Tworzenie banku struktur zależnościowych z wykorzystaniem metody rzutowania ważonego

Alina Wróblewska

Instytut Podstaw Informatyki Polskiej Akademii Nauk

Seminarium ZIL
Warszawa, 8 kwietnia 2013

INNOVATIVE ECONOMY
NATIONAL COHESION STRATEGY

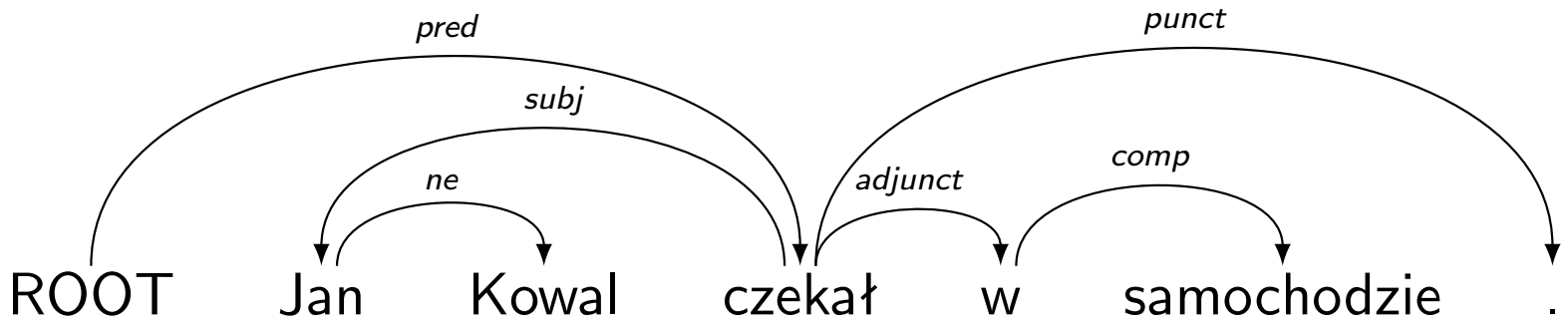
IPI PAN

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND

Outline

Struktura zależnościowa

- Struktura zależnościowa to drzewo rozpinające:
  - każdy wierzchołek ma jedną krawędź wejściową,
  - korzeń nie ma krawędzi wejściowych,
  - nie ma cykli,
- wierzchołki odpowiadają tokenom w zdaniu,
- krawędzie reprezentują relacje składniowe albo semantyczne,
- zakodowana struktura predykatywno-argumentowa.



Analiza zależnościowa

- Wykorzystanie analizy zależnościowej:
  - w systemach odpowiadających na pytania (ang. question answering),
  - w systemach ekstrakcji wiedzy (ang. information extraction),
  - w maszynowym tłumaczeniu.
- Parsery zależnościowe:
  - bazujące na ręcznie stworzonej gramatyce,
  - trenowane na banku ręcznie zaanotowanych struktur zależnościowych,
  - trenowane na automatycznie zaanotowanych strukturach zależnościowych.

Outline

Metoda rzutowania informacji lingwistycznych

- Automatyczne pozyskiwanie zaanotowanych danych w językach ubogich w narzędzia i zasoby lingwistyczne.
- Alternatywa dla ręcznego anototwania tekstów.
- Idea rzutowania:
  1. korpus równoległy (np. polsko-angielski),
  2. automatyczna analiza zdania w języku źródłowym (np. angielskim),
  3. rzutowanie analizy na tłumaczenie w języku docelowym (np. polskim) poprzez przyporządkowania słowne (ang. word alignment).

Problemy i dotychczasowe rozwiązania

- Problemy:
  - błędy w automatycznych przyporządkowaniach na poziomie słów i zdań,
  - błędy w automatycznej analizie tekstu źródłowego,
  - swoboda w tłumaczeniu.
- Dotychczasowe rozwiązania:
  - ręczne reguły korygujące rzutowane analizy,

    Hwa et al. (2005) – miara F z 33,9% do 65,7% dla hiszpańskiego i z 26,3% do 52,4% dla chińskiego,
  - filtrowanie potencjalnie błędnych analiz,

    Hwa et al. (2005) – z 98.000 do 20.000 dla pary angielski-hiszpański i z 240.000 do 50.000 dla pary angielski-chiński,

    Jiang and Liu (2009) – z 5,6 milionów do 500.000 chińskich drzew,
  - rzutowanie oparte na cechach morfo-składniowych tokenów docelowych i cechach przyporządkowań słownych z wykorzystaniem algorytmu EM,

    Smith and Eisner (2009) – jakość parsowania 68,5% dla niemieckiego i 64,8% dla hiszpańskiego.

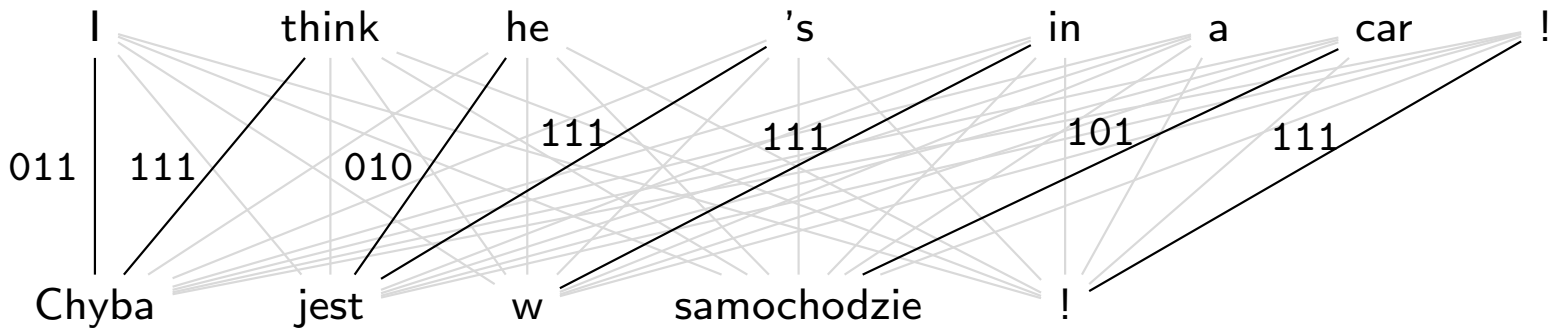
Outline

Rzutowanie ważone

1. Modyfikacja przyporządkowań słownych i przypisanie linkom wag,
2. przypisanie wag rzutowanym relacjom zależnościowym,
3. wybieranie maksymalnych drzew rozpinających (MST).

Ważone przyporządkowania słowne

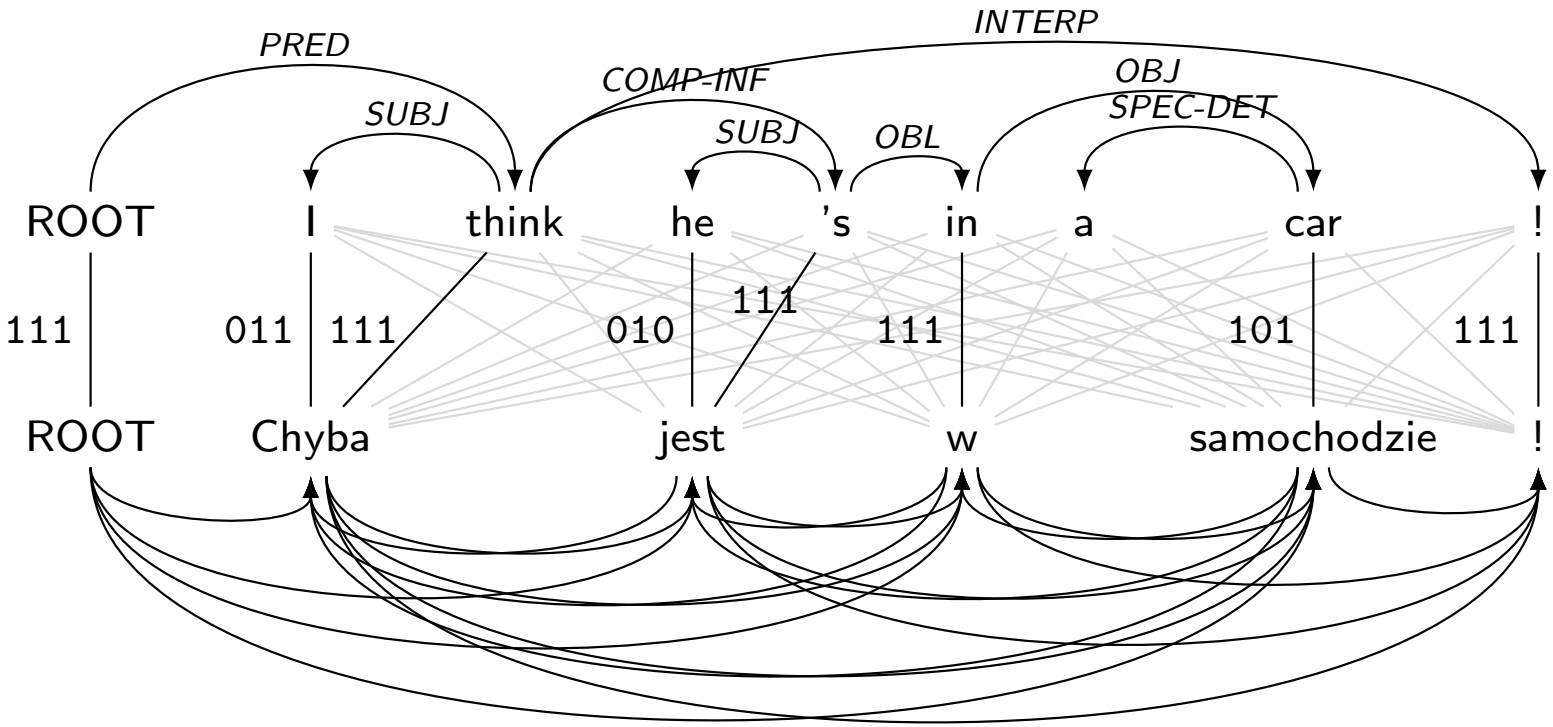
- Pełny graf dwudzielny pomiędzy tokenami polskimi i angielskimi,
- etykietowanie krawędzi grafu na podstawie obecności linków w automatycznych przyporządkowaniach słownych, np. 011 – link nieobecny w $A_{\mathbf{pl}\rightarrow\mathbf{en}}$, ale obecny w $A_{\mathbf{en}\rightarrow\mathbf{pl}}$ i $A_{\mathbf{gdfa}}$,



- krawędziom grafu zostają przypisane wagi (0–3) w zależności od liczby linków, np. jeśli brak linków 000 – 0; 100 albo 010 – 1; 101 albo 011 – 2; 111 – 3.

Rzutowanie relacji (1)

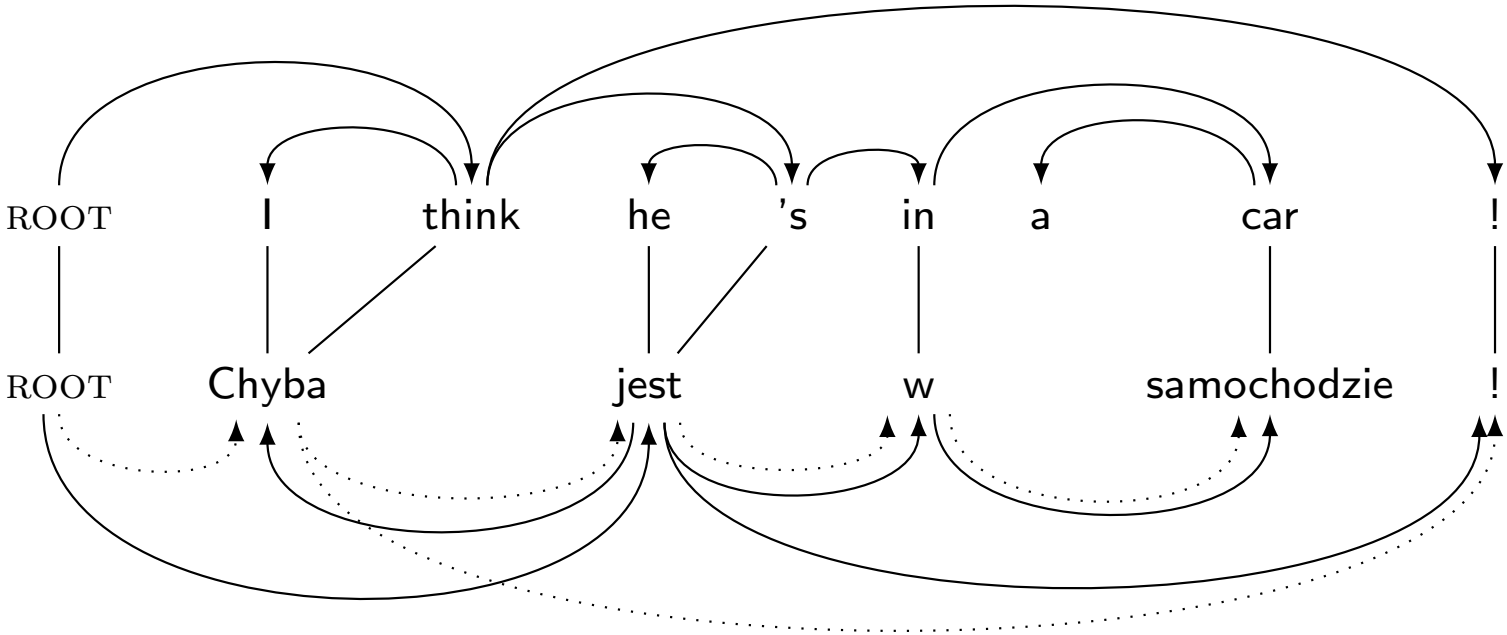
- Rzutowanie wszystkich relacji angielskich poprzez linki, z których przynajmniej jeden nie jest zerowy,
- rzutowane relacje zależnościowe tworzą multigrafy skierowane.



Rzutowanie relacji (2)

Rzutowanie poprzez linki z których jeden jest zerowy:

- liczba rzutowanych krawędzi zwiększa się o potencjalnie błędne krawędzie, ale
- być może uda się poprawić błędne drzewa będące wynikiem rozbieżności w tłumaczeniu.



Intuicyjne przypisywanie wag

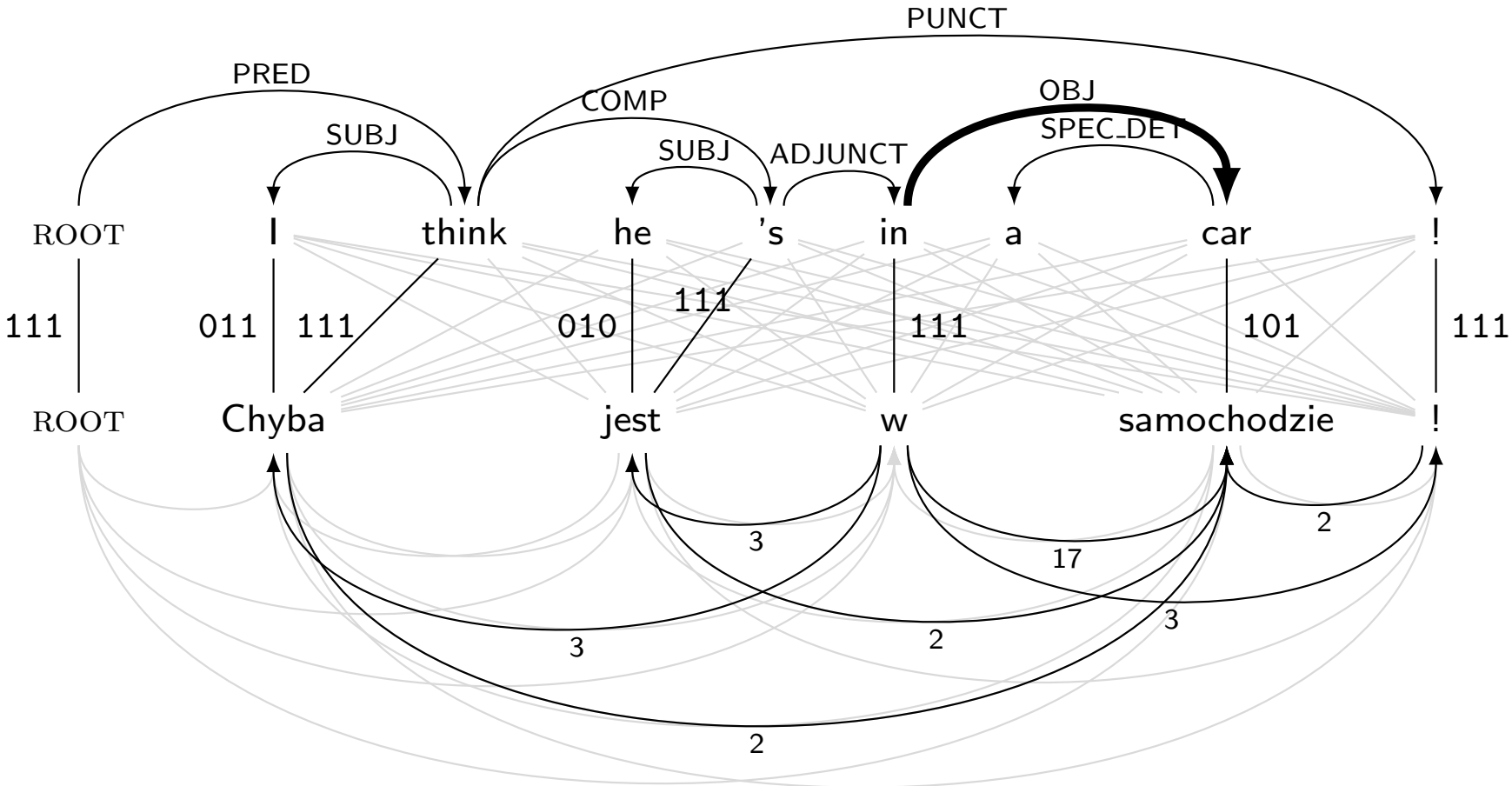
- Etykietowanie krawędzi w multigrafach, np. (3, 3, PRED, 1),
- przypisywanie wag krawędziom w multigrafach na podstawie wagi linków oraz częstości rzutowania danej relacji poprzez tą samą parę linków,

$$s_a = (d + g + 2dg) \times f \tag{1}$$

s_a – waga krawędzi
d – waga linku łączącego podrzędniki
g – waga linku łączącego nadrzędniki
f – częstość rzutowania krawędzi

- np. krawędzi (3, 3, PRED, 1) zostanie przypisana waga $24 = (3 + 3 + 18) \times 1$

Multigraf z początkowymi wagami



Automatyczne przypisywanie wag

- Założenie: maksymalne drzewo rozpinające jest ekwiwalentem struktury zależnościowej.
- Procedura:
  1. wybranie k-najlepszych drzew rozpinających,
  2. oszacowanie rozkładu prawdopodobieństwa typów krawędzi w wybranych drzewach przy pomocy algorytmu EM,
  3. aktualizacja wag na krawędziach w multigrafach,
  4. znalezienie ostatecznych maksymalnych drzew rozpinających w zaktualizowanych multigrafach.

Wybór maksymalnych drzew rozpinających (MST)

- MST to drzewo rozpinające danego grafu o największej możliwej sumie krawędzi.
- Algorytmy wyboru MST w grafach skierowanych:
  - Chu and Liu (1965) oraz Edmonds (1967),
  - Camerini et al. (1980) – wybiera k-najlepszych MSTs.
- Ograniczenia w wyborze MST:
  - tylko jedna krawędź wychodząca z korzenia,
  - krawędź wychodząca z korzenia prowadzi do podrzędnika innego niż kublik,
  - mniej niż 45% krawędzi ma dystans pomiędzy podrzędnikiem i nadrzędnikiem większy od 5.

Algorytm wyboru typów krawędzi (1)

- Adaptacja algorytmu EM wybierającego najbardziej prawdopodobne ramy walencyjne (Dębowski, 2009).
- Wejście: zbiory krawędzi A_i w k-najlepszych drzewach rozpinających dla m wyselekcjonowanych grafów, dla $i = 1, ..., m$.
- Iteracja:

$$p_{ij}^{(t)} = \begin{cases} \dfrac{p_j^{(t)}}{\sum_{j' \in A_i} p_{j'}^{(t)}} & \text{, if } j \in A_i \\ 0 & \text{, otherwise.} \end{cases} \tag{2}$$

$p_{ij}^{(t)}$ – normalizacja prawdopodobieństwa krawędzi typu j w i-tym zbiorze krawędzi w iteracji t.

$$p_j^{(t+1)} = \frac{1}{m} \sum_{i=1}^{m} p_{ij}^{(t)} \tag{3}$$

$p_j^{(t)}$ – wartość prawdopodobieństwa krawędzi typu j w iteracji t.

Algorytm wyboru typów krawędzi (2)

- Zbiór wszystkich parametrów modelu w iteracji t to $\theta^{(t)} = \left(p_j^{(t)}\right)$,
- parametry modelu początkowego mają wartość 1,
- każda iteracja maksymalizuje funkcję wiarogodności (ang. likelihood function),

$$L^{(t)} := \sum_{i=1}^{M} log \left[\sum_{j \in A_i} p_j^{(t)} \right], \; t \geq 2 \tag{4}$$

- algorytm EM maksymalizuje parametry modelu dopóki $L^{(t)} > L^{(t-1)}$ albo $t = 10$.

Aktualizacja multigrafów

- Możliwe typy krawędzi:
  - część mowy podrzędnika, część mowy nadrzędnika i funkcja gramatyczna (kubliki zostały zastąpione przez formy podstawowe), np. (subst, praet, SUBJ)
  - forma podstawowa podrzędnika, forma podstawowa nadrzędnika i funkcja gramatyczna, np. (samochód, jechać, SUBJ).
- Aktualizacja wag na krawędzi:
  - jeżeli krawędź a jest w wybranych typach krawędzi, to $s_a^* = p_j$,
  - jeżeli krawędź nie jest w wybranych typach krawędzi, to

$$s_a^* = \frac{min(p_j) \times s_a}{\sum_{g=1}^{n} \sum_{a' \in A_g} s_{a'}} \tag{5}$$

s_a^* – nowa waga krawędzi
n – liczba wszystkich multigrafów
A_g –zbiór krawędzi w multigrafie g, dla $g \in \{1, ..., n\}$

Outline

Dane treningowe i preprocessing

- Zbiór treningowy:
  - ok. 5 mln par zdań z równoległych korpusów polsko-angielskich: *Europarl*, *DGT-Translation Memory*, *OPUS*, *Pelcra*, *EUR-Lex*,
  - polski: 10,75 tokenów/zdanie; angielski: 12,51 tokenów/zdanie.
- Przyporządkowania słowne i ich symetryzacja: system *MOSES* (Koehn et al., 2007).
- Parsowanie angielskich zdań: parser *XLE* (Crouch et al., 2011) wykorzystujący angielską gramatykę *LFG*.
- Ewaluacja:
  - testowanie parsera wytrenowanego na automatycznie wygenerowanych strukturach zależnościowych,
  - zbiór testowy: 1000 drzew ze *Składnicy zależnościowej*,
  - miara: UAS (unlabelled attachement score) – liczba tokenów, którym parser zależnościowy przypisał poprawny nadrzędnik.

Trenowanie parsera

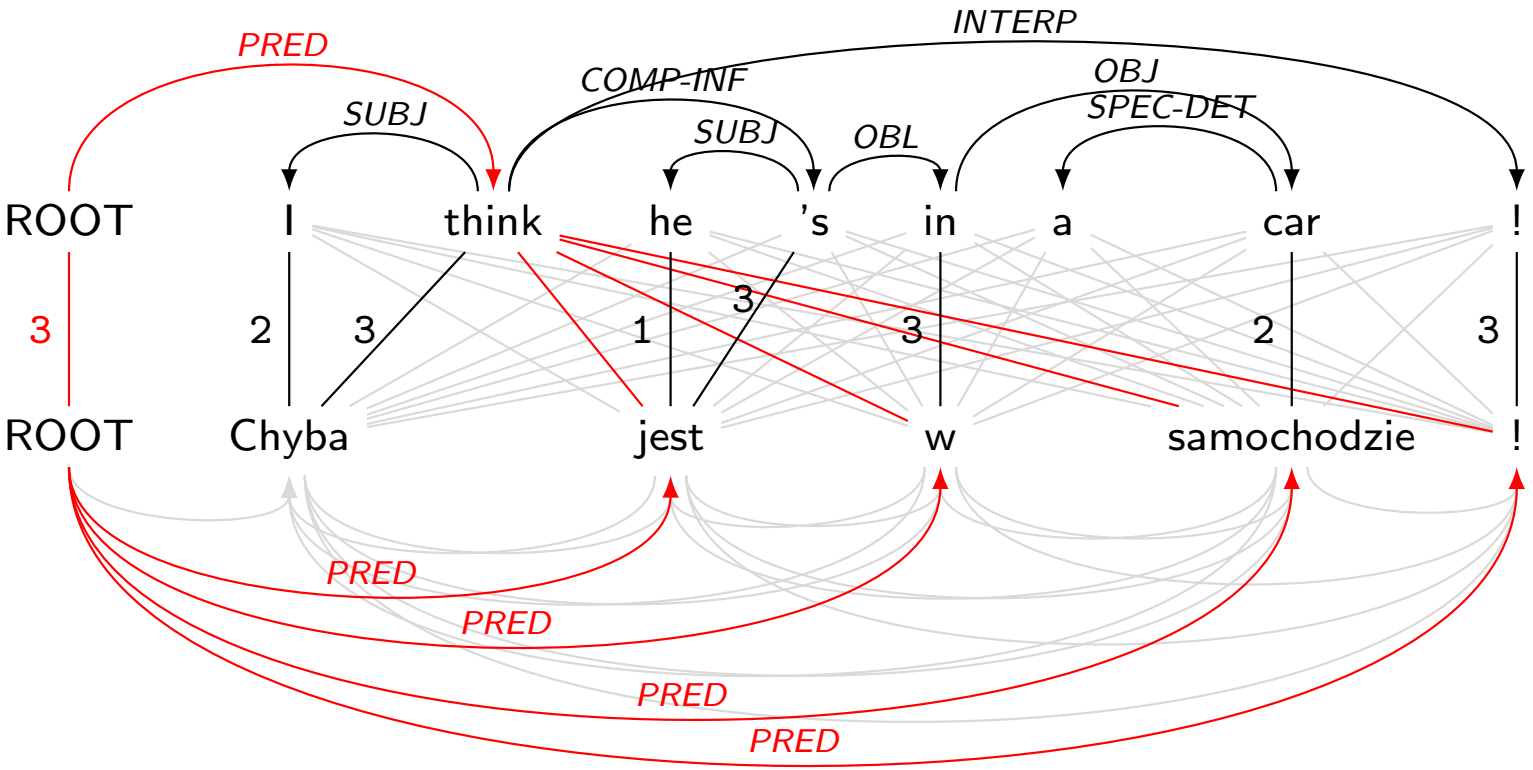
- Ze względów wydajnościowych w eksperymentach została wykorzystana część korpusu: 83 507 równoległych zdań polsko-angielskich.
- Baseline:
  - MSTs wyindukowane z multigrafów z początkowo przypisanymi wagami,
  - 52,7% UAS (bez reguł korygujących i filtrowania).

Wyniki

	EM (k=10)	EM+R (k=10)
baseline	**52,7**	
typ 1	27,4	39,7
typ 2	18,3	
typ 3	25,3	43,7

EM (k=10): 10 iteracji algorytmu EM na k MSTs
EM+R (k=3/10/50): 10 iteracji algorytmu EM na k MSTs wybranych ze zredukowanych grafów (tylko krawędzie k MSTs)
typ 1: pos-podrzędnika, pos-nadrzędnika, funkcja gramatyczna
typ 2: lemat-podrzędnika, lemat-nadrzędnika, funkcja gramatyczna
typ 3: pos/lemat-podrzędnika, pos/lemat-nadrzędnika, funkcja gramatyczna

Problem do rozwiązania



Dziękuję za uwagę!

Camerini, P. M., Fratta, L., and Maffioli, F. (1980). The K Best Spanning Arborescences of a Network. *Networks*, 10:91–110.

Chu, Y. J. and Liu, T. H. (1965). On the Shortest Arborescence of a Directed graph. *Science Sinica*, 14:1396–1400.

Crouch, D., Dalrymple, M., Kaplan, R., King, T., Maxwell, J., and Newman, P. (2011). *XLE Documentation*. Palo Alto Research Center (PARC), Palo Alto, CA.

Dębowski, Ł. (2009). Valence extraction using EM selection and co-occurrence matrices. *Language Resources and Evaluation*, 43(4):301–327.

Edmonds, J. (1967). Optimum Branchings. *Journal of Research of the National Bureau of Standards*, 71B(4):233—-240.

Hwa, R., Resnik, P., Weinberg, A., Cabezas, C., and Kolak, O. (2005). Bootstrapping Parsers via Syntactic Projection across Parallel Texts. *Natural Language Engineering*, 11(3):311–325.

Jiang, W. and Liu, Q. (2009). Automatic Adaptation of Annotation Standards for Dependency Parsing – Using Projected Treebank as Source Corpus. In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT '09)*, pages 25–28.

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 *Annual Meeting of the Association for Computational Linguistics (ACL 2007)*, pages 177–180.

Smith, D. A. and Eisner, J. (2009). Parser Adaptation and Projection with Quasi-Synchronous Grammar Features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 822–831.