Nowe metody ekstrakcji walencji czasowników z tekstów w języku polskim

Łukasz Dębowski, Marcin Woliński
{ldebowsk,wolinski}@ipipan.waw.pl

Instytut Podstaw Informatyki PAN

Problem

- Aby przeprowadzić automatyczną analizę syntaktyczną tekstu, potrzeba różnorakich zasobów (gramatyki, słowników).

> Jasio podziękował Marysi za współpracę.
> Marysia odpowiedziała mu, że nie ma za co.

- Jednym z tych zasobów jest słownik walencyjny:

> ...
> *odpowiedzieć*: np(nom), np(dat), ZE
> ...
> *podziękować*: np(nom), np(dat), za+np(acc)
> ...

- Istnieją takie słowniki na papierze – o niejasnej dokładności.
- Czy lepszy słownik można pozyskać z korpusu tekstów?

Ramy walencyjne

Odpowiedź dziwiła Jasia dwa tygodnie.
Jasia dziwiło, że Marysia tak odpowiedziała.
Jasio dziwił się i dziwił.
Jasio dziwił się i Marysi, i odpowiedzi.
Jasio dziwił się także, że się tak długo dziwi.

$$\mathbf{F(}\textit{dziwić}\mathbf{)} = \left\{ \begin{array}{l} \{\text{np(nom)}, \text{np(acc)}\}\ , \\ \{\text{ZE}, \text{np(acc)}\}\ , \\ \{\text{np(nom)}, \text{sie}\}\ , \\ \{\text{np(nom)}, \text{sie}, \text{np(dat)}\}\ , \\ \{\text{np(nom)}, \text{sie}, \text{ZE}\} \end{array} \right\}$$

Walencje są trudne do opisania

1. Nie wszystkie argumenty mogą współwystępować.
2. Trudno wyliczyć wszystkie ramy.
3. Część argumentów można opuścić.
4. Część argumentów bywa wymagana (czasem warunkowo).
5. Rozróżnienie argumentów i okoliczników jest trudne:

> Odpowiedź brzmiała dziwacznie.

6. Klasy równoważnych argumentów są zależne od czasownika:

> Jasio dziwił się $\left\{ \begin{array}{l} \text{że Marysia tak odpowiedziała.} \\ \text{jej odpowiedzi.} \end{array} \right.$

Podejście Brenta (1993)

Dane: nieanotowany korpus tekstów,
płytki parser zwracający jednoznaczne analizy zdań.

$\mathbf{c(v, f)}$ — liczba parsów z ramą $\mathbf{f}$ i czasownikiem $\mathbf{v}$,
$\mathbf{c(v)} = \sum_{\mathbf{f}} \mathbf{c(v, f)}$ — liczba parsów z czasownikiem $\mathbf{v}$.

Uznajemy, że $\mathbf{f}$ jest ramą czasownika $\mathbf{v}$, gdy

$$\sum_{n=c(v,f)}^{c(v)} \binom{c(v)}{n} p_f^n (1 - p_f)^{c(v)-n} \leq 0.05,$$

gdzie $\mathbf{p_f}$ jest dobierane:

1. pod nadzorem tak, by zminimalizować liczbę błędów,
2. bez nadzoru tak, by rozkład $\mathbf{B(\cdot, p_f)}$ dopasował się do pierwszego skupienia na histogramie czasowników pogrupowanych wg proporcji $\mathbf{c(v, f)/c(v)}$.

Nasze podejście

1. Użyć parsera Świgra do analizy czystego tekstu.
2. Zredukować lasy analiz do lasów ram walencyjnych.
3. Ujednoznacznić las ram za pomocą nowego algorytmu wyboru EM.
4. Zliczyć wystąpienia ram i czasowników.
5. Zastosować uczenie pod nadzorem do ustalenia zbiorów możliwych i obligatoryjnych argumentów.
6. Użyć formalizmu macierzy współwystąpień i uczenia pod nadzorem do naprawienia zbiorów całych ram.

Parsowanie

Teksty do ekstrakcji walencji pochodziły z Korpusu IPI PAN.

Analizowaliśmy je zmodyfikowaną Świgrą:

- dowolny czasownik mógł mieć $\leq$ **1** podmiot i dowolnie wiele innych argumentów.

Do ekstrakcji wybraliśmy zdania:

- długości $\leq$ **15** słów,
- analizowane w $\leq$ **1** minutę,
- mające $\leq$ **40** parsów na zdanie elementarne.

Braliśmy $\leq$ **5000** zdań dla jednego czasownika.

Redukcja analiz do ram walencyjnych

Usunięcie niektórych analiz:

- zawierających *to*, *co*, *nic*,
- zawierających skrajnie nieprawdopodobne interpretacje słów.

Przekształcenie pozostałych analiz:

- usunięcie fraz nie będących zależnikami orzeczenia,
- usunięcie zaimka *sam*,
- dodanie podmiotu domyślnego i wyrażonego przez *się*,
- naprawienie dopełniacza negacji,
- oznaczenie niektórych fraz jako okoliczników,
- wykreślenie lematów.

Bank lasów ram walencynych

- Bank zawiera **510 743** zdania elementarne.
- Świgra rozpatrzyła około **2.6** raza tyle zdań.
- Około **3.3** milionów słów bieżących.

```
'Kto zastąpi piekarza?'
zastąpić :np:acc: :np:nom:
zastąpić :np:gen: :np:nom:
'Nie płakał na podium.'
płakać :np:nom: :prepnp:na:acc:
płakać :np:nom: :prepnp:na:loc:
```

Algorytm wyboru EM

$\mathbf{Y_i}$ — las ram walencyjnych dla $\mathbf{i}$-tego zdania, $\mathbf{i = 1, 2, \ldots, M}$.
$\mathbf{p_j^{(n)}}$ — wypadkowa częstość ramy $\mathbf{j}$ w $\mathbf{n}$-tej iteracji, $\mathbf{p_j^{(1)} = 1}$.

$$\mathbf{p_{ji}^{(n)}} = \begin{cases} \mathbf{p_j^{(n)}} / \sum_{\mathbf{j' \in Y_i}} \mathbf{p_{j'}^{(n)}}, & \mathbf{j \in Y_i}, \\ \mathbf{0}, & \text{inaczej}, \end{cases}$$

$$\mathbf{p_j^{(n+1)}} = \sum_{\mathbf{i=1}}^{\mathbf{M}} \mathbf{p_{ji}^{(n)}}.$$

wybieranie losowe	akuratność
najkrótszej ramy o największym $\mathbf{p_{ji}^{(n)}}$	**72.6%**
ramy o największym $\mathbf{p_{ji}^{(n)}}$	**72.4%**
najkrótszej ramy	**57.5%**
na ślepo	**46.9%**

Test na 190 zdaniach, 500 prób Monte Carlo, $\mathbf{n = 10}$.

Matematycy są jak Francuzi: cokolwiek im się powie, od razu przekładają to na swój własny język i wówczas staje się to zupełnie czymś innym.

— *J. W. Goethe*

Krótkie wprowadzenie do ogólnego algorytmu EM

Dempster, Laird, and Rubin (1977):

- $\mathbf{Y}$ — zmienna obserwowana,
- $\boldsymbol{\theta}$ — nieznany parametr do oszacowania,
- $\mathbf{P}(\mathbf{Y}|\boldsymbol{\theta})$ — funkcja wiarogodności (rozkład $\mathbf{Y}$ dla każdego $\boldsymbol{\theta}$).

Estymator największej wiarogodności: $\boldsymbol{\theta}_{\mathrm{MLE}} = \mathbf{arg\,max}_{\boldsymbol{\theta}}\, \mathbf{P}(\mathbf{Y}|\boldsymbol{\theta})$.

Gdy nie możemy go policzyć, możemy postąpić tak:

- $\mathbf{Z}$ — dyskretna zmienna ukryta o prostym rozkładzie,
- cross-entropia

$$\mathbf{Q}(\theta', \theta'') = \sum_{\mathbf{z}} \mathbf{P}(\mathbf{Z} = \mathbf{z}|\mathbf{Y}, \theta') \log \mathbf{P}(\mathbf{Z} = \mathbf{z}, \mathbf{Y}|\theta''),$$

- obrawszy $\boldsymbol{\theta}_1$, iterujemy $\boldsymbol{\theta}_{\mathbf{n}+1} = \mathbf{arg\,max}_{\boldsymbol{\theta}}\, \mathbf{Q}\left(\boldsymbol{\theta}_{\mathbf{n}}, \boldsymbol{\theta}\right)$.

Łatwo udowodnić, że $\mathbf{P}\left(\mathbf{Y}|\boldsymbol{\theta}_{\mathbf{n}+1}\right) \geq \mathbf{P}\left(\mathbf{Y}|\boldsymbol{\theta}_{\mathbf{n}}\right)$.

Algorytm wyboru EM w ujęciu probabilistycznym

- $\mathbf{Z} = (\mathbf{Z}_1, \mathbf{Z}_2, \ldots, \mathbf{Z}_M)$, gdzie $\mathbf{Z}_i : \Omega \to \mathbf{J}$ — poprawne ramy.
- $\mathbf{Y} = (\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_M)$, gdzie $\mathbf{Y}_i : \Omega \to 2^{\mathbf{J}} \setminus \emptyset$ — lasy ram.
- $\theta = (\mathbf{p}_j)_{j \in J}$, gdzie $\mathbf{p}_j$ — p-stwo ramy $\mathbf{j}$,

$$\mathbf{P}(\mathbf{Z}_i = \mathbf{j} | \theta) = \mathbf{p}_j.$$

Algorytm wyboru EM staje się implementacją algorytmu EM, gdy

$$\mathbf{P}(\mathbf{Z} = (\mathbf{j}_1, \mathbf{j}_2, \ldots, \mathbf{j}_M), \mathbf{Y} | \theta) = \prod_{i=1}^{M} \mathbf{P}(\mathbf{Z}_i = \mathbf{j}_i, \mathbf{Y}_i | \theta),$$

$$\mathbf{P}(\mathbf{Y}_i = \mathbf{A} | \mathbf{Z}_i, \theta) = \begin{cases} \mathbf{g}(\mathbf{A}), & \mathbf{Z}_i \in \mathbf{A}, \\ 0, & \text{inaczej}. \end{cases}$$

Na przykład, możemy wziąć $\mathbf{g}(\mathbf{A}) = \mathbf{q}^{|\mathbf{A}|-1}(1-\mathbf{q})^{|\mathbf{J}|-|\mathbf{A}|}$.

Co maksymalizuje algorytm wyboru EM?

Niezależnie od postaci funkcji $\mathbf{g}(\cdot)$, założenie

$$\mathbf{P}(\mathbf{Y_i} = \mathbf{A}|\mathbf{Z_i}, \theta) = \begin{cases} \mathbf{g}(\mathbf{A}), & \mathbf{Z_i} \in \mathbf{A}, \\ \mathbf{0}, & \text{inaczej}, \end{cases}$$

implikuje $\mathbf{P}(\mathbf{Y_i}|\theta) = \mathbf{g}(\mathbf{Y_i})\mathbf{P}(\mathbf{Z_i} \in \mathbf{Y_i}|\theta)$.

$$\mathbf{P}(\mathbf{Y}|\theta) = \prod_{\mathbf{i}=1}^{\mathbf{M}} \mathbf{P}(\mathbf{Y_i}|\theta) = \prod_{\mathbf{i}=1}^{\mathbf{M}} \frac{\mathbf{P}(\mathbf{Z_i} \in \mathbf{Y_i}|\theta)}{\mathbf{g}(\mathbf{Y_i})}.$$

Zatem $\mathbf{L}^{(\mathbf{n}+1)} \geq \mathbf{L}^{(\mathbf{n})}$ dla $\mathbf{L}^{(\mathbf{n})} = \sum_{\mathbf{i}=1}^{\mathbf{M}} \log \left[\sum_{\mathbf{j} \in \mathbf{Y_i}} \mathbf{p}_{\mathbf{j}}^{(\mathbf{n})} \right].$

Algorytmu wyboru EM można użyć w b. wielu zadaniach NLP. Np. do dezambiguacji morfologicznej bez próby uczącej:

$\mathbf{Y_i}$ — możliwe interpretacje dla $\mathbf{i}$-tego okazu, $\mathbf{i = 1, 2, ..., M}$.
$\mathbf{p_j^{(n)}}$ — częstość interpretacji $\mathbf{j}$ w $\mathbf{n}$-tej iteracji.

$$\mathbf{p_{ji}^{(n)}} = \begin{cases} \mathbf{p_j^{(n)}} / \sum_{\mathbf{j' \in Y_i}} \mathbf{p_{j'}^{(n)}}, & \mathbf{j \in Y_i}, \\ \mathbf{0}, & \text{inaczej}, \end{cases}$$
$$\mathbf{p_j^{(n+1)}} = \sum_{\mathbf{i=1}}^{\mathbf{M}} \mathbf{p_{ji}^{(n)}}.$$

Wybór $\mathbf{p_j^{(1)}}$ jest niestotny, gdyż zbiór maksimów jest wypukły.

Proto-słownik, próba ucząca i testowa

Proto-słownik (zliczenia parsów po ujednoznacznieniu przez EM):

```
'przyłapać' => {
    'np(acc),np(gen),np(nom)' => 1,
    'na+np(loc),np(nom),sie' => 1,
    'na+np(loc),np(gen),np(nom)' => 1,
    'np(acc),np(nom)' => 4,
    'adv,np(nom)' => 1,
    'na+np(loc),np(acc),np(nom)' => 3
}
```

Próba ucząca:

- walencje **1166** czasowników wg słownika Świdzińskiego.

Próba testowa:

- walencje **203** czasowników wg słowników Polańskiego (1980), Świdzińskiego (1994) i Bańki (2000).

Nowy opis walencji

$$\mathbf{F}(\textit{przyłapać}) = \left\{ \begin{array}{l} \{\mathrm{np(nom), np(acc)}\}, \\ \{\mathrm{np(nom), np(acc), na{+}np(loc)}\}, \\ \{\mathrm{np(nom), sie, na{+}np(loc)}\} \end{array} \right\}$$

Opisujemy **F(v)** w sposób przybliżony za pomocą trzech obiektów:

1. **L(v)** — zbiór argumentów czasownika **v**,
2. **E(v)** — zbiór obligatoryjnych argumentów czasownika **v**,
3. $\mathbf{M(v)} : \mathbf{L(v)} \times \mathbf{L(v)} \to \{\leftarrow, \rightarrow, \leftrightarrow, \times, \perp\}$
   — macierz współwystąpień argumentów czasownika **v**.

Wartości macierzy współwystąpień słabo zależą od czasownika.

Zbiory argumentów

$$\mathbf{F}(\textit{przyłapać}) = \left\{ \begin{array}{l} \{\mathrm{np(nom)}, \mathrm{np(acc)}\}, \\ \{\mathrm{np(nom)}, \mathrm{np(acc)}, \mathrm{na{+}np(loc)}\}, \\ \{\mathrm{np(nom)}, \mathrm{sie}, \mathrm{na{+}np(loc)}\} \end{array} \right\}$$

$$\mathbf{L(v)} := \bigcup_{\mathbf{f} \in \mathbf{F(v)}} \mathbf{f}$$

$$\mathbf{E(v)} := \bigcap_{\mathbf{f} \in \mathbf{F(v)}} \mathbf{f}$$

Macierz współwystąpień

$$\mathbf{F}(\textit{przyłapać}) = \left\{ \begin{array}{l} \{\mathrm{np(nom), np(acc)}\}\,, \\ \{\mathrm{np(nom), np(acc), na{+}np(loc)}\}\,, \\ \{\mathrm{np(nom), sie, na{+}np(loc)}\} \end{array} \right\}$$

Niech $\langle \mathbf{a} \rangle := \{\mathbf{f} \in \mathbf{F}(\mathbf{v}) \mid \mathbf{a} \in \mathbf{f}\}$ oraz

$$\begin{array}{rcll}
\mathbf{a} \times \mathbf{b} & \Longleftrightarrow & \langle \mathbf{a} \rangle \cap \langle \mathbf{b} \rangle = \emptyset, & \text{(wykluczanie)} \\
\mathbf{a} \leftrightarrow \mathbf{b} & \Longleftrightarrow & \langle \mathbf{a} \rangle = \langle \mathbf{b} \rangle\,, & \text{(współwystępowanie)} \\
\mathbf{a} \rightarrow \mathbf{b} & \Longleftrightarrow & [\langle \mathbf{a} \rangle \subset \langle \mathbf{b} \rangle \wedge \langle \mathbf{a} \rangle \neq \langle \mathbf{b} \rangle]\,, & \text{(implikacja prawa)} \\
\mathbf{a} \leftarrow \mathbf{b} & \Longleftrightarrow & [\langle \mathbf{a} \rangle \supset \langle \mathbf{b} \rangle \wedge \langle \mathbf{a} \rangle \neq \langle \mathbf{b} \rangle]\,, & \text{(implikacja lewa)} \\
\mathbf{a} \perp \mathbf{b} & \Longleftrightarrow & \langle \mathbf{a} \rangle \setminus \langle \mathbf{b} \rangle\,, \langle \mathbf{a} \rangle \cap \langle \mathbf{b} \rangle\,, \langle \mathbf{b} \rangle \setminus \langle \mathbf{a} \rangle \neq \emptyset. & \text{(niezależność)}
\end{array}$$

$$\mathbf{M}(\mathbf{v})_{\mathbf{ab}} := \mathrm{R} \Longleftrightarrow \mathbf{a}\,\mathrm{R}\,\mathbf{b}$$

Przykład

$$\mathbf{F}(\textit{przyłapać}) = \left\{ \begin{array}{l} \{\text{np(nom)}, \text{np(acc)}\}\ , \\ \{\text{np(nom)}, \text{np(acc)}, \text{na+np(loc)}\}\ , \\ \{\text{np(nom)}, \text{sie}, \text{na+np(loc)}\} \end{array} \right\}$$

M(*przyłapać*)	np(nom)	np(acc)	sie	na+np(loc)
np(nom)	$\leftarrow\!\!\!\!\!\uparrow$	$\leftarrow$	$\leftarrow$	$\leftarrow$
np(acc)	$\uparrow$	$\leftarrow\!\!\!\!\!\uparrow$	$\times$	$\perp$
sie	$\uparrow$	$\times$	$\leftarrow\!\!\!\!\!\uparrow$	$\perp$
na+np(loc)	$\uparrow$	$\perp$	$\perp$	$\leftarrow\!\!\!\!\!\uparrow$

Porównanie macierzy (Bańko vs. Świdziński)

Liczby trójek $(\mathbf{v}, \mathbf{a}, \mathbf{b})$ o zadanych wartościach $\mathbf{M}(\mathbf{v})_{\mathbf{ab}}$:

		Bań.					
		$\times$	$\leftarrow$	$\rightarrow$	$\leftrightarrow$	$\perp$	N/A
Świ.	$\times$	364	2	2	–	30	724
	$\leftarrow$	4	253	1	2	18	176
	$\rightarrow$	4	1	253	2	18	176
	$\leftrightarrow$	–	25	25	410	2	124
	$\perp$	16	6	6	–	28	38
	N/A	1242	230	230	167	102	

Wysoka zgodność słowników

Liczby trójek $(\mathbf{v}, \mathbf{a}, \mathbf{b})$ pojawiających się w parach słowników:

	$\mathbf{\Sigma}$	równe $\mathbf{M(v)_{ab}}$	różne $\mathbf{M(v)_{ab}}$	zgodność
w Bań. i Pol.	1383	1187	196	86%
w Bań. i Świ.	1472	1308	164	89%
w Pol. i Świ.	1449	1283	166	89%

Słowniki walencyjne podają dość odmienne zbiory argumentów dla tych samych czasowników. Jednak łączliwość argumentów wg tych samych słowników jest bardzo podobna.

Najczęstsze wartości macierzy

	np(nom)	np(acc)	advp	np(dat)	np(inst)	w+np(loc)	do+np(gen)	na+np(acc)	z+np(gen)
np(acc)	↑:66%								
adv	↑:94%	↑:57%							
np(dat)	↑:91%	↑:50%	×:72%						
np(inst)	↑:98%	↑:57%	×:85%	×:64%					
w+np(loc)	↑:91%	↑:61%	×:92%	×:71%	×:85%				
do+np(gen)	↑:94%	↑:48%	×:92%	×:80%	×:89%	×:100%			
na+np(acc)	↑:96%	↑:64%	×:96%	×:85%	×:86%	×:90%	×:100%		
z+np(gen)	↑:100%	↑:65%	×:94%	×:91%	×:100%	×:100%	×:92%	×:100%	
ZE	↑:91%	×:76%	×:86%	⊥:64%	×:100%	×:100%	×:80%	×:71%	⊥:100%

W dodatku łączliwość argumentów słabo zależy od czasownika.

Czyszczenie proto-słownika

1. Obliczyć $\mathbf{L(v)}$ i $\mathbf{E(v)}$ z $\mathbf{F(v)}$ w proto-słowniku.
2. Oczyścić $\mathbf{L(v)}$ i $\mathbf{E(v)}$. Następnie zrekonstruować
$$\mathbf{F(v)} := \{(\mathbf{f} \cup \mathbf{E(v)}) \cap \mathbf{L(v)} \mid \mathbf{f} \in \mathbf{F(v)}\}.$$
3. Obliczyć $\mathbf{M(v)}$ z bieżącego $\mathbf{F(v)}$.
4. Oczyścić $\mathbf{M(v)}$. Następnie zrekonstruować
$$\mathbf{F(v)} := \left\{ \mathbf{f} \in 2^{\mathbf{L(v)}} \,\middle|\, \begin{array}{l} \forall_{\mathbf{a} \in \mathbf{E(v)}}\, \mathbf{a} \in \mathbf{f}, \\ \forall_{\mathbf{a},\mathbf{b} \in \mathbf{L(v)}}\, \phi(\mathbf{f}, \mathbf{M(v)}, \mathbf{a}, \mathbf{b}) \end{array} \right\},$$
gdzie
$$\phi(\mathbf{f}, \mu, \mathbf{a}, \mathbf{b}) := \begin{cases} \neg(\mathbf{a} \in \mathbf{f} \wedge \mathbf{b} \in \mathbf{f}), & \mu_{\mathbf{ab}} = \times, \\ \mathbf{a} \in \mathbf{f} \iff \mathbf{b} \in \mathbf{f}, & \mu_{\mathbf{ab}} = \leftrightarrow, \\ \mathbf{a} \in \mathbf{f} \implies \mathbf{b} \in \mathbf{f}, & \mu_{\mathbf{ab}} = \rightarrow, \\ \mathbf{a} \in \mathbf{f} \impliedby \mathbf{b} \in \mathbf{f}, & \mu_{\mathbf{ab}} = \leftarrow, \\ \text{prawda}, & \mu_{\mathbf{ab}} = \perp. \end{cases}$$

Czyszczenie zbiorów argumentów

Argument **a** jest uznawany za możliwy dla czasownika **v**, jeżeli

$$\mathbf{c}(\mathbf{v}, \mathbf{a}) \geq \mathbf{p_a c}(\mathbf{v}) + \mathbf{1},$$

zaś za obligatoryjny dla czasownika **v**, jeżeli jest możliwy oraz

$$\mathbf{c}(\mathbf{v}) - \mathbf{c}(\mathbf{v}, \mathbf{a}) \geq \mathbf{p_{\neg a} c}(\mathbf{v}) + \mathbf{1},$$

gdzie

- $\mathbf{c}(\mathbf{v}, \mathbf{a})$ — liczba parsów zawierających argument **a** i czasownik **v**,
- $\mathbf{c}(\mathbf{v})$ — liczba parsów zawierających czasownik **v**,
- $\mathbf{p_a}$ oraz $\mathbf{p_{\neg a}}$ — parametry dobrane tak, aby zminimalizować liczbę błędów dla czasowników z próby uczącej.

Czyszczenie macierzy współwystąpień

- S — relacja $\mathbf{M(v)_{ab}}$ dla nieoczyszczonej macierzy
- R — najczęstsza relacja między **a** i **b** w próbie uczącej

	zgodność z próbą testową
$\mathbf{M(v)_{ab}} := \mathrm{S}$	77%
$\mathbf{M(v)_{ab}} := \mathrm{R}$	80%
$\mathbf{M(v)_{ab}} := \mathrm{T}$	83%

- $\mathrm{T} = \begin{cases} \mathrm{R}, & \mathbf{C(a\,\mathrm{R}\,b)} \geq \mathbf{p}_{\mathrm{S}\Rightarrow\mathrm{R}}\mathbf{C(ab)} + \mathbf{t}_{\mathrm{S}\Rightarrow\mathrm{R}}, \\ \mathrm{S}, & \text{inaczej}, \end{cases}$
- $\mathbf{C(ab)}$ — liczba czasowników o argumentach **a** i **b**
- $\mathbf{C(a\,\mathrm{R}\,b)}$ — liczba czasowników, dla których $\mathbf{a}\,\mathrm{R}\,\mathbf{b}$
- $\mathbf{p}_{\mathrm{S}\Rightarrow\mathrm{R}}$ oraz $\mathbf{t}_{\mathrm{S}\Rightarrow\mathrm{R}}$ — parametry optymalizowane

1. Wprowadzenie
2. Nowa metoda ekstrakcji
3. Ocena metody

Porównanie z wcześniejszym eksperymentem

Fast i Przepiórkowski (2005):

- walencje pozyskiwane z Korpusu IPI PAN,
- płytki parser,
- metoda Brenta (1993) (BMP),
- trenowanie i testowanie na słowniku Świdzińskiego,
- argumentami tylko frazy nominalne i przyimkowe.

	REC	PRE
Fast i Przepiórkowski (2005)	**48%**	**49%**
nasze wyniki (rzut na NP i PP)		
— protosłownik	**73%**	**19%**
— po oczyszczeniu zbiorów argumentów	**48%**	**64%**
— po oczyszczeniu macierzy współwystąpień	**43%**	**74%**
— powtórzenie BMP	**33%**	**85%**

Niedomogi proponowanego podejścia

- Rozkład fałszywych obserwacji pozytywnych:

FP	ogółem	argument climbing	błędy w słowniku
np(acc)	9	4	1
sie	4	0	4
np(dat)	11	3	6

- Czyszczenie macierzy współwystąpień powoduje, że wiele ich elementów staje się niezależnych od czasownika.
- Wykluczanie ZE $\times$ np(nom) jest nietypowe, a warunkowane nieobecnością *się* jest nieopisywalne:

$$\mathbf{F}(\mathit{dziwić}) = \left\{ \begin{array}{l} \{\mathrm{np(nom)}, \mathrm{np(acc)}\}, \\ \{\mathrm{ZE}, \mathrm{np(acc)}\}, \\ \{\mathrm{np(nom)}, \mathrm{sie}\}, \\ \{\mathrm{np(nom)}, \mathrm{sie}, \mathrm{np(dat)}\}, \\ \{\mathrm{np(nom)}, \mathrm{sie}, \mathrm{ZE}\} \end{array} \right\}$$

Porównanie z trzema słownikami testowymi

Liczby par **(v, f)**: Liczby par **(v, a)**:

a) po oczyszczeniu macierzy współwystąpień:

F	AE	Bań.	Pol.	Świ.	MV
AE	658				
Bań.	418	1646			
Pol.	359	771	1519		
Świ.	363	758	770	1359	
MV	394	984	996	983	1209
recall	0.33	0.81	0.82	0.81	
precision	0.6	0.6	0.66	0.72	

L	AE	Bań.	Pol.	Świ.	MV
AE	674				
Bań.	603	1330			
Pol.	586	956	1320		
Świ.	581	899	954	1251	
MV	600	1056	1111	1054	1211
recall	0.5	0.87	0.92	0.87	
precision	0.89	0.79	0.84	0.84	

b) po zastosowaniu metody Brenta:

F	AE	Bań.	Pol.	Świ.	MV
AE	413				
Bań.	311	1622			
Pol.	275	759	1498		
Świ.	294	743	750	1323	
MV	311	965	972	956	1178
recall	0.26	0.82	0.83	0.81	
precision	0.75	0.59	0.65	0.72	

L	AE	Bań.	Pol.	Świ.	MV
AE	582				
Bań.	524	1308			
Pol.	520	941	1296		
Świ.	521	883	930	1220	
MV	530	1038	1085	1027	1182
recall	0.45	0.88	0.92	0.87	
precision	0.91	0.79	0.84	0.84	

Dziękujemy!