińczuk M., Stanek M., Piasecki M., Musiał A. (2012). *Rich Set of Features for Proper Name Recognition in Polish Texts.* [w:] Bouvry P., Klopotek M.A., Leprevost F., Marciniak M., Mykowiecka A., Rybiński H. (red.), *Security and Intelligent Information Systems* (s. 332–344). Berlin, Heidelberg: Springer.

[79] McDermott D. (2007). *Artificial intelligence and consciousness.* [w:] Zelazo P.D., Moscovitch M., Thompson E. (red.), *The Cambridge handbook of consciousness* (s. 117–150). Cambridge: Cambridge University Press.

[80] Medelyan O., Legg C. (2008). *Integrating Cyc and Wikipedia: Folksonomy meets rigorously defined common-sense.* [w:] Parsons S., Sellmann M. (red.), *Proceedings of the WIKI-AI: Wikipedia and AI Workshop at the AAAI* (s. 13-18).

[81] Medelyan O., Milne D., Legg C., Witten I.H. (2009). *Mining meaning from Wikipedia.* „International Journal of Human-Computer Studies" 67/9, s. 716–754.

[82] Mendes P.N., Jakob M., García-Silva A., Bizer C. (2011). *DBpedia Spotlight: shedding light on the Web of documents.* [w:] Ghidini C., Stefanie Lindstaedt A.-C.N.N., Pellegrini T. (red.), *Proceedings of the 7th International Conference on Semantic Systems* (s. 1–8).

[83] Mihalcea R., Csomai A. (2007). *Wikify!: linking documents to encyclopedic knowledge.* [w:] Laender A.H.F., Falcão A.O., Olsen Ø.H., Silva M.J., Baeza-Yates R., McGuinness D.L., Olstad B. (red.), *Proceedings of the sixteenth ACM conference on information and knowledge management* (s. 233–242).

[84] Mihalcea R. (2007). *Using Wikipedia for automatic word sense disambiguation.* [w:] Sidner C., Schultz T., Stone M., Zhai C.X. (red.), *Proceedings of The Conference of the North American Chapter of the Association for Computational Linguistics* (s. 196–203).

[85] Miller G.A. (1998). *Nouns in WordNet* [w:] Fellbaum C. (red.), *WordNet an Electronic Lexical Database* (s. 23–46). Cambridge, MA: The MIT Press.

[86] Milne D. (2009). *An open-source toolkit for mining Wikipedia.* [w:] Blagojevic R. (red.), *Proceedings of 7th New Zealand Computer Science Research Student Conference* (s. 222-239).

[87] Milne D., Witten I.H. (2008). *Learning to link with Wikipedia.* [w:] Shanahan J.G., Amer-Yahia S., Manolescu I., Zhang Y., Evans D.A., Kolcz A., Choi K.-S., Chowdury A. (red.), *Proceeding of the 17th ACM conference on Information and knowledge management* (s. 509–518).

[88] Mintz M., Bills S., Snow R., Jurafsky D. (2009). *Distant supervision for relation extraction without labeled data.* [w:] Su K.-Y. (red.), *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2* (s. 1003–1011).

[89] Mitkov R. (2003). *The Oxford handbook of computational linguistics.* Oxford: Oxford University Press.

[90] Miłkowski M. (2010). *Developing an open-source, rule-based proofreading tool.* „Software: Practice and Experience" 40/7, s. 543–566.

[91] Miłkowski M., Lipski J. (2009). *Using SRX standard for sentence segmentation in LanguageTool.* [w:] Vetulani Z. (red.), *Proceedings of Human Language Technologies as a Challenge for Computer Science and Linguistics* (s. 556–560).

[92] Moens M.F. (2006). *Information extraction: algorithms and prospects in a retrieval context.* Berlin, Heidelberg: Springer.

[93] Mykowiecka A., Kupść A., Marciniak M. (2005). *Rule-based medical content extraction and classification.* [w:] Kłopotek M.A., Wierzchoń S.T., Trojanowski K. (red.), *Intelligent Information Processing and Web Mining* (s. 237–245).

[94] Mykowiecka A., Marciniak M., Podsiadły-Marczykowska T. (2007). *"Data-Driven" Ontologies for an Information Extraction System from Polish Mammography Reports.* [w:] Musen M. (red.), *Proceedings of the 10th International Protégé Conference* (s. 1–3).

[95] Màrquez L., Escudero G., Martínez D., Rigau G. (2006). *Supervised Corpus-Based Methods for WSD* [w:] Eneko A., Edmonds P. (red.), *Word Sense Disambiguation: Algorithms and Applications* (s. 167–216). Berlin, Heidelberg: Springer.

[96] NIST (2008). *Automatic Content Extraction 2008 Evaluation Plan (ACE08).* Dostępne <http://www.itl.nist.gov/iad/mig/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf>

[97] Niles I., Pease A. (2003). *Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology.* [w:] Arabnia H.R., Hashemi R.R., Vert G., Chennamaneni A., Solo A.M.G. (red.), *Proceedings of the 2003 International Conference on Information and Knowledge Engineering* (s. 412–416).

[98] Niles I., Pease A. (2001). *Towards a standard upper ontology.* [w:] Guarino N., Smith B., Welty C. (red.), *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001* (s. 2–9).

[99] Nothman J., Ringland N., Radford W., Murphy T., Curran J.R. (2013). *Learning multilingual named entity recognition from Wikipedia.* „Artificial Intelligence" 194, s. 151–175.

[100] Ogden C.K., Richards I.A. (1923). *The Meaning of Meaning.* Orlando, Florida: Harcourt Brace Jovanovich.

[101] Ogrodniczuk M., Kopeć M. (2011). *End-to-end coreference resolution baseline system for Polish.* [w:] Vetulani Z. (red.), *Proceedings of the 5th Language & Technology Conference* (s. 167–171).

[102] Pantel P., Pennacchiotti M. (2006). *Espresso: Leveraging generic patterns for automatically harvesting semantic relations.* [w:] Carpuat M., Duh K. (red.), *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics* (s. 113–120).

[103] Paulheim H., Bizer C. (2013). *Type inference on noisy RDF data.* [w:] Alani H., Kagal L., Fokoue A., Groth P., Biemann C., Parreira J.X., Aroyo L., Noy N., Welty C., Janowicz K. (red.), *The Semantic Web–ISWC 2013* (s. 510–525). Berlin, Heidelberg: Springer.

[104] Peirce C.S. (1909). *Manuscript 514.* Niepublikowany manuskrypt.

[105] Piasecki M., Broda B. (2007). *Semantic similarity measure of Polish nouns based on linguistic features.* [w:] Abramowicz W. (red.), *Business Information Systems* (s. 381–390).

[106] Piasecki M. (2006). *Hand-written and automatically extracted rules for Polish tagger.* [w:] Sojka P., Kopeček I., Pala K. (red.), *Text, Speech and Dialogue* (s. 205–212).

[107] Piasecki M., Koczan P. (2007). *Environment supporting construction of the Polish Wordnet.* [w:] Vetulani Z. (red.), *Proceedings of the 3rd Language and Technology Conference* (s. 519–523).

[108] Piasecki M., Szpakowicz S., Broda B. (2009). *A WordNet from the Ground Up.* Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.

[109] Piasecki M., Szpakowicz S., Marcińczuk M., Broda B. (2008). *Classification-based filtering of semantic relatedness in hypernymy extraction.* [w:] Nordström B., Ranta A. (red.), *Advances in Natural Language Processing* (s. 393–404). Berlin, Heidelberg: Springer.

[110] Piasecki M. (2007). *Polish tagger TaKIPI: Rule based construction and optimisation.* „Task Quarterly" 11/1-2, s. 151–167.

[111] Pietras P. (2009). *Ekstrakcja leksykalna* [w:] Lubaszewski W. (red.), *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu* (s. 187–240). Kraków: Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH.

[112] Pisarek P. (2009). *Słownik fleksyjny* [w:] Lubaszewski W. (red.), *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu* (s. 37–68). Kraków: Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH.

[113] Piskorski J., Homola P., Marciniak M., Mykowiecka A., Przepiórkowski A., Woliński M. (2004). *Information extraction for Polish using the SProUT platform.* [w:] Kłopotek M.A., Wierzchoń S.T., Trojanowski K. (red.), *Intelligent Information Processing and Web Mining, Advances in Soft Computing* (s. 227–236).

[114] Piskorski J. (2004). *Automatic named-entity recognition for Polish.* [w:] Bolc L., Michalewicz Z., Nishida T. (red.), *Proceedings of the International Workshop on Intelligent Media Technology for Communicative Intelligence* (s. 122–133).

[115] Piskorski J. (2004). *Extraction of Polish Named-Entities.* [w:] Lino M.T., Xavier M.F., Ferreira F., Costa R., Silva R. (red.), *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC* (s. 313–316).

[116] Piskorski J. (2005). *Named-entity recognition for Polish with SProUT.* [w:] Bolc L., Michalewicz Z., Nishida T. (red.), *Intelligent Media Technology for Communicative Intelligence* (s. 122–133). Berlin, Heidelberg: Springer.

[117] Pohl A. (2009). *Automatic Construction of the Polish Nominal Lexicon for the OpenCyc Ontology.* [w:] Kłopotek M.A., Przepiórkowski A., Wierzchoń S.T., Trojanowski K. (red.), *Recent Advances in Intelligent Information Systems* (s. 51–64). Warszawa: EXIT.

[118] Pohl A. (2012). *Classifying the Wikipedia Articles into the OpenCyc Taxonomy.* [w:] Rizzo G., Mendes P., Charton E., Hellmann S., Kalyanpur A. (red.), *Proceedings of the Web of Linked Entities Workshop in conjuction with the 11th International Semantic Web Conference* (s. 5–16).

[119] Pohl A. (2009). *Rozstrzyganie wieloznaczności, maszynowa reprezentacja znaczenia wyrazu i ekstrakcja znaczeń* [w:] Lubaszewski W. (red.), *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu* (s. 241–255). Kraków: Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH.

[120] Pohl A. (2010). *The Polish Cyc lexicon as a bridge between Polish language and the Semantic Web.* [w:] Ganzha M., Paprzycki M. (red.), *Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference on* (s. 485–492).

[121] Pohl A. (2010). *The Semi-automatic Construction of the Polish Cyc Lexicon.* „Investigationes Linguisticae" 21, s. 17–38.

[122] Pohl A. (2012). *An Ontology-based Method for an Efficient Acquisition of Relation Extraction Training and Testing Examples.* [w:] Bouvry P., Klopotek M.A., Leprevost F., Marciniak M., Mykowiecka A., Rybiński H. (red.), *Security and Intelligent Information Systems* (s. 318–331).

[123] Pohl A. (2012). *Improving the Wikipedia Miner Word Sense Disambiguation Algorithm.* [w:] Ganzha M., Paprzycki M. (red.), *Proceedings of Federated Conference on Computer Science and Information Systems 2012* (s. 241–248).

[124] Pohl A. (2006). *Mapowanie ontologii na przykładzie Cyc i Słownika Semantycznego Języka Polskiego.* Praca magisterska, Kraków: Akademia Górniczo-Hutnicza.

[125] Pohl A. (2012). *ROD – Ruby Object Database.* „Studia Informatica" 33/2A, s. 281-298.

[126] Pooley D., Raya R.M. (2008). *SRX 2.0 Specification.* Dostępne <http://www.ttt.org/oscarStandards/srx/>

[127] Przepiórkowski A., Bańko M., Górski R.L., Lewandowska-Tomaszczyk B. (2012). *Narodowy Korpus Języka Polskiego.* Warsaw: Wydawnictwo Naukowe PWN.

[128] Przepiórkowski A. (2005). *The IPI PAN Corpus in numbers.* [w:] Vetulani Z. (red.), *Proceedings of the 2nd Language & Technology Conference* (s. 27–31).

[129] Przepiórkowski A., Górski R.L., Łaziński M., Pęzik P. (2009). *Recent Developments in the National Corpus of Polish.* [w:] Levická J., Garabík R. (red.), *NLP, Corpus Linguistics, Corpus Based Grammar Research: Proceedings of the Fifth International Conference* (s. 302–309).

[130] Przepiórkowski A. (2004). *Korpus IPI PAN. Wersja wstępna.* Warszawa: Instytut Podstaw Informatyki PAN.

[131] Quinlan J.R. (1993). *C4.5: programs for machine learning.* Burlington, Massachusetts: Morgan Kaufmann.

[132] Radziszewski A., Maziarz M. (2011). *Developing free morphological data for Polish.* „Cognitive Studies — Etudes Cognitives" 11, s. 201-212.

[133] Riloff E., Jones R., others (1999). *Learning dictionaries for information extraction by multi-level bootstrapping.* [w:] Hendler J., Subramanian D., Uthurusamy R., Hayes-Roth B. (red.), *Proceedings of the sixteenth National Conference on Artificial Intelligence* (s. 474–479).

[134] Riloff E., Lorenzen J. (1999). *Extraction-based text categorization: Generating domain-specific role relationships automatically.* [w:] Strzalkowski T. (red.), *Natural Language Information Retrieval* (s. 167–196). Dordrecht: Kluwer Academic Publishers.

[135] Rosch E.H. (1973). *On the internal structure of perceptual and semantic categories* [w:] Moore T.E. (red.), *Cognitive Development and the Acquisition of Language* (s. 111-144). New York: Academic Press.

[136] Russell S.J., Norvig P. (2010). *Artificial Intelligence: A Modern Approach.* Upper Saddle River, New Jersey: Prentice Hall.

[137] Sager N. (1981). *Natural language information processing.* Boston: Addison-Wesley Publishing Company.

[138] Sarjant S., Legg C., Robinson M., Medelyan O. (2009). *All You Can Eat Ontology-Building: Feeding Wikipedia to Cyc.* [w:] Yates R.B., Berendt B., Bertino E., Peng L.E. (red.), *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 01* (s. 341–348).

[139] Saussure F.de (1916). *Cours de linguistique générale.* Paris: Payot.

[140] Savary A., Chojnacka-Kuraś M., Wesołek A., Skowrońska D., Śliwiński P. (2012). *Anotacja jednostek nazewniczych* [w:] Przepiórkowski A., Bańko M., Górski R.L., Lewandowska-Tomaszczyk B. (red.), *Narodowy Korpus Języka Polskiego* (s. 129–168). Warszawa: Wydawnictwo Naukowe PWN.

[141] Savary A., Waszczuk J., Przepiórkowski A. (2010). *Towards the Annotation of Named Entities in the National Corpus of Polish.* [w:] Calzolari N., Choukri K., Maegaard B., Mariani J., Odijk J., Piperidis S., Rosner M., Tapias D. (red.), *Proceedings of the Seventh International Conference on Language Resources and Evaluation* (s. 3622–3629).

[142] Schank R., Abelson R., Schank R.C. (1977). *Scripts Plans Goals and Understanding. An Inquiry into Human Knowledge Structures.* Hillsdale, New Jersey: LEA.

[143] Schank R.C., Riesbeck C.K. (1981). *Inside Computer Understanding: Five Programs Plus Five Miniatures.* Hillsdale, New Jersey: LEA.

[144] Shapiro S.C. (1971). *A Net Structure for Semantic Information Storage, Deduction and Retrieval.* [w:] Cooper D.C. (red.), *IJCAI* (s. 512–523).

[145] Soderland S. (1999). *Learning information extraction rules for semi-structured and free text.* „Machine learning" 34/1, s. 233–272.

[146] Sowa J.F. (1992). *Semantic Networks.* Dostępne <http://www.jfsowa.com/pubs/semnet.htm>

[147] Suchanek F.M. (2008). *Automated construction and growth of a large ontology.* Praca doktorska, Saarbrücken, Germany: Saarbrücken University.

[148] Suchanek F.M., Kasneci G., Weikum G. (2008). *YAGO: A large ontology from Wikipedia and WordNet.* „Web Semantics: Science, Services and Agents on the World Wide Web" 6/3, s. 203–217.

[149] Suchanek F.M., Kasneci G., Weikum G. (2007). *YAGO: a core of semantic knowledge.* [w:] Williamson C., Zurko M.E., Patel-Schneider P., Shenoy P. (red.), *Proceedings of the 16th international conference on World Wide Web* (s. 697–706).

[150] Suchanek F.M., Sozio M., Weikum G. (2009). *SOFIE: a self-organizing framework for information extraction.* [w:] Quemada J., León G., Maarek Y., Nejdl W. (red.), *Proceedings of the 18th international conference on World wide web* (s. 631–640).

[151] Tarski A. (1944). *The semantic conception of truth and the foundations of semantics.* „Philosophy and phenomenological research" 4/3, s. 341–376.

[152] Tesnière L., Fourquet J. (1959). *Eléments de syntaxe structurale.* Paris: Klincksieck.

[153] Toutanova K., Manning C.D. (2000). *Enriching the knowledge sources used in a maximum entropy part-of-speech tagger.* [w:] Schütze H., Su K.-Y. (red.), *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13* (s. 63–70).

[154] Turing A.M. (1950). *Computing machinery and intelligence.* „Mind" 59/236, s. 433–460.

[155] Ullman J., Widom J. (2000). *Podstawowy wykład z systemów baz danych.* Warszawa: Wydawnictwo Naukowo-Techniczne.

[156] Vetulani Z., Walkowska J., Obrębski T., Marciniak J., Konieczka P., Rzepecki P. (2009). *An Algorithm for Building Lexical Semantic Network and Its Application to PolNet-Polish WordNet Project.* [w:] Vetulani Z. (red.), *Human Language Technology. Challenges of the Information Society* (s. 369–381).

[157] Walas M., Jassem K. (2010). *Named entity recognition in a Polish question answering system.* [w:] Kłopotek M.A., Marciniak M., Mykowiecka A., Penczek W., Wierzchoń S.T. (red.), *Intelligent Information Systems* (s. 181–192).

[158] Waszczuk J. (2012). *Harnessing the CRF complexity with domain-specific constraints. The case of morphosyntactic tagging of a highly inflected language.* [w:] Kay M., Boitet C. (red.), *Proceedings of COLING* (s. 2789–2804).

[159] Witten I.H., Bell T.C. (1991). *The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression.* „Information Theory, IEEE Transactions on" 37/4, s. 1085–1094.

[160] Witten I.H., Milne D. (2008). *An effective, low-cost measure of semantic relatedness obtained from Wikipedia links.* [w:] Bunescu R., Gabrilovich E., Mihalcea R. (red.), *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press, Chicago, USA* (s. 25–30).

[161] Woliński M., Miłkowski M., Ogrodniczuk M., Przepiórkowski A., Szałkiewicz Ł. (2012). *PoliMorf: a (not so) new open morphological dictionary for Polish.* [w:] Calzolari N., Choukri K., Declerck T., Doğan M.U., Maegaard B., Mariani J., Moreno A., Odijk J., Piperidis S. (red.), *Proceedings of Eighth International Conference on Language Resources and Evaluation* (s. 860–864).

[162] Woliński M. (2004). *System znaczników morfosyntaktycznych w korpusie IPI PAN.* „Polonica" XII, s. 39–54.

[163] Woliński M. (2006). *Morfeusz—a practical tool for the morphological analysis of Polish.* [w:] Kłopotek M.A., Wierzchoń S.T., Trojanowski K. (red.), *Intelligent Information Processing and Web Mining* (s. 511–520).

[164] Ziółko B., Manandhar S., Wilson R.C., Ziółko M., Gałka J. (2008). *Application of HTK to the Polish Language.* [w:] IEEE (red.), *Audio, Language and Image Processing. International Conference on* (s. 1759–1764).

Spis rysunków

Spis tablic

Dodatki

A. Kompletna lista par symboli połączonych predykatem `#$anatomicalParts`

Tablica A.1: Lista par pojęć dla predykatu `#$anatomicalParts` zaczerpnięta z ontologii ResearchCyc – zbiór $\boldsymbol{C_R}$: $label_1$ – nazwa pierwszego argumentu, $label_2$ – nazwa drugiego argumentu.

$\boldsymbol{label_1}$	$\boldsymbol{label_2}$
#$CarnivoreOrder	#$Tail-BodyPart
#$HomoSapiens	#$Hand
#$Virus	#$Capsid
#$ParameciumCaudatum	#$aura:ContractileVacuole
#$Archaea	#$ArchaealCellWall
#$BotanicalCell	#$BotanicalCellWall
#$GramNegativeBacillusBacterium	#$BacterialCellWall-GramNegative
#$GramPositiveBacillusBacterium	#$BacterialCellWall-GramPositive
#$HomoSapiens	#$Eyebrow
#$HomoSapiens	#$Finger
#$HomoSapiens	#$Toe
#$HomoSapiens	#$Brain
#$Mosquito	#$InsectProboscis
#$Butterfly	#$InsectProboscis
#$Person	#$Hand
#$Primate	#$Hand
#$Cat	#$SpinalColumn
(#$FemaleFn #$Mammal)	#$MammaryGland
#$Vertebrate	#$Head-Vertebrate
#$Primate	#$Leg
#$Mammal	#$Groin
#$Mammal	#$Nose
(#$MaleFn #$Mammal)	#$Penis
(#$MaleFn #$Reptile)	#$Penis
#$Turtle	#$Beak
#$Cetacean	#$Blowhole
#$Platypus	#$Beak

$label_1$	$label_2$
#$Vertebrate	#$SpinalColumn
#$Mammal	#$Foot-AnimalBodyPart
#$Vertebrate	#$VertebrateSkeleton
#$Bird	#$Bill-Birds
#$Vertebrate	#$Trunk-BodyCore
#$Gopher-Rodent	#$Claw
#$Bird	#$Plumage
#$Vertebrate	#$MusculoskeletalSystem
#$AirBreathingVertebrate	#$RespiratoryTract
#$Bird	(#$GroupFn #$Feather)
#$Horse	#$Withers
#$Plant-Woody	#$Leaf
#$Cactus	#$Thorn
#$FloweringPlant	#$Stem
#$Tree-ThePlant	#$TreeBranch
#$Tree-ThePlant	#$TreeCanopy
#$Scorpion	#$Stinger
#$Arthropod	#$Leg
#$Fish	#$Gill
#$Vertebrate	#$Trachea
#$Rodent	#$Whisker
#$Fish	#$Fin
#$Wasp	#$Stinger
#$ClawedLobster	#$Pincer
#$Vertebrate	#$Pharynx
#$Primate	#$Shoulder
#$Mammal	#$Eyelash
#$Mammal	#$Nostril
#$Crab	#$Arm
#$Arthropod	#$Exoskeleton
#$Arthropod	#$Antenna-AnimalBodyPart
#$Mammal	#$HairOnHead
#$Elephant	#$Trunk-TheAppendage
#$SeaMammal	#$Blubber
#$MaleAnimal	#$ReproductiveSystem-Male
#$Cetacean	#$Fin
#$Bull-Cattle	#$Horn-AnimalBodyPart
#$Bear-Animal	#$Claw
#$SeaMammal	#$Flipper
#$Animal	#$VisualSystem
#$Marsupial	#$Whisker
#$Crab	#$Pincer

$label_1$	$label_2$
#$Animal	#$DigestiveSystem
#$Bee	#$Stinger
#$Bird	#$MobOfFeathers
#$FemaleAnimal	#$ReproductiveSystem-Female
#$Reptile	#$Tail-BodyPart
#$FelidaeFamily	#$Claw
#$Mammal	#$Diaphragm-BodyPart
#$Vertebrate	#$Mouth
#$Vertebrate	#$Skin
#$Vertebrate	#$Appendage-AnimalBodyPart
#$Vertebrate	#$Cornea
#$Primate	#$Arm
#$Primate	#$Hand
#$Octopus	#$Tentacle
#$EquineAnimal	#$Tail-BodyPart
#$Mammal	#$MobOfHair-Mammal
#$Animal	#$ReproductiveSystem
#$BirdOfPrey	#$Talon
#$Vertebrate	(#$OrganismPartTypeFn #$Vertebrate #$Eye)
#$Goose-Bird	(#$OrganismPartTypeFn #$Goose-BirdLiver)
#$Mammoth	#$Trunk-TheAppendage
#$Mastodon	#$Trunk-TheAppendage
#$ScaledAnimal	#$Scale-AnimalBodyPart
#$Primate	#$Back-AnimalBodyPart
#$SandFly	#$InsectProboscis

B. Lista par symboli połączonych predykatem **#$anatomicalParts** przetłumaczonych na język polski

1. Związki specyficzne (semantyczne):
   - #$Bear-Animal (*niedźwiedź*) – #$Claw (*pazur*)
   - #$Bee (*pszczoła*) – #$Stinger (*żądło*)
   - #$Bull-Cattle (*byk*) – #$Horn-AnimalBodyPart (*róg*)
   - #$Butterfly (*motyl*) – #$InsectProboscis (*trąbka*)
   - #$Cactus (*kaktus*) – #$Thorn (*kolec*)
   - #$Cat (*kot domowy*) – #$SpinalColumn (*kręgosłup*)
   - #$Crab (*krab*) – #$Arm (*ramię*)
   - #$Crab (*krab*) – #$Pincer (*szczypce*)
   - #$Elephant (*słoń*) – #$Trunk-TheAppendage (*trąba*)
   - #$Goose-Bird (*gęś*) – #$(OrganismPartTypeFn Goose-Bird Liver) (*gęsia wątroba*)
   - #$Gopher-Rodent (*suseł*) – #$Claw (*pazur*)
   - #$HomoSapiens (*człowiek rozumny*) – #$Eyebrow (*brew*)
   - #$HomoSapiens (*człowiek rozumny*) – #$Finger (*palec*)
   - #$HomoSapiens (*człowiek rozumny*) – #$Toe (*palec u nogi*)
   - #$Lobster (*homar*) – #$Pincer (*szczypce*)
   - #$Mammoth (*mamut*) – #$Trunk-TheAppendage (*trąba*)
   - #$Marsupial (*torbacz*) – #$Whisker (*włos czuciowy*)
   - #$Mastodon (*mastodont*) – #$Trunk-TheAppendage (*trąba*)
   - #$Mosquito (*komar*) – #$InsectProboscis (*trąbka*)
   - #$Octopus (*ośmiornica*) – #$Tentacle (*czułek*)
   - #$Person (*osoba*) – #$Hand (*ręka*)
   - #$Platypus (*dziobak*) – #$Beak (*dziób*)
   - #$SandFly (*moskit*) – #$InsectProboscis (*trąbka*)
   - #$Scorpion (*skorpion*) – #$Stinger (*żądło*)
   - #$Turtle (*żółw*) – #$Beak (*dziób*)
   - #$Wasp (*osa*) – #$Stinger (*żądło*)

2. Związki pośrednie (semantyczne):
   - `#$Arthropod` (*stawonóg*) – `#$Antenna-AnimalBodyPart` (*czułek*)
   - `#$Arthropod` (*stawonóg*) – `#$Exoskeleton` (*szkielet zewnętrzny*)
   - `#$Arthropod` (*stawonóg*) – `#$Leg` (*noga*)
   - `#$BirdOfPrey` (*ptak drapieżny*) – `#$Talon` (*szpon*)
   - `#$Bird` (*ptak*) – `#$Bill-Birds` (*dziób*)
   - `#$Bird` (*ptak*) – `#$(GroupFn Feather)` (*pióra*)
   - `#$Bird` (*ptak*) – `#$MobOfFeathers` (*upierzenie*)
   - `#$Bird` (*ptak*) – `#$Plumage` (*upierzenie*)
   - `#$Cetacean` (*waleń*) – `#$Blowhole` (*nozdrze*)
   - `#$Cetacean` (*waleń*) – `#$Fin` (*płetwa*)
   - `#$EquineAnimal` (*koniowaty*) – `#$Tail-BodyPart` (*ogon*)
   - `#$FelidaeFamily` (*kotowate*) – `#$Claw` (*pazur*)
   - `#$Fish` (*ryba*) – `#$Fin` (*płetwa*)
   - `#$Fish` (*ryba*) – `#$Gill` (*skrzele*)
   - `#$FloweringPlant` (*roślina okrytonasienna*) – `#$Stem` (*łodyga*)
   - `#$Plant-Woody` (*roślina drzewiasta*) – `#$Leaf` (*liść*)
   - `#$Reptile` (*gad*) – `#$Tail-BodyPart` (*ogon*)
   - `#$Rodent` (*gryzoń*) – `#$Whisker` (*włos czuciowy*)
   - `#$Tree-ThePlant` (*drzewo*) – `#$TreeBranch` (*gałąź*)
   - `#$Tree-ThePlant` (*drzewo*) – `#$TreeCanopy` (*korona*)
3. Związki abstrakcyjne (ontologiczne):
   - `#$Animal` (*zwierzę*) – `#$DigestiveSystem` (*system trawienny*)
   - `#$Animal` (*zwierzę*) – `#$ReproductiveSystem` (*układ rozrodczy*)
   - `#$Animal` (*zwierzę*) – `#$VisualSystem` (*system wzrokowy*)
   - `#$FemaleAnimal` (*samica*) – `#$ReproductiveSystem-Female` (*żeński układ rozrodczy*)
   - `#$(FemaleFn Mammal)` (*samica ssak*) – `#$MammaryGland` (*gruczoł sutkowy*)
   - `#$MaleAnimal` (*samiec*) – `#$ReproductiveSystem-Male` (*męski układ rozrodczy*)
   - `#$(MaleFn Mammal)` (*samiec ssak*) – `#$Penis` (*penis*)
   - `#$(MaleFn Reptile)` (*samiec gada*) – `#$Penis` (*penis*)
   - `#$Mammal` (*ssak*) – `#$Diaphragm-BodyPart` (*przepona*)
   - `#$Mammal` (*ssak*) – `#$Eyelash` (*rzęsa*)
   - `#$Mammal` (*ssak*) – `#$Foot-AnimalBodyPart` (*stopa*)
   - `#$Mammal` (*ssak*) – `#$Groin` (*pachwina*)
   - `#$Mammal` (*ssak*) – `#$HairOnHead` (*włosy na głowie*)
   - `#$Mammal` (*ssak*) – `#$MobOfHair-Mammal` (*owłosienie*)

- `#$Mammal` (*ssak*) – `#$Nose` (*nos*)
- `#$Mammal` (*ssak*) – `#$Nostril` (*nozdrze*)
- `#$Primate` (*ssak naczelny*) – `#$Arm` (*ramię*)
- `#$Primate` (*ssak naczelny*) – `#$Back-AnimalBodyPart` (*grzbiet*)
- `#$Primate` (*ssak naczelny*) – `#$Hand` (*ręka*)
- `#$Primate` (*ssak naczelny*) – `#$Leg` (*noga*)
- `#$Primate` (*ssak naczelny*) – `#$Shoulder` (*bark*)
- `#$ScaledAnimal` (*zwierzę pokryte łuskami*) – `#$Scale-AnimalBodyPart` (*łuska*)
- `#$SeaMammal` (*ssak morski*) – `#$Blubber` (*tłuszcz*)
- `#$SeaMammal` (*ssak morski*) – `#$Flipper` (*płetwa*)
- `#$Vertebrate` (*kręgowiec*) – `#$Appendage-AnimalBodyPart` (*członek ciała*)
- `#$Vertebrate` (*kręgowiec*) – `#$Cornea` (*rogówka*)
- `#$Vertebrate` (*kręgowiec*) – `#$Head-Vertebrate` (*głowa kręgowca*)
- `#$Vertebrate` (*kręgowiec*) – `#$Mouth` (*usta*)
- `#$Vertebrate` (*kręgowiec*) – `#$(OrganismPartTypeFn Vertebrate Eye)` (*oko kręgowca*)
- `#$Vertebrate` (*kręgowiec*) – `#$Pharynx` (*gardło*)
- `#$Vertebrate` (*kręgowiec*) – `#$Skin` (*skóra*)
- `#$Vertebrate` (*kręgowiec*) – `#$SpinalColumn` (*kręgosłup*)
- `#$Vertebrate` (*kręgowiec*) – `#$Trachea` (*tchawica*)
- `#$Vertebrate` (*kręgowiec*) – `#$Trunk-BodyCore` (*tułów*)
- `#$Vertebrate` (*kręgowiec*) – `#$VertebrateSkeleton` (*kościec wewnętrzny*)

C. Część taksonomia ontologii Cyc zakorzeniona w pojęciu **#$Bird** i przetłumaczona na język polski

- #$Peacock (*paw*)
- #$Woodpecker (*dzięcioł*)
- #$Kingfisher (*zimorodek*)
- #$Stork (*bocian*)
- #$Wren (*strzyżyk*)
- #$Crow (*wrona*)
- #$Cuckoo (*kukułka*)
- #$Vulture (*sęp*)
- #$Pheasant (*bażant*)
- #$Pigeon (*gołąb*)
- #$Toucan (*tukan*)
- #$Raven (*kruk*)
- #$Thrush-Bird (*drozd*)
- #$Chickadee (*sikorka*)
- #$Ibis (*ibis*)
- #$Nightingale (*słowik rdzawy*)
- #$Coot (*łyska*)
- #$Hummingbird (*koliber*)
- #$Parrot (*papuga*)
  - #$Parakeet (*papużka*)
  - #$Macaw (*ara*)
  - #$AfricanGreyParrot (*papuga popielata*)
- #$BirdOfPrey (*ptak drapieżny*)

    - #$Owl (*sowa*)
    - #$Condor (*kondor*)
- #$Waterfowl (*ptak wodny*)
    - #$Goose-Bird (*gęś*)
        - #$Gosse-Domestic (*gęś domowa*)
        - #$HawaiianGoose (*bernikla kanadyjska*)
        - #$CanadaGoose (*bernikla hawajska*)
    - #$Duck (*kaczka*)
        - #$MallardDuck (*krzyżówka*)
    - #$Penguin (*pingwin*)
    - #$Pelican (*pelikan*)
    - #$Swan (*łabędź*)
    - #$Grebe-Western (*perkoz*)
- #$ShoreBird (*ptak nadbrzeżny*)
    - #$Heron (*czapla*)
    - #$Seagull (*mewa*)
    - #$Pelican (*pelikan*)
- #$Poultry (*drób*)
    - #$Goose-Domestic (*gęś domowa*)
    - #$Chicken (*kura*)
    - #$Duck (*kaczka*)
    - #$Turkey-Bird (*indyk*)

D. Wzorce formalne relacji *całość-część* o $CD_P \geq 2$

Tablica D.1: Lista wzorców formalnych relacji *całość-część* o $CD_P \geq 2$: `dir` – kolejność argumentów, `arg_l` – argument lewy, `arg_r` – argument prawy, `inner` – kontekst wewnętrzny, CT_P – liczba różnych zdań, w których pojawił się wzorzec, CD_P – liczba różnych par argumentów pasujących do wzorca.

dir	**arg_l**	**arg_r**	**inner**	$\boldsymbol{CT_P}$	$\boldsymbol{CD_P}$
rl	subst:sg:gen:f	subst:sg:gen:m1	--	32	13
rl	subst:sg:gen:f	subst:sg:gen:m2	--	25	16
lr	subst:sg:acc:m2	subst:pl:acc:m3	za	25	2
rl	subst:sg:nom:f	subst:sg:gen:m2	--	20	12
rl	subst:sg:acc:f	subst:sg:gen:m1	--	15	13
rl	subst:sg:gen:m3	subst:sg:gen:m1	--	15	7
rl	subst:sg:nom:m3	subst:sg:gen:m1	--	14	4
rl	subst:pl:gen:f	subst:sg:gen:m1	--	13	4
rl	subst:sg:gen:f	subst:sg:gen:f	--	13	11
rl	subst:sg:nom:f	subst:sg:gen:m1	--	12	8
rl	subst:sg:acc:f	subst:sg:gen:m2	--	12	7
rl	subst:pl:acc:m3	subst:pl:gen:m1	--	12	2
rl	subst:sg:nom:m3	subst:sg:gen:m2	--	11	9
rl	subst:pl:gen:m3	subst:sg:gen:f	--	10	4
rl	subst:sg:acc:m3	subst:sg:gen:f	--	9	6
rl	subst:sg:acc:f	subst:sg:gen:f	--	8	4
rl	subst:sg:gen:m3	subst:sg:gen:m2	--	8	7
rl	subst:sg:nom:f	subst:sg:gen:f	--	8	7
rl	subst:pl:gen:n	subst:sg:gen:m1	--	8	5
rl	subst:sg:loc:f	subst:sg:gen:f	--	8	7
rl	subst:sg:gen:m3	subst:sg:gen:f	--	7	6
rl	subst:pl:acc:m3	subst:sg:gen:f	--	7	5
rl	subst:sg:acc:m3	subst:sg:gen:m3	--	7	6
rl	subst:pl:nom:m3	subst:sg:gen:m3	--	7	4
rl	subst:sg:nom:m3	subst:sg:gen:m3	--	7	3
rl	subst:pl:nom:m3	subst:sg:gen:f	--	7	4
rl	subst:sg:loc:f	subst:sg:gen:m1	--	7	5

dir	arg_l	arg_r	inner	CT_P	CD_P
rl	subst:pl:loc:f	subst:pl:gen:f	--	6	6
rl	subst:pl:nom:f	subst:pl:gen:f	--	6	5
rl	subst:pl:acc:f	subst:pl:gen:m1	--	6	5
lr	subst:sg:gen:m2	subst:pl:acc:m3	za	6	2
rl	subst:sg:loc:m3	subst:sg:gen:m1	--	6	3
rl	subst:pl:nom:f	subst:sg:gen:m1	--	6	6
rl	subst:sg:gen:f	subst:sg:gen:n	--	6	3
rl	subst:sg:gen:f	subst:pl:gen:m2	--	6	6
rl	subst:pl:gen:f	subst:sg:gen:f	--	6	4
rl	subst:pl:acc:m3	subst:pl:gen:m3	--	6	3
rl	subst:sg:acc:f	subst:sg:gen:m2	z	6	5
lr	subst:pl:dat:m1	subst:pl:acc:m3	--	6	4
rl	subst:sg:loc:f	subst:sg:gen:m2	--	5	4
rl	subst:pl:loc:m3	subst:pl:gen:m1	--	5	3
rl	subst:pl:gen:n	subst:sg:gen:m2	--	5	3
rl	subst:pl:nom:m3	subst:pl:gen:f	--	5	4
rl	subst:pl:gen:m3	subst:pl:gen:f	--	5	5
rl	subst:sg:nom:m3	subst:sg:gen:f	--	5	4
rl	subst:pl:loc:m3	subst:sg:gen:m3	--	5	4
rl	subst:sg:inst:m3	subst:sg:gen:f	--	5	4
rl	subst:pl:gen:m3	subst:sg:gen:m3	--	4	3
rl	subst:sg:inst:m3	subst:sg:gen:m1	--	4	3
lr	subst:sg:voc:m2	subst:pl:acc:m3	--	4	3
rl	subst:pl:inst:f	subst:pl:gen:f	--	4	3
lr	subst:sg:voc:m2	subst:sg:gen:f	--	4	3
rl	subst:pl:acc:m3	subst:sg:gen:m2	--	4	3
rl	subst:sg:inst:f	subst:sg:gen:m2	--	4	3
rl	subst:pl:nom:f	subst:sg:gen:m2	--	4	4
rl	subst:pl:gen:m3	subst:sg:gen:m2	--	4	2
rl	subst:pl:loc:m3	subst:sg:gen:f	--	4	4
lr	subst:sg:dat:f	subst:sg:gen:f	--	4	3
rl	subst:pl:nom:m3	subst:pl:gen:m3	--	4	3
rl	subst:pl:inst:m3	subst:sg:gen:f	--	4	4
lr	subst:sg:nom:m1	subst:sg:gen:m3	doznał wstrząśnienia	4	2
rl	subst:pl:inst:m3	subst:sg:gen:m3	--	4	2
lr	subst:sg:acc:f	subst:pl:inst:f	--	4	2
rl	subst:pl:acc:m3	subst:sg:gen:m1	--	4	4
lr	subst:sg:dat:m2	subst:sg:acc:m3	--	4	2
rl	subst:pl:acc:m3	subst:sg:gen:m3	--	4	2
rl	subst:sg:acc:m3	subst:sg:gen:m2	--	4	3
rl	subst:pl:loc:f	subst:sg:gen:m1	--	4	3

dir	arg_l	arg_r	inner	CT_P	CD_P
rl	subst:sg:acc:f	subst:sg:dat:m1	--	4	3
rl	subst:sg:gen:f	subst:pl:gen:m1	--	4	4
rl	subst:pl:loc:m3	subst:pl:gen:f	--	4	4
rl	subst:pl:loc:f	subst:sg:gen:f	--	4	4
rl	subst:sg:gen:f	subst:sg:gen:m3	--	4	4
rl	subst:pl:nom:f	subst:pl:gen:m3	--	4	2
rl	subst:pl:nom:f	subst:sg:gen:f	--	4	4
rl	subst:pl:acc:f	subst:pl:gen:m3	--	4	3
rl	subst:sg:inst:f	subst:sg:gen:f	--	4	4
rl	subst:sg:gen:f	subst:pl:gen:f	--	3	2
rl	subst:pl:acc:f	subst:pl:gen:f	--	3	3
rl	subst:sg:loc:f	subst:sg:gen:m3	--	3	3
rl	subst:pl:loc:f	subst:sg:gen:m3	--	3	3
lr	subst:sg:gen:f	subst:sg:gen:f	i	3	2
rl	subst:pl:gen:f	subst:sg:gen:m2	--	3	2
rl	subst:sg:loc:m3	subst:sg:gen:f	--	3	3
lr	subst:sg:dat:m2	subst:sg:gen:n	z	3	3
lr	subst:sg:nom:m1	subst:sg:inst:f	z siwą	3	2
rl	subst:pl:nom:m3	subst:sg:gen:m2	--	3	3
lr	subst:sg:acc:m1	subst:pl:inst:f	--	3	2
lr	subst:sg:acc:m1	subst:sg:inst:f	--	3	3
lr	subst:sg:nom:m1	subst:sg:inst:f	uderzył	3	3
rl	subst:pl:loc:n	subst:sg:gen:f	--	3	2
lr	subst:sg:nom:m1	subst:sg:gen:m3	doznał wstrząsu	3	2
rl	subst:sg:gen:m3	subst:pl:gen:n	u	3	2
rl	subst:sg:inst:m3	subst:sg:gen:m2	--	3	2
rl	subst:pl:loc:n	subst:sg:gen:m1	--	3	3
rl	subst:pl:acc:n	subst:sg:gen:m1	--	3	3
rl	subst:pl:gen:n	subst:pl:gen:m1	--	3	2
lr	subst:sg:nom:m2	subst:sg:acc:f	--	3	3
rl	subst:sg:nom:f	subst:sg:gen:m2	z	3	3
lr	subst:sg:nom:f	subst:sg:gen:m3	doznała złamania	3	3
lr	subst:sg:acc:m1	subst:sg:acc:f	w	3	3
lr	subst:pl:nom:m2	subst:sg:gen:f	--	3	2
rl	subst:pl:acc:f	subst:sg:gen:f	--	3	3
rl	subst:pl:inst:f	subst:sg:gen:f	--	3	3
lr	subst:sg:nom:m1	subst:sg:gen:f	bez	3	3
rl	subst:sg:loc:m3	subst:sg:gen:m2	--	3	3
lr	subst:sg:dat:m1	subst:pl:gen:f	do	3	2
rl	subst:sg:acc:m3	subst:sg:dat:m1	--	2	2
lr	subst:sg:dat:m1	subst:sg:acc:m3	--	2	2
lr	subst:sg:nom:m1	subst:sg:acc:m3	ma złamany	2	2

dir	arg_l	arg_r	inner	CT_P	CD_P
lr	subst:sg:nom:m1	subst:sg:gen:m3	--	2	2
rl	subst:pl:nom:n	subst:sg:gen:m1	--	2	2
rl	subst:sg:gen:m3	subst:pl:dat:m1	--	2	2
lr	subst:sg:nom:m1	subst:sg:gen:m3	doznał złamania	2	2
rl	subst:sg:acc:f	subst:sg:gen:m2	białego	2	2
lr	subst:sg:nom:m1	subst:pl:gen:f	nie ma	2	2
rl	subst:sg:nom:f	subst:pl:gen:m1	--	2	2
lr	subst:sg:gen:m1	subst:sg:loc:f	w	2	2
rl	subst:sg:acc:f	subst:pl:dat:m1	--	2	2
lr	subst:sg:nom:m1	subst:sg:acc:m3	złamał	2	2
rl	subst:pl:loc:f	subst:pl:gen:m1	--	2	2
rl	subst:sg:gen:f	subst:sg:voc:m2	--	2	2
rl	subst:pl:loc:f	subst:pl:gen:m1	przez dwóch	2	2
lr	subst:sg:nom:m1	subst:sg:inst:f	--	2	2
rl	subst:sg:acc:f	subst:sg:gen:f	z	2	2
rl	subst:pl:acc:f	subst:sg:gen:m3	--	2	2
lr	subst:sg:nom:m1	subst:sg:gen:m3	doznał urazu głowy i	2	2
rl	subst:sg:acc:m3	subst:sg:gen:m1	--	2	2
rl	subst:sg:gen:m3	subst:sg:nom:m1	--	2	2
lr	subst:sg:dat:m1	subst:sg:gen:m3	z	2	2
lr	subst:sg:nom:f	subst:sg:gen:m3	z podejrzeniem wstrząśnienia	2	2
lr	subst:sg:acc:f	subst:sg:gen:m3	ze wstrząsem	2	2
lr	subst:pl:nom:m2	subst:sg:gen:f	bez	2	2
lr	subst:sg:acc:f	subst:sg:gen:m3	z podejrzeniem wstrząsu	2	2
lr	subst:sg:dat:m1	subst:sg:acc:f	--	2	2
rl	subst:pl:gen:f	subst:pl:gen:m2	--	2	2
rl	subst:pl:acc:m3	subst:sg:loc:f	na	2	2
lr	subst:pl:gen:m1	subst:sg:inst:f	uderzył go	2	2
rl	subst:pl:nom:m3	subst:pl:gen:n	--	2	2
rl	subst:sg:gen:f	subst:pl:gen:m1	starych	2	2
lr	subst:sg:dat:n	subst:sg:nom:m3	--	2	2
lr	subst:sg:nom:f	subst:sg:inst:f	została uderzona	2	2
lr	subst:sg:gen:m1	subst:sg:inst:m3	--	2	2
lr	subst:sg:acc:m2	subst:sg:acc:m3	w	2	2
rl	subst:pl:acc:f	subst:sg:gen:m1	--	2	2
rl	subst:sg:gen:n	subst:sg:gen:m3	--	2	2
rl	subst:pl:gen:n	subst:sg:gen:m3	--	2	2
rl	subst:sg:gen:m3	subst:sg:gen:m3	--	2	2
rl	subst:sg:loc:m3	subst:sg:gen:m3	--	2	2

dir	arg_l	arg_r	inner	CT_P	CD_P
lr	subst:sg:dat:f	subst:sg:gen:n	nóż do	2	2
lr	subst:sg:nom:m1	subst:sg:inst:f	z czarną	2	2
rl	subst:sg:loc:f	subst:sg:nom:m1	--	2	2
lr	subst:sg:dat:m2	subst:sg:acc:n	w	2	2
rl	subst:sg:gen:m3	subst:pl:gen:m2	u	2	2
rl	subst:sg:acc:m3	subst:sg:nom:m1	--	2	2
rl	subst:pl:loc:f	subst:sg:gen:n	--	2	2

Tablica D.2: Oznaczenia wykorzystywane do opisu wzorców formalnych.

subst	rzeczownik
sg	liczba pojedyncza
pl	liczba mnoga
nom	mianownik
acc	biernik
dat	celownik
gen	dopełniacz
inst	narzędnik
loc	miejscownik
voc	wołacz
m1	rodzaj męski osobowy
m2	rodzaj męski żywotny
m3	rodzaj męski nieżywotny
f	rodzaj żeński
n	rodzaj nijaki

E. Lista predykatów DBpedii odpowiadających relacji *całość-część*

Tablica E.1: Lista predykatów występujących w DBpedii, odpowiadających relacji *całość-część*.

Predykat	**Kierunek**
affiliation	inverse
album	inverse
associatedBand	inverse
board	inverse
almaMater	inverse
athletics	direct
capital	direct
childOrganisation	direct
citizenship	inverse
commandStructure	inverse
committeeInLegislature	direct
constructionMaterial	direct
countySeat	direct
employer	inverse
ethnicGroup	inverse
ethnicity	inverse
europeanAffiliation	inverse
europeanParliamentGroup	inverse
formerTeam	inverse
house	inverse
institution	inverse
internationalAffiliation	inverse
isPartOfMilitaryConflict	inverse
keyPerson	direct
leader	direct
league	inverse
majorIsland	direct
memberOfParliament	direct

Predykat	**Kierunek**
militaryBranch	inverse
mountainRange	inverse
nationalAffiliation	inverse
part	inverse
notableCommander	direct
politicalPartyInLegislature	direct
structuralSystem	direct
subregion	direct
subsidiary	direct
team	inverse
youthWing	direct

F. Oznaczenia matematyczne

$\boldsymbol{X}$	zbiór
$2^{\boldsymbol{X}}$	zbiór wszystkich podzbiorów $\boldsymbol{X}$
$\in$	należenie do zbioru
$\times$	iloczyn kartezjański
$\bigcup$	suma zbiorów
$\bigcap$	przecięcie zbiorów
$\sum$	suma
$\wedge$	koniunkcja
$\vee$	alternatywa
$\neg$	negacja
$\forall$	kwantyfikator ogólny
$\exists$	kwantyfikator szczegółowy
$\Rightarrow$	wynikanie
$\Leftrightarrow$	równoważność
$\overset{def}{\Leftrightarrow}$	definicja
$\oplus$	konkatenacja
$\lvert\boldsymbol{X}\rvert$	moc zbioru
$[a;b]$	przedział domknięty a, b
$(a;b)$	przedział otwarty a, b
(a,b)	para a, b
$P(x)$	prawdopodobieństwo
$P(x\lvert y)$	prawdopodobieństwo warunkowe
$f(x), F(x)$	funkcje
Pr	precyzja (ang. *precision*)
Rc	pokrycie (ang. *recall*)