Automatyczna anotacja banku drzew zależnościowych

Alina Wróblewska

Instytut Podstaw Informatyki Polskiej Akademii Nauk

Seminarium ZIL
Warszawa, 17 marca 2014

INNOVATIVE ECONOMY
NATIONAL COHESION STRATEGY

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND

Outline

Analiza zależnościowa

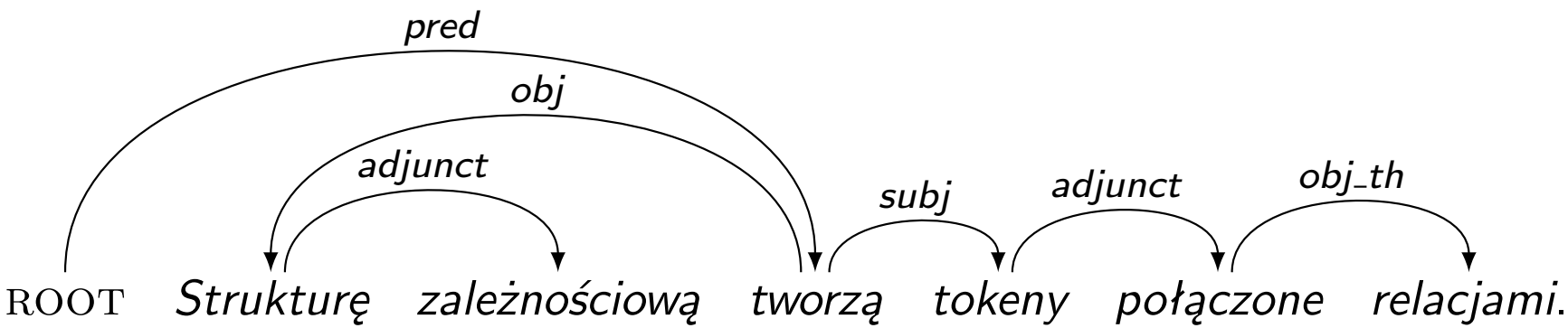
- Parsowanie zależnościowe to analiza składniowo-semantyczna zdania.
- Parsowanie wydobywa strukturę predykatywno-argumentową zdania.
- Analiza zależnościowa jest wykorzystywana m.in. w:
  - systemach dialogowych,
  - systemach ekstrakcji wiedzy,
  - maszynowym tłumaczeniu.
- Parsery zależnościowe:
  1. mogą bazować na ręcznie stworzonej gramatyce,
  2. mogą wykorzystywać metody statystyczne do wytrenowania modelu parsowania.

Statystyczne parsery zależnościowe

- Na podstawie danych treningowych parsery uczą się analizować zdania i generować dla nich odpowiednie struktury zależnościowe.
- Parsery trenowane za pomocą metod z nadzorem na banku ręcznie zaanotowanych struktur zależnościowych:
  - osiągają najlepsze wyniki jak dotychczas,
  - wymagają dużej liczby poprawnie zaanotowanych struktur zależnościowych.
- Parsery trenowane za pomocą metod bez nadzoru:
  - mało efektywne,
  - bardzo duża złożoność obliczeniowa przy trenowaniu i parsowaniu,
  - generują głównie struktury zależnościowe bez etykiet na krawędziach przez co są niewystarczające dla wielu zadań NLP.
- Alternatywą jest trenowanie parserów z nadzorem na automatycznie zaanotowaych strukturach zależnościowych.

Struktura zależnościowa

- Struktura zależnościowa to drzewo rozpinające:
  - każdy wierzchołek ma jedną krawędź wejściową,
  - korzeń nie ma krawędzi wejściowych i jedną krawędź wyjściową,
  - nie ma cykli,
- wierzchołki w drzewie odpowiadają tokenom w zdaniu,
- krawędzie skierowane reprezentują relacje składniowe albo semantyczne:
  - token, z którego wychodzi krawędź, jest nadrzędnikiem tokena, do którego wchodzi dana krawędź,
  - etykiety krawędzi odpowiadają funkcji gramatycznej podrzędnika,
- drzewo koduje strukturę predykatywno-argumentową.



Zalety i niedoskonałości parsowania zależnościowego

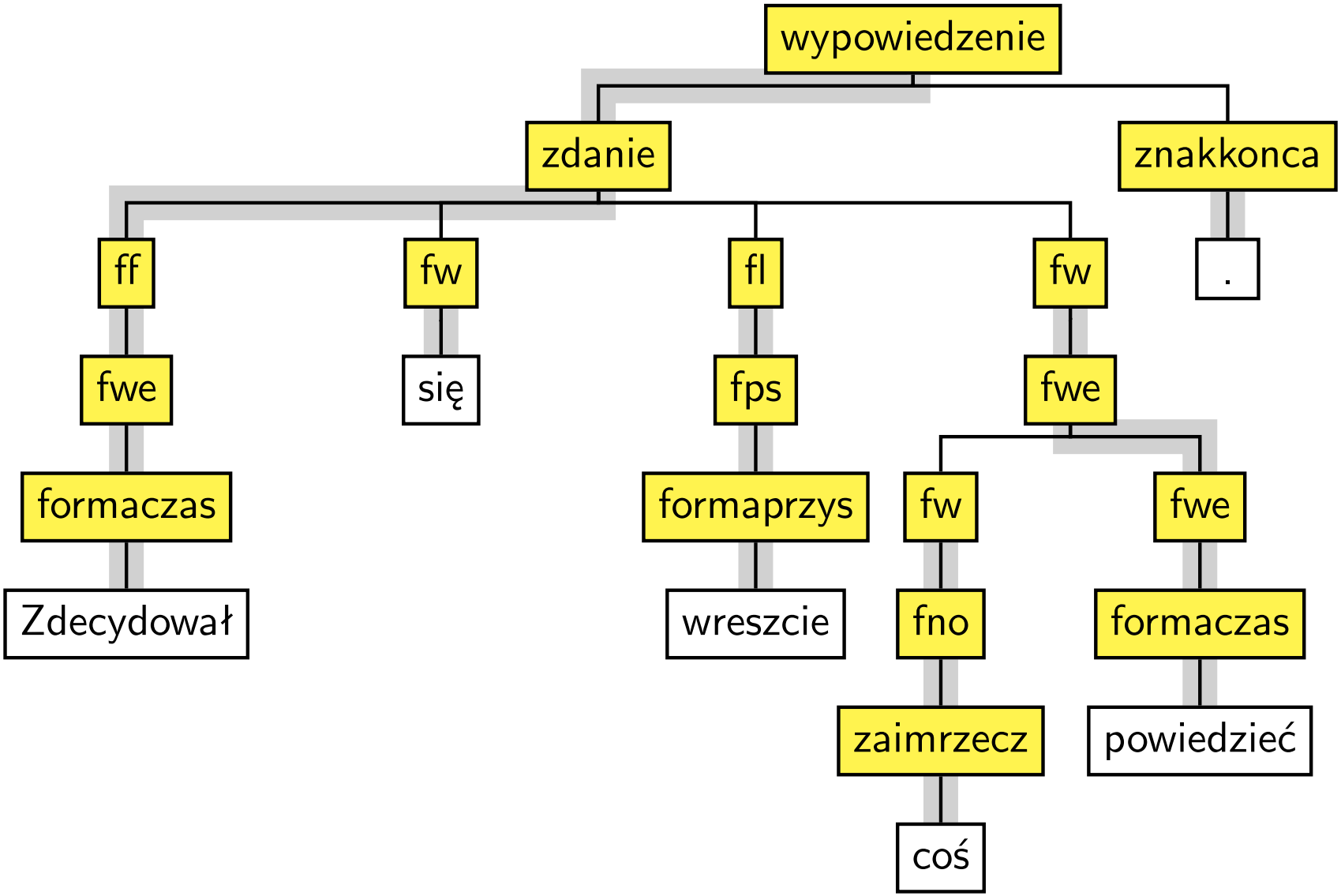
- Zalety:
  - szybkość – zdania mogą być parsowane w czasie liniowym,
  - dyzambiguacja – parsowanie zwraca tylko jedną strukturę dla zdania,
  - łatwość w reprezentowaniu niektórych fenomenów lingwistycznych np. nieciągłości.
- Niedoskonałości:
  - brak możliwości zastąpienia zwracanej struktury, która zawiera błędne relacje, przez inną,
  - ograniczenia i uproszczenia w reprezentowaniu niektórych fenomenów np. podmiot logiczny, podnoszenie podmiotu,
  - duże uzależnienie od jakości tagowania.

Outline

Konwersja drzew składnikowych

- Idea – konwersja drzew składnikowych do postaci drzew zależnościowych,
- dla języka polskiego istnieje bank struktur składnikowych – *Składnica frazowa*,
- bierzemy pod uwagę wszystkie fenomeny składniowe zakodowane w polskich drzewach składnikowych i anotujemy je odpowiednimi zależnościami,
- relacje zależnościowe można stosunkowo łatwo wywieść ze struktur składnikowych z wyróżnionymi elementami głównymi,
- krawędziom w przekonwertowanych drzewach przypisujemy etykiety,
- ostatecznym wynikiem procesu konwersji jest bank 8227 drzew zależnościowych.

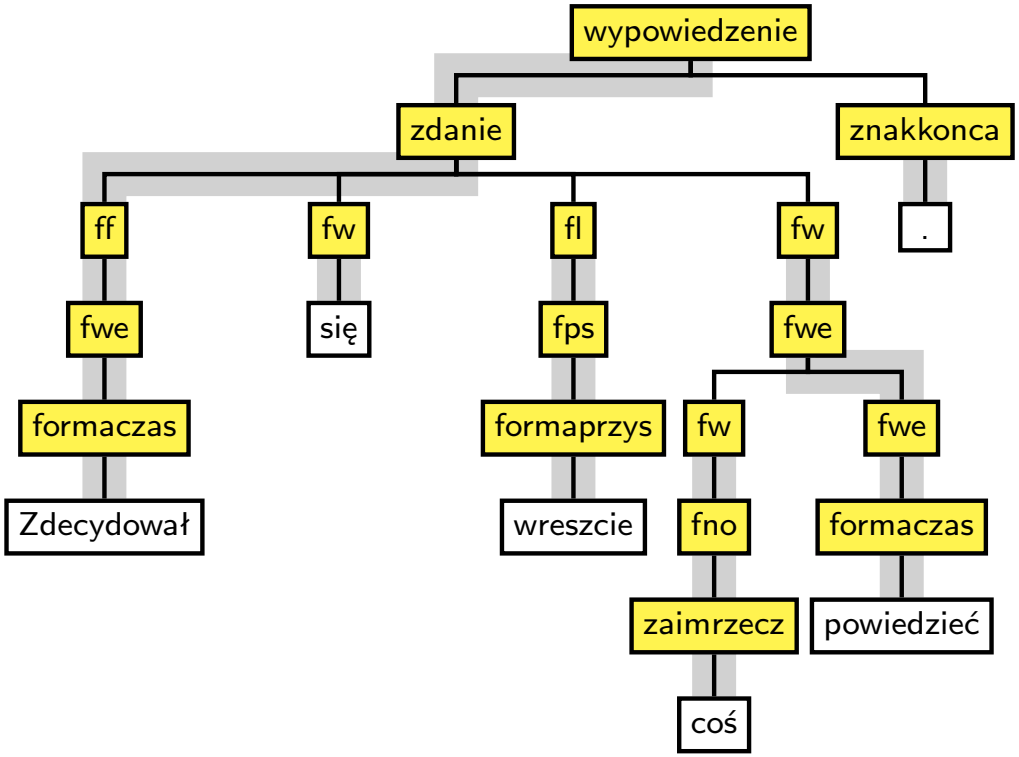
Przykład drzewa składnikowego

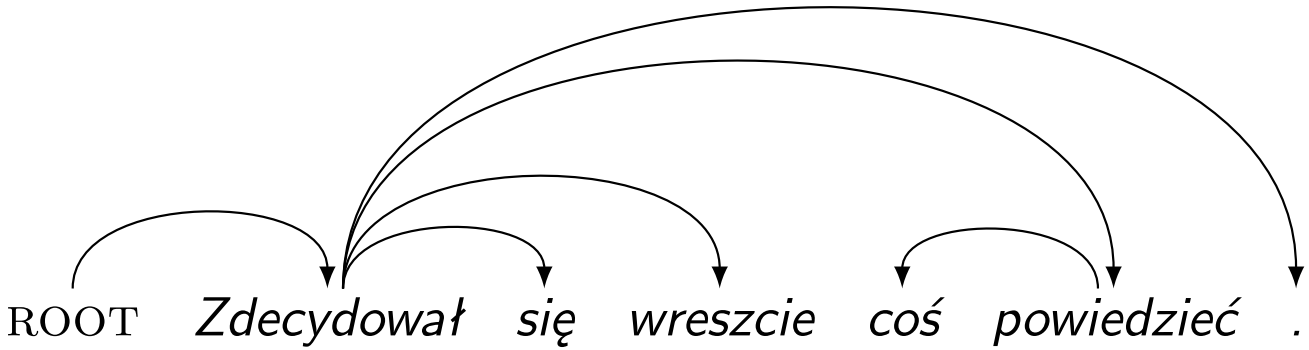


Procedura konwersji

- Węzły leksykalne w drzewach zależnościowych odpowiadają tokenom zakodowanym w składnikowych węzłach terminalnych,
- informacje morfoskładniowe (forma podstawowa, część mowy, cechy morfologiczne) z węzłów terminalnych są rzutowane jako atrybuty poszczególnych tokenów,
- identyfikacja nadrzędnika każdego tokena w drzewie składnikowym:
  1. przechodząc poszczególne krawędzie w kierunku wznoszącym, szukamy pierwszego węzła, który nie jest oznaczony jako głowa frazy,
  2. rozpoczynając od ojca znalezionego węzła i przechodząc krawędzie pomiędzy ojcem i jego synem oznaczonym jako głowa frazy, znajdujemy token, który jest nadrzędnikiem tokena wyjściowego,
- znalezione relacje pomiędzy tokenami tworzą strukturę zależnościową dla danego zdania.

Przykład konwersji

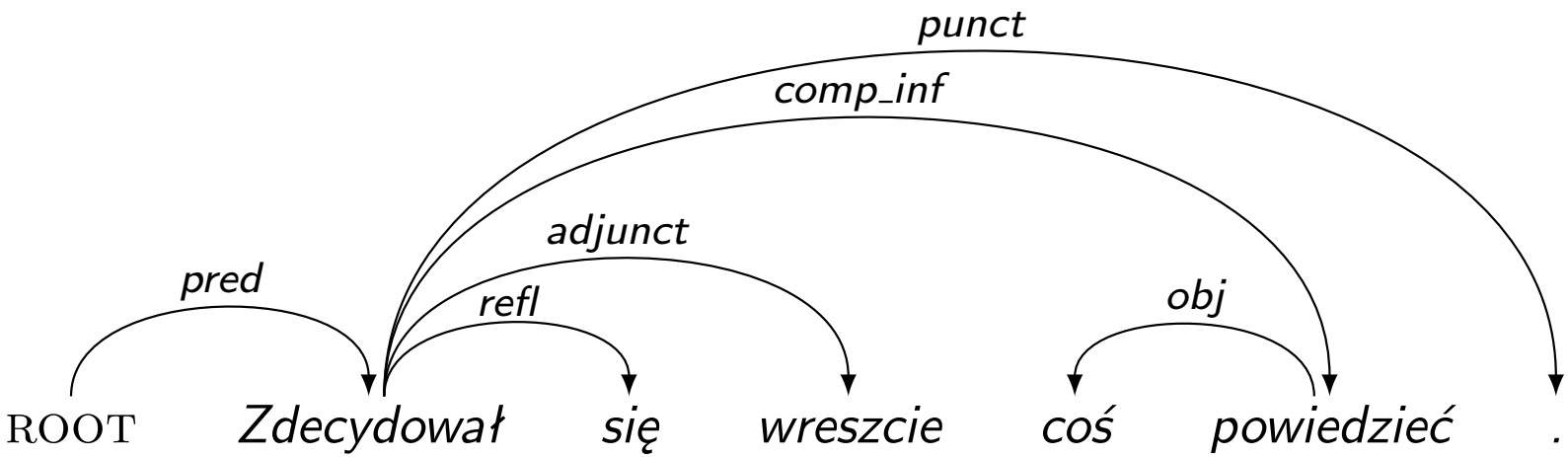




Etykietowanie relacji zależnościowych

- Współczesne systemy parsowania zależnościowego są dostosowane do uczenia modeli na drzewach, których krawędzie mają przypisane etykiety,
- etykietowanie relacji następuje równocześnie z identyfikacją relacji w drzewach składnikowych,
- relacje tworzące ostateczne drzewo zależnościowe mają przypisane etykiety,
- funkcje gramatyczne zakodowane w drzewach składnikowych są bezpośrednio przenoszone do drzew zależnościowych,
- proces etykietowania pozostałych relacji bazuje na regułach,
- reguły wykorzystują informacje zakodowane w węzłach drzew składnikowych odwiedzanych podczas szukania nadrzędnika:
  - informacja czy fraza jest wymagana przez czasownik,
  - informacje o typach fraz (np. werbalna, nominalna),
  - cechy morfoskładniowe (np. część mowy, przypadek, liczba),
  - typy reguł konstrukcyjnych wykorzystanych podczas parsowania zdania (np. *noap* – apozycja).

Przykład przekonwertowanego drzewa zależności



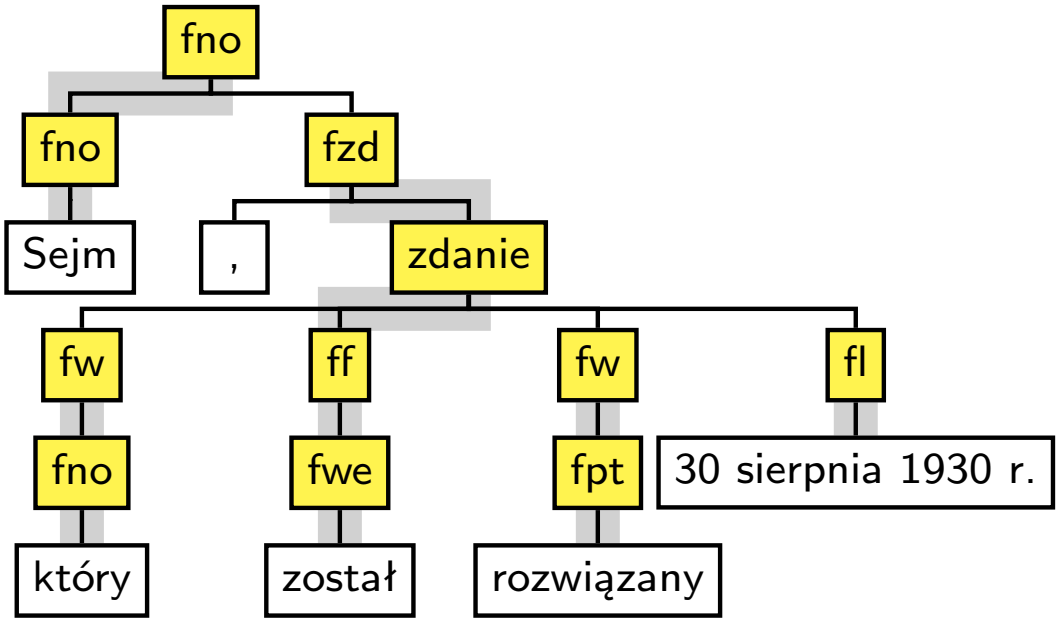
Wybór składnika głównego frazy

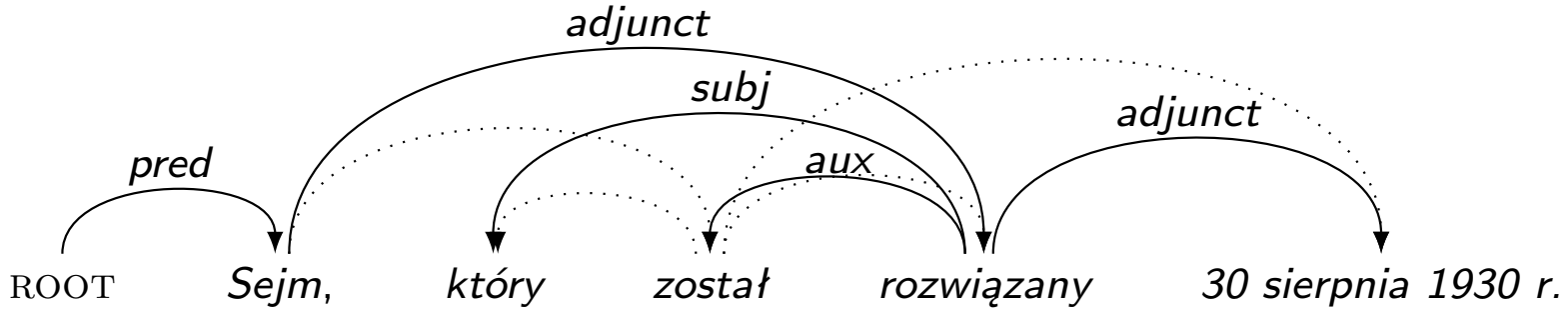
- W większości przypadków frazy drzew składnicowych mają wyróżniony składnik główny,
- we frazach, w których kilka składników zostało wyróżnionych jako elementy główne, musi zostać wybrany jeden składnik,
- reguły wyboru składnika głównego zostały zdefiniowane dla następujących przypadków:
  - warunkowe formy czasownikowe (np. chciał $\xrightarrow{cond}$ by),
  - forma czasownikowa z aglutynatem (np. chcieli $\xrightarrow{aglt}$ śmy, chcieli $\xrightarrow{cond}$ by $\xrightarrow{aglt}$ śmy),
  - analityczne formy czasowników (np. będziemy $\xleftarrow{aux}$ rozmawiać),
  - złożone spójniki współrzędne (np. ani $\xleftarrow{precoord}$ ani) i podrzędne (np. nie tylko $\xleftarrow{adjunct}$ ale także),
  - skróty, wieloczłonowe wyrażenia, serie znaków interpunkcyjnych (np. . $\xrightarrow{punct}$. $\xrightarrow{punct}$.).

Modyfikacje struktur zależnościowych

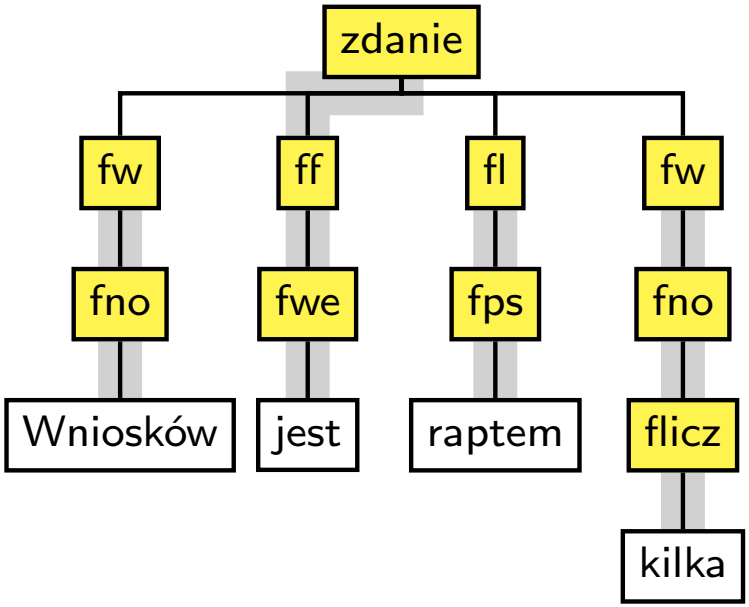
- Niektóre relacje zostały zmodyfikowane w celu dostosowania struktur do schematu anotacji,
- modyfikacje dotyczyły:
  - konstrukcji strony biernej,
  - konstrukcji zawierających frazy nieciągłe,
  - niektórych zdań podrzędnych,
  - fraz z korelatem,
  - zdań ze spójnikami inkorporacyjnymi.

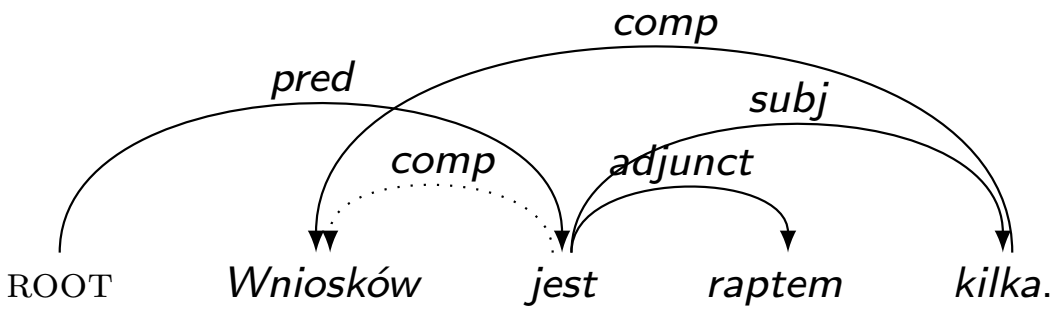
Modyfikacje konstrukcji strony biernej



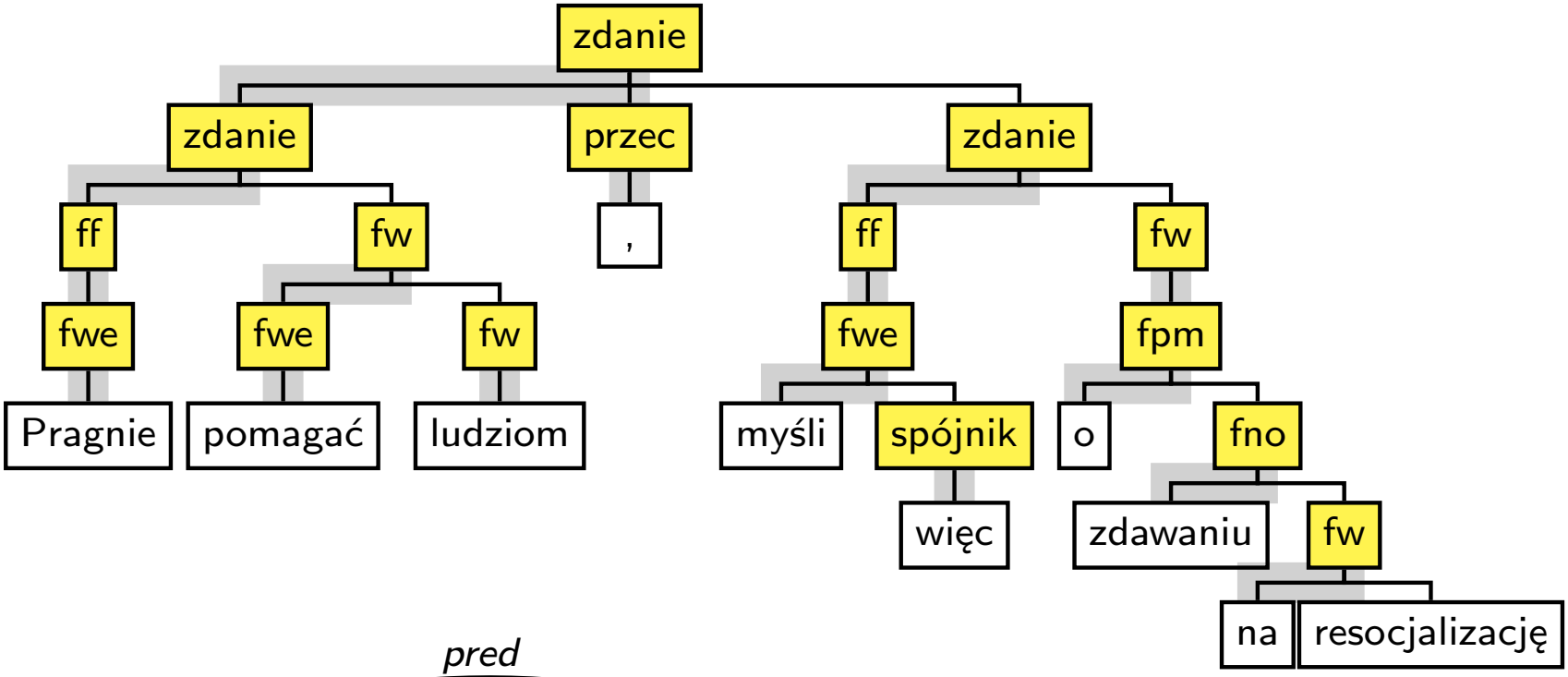


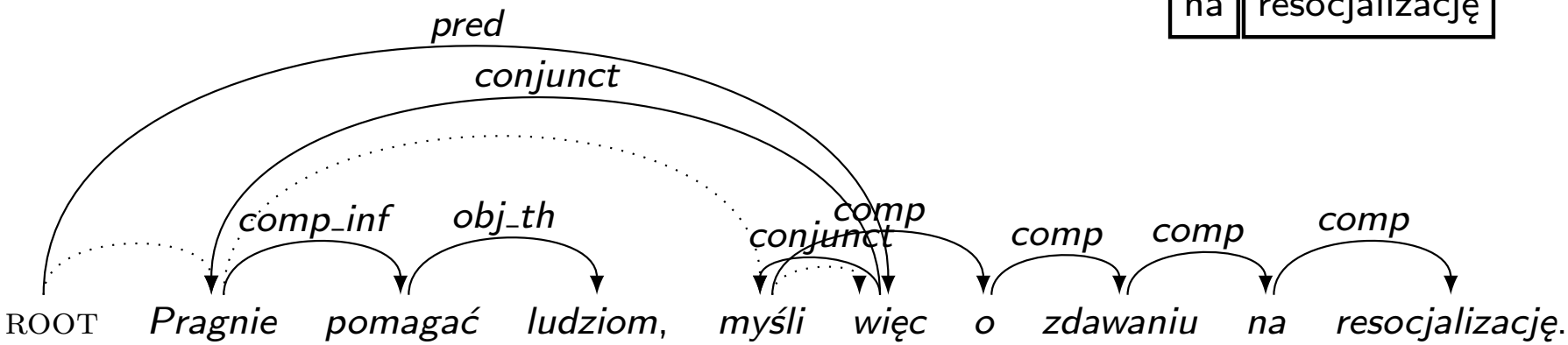
Modyfikacje konstrukcji z frazami nieciągłymi



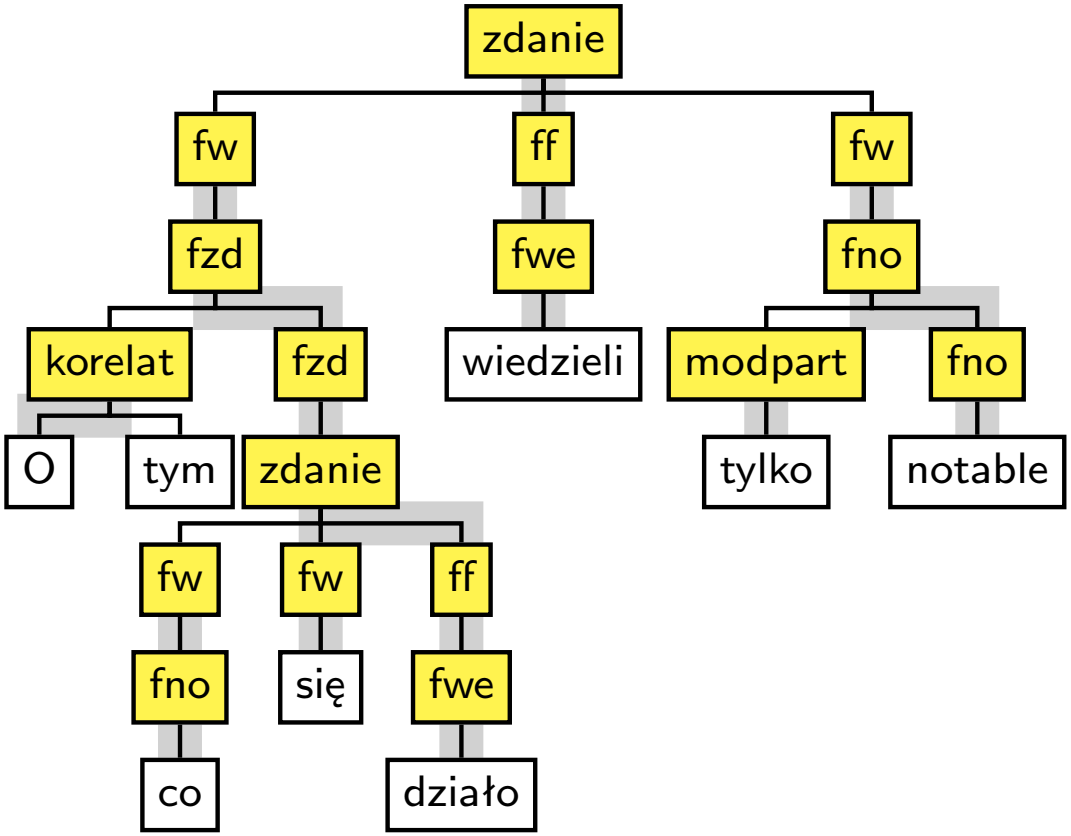


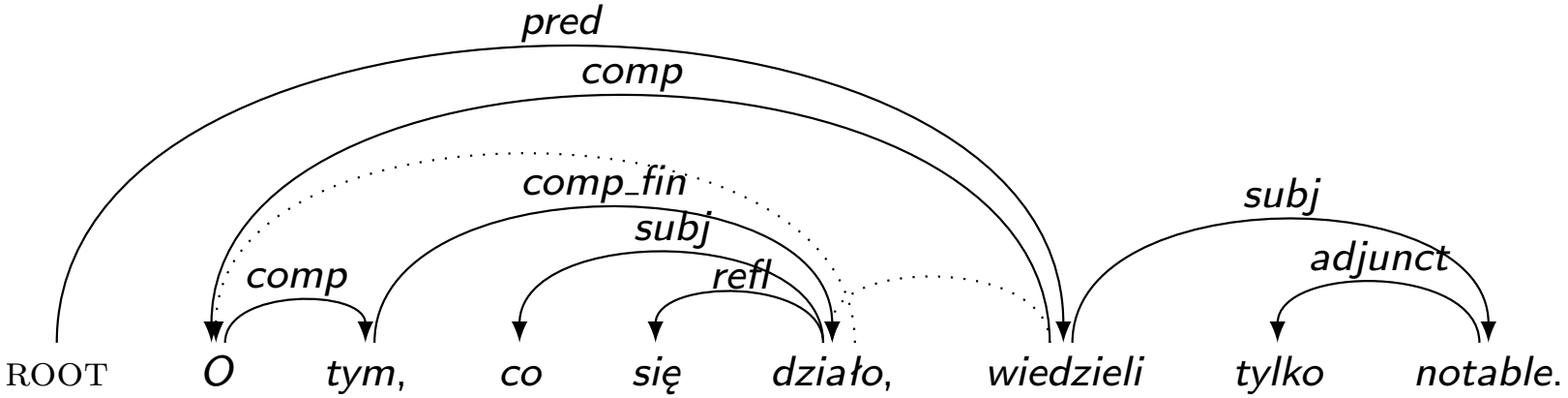
Modyfikacje fraz ze spójnikami inkorporacyjnymi





Modyfikacje fraz z korelatem





Ewaluacja

- W celu oceny jakości pozyskanych drzew zależnościowych została wykorzystana zewnętrzna metoda ewaluacji (ang. 'extrinsic evaluation'),
- ewaluacja zewnętrzna polega na wytrenowaniu parsera na drzewach, a następnie na ocenie wpływu danych treningowych na jakość parsowania,
- wykorzystane systemy parsujące: *MaltParser* (Nivre et al., 2006) i parser *Mate* (Bohnet, 2010),
- zbiory testowe:
  1. 822 drzew zależnościowych ze *Składnicy* (Manual Test),
  2. 822 drzew ze *Składnicy* z automatycznie przypisanymi tagami (Automatic Test),
  3. 100 drzew z czasopism i NKJP (średnio 16,6 tokenów w zdaniu, 2,8% krzyżujących się krawędzi, Additional Test).

Wyniki (1)

Model	Manual Test		Automatic Test		Additional Test	
	UAS	LAS	UAS	LAS	UAS	LAS
Malt default	88.2	80.4	84.6	76.1	72.7	63.3
Malt optimised	90.5	85.4	85.3	78.4	73.3	66.1
Malt automatic	88.4	82.9	87.8	81.6	75.8	68.0
Mate default	**92.7**	**87.2**	88.4	81.0	76.0	69.5
Mate automatic	91.2	85.6	**90.8**	**84.7**	**76.6**	**70.1**

1. Manual Test – zbiór 822 drzew ze *Składnicy* zależnościowej
2. Automatic Test – zbiór 822 drzew ze *Składnicy* z automatycznie zaanotowanymi tokenami
3. Additional Test – zbiór 100 ręcznie zaanotowanych zdań

default – parser trenowany z wykorzystaniem standardowych parametrów
optimised – parser trenowany ze zoptymalizowanym modelem cech
automatic – parser trenowany na drzewach ze *Składnicy* z automatycznymi tokenami

Wyniki (2)

- Praser *Mate* trenowany na drzewach zależnościowych ze *Składnicy* przypisuje:
  1. poprawny nadrzędnik do **92,7**% tokenów, a do **87,2**% tokenów poprawny nadrzędnik oraz poprawną funkcję gramatyczną,
  2. poprawny nadrzędnik do 88,4% tokenów, a do 81% tokenów poprawny nadrzędnik oraz poprawną etykietę,
  3. poprawny nadrzędnik do 76% tokenów, a do 69,5% tokenów poprawny nadrzędnik i poprawną etykietę.
- Parser *Mate* trenowany na drzewach ze *Składnicy* z automatycznie zaanotowanymi tokenami przypisuje:
  1. poprawny nadrzędnik do 91.2% tokenów, a do 85,6% tokenów poprawny nadrzędnik i funkcję gramatyczną,
  2. poprawny nadrzędnik do **90,8**% tokenów, a do **84,7**% tokenów przypisuje poprawny nadrzędnik i funkcję,
  3. poprawny nadrzędnik do **76,6**% tokenów, a do **70,1**% tokenów poprawny nadrzędnik oraz etykietę relacji.
- problematyczne relacje: *coord*, *coord_punct*, *ne*, *pd*, *mwe*, *app*.

Outline

Rzutowanie informacji lingwistycznych

- Alternatywa dla ręcznego anototwania tekstów,
- automatyczny sposób pozyskiwania zaanotowanych danych w językach ubogich w narzędzia i zasoby lingwistyczne,
- idea rzutowania – odwzorowanie anotacji lingwistycznych w zdaniach z części korpusu równoległego w jednym języku na odpowiednie zdania z części korpusu w drugim języku,
- informacje lingwistyczne są rzutowane z wykorzystaniem automatycznie wygenerowanych przyporządkowań słownych (ang. 'word alignment').

Metoda rzutowania ważonego

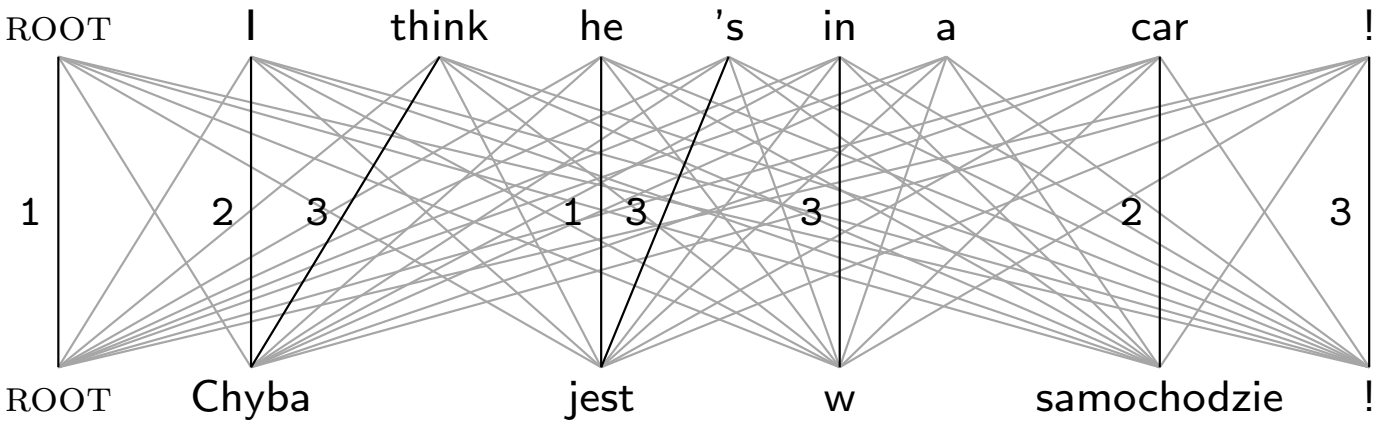
- Procedura rzutowania ważonego obejmuje dwa główne kroki:
  1. **rzutowanie ważone** angielskich relacji zależnościowych na zdanie polskie,
  2. **indukcja ważona** drzew zależnościowych na podstawie zbioru rzutowanych krawędzi.
- Nowatorstwo metody polega na włączeniu czynnika ważenia do procesów rzutowania relacji oraz indukcji struktur zależnościowych.

Rzutowania ważone – procedura

1. Angielskie relacje zależnościowe są rzutowane na odpowiednie zdania polskie poprzez rozbudowany zbiór przyporządkowań (graf dwudzielny przyporządkowań słownych),
2. rzutowane relacje tworzą grafy skierowane dla polskich zdań,
3. rzutowanym krawędziom zostają przypisane intuicyjne wagi.

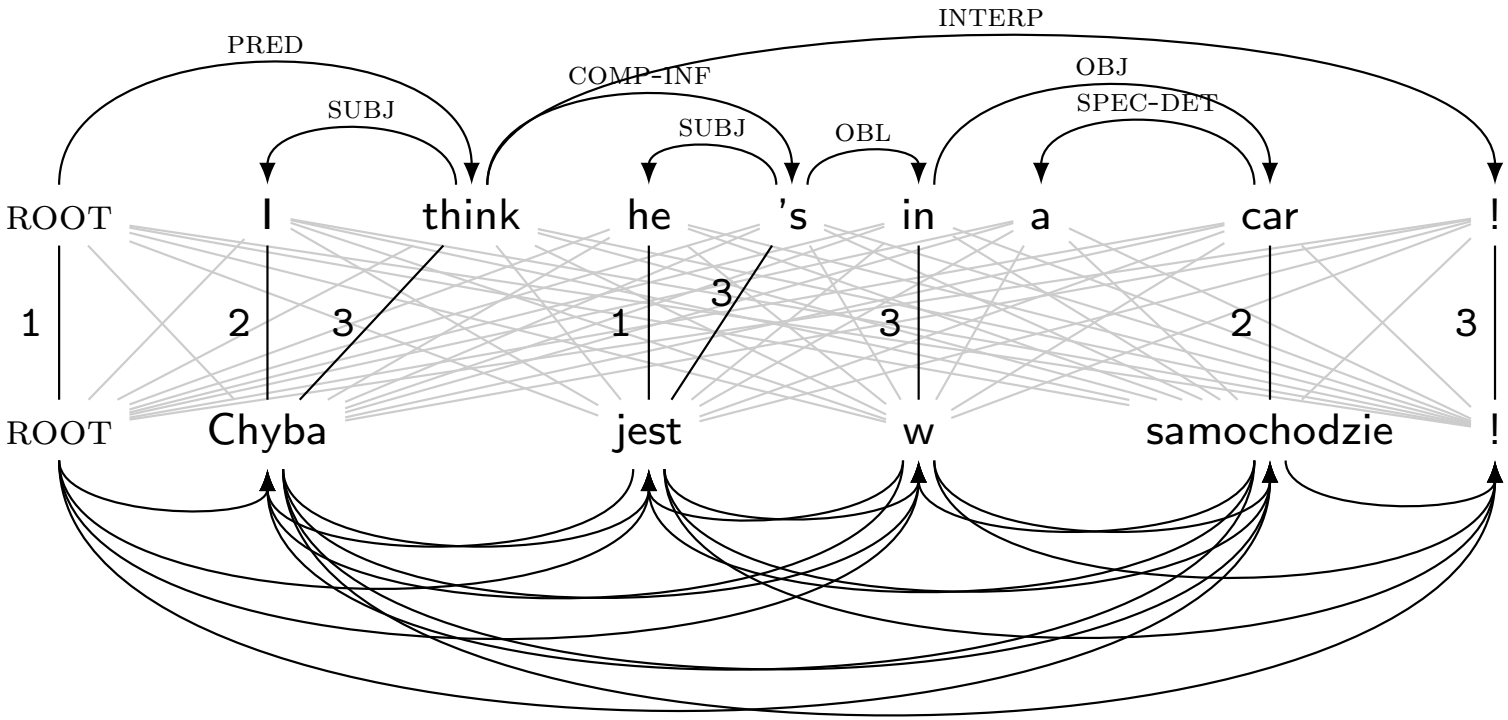
Graf dwudzielny przyporządkowań słownych

- Pełny graf dwudzielny pomiędzy tokenami polskimi i angielskimi,
- krawędzie grafu mają przypisane wagi (0–3) odpowiadające liczbie wystąpień danej krawędzi w trzech zbiorach automatycznych przyporządkowań słownych.



Rzutowanie relacji

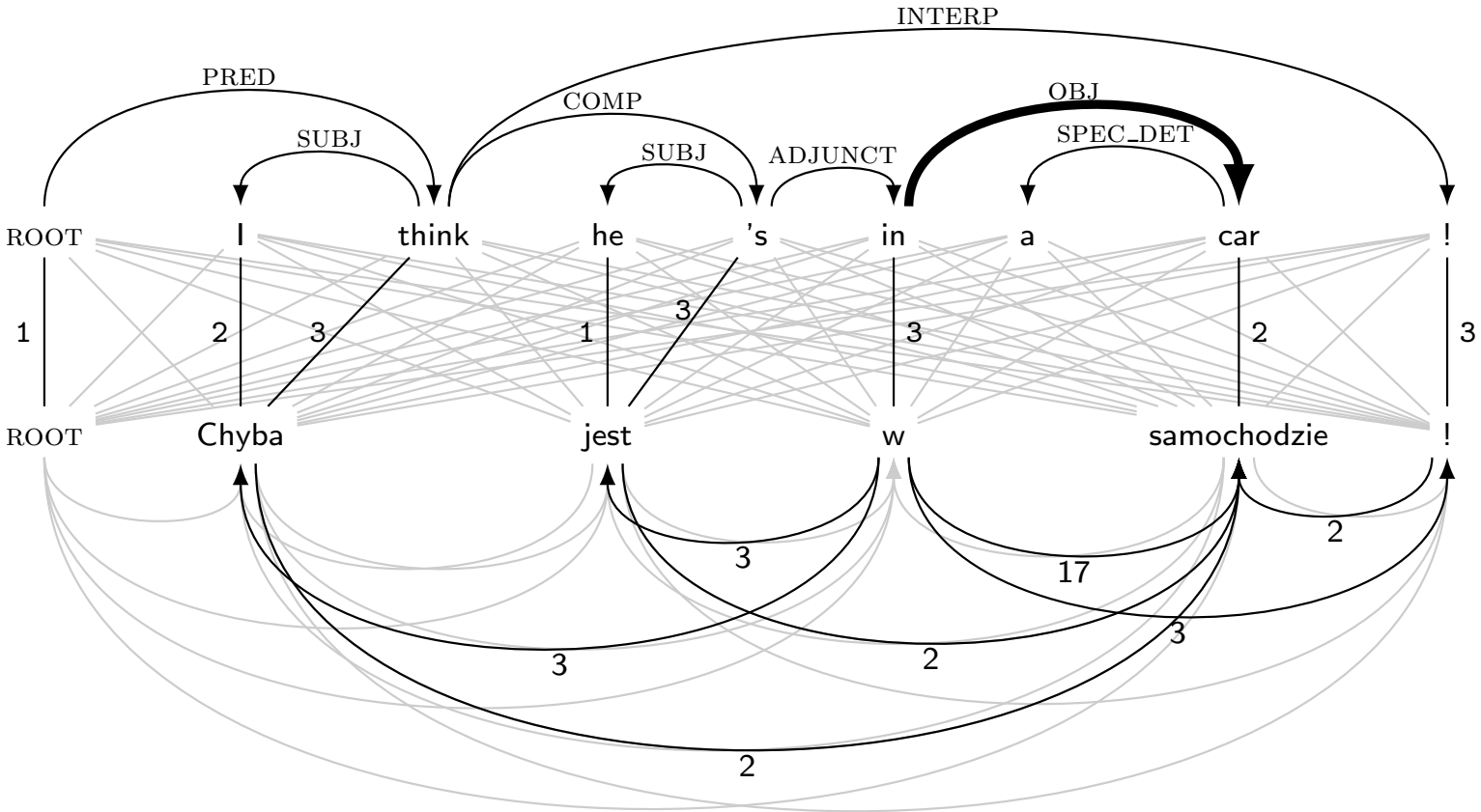
- Rzutowanie wszystkich relacji angielskich poprzez krawędzie grafu dwudzielnego, z których przynajmniej jedna nie jest zerowa,
- rzutowane relacje zależnościowe tworzą multi-grafy skierowane.



Intuicyjne przypisywanie wag

Krawędziom w multi-grafach zostają przypisane wagi s na podstawie wag krawędzi dwudzielnych (w_d, w_g) oraz częstości f rzutowania danej relacji poprzez tą samą parę krawędzi:

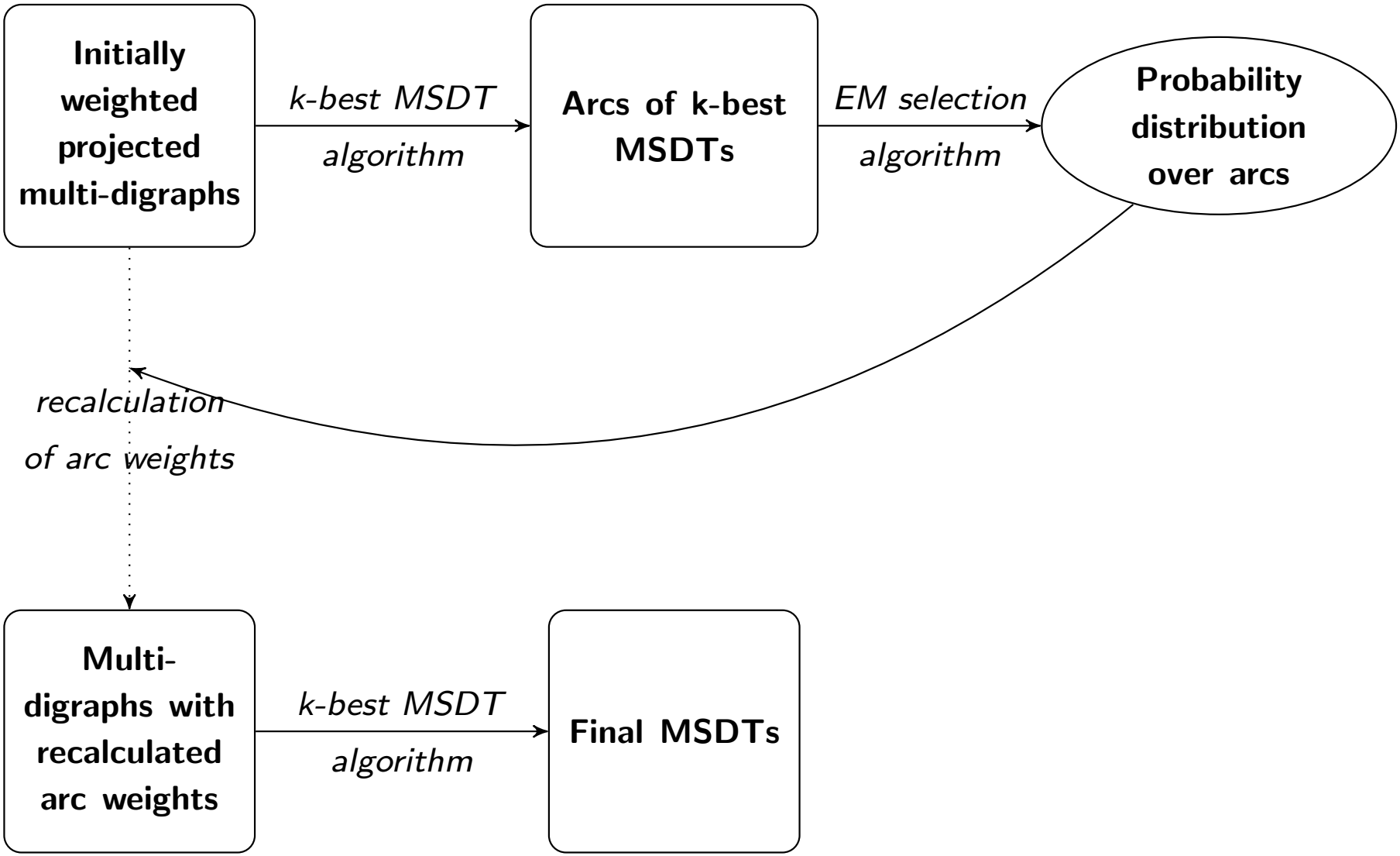
$$s(v_i, v_j, l) = w_d + w_g + 2w_d w_g f \tag{1}$$



Indukcja ważona – procedura

- Polega na szukaniu w grafach skierowanych zawierających rzutowane krawędzie ze zoptymalizowanymi wagami maksymalnych drzew rozpinających, które spełniają kryteria poprawnego drzewa zależności.
- Procedura:
  1. identyfikacja najbardziej prawdopodobnych krawędzi w rzutowanych multi-grafach z intuicyjnymi wagami na krawędziach i optymalizacja wag tych krawędzi,
  2. ekstrakcja maksymalnych rozpinających drzew zależności z multi-grafów ze zoptymalizowanymi wagami.

Schemat procedury indukcji ważonej



Maksymalne rozpinające drzewo zależnościowe

- Maksymalne rozpinające drzewo zależnościowe (MSDT) to maksymalne drzewo rozpinające $T = (V', A')$ w rzutowanym multi-grafie $G = (V, A)$:
  - $A' \subseteq A$,
  - $V' = V = \{v_0, v_1, ..., v_n\}$, gdzie v_i jest i-tym tokenem zdania $S = t_1, ..., t_n$, a v_0 jest dodatkowym wierzchołkiem drzewa,
  - v_0 jest korzeniem drzewa T, tzn. $(v_i, v_0, l) \notin A'$, dla $v_i \in V'$, $l \in L$,
  - v_0 ma tylko jedno dziecko, tzn. jeśli $(v_0, v_i, l) \in A'$, to $(v_0, v_j, l') \notin A'$, dla $v_i \neq v_j$.
- Algorytmy znajdujące maksymalne drzewa zależnościowe:
  - Chu-Liu-Edmonds algorytm (McDonald et al., 2005),
  - algorytm wyboru k najlepszych MSDTs (Camerini et al., 1980).

Optymalizacja wag rzutowanych krawędzi

1. Wybór k najlepszych MSDTs z rzutowanych multi-grafów z intuicyjnymi wagami na krawędziach:
   - jeśli najlepsze wybrane drzewo spełnia kryteria poprawnego drzewa zależnościowego, to wybierane są kolejne MSDTs,
   - jeśli najlepsze wybrane drzewo nie jest MSDT, to drzewa z tego multi-grafu nie są w ogóle brane pod uwagę w trenowaniu,
2. oszacowanie rozkładu prawdopodobieństwa typów krawędzi w zbiorze wybranych MSDTs za pomocą zmodyfikowanej wersji algorytmu EM (Dębowski, 2009),
3. optymalizacja wag krawędzi w rzutowanych multi-grafach:
   - jeśli reprezentacja cech j krawędzi (v_h, v_i, l) ma swoją instancję w oszacowanym rozkładzie prawdopodobieństwa (p_j), to: $s^* := \sqrt{s(v_h, v_i, l)} \times p_j$,
   - w innym przypadku: $s^* := \sqrt{s(v_h, v_i, l)} \times \min_j p_j \times \alpha$, dla jakiegoś $0 < \alpha < 1$

Indukcja banku struktur zależnościowych

- Ostateczne maksymalne rozpinające drzewa zależności są szukane w rzutowanych multi-grafach ze zoptymalizowanymi wagami na krawędziach,
- w rzutowanych multi-grafach skierowanych zostały znalezione prawie 4 miliony maksymalnych drzew rozpinających spełniających kryteria poprawnego drzewa zależnościowego,
- krawędzie w wyindukowanych drzewach zależnościowych nie są etykietowane polskimi funkcjami gramatycznymi (ang. 'unlabelled dependency trees').

Etykietowanie krawędzi w drzewach

- Etykietowanie krawędzi w drzewach jest procesem regułowym,
- reguły biorą pod uwagę rzutowane angielskie funkcje gramatyczne, cechy morfoskładniowe tokenów polskich połączonych relacją oraz informacje walencyjne wyekstrahowane ze słownika *Walenty*,
- proces etykietowania jest trzystopniowy:
  1. funkcje gramatyczne zostają przypisane do argumentów czasownika, jeśli czasownik jest w *Walentym* i cechy morfoskładniowe elementów ramy walencyjnej zgadzają się z cechami tokenów połączonych relacją,
  2. funkcje gramatyczne zostają przypisane do pozostałych podrzędników czasownika, który jest w *Walentym*,
  3. etykiety zostają przypisane do pozostałych jeszcze nienazwanych krawędzi w drzewach zależnościowych.

Etykietowanie krawędzi – przykład reguły *aglutynat*

If $pos_{dep} = aglt$ and $lemma_{dep} =$ ‘być’
  If $pos_{gov} \in [praet, winien]$ and $num_{dep} = num_{gov}$
    then $gf_{dep} := aglt$
  Elif $pos_{gov} \neq comp$ and $lemma_{gov} =$ ‘by’
    then $gf_{dep} := aglt$
  Else $gf_{dep} := dep$

Przykład: $\text{zrobili}_{praet} \xrightarrow{aglt} \text{śmy}_{aglt}$

Modyfikowanie krawędzi w drzewach

- Modyfikowanie krawędzi w drzewach jest procesem regułowym,
- poprawiane są krawędzie kodujące konstrukcje specyficzne dla języka polskiego np. aglutynat,
- poprawiane są krawędzie z rozbieżnymi anotacjami w obydwu językach np. frazy liczebnikowe,
- poprawiane są krawędzie, które powtarzają się często w drzewach i których anotacja wynika z błędnej struktury angielskiej, błędnych przyporządkowań słownych albo błędnego procesu indukcji.
- Przykład reguły *aglutynat*:

```
If a mobile inflection (lemma: być, pos: aglt) is adjacent to
a verb form or a conditional clitic ‘by’ on the left side,
then the left token is annotated as the governor of the mobile
inflection. Otherwise, if there is a verb form immediately
following the mobile inflection, it is its governor.
```

Eksperyment ewaluacyjny

- Dane:
  - ok. 5 mln par zdań z równoległych korpusów polsko-angielskich: *Europarl*, *DGT-Translation Memory*, *OPUS*, *Pelcra*, *EUR-Lex*, literatura,
  - polski: 10,75 tokenów/zdanie; angielski: 12,51 tokenów/zdanie.
- Preprocessing:
  - przyporządkowania słowne i ich symetryzacja: system *MOSES* (Koehn et al., 2007),
  - parsowanie angielskich zdań: parser *XLE* wykorzystujący angielską gramatykę *LFG*,
  - konwersja struktur LFG do postaci drzew zależnościowych.
- Ewaluacja w oparciu o zbiory testowe z poprzedniego eksperymentu.

Wyniki parsera *Mate* (1)

Model	Training Data	Filtering (in %)		Manual Test		Automatic Test		Additional Test	
		non-proj	*dep*	UAS	LAS	UAS	LAS	UAS	LAS
induced	3958556	–	–	73.7	–	72.8	–	63.5	–
labelled	3958556	–	–	74.6	69.4	74.0	68.1	63.7	58.3
modified	3958556	–	–	85.1	79.2	84.0	77.3	74.3	68.5
filtered	3548347	50	30	85.0	79.1	83.9	77.2	74.5	69.2
filtered	3036020	30	30	85.2	79.3	83.6	77.0	74.4	68.5
filtered	2352940	30	10	86.0	80.5	84.7	78.3	76.1	**70.3**
default	7405	–	–	**92.7**	**87.2**	88.4	81.0	76.0	69.5
automatic	7405	–	–	91.2	85.6	**90.8**	**84.7**	**76.6**	70.1

induced	parser trenowany na automatycznie wyindukowanych drzewach zależnościowych
labelled	parser trenowany na wyindukowanych drzewach z etykietami na krawędziach
modified	parser trenowany na wyindukowanych i poprawionych drzewach z etykietami na krawędziach
filtered	parser trenowany na części wyindukowanych i poprawionych drzew z etykietami na krawędziach
default	parser trenowany z wykorzystaniem standardowych parametrów na drzewach ze *Składnicy*
automatic	parser trenowany na drzewach ze *Składnicy* z automatycznymi tokenami

Wyniki parsera *Mate* (2)

- Parser *induced* trenowany na drzewach bez etykiet znajduje poprawne nadrzędniki dla 73,7% tokenów,
- parser *labelled* trenowany na drzewach z etykietami nieznacznie poprawia wyniki UAS parsera *induced*,
- parser *labelled* znajduje poprawne nadrzędniki wraz z poprawną etykietą relacji dla 69,4% tokenów,
- parser *modified* trenowany na poprawionych drzewach znacznie lepiej analizuje zdania,
- parser *filtered* trenowany na zmodyfikowanych i przefiltrowanych drzewach przypisuje poprawne nadrzędniki do 86,0% tokenów, a poprawne nadrzędniki i poprawne etykiety relacji do 80,5% tokenów,
- ewaluacja w oparciu o dłuższe i bardziej skomplikowane struktury pokazała, że parser trenowany na drzewach pozyskanych metodą ważonej indukcji działa nieznacznie lepiej niż parser trenowany na przekonwertowanych drzewach.

Outline

Porównanie metod pozyskiwania drzew

- Jakość parsowania:
  - ewaluacja na części przekonwertowanych drzew: parser trenowany na przekonwertowanych drzewach działa lepiej niż parser trenowany na automatycznie wyindukowanych drzewach,
  - ewaluacja na rozbudowanych zdaniach: obydwa parsery działają gorzej niż w przypadku wcześniejszego testu, ale tym razem parser trenowany na wyindukowanych drzewach analizuje te zdania nieznacznie lepiej.
- Zakres prac ręcznych: obydwie metody wymagają ręcznego stworzenia zbioru reguł.
- Możliwość wykorzystania metody dla innej pary języków:
  - metodę konwersji można wykorzystać tylko w języku, dla którego istnieje bank drzew składnikowych,
  - nawet jeśli istnieje taki bank, to reguły konwersji należy zdefiniować na nowo,
  - metodę indukcji automatycznej można wykorzystać do każdej pary języków, dla których istnieje tekstowy korpus równoległy oraz parser dla jednego z języków.

Dziękuję za uwagę!

Literatura

- Bohnet, B. (2010). Very High Accuracy and Fast Dependency Parsing is not a Contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING 2010, pages 89–97.
- Camerini, P. M., Fratta, L., and Maffioli, F. (1980). The K Best Spanning Arborescences of a Network. *Networks*, 10:91–110.
- Dębowski, Ł. (2009). Valence extraction using EM selection and co-occurrence matrices. *Language Resources and Evaluation*, 43(4):301–327.
- Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, ACL'07, pages 177–180.
- McDonald, R., Pereira, F., Ribarov, K., and Hajič, J. (2005). Non-projective Dependency Parsing using Spanning Tree Algorithms. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HTL'05, pages 523–530.
- Nivre, J., Hall, J., and Nilsson, J. (2006). MaltParser: A Data-Driven Parser-Generator for Dependency Parsing. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, LREC'06, pages 2216–2219.