System NEKST - wykorzystanie narzędzi NLP

Outline

Analiza morfosyntaktyczna

- Biblioteka OpenNLP (model *Maximum Entropy*).
- Korpus NKJP.
- Samo tagowanie, bez detekcji lematu.
- Modyfikacja algorytmu wyszukiwania najlepszego dopasowania (*Beam search*):
  - ograniczenie pamięci do jednego najlepszego wyboru - sekwencyjne przypisywanie tagów,
  - optymalizacja kodu - ograniczenie przeszukiwania przestrzeni tylko do tagów wskazanych jako poprawne przez słownik Polimirf-a.

Analiza morfosyntaktyczna

- Modelowanie kontekstu:
  Zdanie: Premiera spektaklu w teatrze.
  Wygenerowany kontekst (dla słowa *spektaklu*):
  default, w=spektaklu, suf=u, suf=lu, suf=klu, suf=aklu, pre=s, pre=sp, pre=spe, pre=spek, p=Premiera, t=subst:sg:nom:f, pp=*SB*, n=w, nn=teatrze
- Ewaluacja (MaxEnt - 50 iteracji):

Algorytm	PW	PS	PPW	PPS	W/s	S/s
Standard	0.913	0.117	0.930	0.140	4148.4	101.0
Standard-validated	**0.979**	**0.663**	**0.992**	**0.982**	8820.3	214.1
Optimized-validated	0.946	0.298	0.966	0.417	**52236.0**	**1262.4**

- Czas uczenia (procesor Intel 4.5 GHz): ok 48 min.

Wyszukiwarka a formy podstawowe słów

- Potrzeba wyszukiwania bez uwzględnienia formy ortograficznej słów.
  - premier, premiera, premierowi, . . . -> **premier**
- Duży wpływ błędnego przypisania formy podstawowej na wyniki wyszukiwania.
- Bardzo krótkie zapytania, często niezgodne ze składnią języka (lista słów kluczowych).
- W wielu przypadkach sprowadzanie do form podstawowych jest zadaniem ujednoznaczniania sensów słów.

Przykład

Odmiana słowa *damy*

damy	dać	fin:pl:pri:perf	pospolita
damy	dama	subst:pl:nom:f	pospolita

Przykład użycia

damy brylowały na parkiecie
damy podręczniki

Odmiana słowa *premier*

premiera	premier	subst:sg:acc:m1	pospolita
premiera	premiera	subst:sg:nom:f	pospolita

Przykład użycia

spotkanie premiera polski
premiera spektaklu

Rozpoznawanie nazw własnych

- Projekt Liner2 - bazujący na CRF (*ang. Conditional Random Fields*).
- Zastosowanie:
  - indeksowanie wykrytych nazw własnych oraz premiowanie ich wystąpienia w dokumentach,
  - źródło relacji IS-A (tylko typ OSOBA) dla procesu tworzącego taksonomię.
- Problemy:
  - niezadowalająca jakość działania dla klas innych niż OSOBA,
  - wysoka złożoność czasowa algorytmu - najdłużej działający proces anotacji dokumentów w systemie NEKST.

Wykrywanie fraz rzeczownikowych/czasownikowych

- Biblioteka OpenNLP (model *Maximum Entropy*).
- Dane treningowe: Korpus Języka Polskiego Politechniki Wrocławskiej.
- Kontekst budowany jedynie w oparciu o części mowy, lematy oraz formy ortograficzne.
- Ewaluacja:
  - Precyzja: 0.729
  - Przywołanie: 0.695
  - F-Miara: 0.712
- Wynik anotacji wykorzystywany przez algorytm wykrywania faktów typu NP-VP-NP.

Parsowanie zależnościowe

- Zbiór treningowy opracowany w IPI PAN - 8004 zdania.
- Wykorzystane algorytmy:
  - Malt Parser - szybki algorytm o złożoności liniowej.
  - Mate-tools - lepsza jakość działania, większa złożoność.
- Zastosowanie:
  - regułowy algorytm wykrywania faktów typu NP-VP-NP,
  - regułowy algorytm wykrywania relacji IS-A (budowa taksonomii),
  - składowa rankingu wyszukiwania,
  - wykrywanie *Question Focus* w pytaniach.

Parsowanie zależnościowe - możliwe ulepszenia

- Lepsze wyniki po podłączeniu metod generalizujących znaczenie słów, np. poprzez grupowanie (*brown clusters*).
- Najnowsze podejście: połączenie tagowania oraz parsowania zależnościowego w jednym algorytmie.
  - Najczęściej realizacja dwuetapowa: wybór kilku najlepszych wariantów tagowania, a następnie wybór jednego maksymalizującego prawdopodobieństwo rozkładu zależnościowego.
  - Potrzebna spora ilość danych treningowych.
  - Dostępne gotowe oprogramowanie: https://code.google.com/p/deepsyntacticparsing/.

Wykrywanie Question Focus

- Zadanie: wykrycie w zapytaniu frazy określającej przedmiot zapytania (*ang. Question Focus*).
- Przykład: W którym **mieście** urodził się Adam Mickiewicz?
- Zbiór treningowy opracowany w IPI PAN - 583 zdania.
- Zastosowano CRF (*ang. Conditional Random Fields*) - projekt Mallet http://mallet.cs.umass.edu/index.php.
- Kontekst: wynik analizy morfosyntaktycznej + wynik parsowania zależnościowego.
- Wyniki (precyzja):
  - Idealne dopasowanie: 0.609
  - Część wspólna/podzbiór: 0.252
  - Użyteczna anotacja (idealne dopasowanie + Część wspólna/podzbiór): 0.861

Ekstrakcji relacji IS-A

- Zadanie: ekstrakcja relacji hiperonimii/hiponimii `jest typu/należy do klasy` z nieustrukturalizowanych danych tekstowych.
- Trójka (E_1, R, E_2), gdzie E_1, E_2 to frazy rzeczownikowe desygnujące obiekty/encje, a R to fraza czasownikowa desygnująca relację hiperonimii/hiponimii.
- Podejścia do ekstrakcji relacji IS-A dzielą się na dwie kategorie:
  - w oparciu o wzorce tekstowe (najbardziej znane: reguły Hearsta, przykład "W spotkaniu udział wzięli **piłkarze** tacy jak: **Jan Nowak, Jan Kowalski**, którzy reprezentowali klub X"),
  - w oparciu o informację statystyczną o współwystępowaniu fraz rzeczownikowych.

Ekstrakcja relacji IS-A w oparciu o wzorce

- ✓Metody opierające się na wzorcach pozwalają uzyskać wyższą precyzję niż metody statystyczne.
- ✗Charakteryzują się niższym przywołaniem (ekstrahowane relacje muszą jawnie wystąpić w tekście).

Ekstrakcja relacji IS-A w oparciu o wzorce – nowe podejście

Nowa metoda ekstrakcji relacji IS-A oparta o wzorce

- wykorzystanie narzędnikowej i mianownikowej odmiany frazy rzeczownikowej do identyfikacji relacji IS-A,
- wykorzystanie parsera zależnościowego do identyfikacji granic fraz rzeczownikowych biorących udział w relacji.

Liczba ekstrahowanych relacji jest dodatkowo zwiększana przy pomocy nowatorskiej metody nazwanej przez nas wzmacnianie pseudo-podklasami (*pseudo-subclass boosting*). Jest to metoda niezależna od wykorzystywanych wzorców, więc może być stosowana np. we wspomaganiu ekstrakcji regułami Hearsta.

Nasze podejście – metoda ekstrakcji

Typowe konstrukcje wyrażające hiperonimię w języku polskim:

$$NP_1^{Nom} \text{ to } NP_2^{Nom}, \tag{1}$$

$$NP_1^{Nom} \text{ jest } NP_2^{Abl}. \tag{2}$$

Obie są sposobami wyrażenia, że fraza rzeczownikowa NP_1 `jest podklasą/instancją` NP_2. konstrukcja 2 ma swój odpowiednik dla czasu przeszłego:

$$NP_1^{Nom} \text{ był/była/było } NP_2^{Abl}. \tag{3}$$

Nasze podejście – metoda ekstrakcji

Narzędnik i mianownik są w języku polskim „łatwymi" przypadkami dla automatycznej anotacji, ponieważ:

- w mianowniku rzeczownik występuje w formie podstawowej,
- odmiana narzędnika dla rzeczowników jest *regularna* i ma unikalne końcówki:

	masculine	**feminine**
singular	-em	-ą
plural	-ami (-mi)	

Tabela: Końcówki narzędnikowej odmiany rzeczownika w j. polskim

Nasze podejście – metoda ekstrakcji

Proponujemy regułowe podejście do ekstrakcji relacji IS-A składające się z następujących kroków:

1. anotacja zdań częściami mowy i parsowanie zależnościowe,
2. wstępna selekcja drzew zależnościowych do ekstrakcji,
3. filtrowanie słownikowe głównego rzeczownika frazy NP_2,
4. zastosowanie reguł konstruujących nazwę instancji z drzewa zależnościowego NP_1 i nazwę klasy z drzewa zależnościowego NP_2,
5. końcowe filtrowanie wyników.

Nasze podejście – wstępna selekcja drzew zależnościowych do ekstrakcji

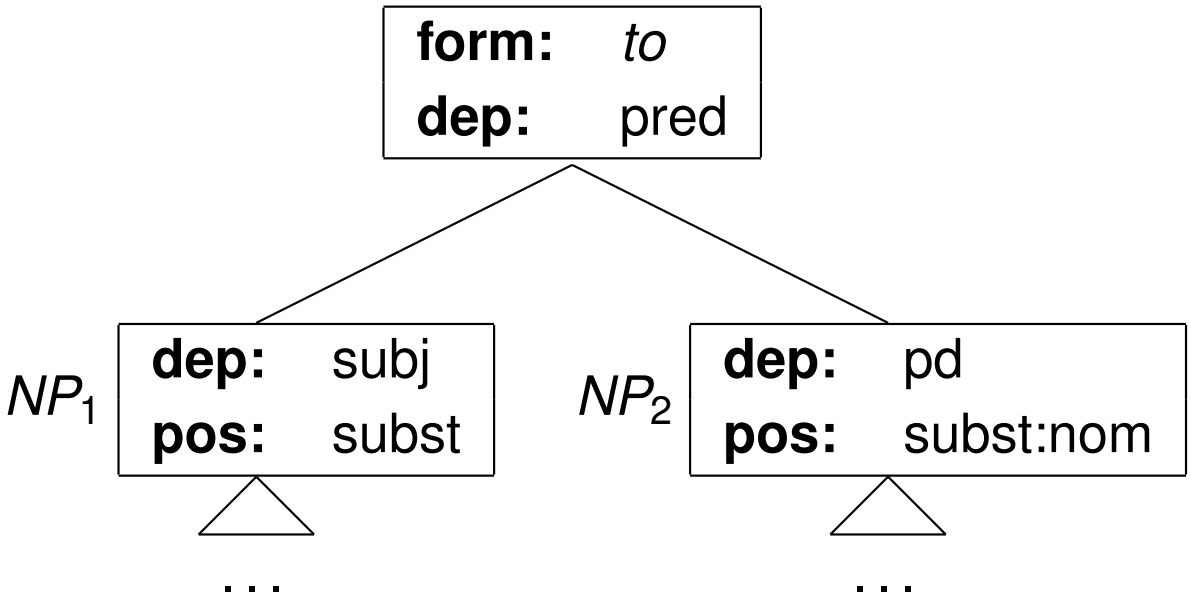
1	Golden	golden	subst	subst	sg:nom:m3	3	subj
2	retriever	retriever	subst	subst	sg:nom:m2	1	app
3	jest	być	fin	fin	sg:ter:imperf	0	pred
4	psem	pies	subst	subst	sg:inst:m2	3	pd
5	myśliwskim	myśliwski	adj	adj	sg:loc:m3:pos	4	adjunct
6	.	.	interp	interp	_	3	punct

Rysunek: Wynik parsowania zależnościowego dla zdania “Golden retriever jest psem myśliwskim", format CoNLL.

Wstępna selekcja drzew zależnościowych do ekstrakcji

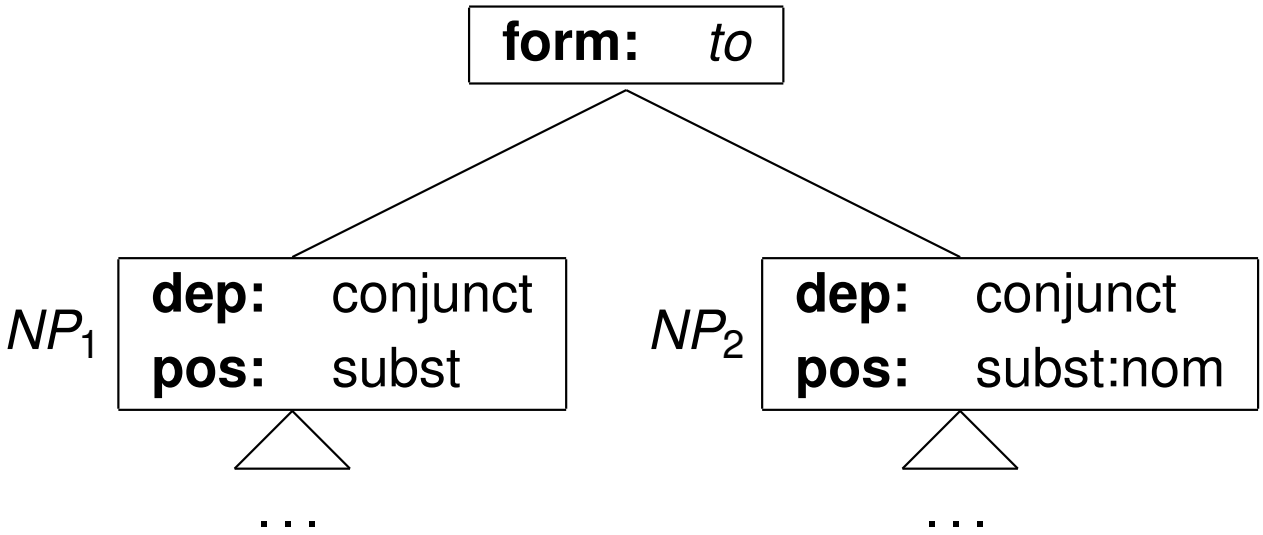
Wstępna selekcja polega na wyborze tylko tych zdań, których drzewo zależnościowe pasuje do jednego ze zdefiniowanych wzorców. Fraza NP_1 konstruowana jest z lewego poddrzewa, a NP_2 z prawego poddrzewa.

Nasze podejście – wstępna selekcja drzew zależnościowych do ekstrakcji - konstrukcja mianownikowa



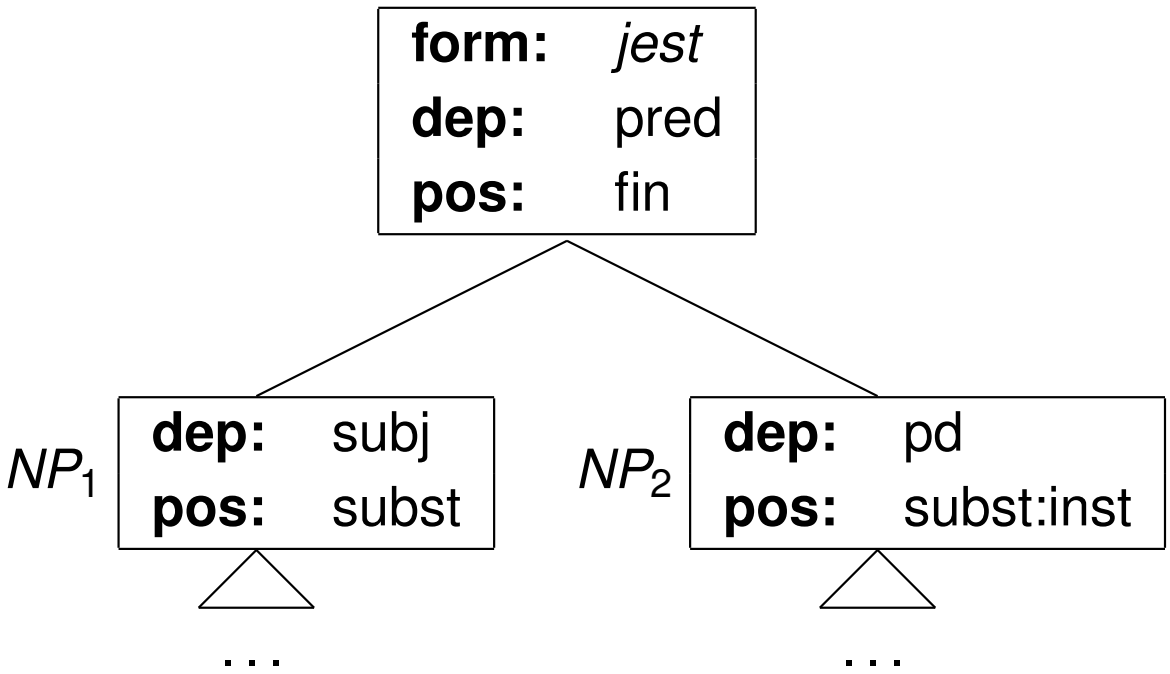
Rysunek: Wzorzec drzewa zależnościowego dla konstrukcji mianownikowej

Nasze podejście – wstępna selekcja drzew zależnościowych do ekstrakcji - konstrukcja mianownikowa (Malt parser)



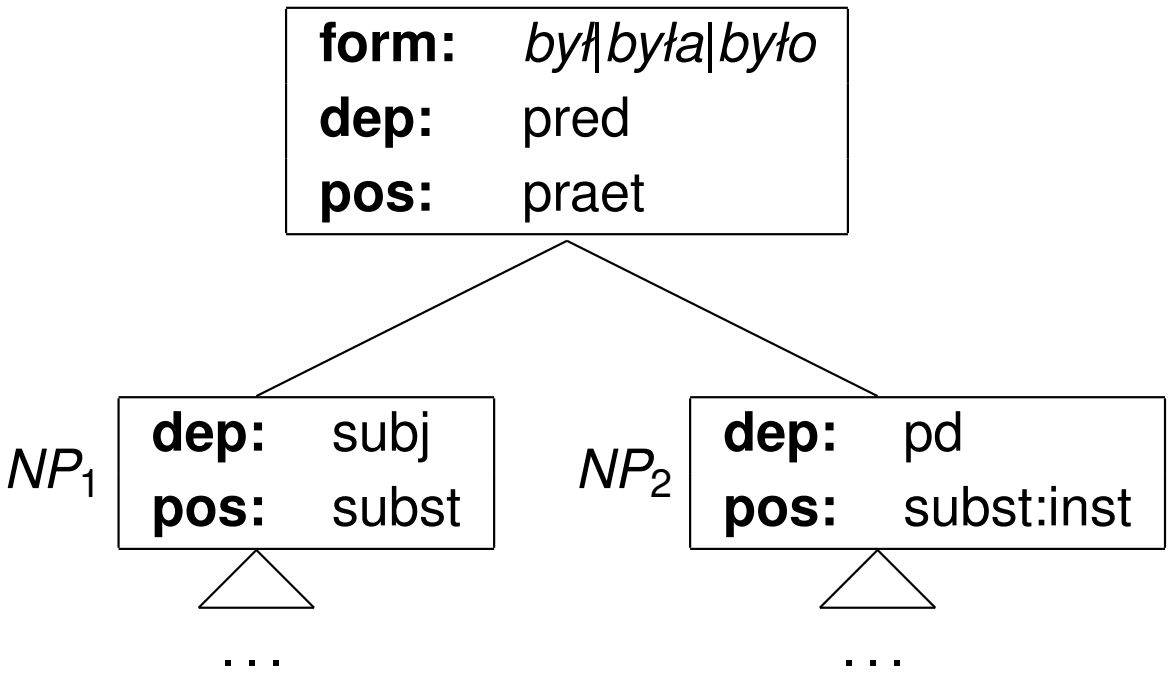
Rysunek: Wzorzec drzewa zależnościowego dla konstrukcji mianownikowej - błędne przypisanie relacji przez Malt parser

Nasze podejście – wstępna selekcja drzew zależnościowych do ekstrakcji



Rysunek: Wzorzec drzewa zależnościowego dla konstrukcji narzędnikowej

Nasze podejście – wstępna selekcja drzew zależnościowych do ekstrakcji - konstrukcja narzędnikowa



Rysunek: Wzorzec drzewa zależnościowego dla konstrukcji (3)

Filtrowanie słownikowe głównego rzeczownika frazy NP_2

Odfiltrowanie zdań, w których głowa NP_2 jest terminem ogólnym, np.:

*Polska jest **przykładem** kraju nadbałtyckiego.*

*Miasto jest **tematem**, na który można długo dyskutować.*

Zastosowanie reguł konstruujących frazy rzeczownikowe

NP_1**[**NP_1^H**-Golden retriever] jest** NP_2**[**NP_2^H**-psem myśliwskim].**

Konstruujemy NP_1 i NP_2 z lematów słów:

- NP_1 - stwórz listę lematów począwszy od głowy NP_1^H aż do relacji (to/jest),
- NP_2 - stwórz listę lematów począwszy od głowy NP_2^H a skończywszy na znaku interpunkcji lub słowie którego część mowy nie należy do zbioru $\{adj, subst, ger\}$ lub część zdania do zbioru $\{adjunct, app, conjunct, obj\}$.,

Końcowe filtrowanie wyników

- Usuwanie relacji IS-A, w których NP_1 zawierających zaimki i odniesienia tekstowe typu *powyższy*/*wspomniany* (odniesienia do encji wymienionych wcześniej w tekście),
- zatrzymywanie tylko relacji z liczbą wystąpień powyżej zadanego progu t lub występujących zarówno w konstrukcjach narzędnikowych jak i mianownikowych.

Wzmacnianie pseudo-podklasami

Liczba ekstrahowanych relacji znacząco spada przy rosnącym progu t. Równoważenie tej odbywa się przy pomocy metody opartej na następującej intuicji:

Intuicja

Niech *I* `IS-A` *C* i *I* `IS-A` *C'* będą wyekstrahowanymi relacjami oraz *C* będzie podciągiem *C'*.

Jest prawdopodobne, że, *C'* jest bardziej szczegółowym typem *I* niż *C*, a więc *C'* jest pseudo-podklasą *C*. Jeżeli tak, to możemy zwiększyć naszą pewność, że relacja *I* `IS-A` *C* jest poprawnie wyekstrahowana.

Wzmacnianie pseudo-podklasami – przykład 1

Ze zdań:

> *Kraków to najchętniej odwiedzane miasto przez turystów w Polsce. Kraków – dawna stolica Polaków jest miastem magicznym.*

dostajemy relacje *Kraków* `IS-A` *miasto* i *Kraków* `IS-A` *miasto magiczne*, z których druga wspiera poprawność pierwszej.

Wzmacnianie pseudo-podklasami

W ogólności, w celu wyliczenia wsparcia pseudo-podklasami dla każdej relacji $R = I \texttt{ IS-A } C$ generujemy listę L:

- list prefiksowych tokenów z C,
- list sufiksowych tokenów z C, które nie rozpoczynają się od przymiotnika.

W zadaniu Map/Reduce dla każdego R emitujemy

- $(I, C) \rightarrow$ liczba wystąpień R,
- $\forall_{c \in L}((I, c) \rightarrow$ liczba wystąpień $R)$

i agregujemy informację o liczbie wystąpień podklas.

Wzmacnianie pseudo-podklasami – przykład 2

```
mukowiscydoza IS-A
 1. choroba
 2.   choroba dziedziczna
 3.   choroba genetyczna
 4.     choroba genetyczna ludzi rasy białej
 5.     choroba genetyczna ogólnoustrojowa
 6.     choroba genetyczna ras biała
 7.     choroba genetyczna układu pokarmowego
 8.   choroba monogenowa
 9.   choroba nieuleczalna
10.   choroba przewlekła
11.   choroba wielonarządowa
12.   choroba wieloukładowa
13.   wieloukładowa choroba
14.     wieloukładowa choroba monogenowa
15. przyczyna wykonywania
16.   przyczyna wykonywania przeszczepu płuca
17. schorzenie
18.   schorzenie genetyczne
```

- Wiersz 13 jest przykładem wzmacniania listą sufiksową.
- Wierze 2–12 wspierają relację *mukowiscydoza* `IS-A` *choroba*, dodatkowo wierze 4–7 wspierają relację *mukowiscydoza* `IS-A` *choroba genetyczny*, itd.

Środowisko testowe

Eksperymenty przeprowadzono na korpusie 500 milionów dokumentów ściągniętych z polskiej części sieci Internet.

Dane były przetwarzane przy pomocy technologii Apache Hadoop Map/Reduce oraz Apache Hive.

Do obliczeń wykorzystano klaster 70 maszyn dysponujący 980 rdzeniami CPU i łączną pamięcią operacyjną 4.375TB.

Eksperymenty – scenariusz

Ewaluacja metody została przeprowadzona w czterech eksperymentach z rosnącą wartością progu odcięcia t.

Precyzja była wyliczana w oparciu o ręczą ocenę około 110 relacji wybranych losowo z wyników każdego z eksperymentów.

Eksperymenty – wyniki

t	**nom/pcb**	**abl/pcb**	**nom i abl/pcb**	**łącznie**	**zysk z pcb**
1	1348993/0	1999418/0	32425/0	3380836	0%
2	95855/16879	198153/44875	32425/14886	403073	23%
3	33085/8361	69097/26083	32425/14886	183937	36%
4	17423/5247	35955/17420	32425/14886	123356	43%

Tabela: Liczba ekstrahowanych relacji dla różnych wartości progu *t*.

t	1	2	3	4
precyzja z pcb	0.61	0.72	0.79	0.81
precyzja bez pcb	0.61	0.71	0.87	0.87

Tabela: Szacowana precyzja ekstrakcji dla różnych wartości progu *t*.

pcb - pseudo class boosting

Uwagi na temat wyników

- Dla progu $t = 2$ wzmacnianie pseudo-podklasami daje 23% zysk w liczbie ekstrakcji przy braku widocznego spadku precyzji.
- Szacowana precyzja metody wzrasta dla rosnącego t, do poziomu około 80%.
- Dla $t = 3$ i $t = 4$ wzrost liczby ekstrakcji dzięki pseudo-podklasom jest okupiony dużym spadkiem precyzji.

Uwagi na temat wyników – analiza błędów

Analiza przypadków błędnie zakwalifikowanych jako relacje `IS-A` wykazała trzy klasy błędów:

- niejawna koreferencja,
- błędne wyznaczenie granic fraz rzeczownikowych,
- stale rosnący słownik służący do wstępnego filtrowania.

Podsumowanie

- Opracowana została nowa metoda ekstrakcji relacji `IS-A` ze wzorców dla języka polskiego, niezależna od popularnych reguł Hearsta.
- Opracowana została prosta lecz nowatorska metoda zwiększania liczby ekstrahowanych relacji (niezależna od stosowanego mechanizmu ekstrakcji) nazwana *wzmacnianiem pseudo-podklasami*.
- Jak wykazały eksperymenty, metoda osiąga zadowalającą precyzję (chociaż jest pole do poprawy) i umożliwia ekstrahowanie dużej liczby relacji taksonomicznych.