Tager morfoskładniowy Concraft[-pl]

Jakub Waszczuk

Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland

22 kwietnia 2013

Plan prezentacji

Tagowanie – podstawowe pojęcia

Znacznik morfoskładniowy (tag, interpretacja)

Wartości atrybutów gramatycznych (klasy oraz kategorii).

Analiza morfoskładniowa (częściowa)

Określenie wszystkich możliwych interpretacji morfoskładniowych napotkanych w tekście słów.

Ujednoznacznianie morfoskładniowe (częściowe)

Wybranie interpretacji poprawnych w zadanym kontekście.

Założenie: w wyniku analizy otrzymujemy jednoznaczny, ustalony podział na słowa.

Tagowanie – podstawowe pojęcia

Znacznik morfoskładniowy (tag, interpretacja)

Wartości atrybutów gramatycznych (klasy oraz kategorii).

Analiza morfoskładniowa (częściowa)

Określenie wszystkich możliwych interpretacji morfoskładniowych napotkanych w tekście słów.

Ujednoznacznianie morfoskładniowe (częściowe)

Wybranie interpretacji poprawnych w zadanym kontekście.

Założenie: w wyniku analizy otrzymujemy jednoznaczny, ustalony podział na słowa.

Adaptacja CRF-ów

Problemy

- Liniowe CRF-y (ang. linear-chain conditional random fields) zbyt wolne w kontekście języka o liczbie tagów rzędu 10^3.
- Brak gwarancji spójności pomiędzy wynikami analizy a ujednoznacznianiem.

Pomysły

- Ograniczenie przestrzeni wyszukiwania do zbioru sekwencji tagów zgodnych z wynikami analizy.
- Zgadywanie potencjalnych interpretacji słów OOV (ang. out-of-vocabulary).

Adaptacja CRF-ów

Problemy

- Liniowe CRF-y (ang. linear-chain conditional random fields) zbyt wolne w kontekście języka o liczbie tagów rzędu 10^3.
- Brak gwarancji spójności pomiędzy wynikami analizy a ujednoznacznianiem.

Pomysły

- Ograniczenie przestrzeni wyszukiwania do zbioru sekwencji tagów zgodnych z wynikami analizy.
- Zgadywanie potencjalnych interpretacji słów OOV (ang. out-of-vocabulary).

Plan prezentacji

Liniowy model CRF

Wejście

Ciąg słów $\mathbf{x} = (x_1, \ldots, x_n)$,

Wyjście

Ciąg etykiet $\mathbf{y} = (y_1, \ldots, y_n)$,

Parametry

Zbiór parametrów modelu $\theta = \{\theta_k \in \mathbb{R}\}_{k=1}^{K}$,

Cechy

Funckja $f_k(\mathbf{x}, \mathbf{y}, i)$ o wartościach ze zbioru $\{0, 1\}$ dla $i \in \{1, \ldots, n\}$ oraz $k \in \{1, \ldots, K\}$,

Liniowy model CRF

Potencjał

Dodania funkcja $\phi_\theta(\mathbf{x}, \mathbf{y}) = \exp\left(\sum_{i=1}^{n}\sum_{k=1}^{K}\theta_k f_k(\mathbf{x}, \mathbf{y}, i)\right)$, która mierzy poprawność tagów $\mathbf{y}$ w kontekście zdania $\mathbf{x}$.

Przestrzeń potencjalnych sekwencji wyjściowych

$\mathbf{Y} = \prod_{i=1}^{n} Y$ gdzie Y to tagset a n stanowi długość zdania.

Prawdopodobieństwo

$$p_\theta(\mathbf{y}|\mathbf{x}) = \phi_\theta(\mathbf{x}, \mathbf{y})/Z_\theta(\mathbf{x})$$

$$Z_\theta(\mathbf{x}) = \sum_{\mathbf{y}\in\mathbf{Y}} \phi_\theta(\mathbf{x}, \mathbf{y})$$

Ograniczony liniowy model CRF

Ograniczenia

$\mathbf{r} = (r_1, \ldots, r_n)$ takie, że $r_i \neq \emptyset$ oraz $r_i \subseteq Y$ dla $i \in \{1, \ldots, n\}$.

Przestrzeń potencjalnych sekwencji wyjściowych

$\mathbf{Y}(\mathbf{r}) = \prod_{i=1}^{n} r_i$

Prawdopodobieństwo

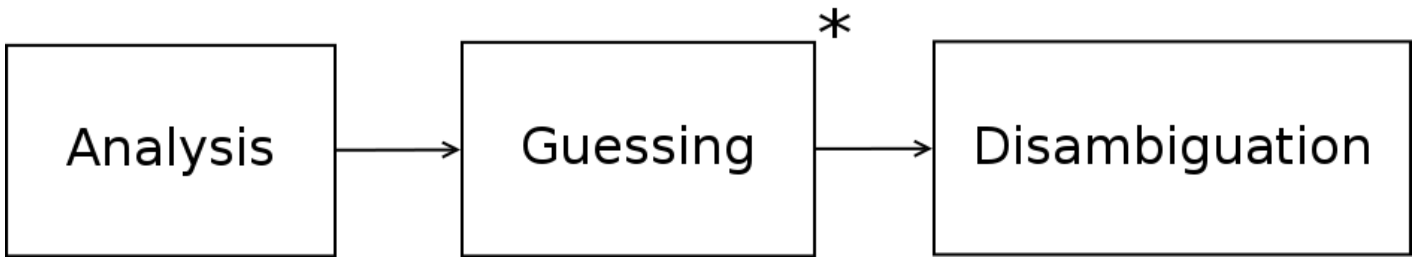
$$p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{r}) = \begin{cases} \phi_\theta(\mathbf{x}, \mathbf{y}) / Z_\theta(\mathbf{x}, \mathbf{r}) & \text{jeśli } \mathbf{y} \in \mathbf{Y}(\mathbf{r}) \\ 0 & \text{wpp.} \end{cases}$$

$$Z_\theta(\mathbf{x}, \mathbf{r}) = \sum_{\mathbf{y} \in \mathbf{Y}(\mathbf{r})} \phi_\theta(\mathbf{x}, \mathbf{y})$$

Rozkład problemu tagowania

Zgadywanie kontekstowe

- Odrzucenie mniej prawdopodobnych interpretacji w zadanym kontekście, a zarazem
- Pozostawienie wszystkich poprawnych interpretacji.



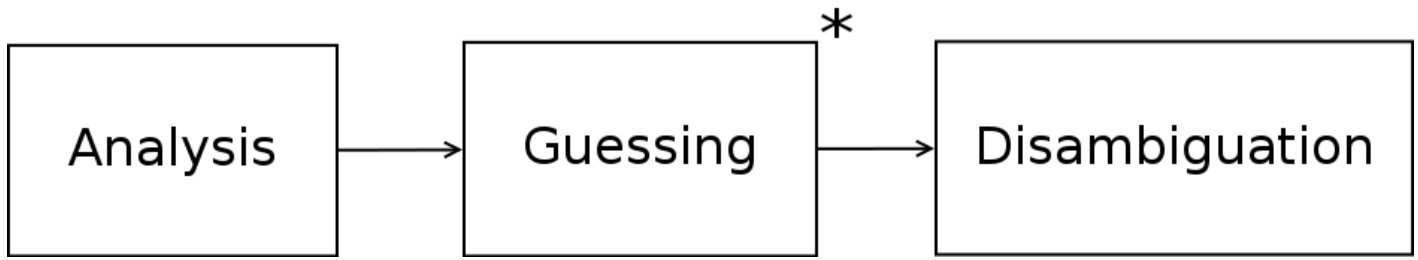
Zalety rozkładu

- Potencjalna prostota problemu zgadywania,
- Ujednoznacznianie właściwe przeprowadzane na ograniczonej przestrzeni rozwiązań.

Rozkład problemu tagowania

Zgadywanie kontekstowe

- Odrzucenie mniej prawdopodobnych interpretacji w zadanym kontekście, a zarazem
- Pozostawienie wszystkich poprawnych interpretacji.



Zalety rozkładu

- Potencjalna prostota problemu zgadywania,
- Ujednoznacznianie właściwe przeprowadzane na ograniczonej przestrzeni rozwiązań.

Model zgadywania kontekstowego

Ograniczony model CRF pierwszego rzędu. Każde słowo reprezentowane jest przez opisujący je zbiór obserwacji.

Cechy

- Przejścia modelują zależności pomiędzy sąsiednimi tagami.
- Unigramy modelują zależności pomiędzy tagami oraz odpowiadającymi im obserwacjami.

$$f_k(\mathbf{x}, \mathbf{y}, i) = \begin{cases} \mathbf{1}(y_i = u, o \in x_i) & k \text{ identyfikuje unigram } (u, o) \\ \mathbf{1}(y_i = u, y_{i-1} = v) & k \text{ identyfikuje przejście } (v, u) \end{cases}$$

Model zgadywania kontekstowego

Zgadywanie

Zgadywanie odbywa się w oparciu o rozkłady brzegowe:

$$p_\theta(y_i = t|\mathbf{x}, \mathbf{r}) = \sum_{\mathbf{y} \in \mathbf{Y}(\mathbf{r}) \,:\, y'_i = t} p_\theta(\mathbf{y}'|\mathbf{x}, \mathbf{r})$$

Schemat obserwacji dla polskiego

Wszystkie obserwacje dotyczą aktualnie rozpatrywanego słowa:

- Zmniejszone prefiksy i sufiksy długości 1 oraz 2,
- Czy słowo jest OOV,
- Skompresowany kształt słowa („Coling-2012” $\rightarrow$ „ullllllxdddd” $\rightarrow$ „ulxd”) oraz informacja, czy słowo znajduje się na początku zdania, połączone w jedną obserwację.

Model zgadywania kontekstowego

Zgadywanie

Zgadywanie odbywa się w oparciu o rozkłady brzegowe:

$$p_\theta(y_i = t|\mathbf{x}, \mathbf{r}) = \sum_{\mathbf{y} \in \mathbf{Y}(\mathbf{r}) \,:\, y'_i = t} p_\theta(\mathbf{y}'|\mathbf{x}, \mathbf{r})$$

Schemat obserwacji dla polskiego

Wszystkie obserwacje dotyczą aktualnie rozpatrywanego słowa:

- Zmniejszone prefiksy i sufiksy długości 1 oraz 2,
- Czy słowo jest OOV,
- Skompresowany kształt słowa („Coling-2012” $\rightarrow$ „ullllllxdddd” $\rightarrow$ „ulxd”) oraz informacja, czy słowo znajduje się na początku zdania, połączone w jedną obserwację.

Strategie zgadywania

Zakres

- Zgadywanie interpretacji słów nieznanych,
- Zgadywanie interpretacji wszystkich słów.

Metoda

- Wybranie k najbardziej prawdopodobnych tagów,
- Odrzucenie interpretacji o prawdopodobieństwie niższym niż zadany próg.

Przykład zgadywania

słowo	obserwacje	poprawny tag	interpretacje
Szef /*Head*	{ 1.S, 2.Sz , 3.f, 4.ef , 5.True , 6.ul-True }	subst:sg:nom:m1	{ subst:sg:nom:m1 }
administracji /*[of] administration*	{ 1.a, 2.ad , 3.i, 4.ji , 5.True , 6.l-False }	subst:sg:gen:f	{ subst:sg:gen:f , subst:sg:dat:f , subst:sg:loc:f , subst:pl:gen:f }
Wołodymyr	{ 1.W, 2.Wo , 3.r, 4.yr , 5.False , 6.ul-False }	subst:sg:nom:m1	*U*
Łatwyn	{ 1.Ł, 2.Ła , 3.n, 4.yn , 5.False , 6.ul-False }	subst:sg:nom:m1	*U*

Przykład zgadywania

słowo	interpretacje
Szef	{ **subst:sg:nom:m1** }
administracji	{ **subst:sg:gen:f** , subst:sg:dat:f , subst:sg:loc:f , subst:pl:gen:f }
Wołodymyr	{ subst:sg:nom:m2, subst:sg:nom:n, subst:sg:gen:m3 , subst:pl:nom:m1, subst:sg:nom:m3, **subst:sg:nom:m1** , adj:sg:gen:m3:pos, subst:sg:loc:f, qub, brev:npun }
Łatwyn	{ subst:sg:gen:m3, subst:sg:nom:m2, subst:sg:nom:n , subst:sg:gen:m1, subst:sg:gen:f, subst:sg:nom:m3 , subst:sg:acc:n, **subst:sg:nom:m1**, subst:pl:gen:m1 , subst:sg:nom:f }

Tablica : Wybór 10 najbardziej prawdopodobnych interpretacji słów nieznanych.

Model ujednoznaczniania

Ograniczony, **warstwowy** model CRF **drugiego** rzędu. Każde słowo reprezentowane jest przez opisujący je zbiór obserwacji. Niech $\{1, \ldots, L\}$ dla $L > 0$ oznacza zbiór identyfikatorów warstw.

Cechy w warstwie $l \in \{1, \ldots, L\}$

- Przejścia modelują zależności pomiędzy trzema następującymi po sobie tagami obciętymi do l-tej warstwy.
- Unigramy modelują zależności pomiędzy tagami obciętymi do l-tej warstwy oraz odpowiadającymi im obserwacjami.

Poszczególne warstwy **nie są** od siebie **niezależne**.

Model ujednoznaczniania

Cechy formalnie

Niech $y(l)$ oznacza część tagu y przypisaną do l-tej warstwy:

$$f_k(\mathbf{x}, \mathbf{y}, i) = \begin{cases} \mathbf{1}\left(y_i(l) = u, o \in x_i\right) & k \text{ identyfikuje unigram } (l, u, o) \\ \mathbf{1}\left(y_i(l) = u\right. & \\ \quad , y_{i-1}(l) = v & \\ \quad \left., y_{i-2}(l) = w\right) & k \text{ identyfikuje przejście } (l, w, v, u) \end{cases}$$

Ujednoznacznianie

Ujednoznacznianie odbywa się w oparciu o modelowane prawdopodobieństwo:

$$\mathbf{y}^* = \underset{\mathbf{y} \in \mathbf{Y}(\mathbf{r})}{\arg\max} \left(p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{r})\right)$$

Model ujednoznaczniania

Cechy formalnie

Niech $y(l)$ oznacza część tagu y przypisaną do l-tej warstwy:

$$f_k(\mathbf{x}, \mathbf{y}, i) = \begin{cases} \mathbf{1}\left(y_i(l) = u, o \in x_i\right) & k \text{ identyfikuje unigram } (l, u, o) \\ \mathbf{1}\left(y_i(l) = u\right. & \\ \quad , y_{i-1}(l) = v & \\ \quad \left., y_{i-2}(l) = w\right) & k \text{ identyfikuje przejście } (l, w, v, u) \end{cases}$$

Ujednoznacznianie

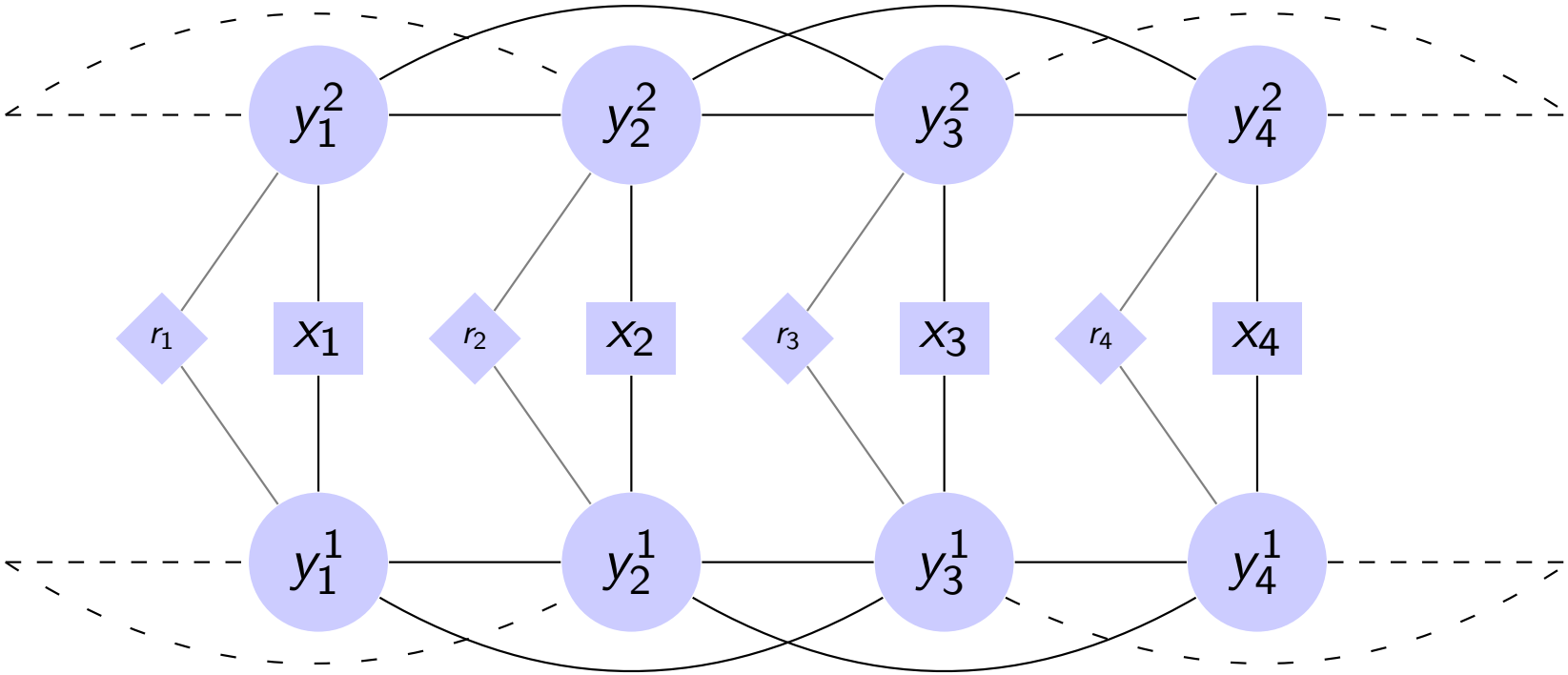
Ujednoznacznianie odbywa się w oparciu o modelowane prawdopodobieństwo:

$$\mathbf{y}^* = \underset{\mathbf{y} \in \mathbf{Y}(\mathbf{r})}{\arg\max} \left(p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{r})\right)$$

Model ujednoznaczniania

Podział dla polskiego

- Warstwa (1): część mowy, przypadek oraz osoba
- Warstwa (2): pozostałe kategorie gramatyczne



Model ujednoznaczniania

Schemat obserwacji dla polskiego

Zbiór obserwacji określony dla aktualnego słowa (na pozycji i) składa się ze zmniejszonych (z dużymi literami zamienionymi na małe) form ortograficznych na pozycjach $i-1$, i and $i+1$.

Dodatkowo, jeśli aktualne słowo jest OOV, zbiór obserwaji zostaje rozszerzony o:

- Zmniejszone prefiksy aktualnego słowa długości 1, 2 oraz 3,
- Zmniejszone sufiksy aktualnego słowa długości 1, 2 oraz 3,
- Skompresowany kształt słowa oraz informacja, czy słowo znajduje się na początku zdania, połączone w jedną obserwację.

Model ujednoznaczniania

Schemat obserwacji dla polskiego

Zbiór obserwacji określony dla aktualnego słowa (na pozycji i) składa się ze zmniejszonych (z dużymi literami zamienionymi na małe) form ortograficznych na pozycjach $i-1$, i and $i+1$.

Dodatkowo, jeśli aktualne słowo jest OOV, zbiór obserwaji zostaje rozszerzony o:

- Zmniejszone prefiksy aktualnego słowa długości 1, 2 oraz 3,
- Zmniejszone sufiksy aktualnego słowa długości 1, 2 oraz 3,
- Skompresowany kształt słowa oraz informacja, czy słowo znajduje się na początku zdania, połączone w jedną obserwację.

Trenowanie

Estymacja parametrów modelu CRF

Maksymalizacja wiarygodności (ang. likelihood) parametrów modelu względem zbioru treningowego przy użyciu metody SGD (ang. stochastic gradient descent). Przyjmujemy *a priori* rozkłady normalne na parametrach.

Trenowanie

- Każde zdanie korpusu poddajemy ponownej analizie morfoskładniowej.
- Trenujemy model zgadywania i wykorzystujemy go do określenia k najbardziej prawdopodobnych interpretacji dla każdego słowa OOV w korpusie.
- Trenujemy model ujednoznaczniania na korpusie ze zgadniętymi interpretacjami słów OOV.

Trenowanie

Estymacja parametrów modelu CRF

Maksymalizacja wiarygodności (ang. likelihood) parametrów modelu względem zbioru treningowego przy użyciu metody SGD (ang. stochastic gradient descent). Przyjmujemy *a priori* rozkłady normalne na parametrach.

Trenowanie

- Każde zdanie korpusu poddajemy ponownej analizie morfoskładniowej.
- Trenujemy model zgadywania i wykorzystujemy go do określenia k najbardziej prawdopodobnych interpretacji dla każdego słowa OOV w korpusie.
- Trenujemy model ujednoznaczniania na korpusie ze zgadniętymi interpretacjami słów OOV.

Plan prezentacji

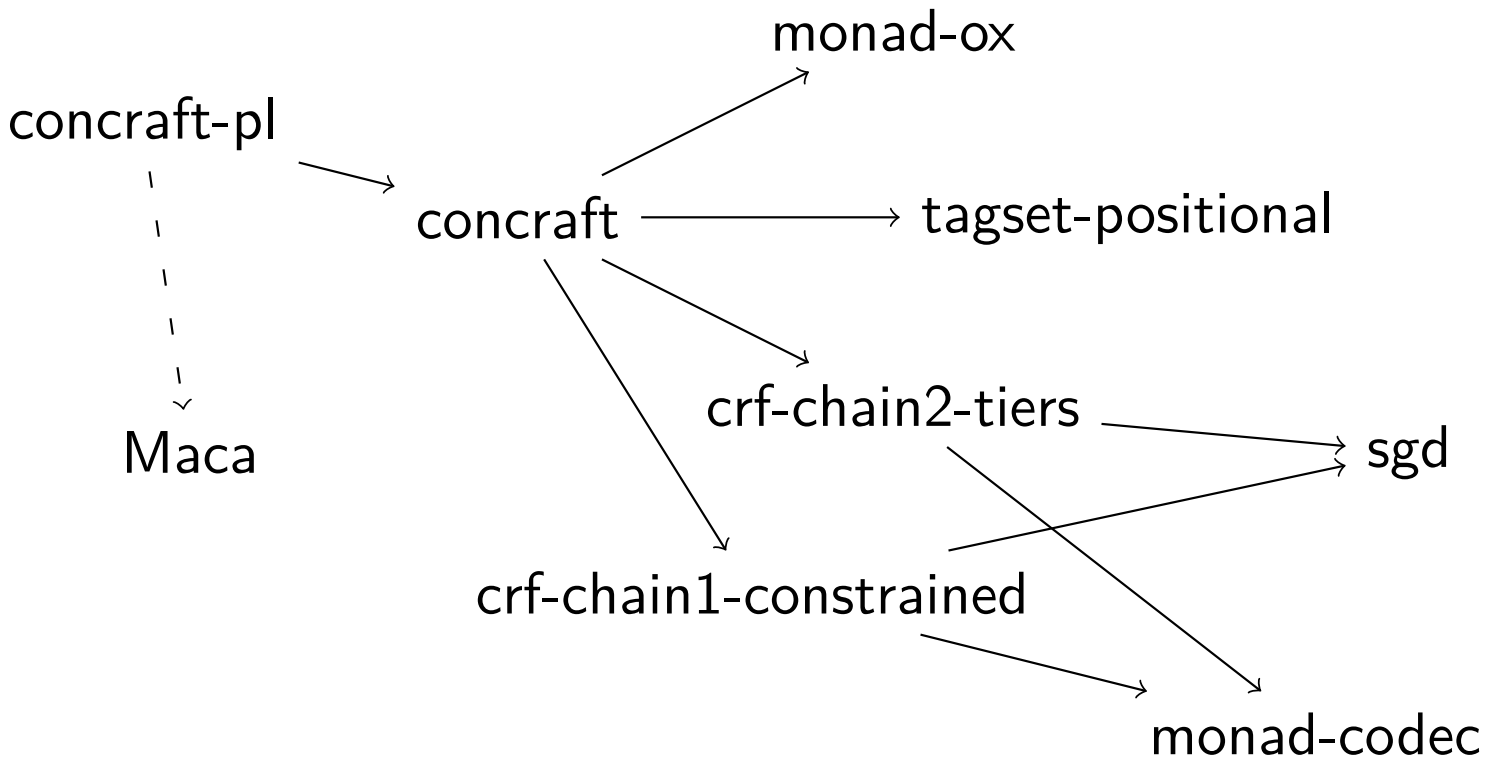
Implementacja

Informacje ogólne

- Concraft: biblioteka ujednoznaczniania
- Concraft-pl: tager morfoskładniowy (biblioteka + narzędzia)
- Dostępne na 2-klauzulowej licencji BSD
- Napisane w języku Haskell
- Kompilator dedykowany: GHC
- Strona domowa: `http://zil.ipipan.waw.pl/Concraft`
- Repozytorium: `https://github.com/kawu/concraft-pl`

Biblioteki

Biblioteki odpowiedzialne za poszczególne podzadania udostępnione na serwerze Hackage:



Założenia

Concraft ma docelowo przyjmować jak najmniej założeń na temat postaci poziomu morfoskładniowego. Założenia te wyrażają się na poziomie interfejsu biblioteki.

Do tagów przypisane są nieujemne wagi poprawności w kontekście zdania.

```
data Seg = Seg
    { orth  :: Text
    , oov   :: Bool
    , tags  :: Map Tag Double }
```

Concraft automatyzuje reanalizę danych w ramach trenowania modelu.

```
type Analyse = Text -> [Seg]
```

Założenia

Concraft ma docelowo przyjmować jak najmniej założeń na temat postaci poziomu morfoskładniowego. Założenia te wyrażają się na poziomie interfejsu biblioteki.

Do tagów przypisane są nieujemne wagi poprawności w kontekście zdania.

```
data Seg = Seg
    { orth  :: Text
    , oov   :: Bool
    , tags  :: Map Tag Double }
```

Concraft automatyzuje reanalizę danych w ramach trenowania modelu.

```
type Analyse = Text -> [Seg]
```

Założenia

Concraft ma docelowo przyjmować jak najmniej założeń na temat postaci poziomu morfoskładniowego. Założenia te wyrażają się na poziomie interfejsu biblioteki.

Do tagów przypisane są nieujemne wagi poprawności w kontekście zdania.

```
data Seg = Seg
    { orth  :: Text
    , oov   :: Bool
    , tags  :: Map Tag Double }
```

Concraft automatyzuje reanalizę danych w ramach trenowania modelu.

```
type Analyse = Text -> [Seg]
```

Założenia

Przyjmujemy tagset pozycyjny:

```
type POS, Attr, AttrVal = Text
type Optional           = Bool

data Tag = Tag
    { pos   :: POS
    , atts  :: Map Attr AttrVal }

data Tagset = Tagset
    { domains   :: Map Attr (Set AttrVal)
    , rules     :: Map POS [(Attr, Optional)] }
```

Concraft-pl

Maca [Radziszewski and Śniatowski, 2011]

- Podział na zdania, analiza morfoskładniowa (Morfesz SGJP) oraz podział na słowa.
- Concraft-pl uruchamia pulę instancji Macy w trybie interactywnym (liczba instancji zależy od `+RTS -N`).
- Komunikacja odbywa się przez stdin/stdout.

Użycie Macy nie jest obowiązkowe, ale jeśli użytkownik chce wykorzystać własny łańcuch przetwarzania, powinien:

- Przeprowadzić reanalizę korpusu treningowego,
- Wytrenować własny model.

Concraft-pl

Maca [Radziszewski and Śniatowski, 2011]

- Podział na zdania, analiza morfoskładniowa (Morfesz SGJP) oraz podział na słowa.
- Concraft-pl uruchamia pulę instancji Macy w trybie interactywnym (liczba instancji zależy od `+RTS -N`).
- Komunikacja odbywa się przez stdin/stdout.

Użycie Macy nie jest obowiązkowe, ale jeśli użytkownik chce wykorzystać własny łańcuch przetwarzania, powinien:

- Przeprowadzić reanalizę korpusu treningowego,
- Wytrenować własny model.

Concraft-pl

Trenowanie

- Przechowywanie przetworzonych danych treningowych w pamięci lub na dysku
- Opcja: usunięcie nieistotnych cech
- Trenowanie równoległe (opcja `+RTS -N`)
- Czas trenowania przy standardowych parametrach: 5 h*

Tagowanie

- Tagowanie stdin (wada: czas wczytywania modelu, 10 s*)
- Tryb klient/serwer (proof-of-concept)
- Szybość tagowania: 1250 słów/s*

* Intel Core i5-3210M (2.50GHz), 8GB RAM (1600Mhz)

Concraft-pl

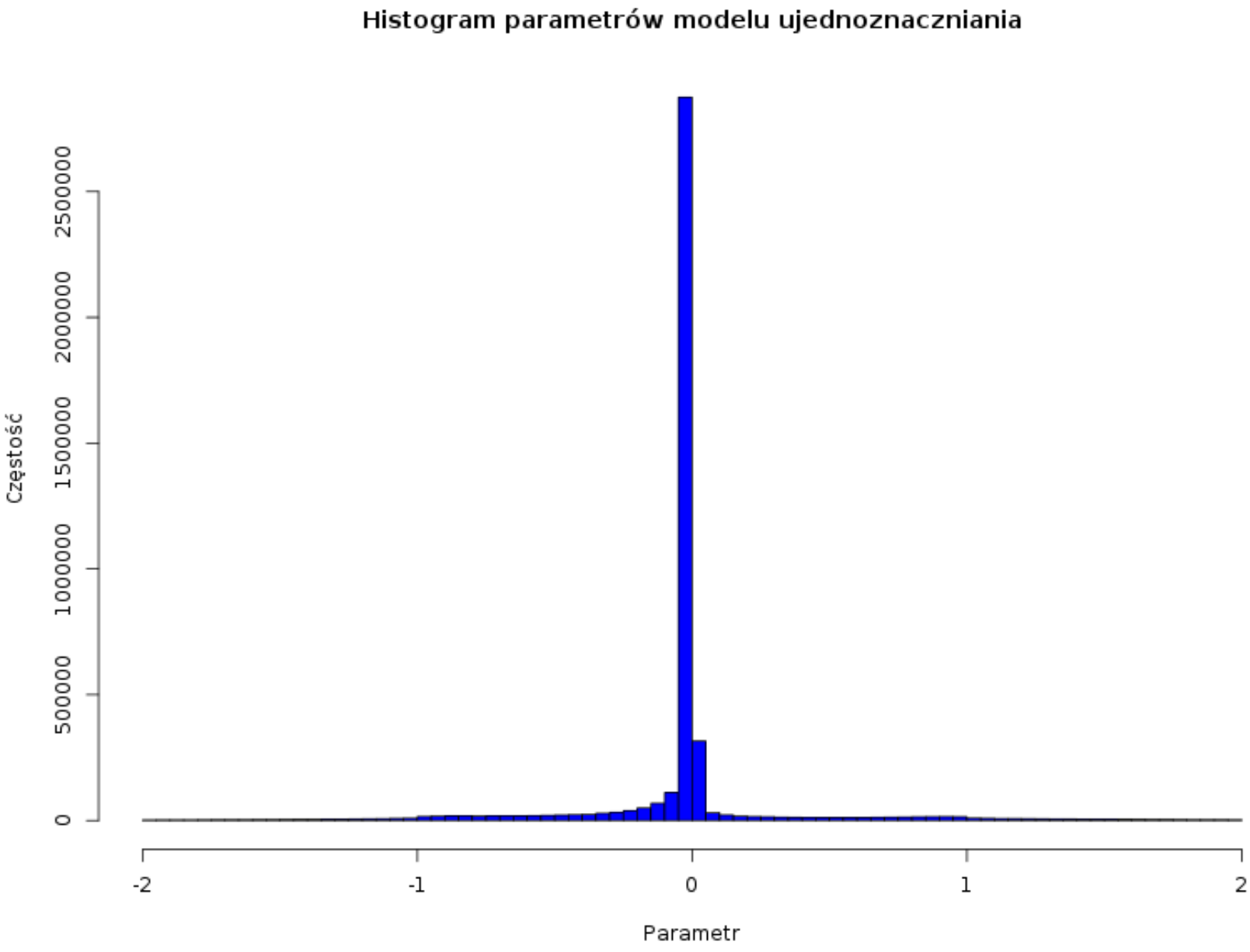
Trenowanie

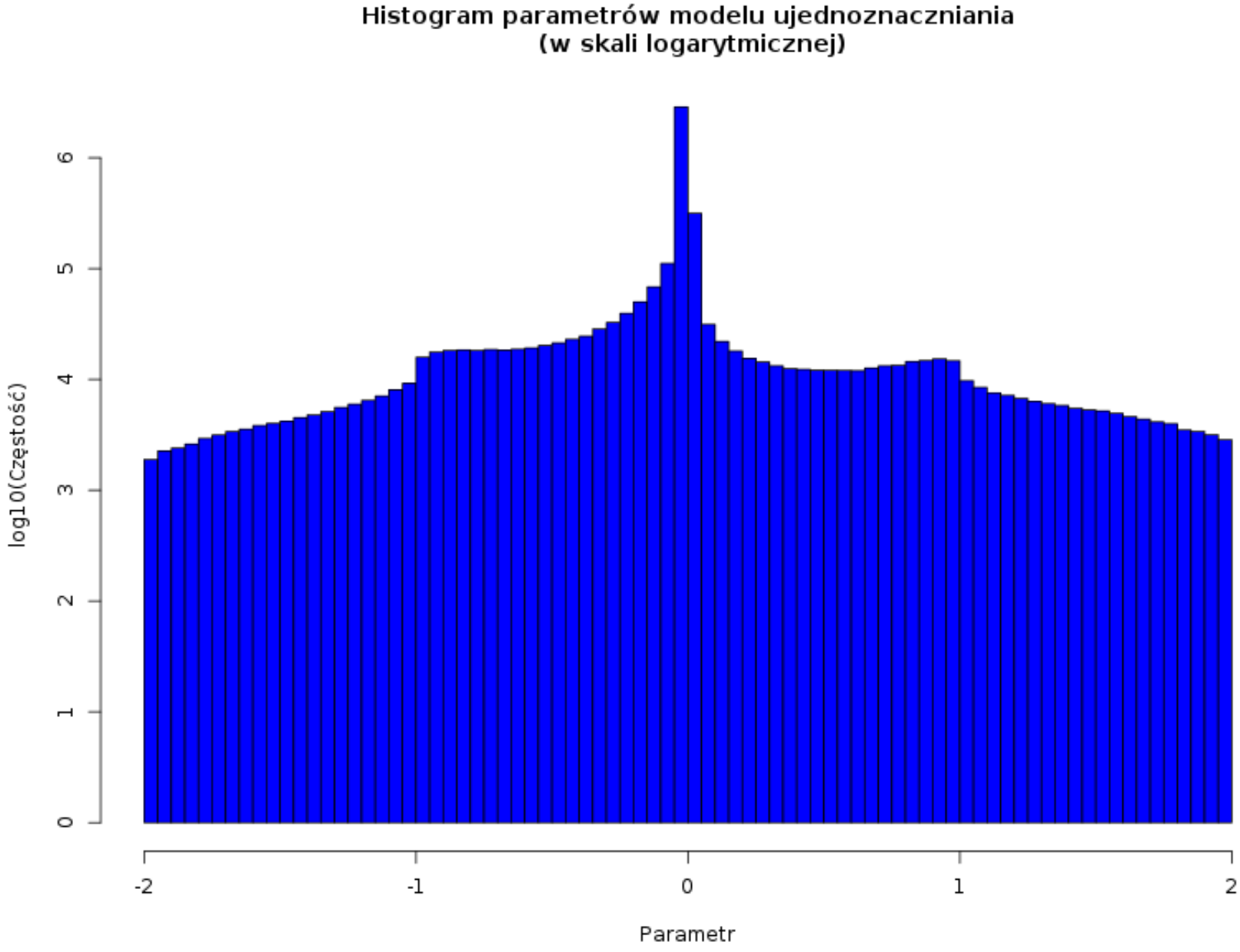
- Przechowywanie przetworzonych danych treningowych w pamięci lub na dysku
- Opcja: usunięcie nieistotnych cech
- Trenowanie równoległe (opcja `+RTS -N`)
- Czas trenowania przy standardowych parametrach: 5 h*

Tagowanie

- Tagowanie stdin (wada: czas wczytywania modelu, 10 s*)
- Tryb klient/serwer (proof-of-concept)
- Szybość tagowania: 1250 słów/s*

* Intel Core i5-3210M (2.50GHz), 8GB RAM (1600Mhz)


Histogram parametrów modelu ujednoznaczniania
Częstość
0
500000
1000000
1500000
2000000
2500000
-2
-1
0
1
2
Parametr


Histogram parametrów modelu ujednoznaczniania
(w skali logarytmicznej)
log10(Częstość)
0
1
2
3
4
5
6
-2
-1
0
1
2
Parametr

Plan prezentacji

Porównanie

WCRFT [Radziszewski, 2013]

- Kaskada modeli CRF – osobny model dla każdegu atrybutu gramatycznego.
- Modele pierwszego rzędu (relacje dalszego zasięgu reprezentowane są na poziomie obserwacji).
- Ujednoznacznianie wieloprzebiegowe – brak propagacji wyników ujednoznaczniania do niższych warstw.
- Arbitralne decyzje w przypadku niezgodności z wynikami analizy morfoskładniowej.

Porównanie

TaKIPI – zgadywacz [Piasecki and Radziszewski, 2007]

- Określa potencjalne interpretacje na podstawie sufiksów.
- Nie bierze pod uwagę kontekstu.
- Proponuje dla zadanego słowa OOV tylko jedną interpretację (ale za to z formą podstawową).

Pantera [Acedański, 2010]

- Warstwowy tager Brilla dostosowany do języków fleksyjnych.
- Konfigurowalna liczba warstw.
- Ujednoznacznianie wieloprzebiegowe – brak propagacji wyników ujednoznaczniania do niższych warstw.

Porównanie

TaKIPI – zgadywacz [Piasecki and Radziszewski, 2007]

- Określa potencjalne interpretacje na podstawie sufiksów.
- Nie bierze pod uwagę kontekstu.
- Proponuje dla zadanego słowa OOV tylko jedną interpretację (ale za to z formą podstawową).

Pantera [Acedański, 2010]

- Warstwowy tager Brilla dostosowany do języków fleksyjnych.
- Konfigurowalna liczba warstw.
- Ujednoznacznianie wieloprzebiegowe – brak propagacji wyników ujednoznaczniania do niższych warstw.

Porównanie

[Smith et al., 2005] – model dla czeskiego

- Model *source-channel* gwarantujący zgodność ujednoznaczniania z wynikami analizy.
- Rozwiązuje niejednoznaczności segmentacji na poziomie słów.
- Przeprowadza lematyzację.
- Ujednoznacznia tylko cztery główne atrybuty gramatyczne.
- Podział cech modelu na 5 klas i niezależna estymacja parametrów dla poszczególnych klas.
- Niezbyt wysoka dokładność ujednoznaczniania.
- Czas trenowania samego modelu części mowy (w 2005 roku): dwa tygodnie.

Porównanie

[Smith et al., 2005] – model dla czeskiego

- Model *source-channel* gwarantujący zgodność ujednoznaczniania z wynikami analizy.
- Rozwiązuje niejednoznaczności segmentacji na poziomie słów.
- Przeprowadza lematyzację.
- Ujednoznacznia tylko cztery główne atrybuty gramatyczne.
- Podział cech modelu na 5 klas i niezależna estymacja parametrów dla poszczególnych klas.
- Niezbyt wysoka dokładność ujednoznaczniania.
- Czas trenowania samego modelu części mowy (w 2005 roku): dwa tygodnie.

Ewaluacja

Metodologia

- 10-krotna walidacja krzyżowa.
- Ponowna segmentacja (na poziomie zdań i słów) oraz analiza danych. Dotyczy zarówno części przeznaczonej do oceny tagera (metodologia) jak i danych treninowych (jakość).

Zalety

- Ewaluacja pełnego procesu (segmentacja + analiza + ujednoznacznianie) tagowania tekstu.
- Badanie wrażliwości modułu ujednoznaczniającego na popełnione wcześniej błędy.

Ewaluacja

Metodologia

- 10-krotna walidacja krzyżowa.
- Ponowna segmentacja (na poziomie zdań i słów) oraz analiza danych. Dotyczy zarówno części przeznaczonej do oceny tagera (metodologia) jak i danych treninowych (jakość).

Zalety

- Ewaluacja pełnego procesu (segmentacja + analiza + ujednoznacznianie) tagowania tekstu.
- Badanie wrażliwości modułu ujednoznaczniającego na popełnione wcześniej błędy.

Wyniki ewaluacji

Ewaluacja narzędzi została przeprowadzona na **milionownym podkorpusie NKJP** oraz względem **tego samego podziału** korpusu na 10 części.

Tagger	Acc_{dis}	Acc_{low}	Acc_{upp}	Acc^{K}_{low}	Acc^{U}_{low}
WMBT	93.00	87.50	87.82	89.78	13.57
Pantera	92.95	88.99	89.28	91.27	14.74
WMBT+u	–	89.71	90.04	91.20	41.45
WCRFT	–	90.34	90.67	91.89	40.13
WCRFT-NG	–	90.69	91.01	91.95	49.84
Concraft	≈ 94	**91.12**	**91.44**	**92.10**	**59.19**

Tablica : Średnia dokładność w % uzyskana przez poszczególne narzędzia podczas walidacji krzyżowej.

Bibliografia

Acedański, S. (2010).

A Morphosyntactic Brill Tagger for Inflectional Languages.
In Advances in Natural Language Processing, (Loftsson, H., Rögnvaldsson, E. and Helgadóttir, S., eds), vol. 6233, of Lecture Notes in Computer Science pp. 3–14, Springer.

Piasecki, M. and Radziszewski, A. (2007).

Polish Morphological Guesser Based on a Statistical A Tergo Index.
In Proceedings of the International Multiconference on Computer Science and Information Technology — 2nd International Symposium Advances in Artificial Intelligence and Applications (AAIA'07) pp. 247–256,.

Radziszewski, A. (2013).

A tiered CRF tagger for Polish.
In Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions, (R. Bembenik, Ł. Skonieczny, H. R. M. K. M. N., ed.),. Springer Verlag.

Radziszewski, A. and Śniatowski, T. (2011).

Maca — a configurable tool to integrate Polish morphological data.
In Proceedings of the Second International Workshop on Free/Open-Source Rule-Based Machine Translation.

Smith, N. A., Smith, D. A. and Tromble, R. W. (2005).

Context-based morphological disambiguation with random fields.
In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing HLT '05 pp. 475–482, Association for Computational Linguistics, Stroudsburg, PA, USA.

Waszczuk, J. (2012).

Harnessing the CRF complexity with domain-specific constraints. The case of morphosyntactic tagging of a highly inflected language.
In Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012) pp. 2789–2804,, Mumbai.