# PoliTa – multitager morfosyntaktyczny dla języka polskiego

Łukasz Kobyliński

Instytut Podstaw Informatyki Polskiej Akademii Nauk
ul. Jana Kazimierza 5, 01-248 Warszawa, Poland

3.03.2014

# Plan


1. Wprowadzenie
2. Stan obecny
3. Łączenie wyników tagerów
4. Uwzględnianie kontekstu
5. Podsumowanie

# Wprowadzenie

## Znakowanie morfosyntaktyczne – fundamentalne zadanie w obszarze przetwarzania języka naturalnego

- przypisywanie znaczników (tagów) morfosyntaktycznych do tokenów w tekście,
- jakość znakowania ma bezpośredni wpływ na skuteczność innych narzędzi, działających na wyższych poziomach przetwarzania,
- zadanie relatywnie łatwe dla języka angielskiego (ok. 200 możliwych znaczników w tagsecie Brown Corpus):
  - np. candidates [candidate:NNS],
  - dokładność najlepszego tagera ok. 97%,
- dużo trudniejsze w jęz. polskim (ponad 4000 możliwych tagów):
  - e.g. kandydaci [kandydat:subst:pl:nom:m1].
  - dokładność najlepszego tagera ok. 91%.

# Dokładność tagera a liczba błędów

## Czy różnica dokładności 1% to dużo?

- przeciętna książka: 80 000 tokenów
  1% błędów tagera przekłada się na 800 nieprawidłowo oznakowanych tokenów,
- ręcznie oznakowany fragment Narodowego Korpusu Języka Polskiego (NKJP1M): 1 215 513 tokenów
  1% błędów tagera przekłada się na 12 155 nieprawidłowo oznakowanych tokenów w korpusie,
- pełny korpus NKJP: 1,8 mld tokenów
  1% błędów to 18 000 000 nieprawidłowo oznakowanych tokenów.

# Kontekst projektowy

## Projekt Narodowego Centrum Nauki

- „Automatyczne wykrywanie i korekcja błędów anotacyjnych w polskich korpusach językowych” (umowa nr 2011/01/N/ST6/01107)
- czas trwania: styczeń 2012 – czerwiec 2014
- wykonawcy: Ł. Kobyliński (kierownik), A. Przepiórkowski (opiekun), Ł. Szałkiewicz (lingwista)

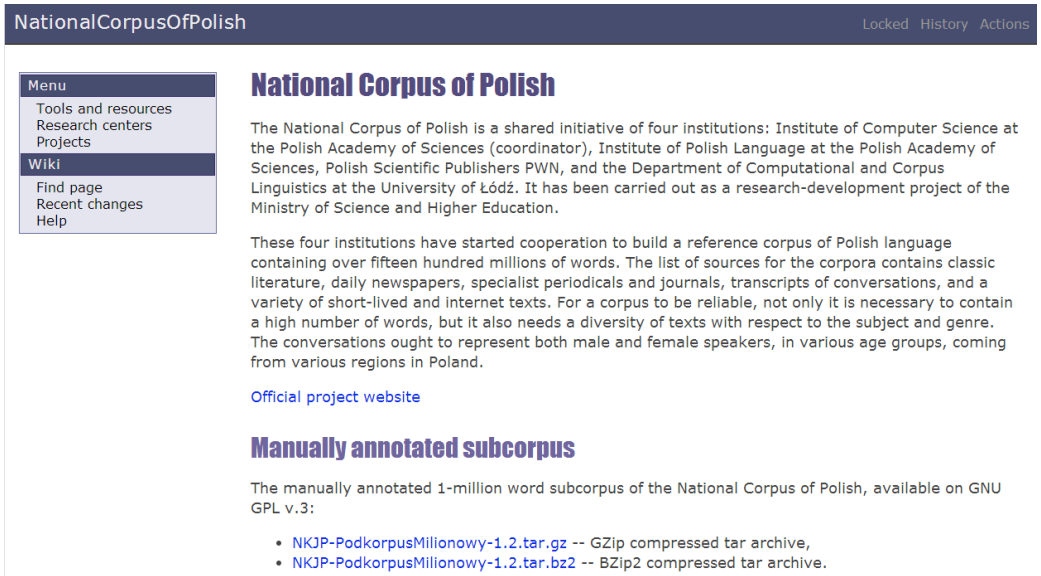
## W rzeczywistości dwa zadania składowe:

- poprawianie podkorpusu ręcznie anotowanego (NKJP1M),
- poprawianie korpusu pełnego NKJP.

# Poprawianie podkorpusu NKJP1M

## Udostępniona została wersja 1.2 korpusu, dostępna na CLIP

- poprawki ręczne warstwy morfosyntaktycznej (Łukasz Szałkiewicz),
- poprawki ręczne warstwy słów składowych (Alicja Wójcicka, niezależnie od tego projektu),
- poprawki automatyczne.



# Poprawianie pełnego korpusu NKJP

**Inne zadanie, niż poprawianie podkorpusu ręcznie anotowanego**

- nie można zastosować metody nienadzorowanej, wyszukującej niekonsekwencji w anotacji (bo anotacja automatyczna),
- można dążyć do wyeliminowania jak największej liczby błędów z NKJP 1M i ponownie wyuczyć tager, który posłuży do nowej anotacji pełnego korpusu,
- można zwiększyć jakość tagowania automatycznego i ponownie oznakować korpus.

# Stan obecny

# Tagery morfostynaktyczne dla języka polskiego

## Tagery uwzględniające tagset NKJP

- Pantera [Acedański 2010] – adaptacja algorytmu Brilla do języków bogatych morfologicznie, takich jak polski,
- WMBT [Radziszewski and Śniatowski 2011] – tager oparty na uczeniu pamięciowym, rozbudowany o wielowarstwowość dla uwzględnienia wielu atrybutów znakowania w języku polskim,
- Concraft [Waszczuk 2012] – tager warstwowy, oparty na Conditional Random Fields (CRF); wyniki dezambiguacji morfosyntaktycznej przekazywane są z jednej warstwy do drugiej,
- WCRFT [Radziszewski 2013] – również oparty na CRF; osobne modele wykorzystywane są do dezambiguacji poszczególnych atrybutów opisu morfosyntaktycznego.

# Typowe błędy popełniane przez tagery

**gdera**

- Pani Hela nadal dobrotliwie gdera. [gdera:subst:sg:nom:m1]

**łup**

- gdzie złoczyńcy próbowali spieniężyć łup [łupa:subst:pl:gen:f]

**czasowniki z pierwszej setki listy frekwencyjnej**

- maić, mamić, nizać, chybać, sposobić, mieść, udziać

**kup**

- I kuszą: wstąp, kup, przecież to czas świąt, wielkiego kupowania, prezentów. [kupa:subst:pl:gen:f]

**[case=voc]**

- Były panie montażystki [pan:subst:sg:voc:m1]
- Za męstwo w wojnie polsko–bolszewickiej 1920 r. został odznaczony [Polska:subst:sg:voc:f]

# Motywacja

**Znakowanie jest rodzajem klasyfikacji**
– wybieranie jednej z predefiniowanych klas (tagów) dla każdego z przykładów (słów/tokenów).

**Łączenie klasyfikatorów, mające na celu uzyskanie klasyfikacji lepsej jakości jest zagadnieniem dobrze znanym w obszarze uczenia maszynowego.**

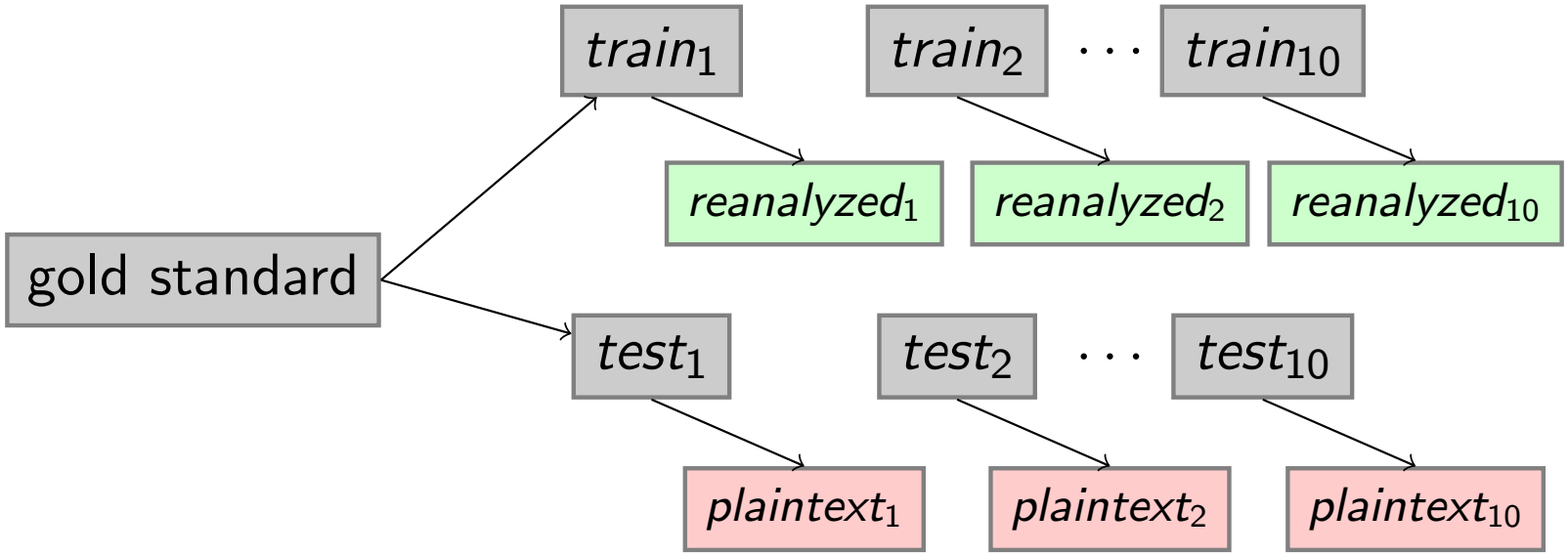
- bagging,
- boosting,
- stacking,
- ...

**W związku z tym, że mamy dostępnych kilka różnych tagerów, wartościowe może okazać się utworzenie komitetu klasyfikatorów..**

# Metoda ewaluacji (1)

## Ewaluacja na czystym tekście

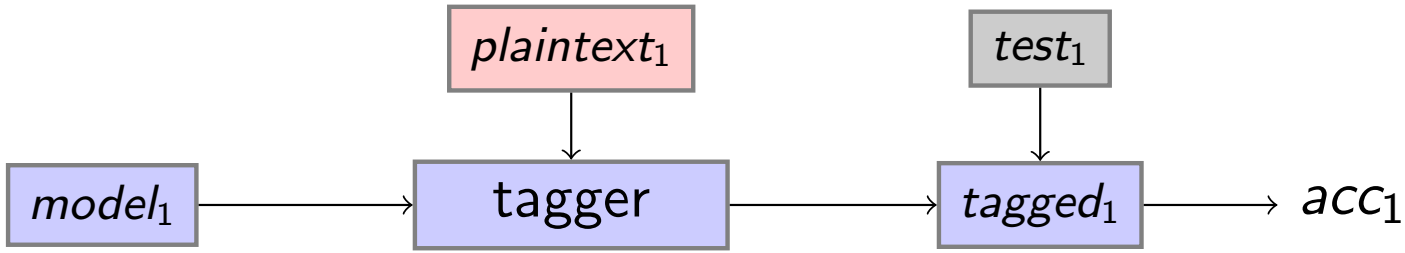
- [Radziszewski and Acedański 2012] zaproponowali procedurę ewaluacji tagerów, która przyjmuje czysty tekst jako podstawę porównania,
- odpowiada to realnemu użyciu tagerów, kiedy nie jest dostępna segmentacja tekstu i analiza morfologiczna.



# Metoda ewaluacji (2)

## Uczenie tagera


## Ewaluacja tagera




$$Acc = \sum_{i=1}^{10} acc_i/10.$$

# Metoda ewaluacji (3)

## Miara jakości znakowania

- ze względu na możliwość wystąpienia różnic w segmentacji pomiędzy wynikiem znakowania, a złotym standardem, wykorzystujemy dolne ograniczenie trafności (*accuracy lower bound*, $Acc_{lower}$) do oceny dokładności tagerów,
- miara ta karze wszelkie zmiany segmentacyjne w stosunku do złotego standardu i traktuje takie tokeny jako sklasyfikowane błędnie,
- token traktowany jest jako oznakowany prawidłowo, jeśli zbiór jego interpretacji ma niepuste przecięcie ze zbiorem interpretacji zwracanych przez tager,
- niezależne sprawdzamy dokładność dla znanych ($Acc^{K}_{lower}$) i nieznanych słów ($Acc^{U}_{lower}$), aby ocenić skuteczność ew. modułów odgadywania.

# Ewaluacja pojedynczych tagerów

**Eksperymenty na milionowym podkorpusie Narodowego Korpusu Języka Polskiego, ver. 1.1, 10-krotna walidacja krzyżowa.**

| n | Tager | $Acc_{lower}$ | $Acc^{K}_{lower}$ | $Acc^{U}_{lower}$ |
|---|---|---|---|---|
| 1 | Pantera | 88.95% | 91.22% | 15.19% |
| 2 | WMBT | 90.33% | 91.26% | 60.25% |
| 3 | WCRFT | 90.76% | 91.92% | 53.18% |
| 4 | Concraft | 91.07% | 92.06% | 58.81% |

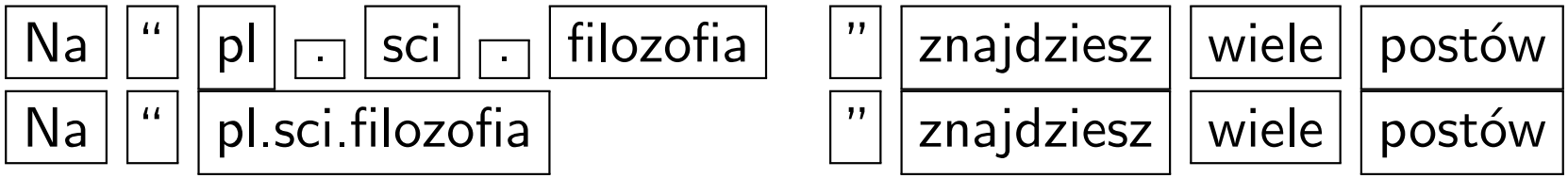
- $Acc_{lower}$ – łączna dokładność,
- $Acc^{K}_{lower}$ – dokładność dla znanych słów,
- $Acc^{U}_{lower}$ – dokładność dla słów nieznanych.

# Łączenie wyników tagerów

# Ujednoznacznianie segmentacji

## Zrównoleglenie rezultatu działania tagerów

- każdy z tagerów może potencjalnie wykorzystywać wewnętrznie inną segmentację,
- przed uzgodnieniem znakowania poszczególnych tokenów, konieczne jest zrównoleglenie i ujednoznacznienie segmentacji,
- zrównoleglenie następuje na poziomie czystego tekstu (form ortograficznych):



## Głosowanie nad segmentacją

- w przypadku różnicy pomiędzy tagerami w sposobie jaki dany fragment tekstu został podzielony na tokeny, segmentacja tego fragmentu jest ujednoznaczniana,
- ujednoznacznienie następuje poprzez proste głosowanie – każdy tager ma jeden głos.

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Na | “ | pl | . | sci | . | filozofia | ” | znajdziesz | wiele | postów |
| Na | “ | pl | . | sci | . | filozofia | ” | znajdziesz | wiele | postów |
| Na | “ | pl | . | sci | . | filozofia | ” | znajdziesz | wiele | postów |
| Na | “ | pl.sci.filozofia | | | | | ” | znajdziesz | wiele | postów |

# Analiza rezultatu działania tagerów

## Porównanie wyników

- Wszystkie zwracają prawidłowy tag: **82,78%**
  unikam fin:sg:pri:imperf

  fin:sg:pri:imperf+ fin:sg:pri:imperf+ fin:sg:pri:imperf+ fin:sg:pri:imperf+
- Większość zwraca prawidłowy tag: **7,95%**
  kapitalistów subst:pl:gen:m1

  subst:pl:gen:m1+ subst:pl:gen:m1+ subst:pl:gen:m1+ subst:pl:acc:m1-
- Równowaga w głosowaniu: **2,71%**
  powolny adj:sg:nom:m3:pos

  adj:sg:nom:m3:pos+ adj:sg:nom:m3:pos+ adj:sg:acc:m3:pos- adj:sg:acc:m3:pos-
- Prawidłowy tag w mniejszości: **2,38%**

  twarzy subst:sg:loc:f subst:sg:gen:f- subst:sg:gen:f- subst:sg:gen:f- subst:sg:loc:f+
- Wszystkie się mylą: **4.18%**
  biurka subst:pl:nom:n subst:pl:acc:n- subst:pl:acc:n- subst:sg:gen:n- subst:pl:acc:n-
  (Peggy) McCreary subst:sg:nom:f

  subst:sg:gen:f- subst:sg:gen:n- subst:sg:nom:n- subst:sg:acc:m1-

# Komplementarność tagerów (1)

**Komplementarność tagerów [Brill and Wu 1998] jest miarą (nie)podobieństwa zbiorów błędów popełnianych przez dwa tagery $A$ i $B$.**

Wartością komplementarności jest odsetek przypadków kiedy tager $B$ zwraca poprawny tag, podczas gdy tager $A$ się myli:

$$Comp(A, B) = (1 - \frac{e_{AB}}{e_A}) * 100,$$

gdzie $e_{AB}$ jest liczbą przypadków, w których myli się zarówno tager $A$, jak i tager $B$, natomiast $e_A$ jest liczbą błędów popełnianych tylko przez tager $A$.

# Komplementarność tagerów (2)

**Większa wartość – większa różnica w działaniu tagerów $A$ i $B$.**

| A \ B | Pantera | WMBT | WCRFT | Concraft |
|---|---|---|---|---|
| Pantera | 0.00% | 42.33% | 42.16% | 45.22% |
| WMBT | 34.09% | 0.00% | 35.30% | 39.52% |
| WCRFT | 30.78% | 32.25% | 0.00% | 33.97% |
| Concraft | 32.21% | 34.52% | 31.72% | 0.00% |

# Ograniczenia dokładności

**Jakie są ograniczenia oczekiwanej dokładności komitetu tagerów?**

- Dolne ograniczenie – losowy wybór jednego z tagerów za każdym razem, kiedy należy podjąć decyzję,
  - wybór losowy: $Acc_{lower} = 90.30\%$.
- Wartość odniesienia – dokładność najlepszego indywidualnego tagera,
  - Concraft: $Acc_{lower} = 91.07\%$.
- Górne ograniczenie – wybór zawsze prawidłowego tagu z propozycji dostarczanych przez tagery,
  - wyrocznia: $Acc_{lower} = 95.82\%$.

# Głosowanie (1)

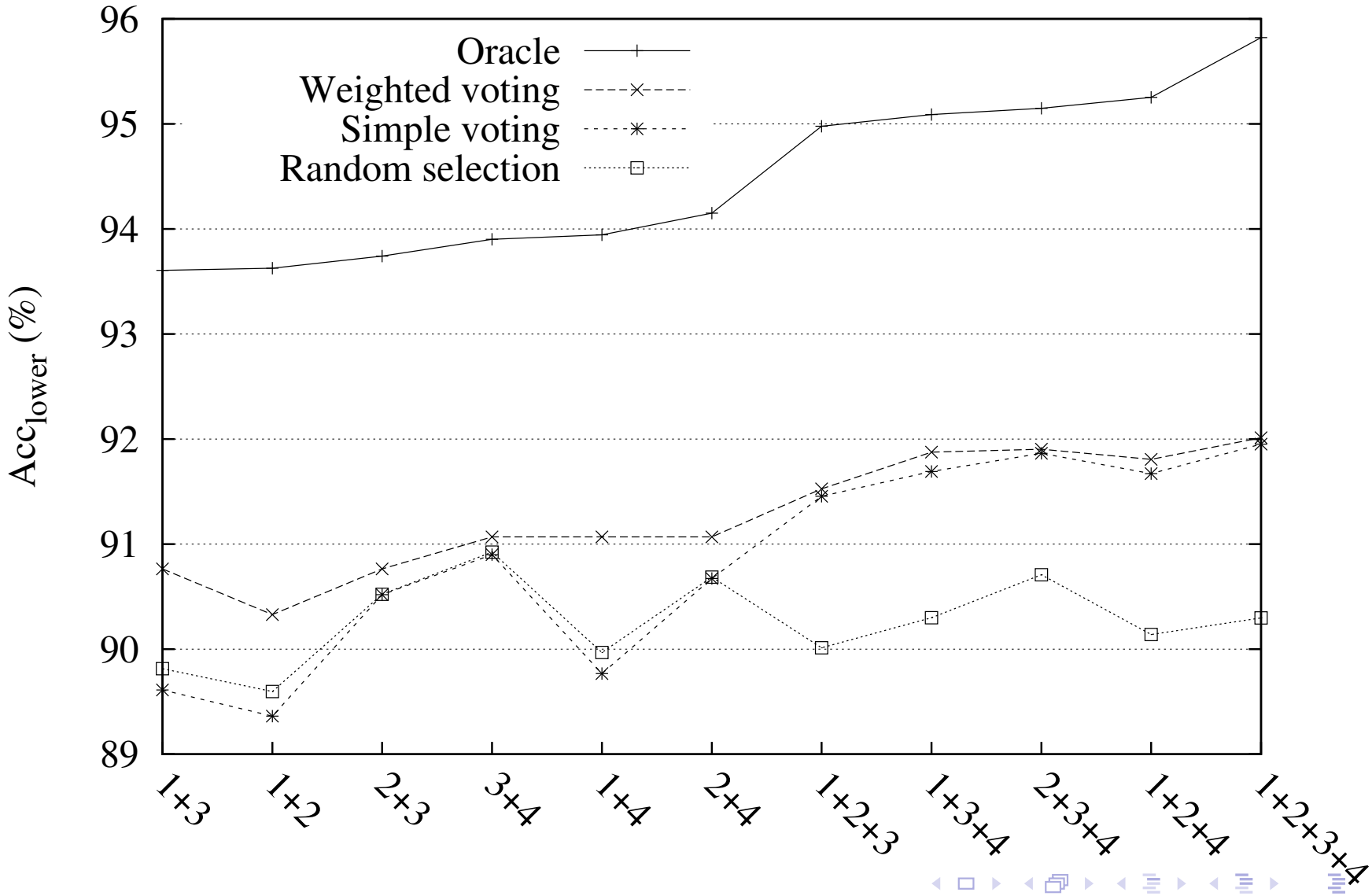
## W jaki sposób dokonać wyboru prawidłowej decyzji w ramach komitetu?

Podejście bezpośrednie – głosowanie:

- proste głosowanie – każdy tager ma jeden głos, większość wygrywa; losowy tager wybierany jest w przypadku równowagi w liczbie głosów,
- głosowanie ważone – każdy tager ma głos o wadze proporcjonalnej do jego własnej dokładności (najlepszy tager wygrywa w przypadku równowagi).

| Metoda | $Acc_{lower}$ | $Acc^{K}_{lower}$ | $Acc^{U}_{lower}$ |
|---|---|---|---|
| Proste głosowanie | 91,95% | 92,87% | 62,18% |
| Głosowanie ważone | 92,01% | 92,91% | 62,81% |

# Dokładność komitetu głosującego



# Głosowanie (2)

**Waga wynikająca z ogólnej dokładności tagera uniemożliwia podjęcie decyzji dostosowanej do niszowych przypadków**

- poszczególne tagery mogą być lepsze niż inne w specyficznych przypadkach, np. dla pewnej klasy tagów lub tokenów,
- możemy zebrać bardziej szczegółowe statystyki dotyczące działania poszczególnych tagerów na danych uczących,
- wykorzystać tę wiedzę przy podejmowaniu decyzji na temat właściwego znakowania przy testowaniu.

**Sprawdźmy jak zachowują się tagery dla poszczególnych klas gramatycznych.**

# Podział na klasy gramatyczne

| | | $Acc_{lower}$ (%) | | | |
|---|---|---|---|---|---|
| klasa | liczność | PANTERA | WMBT | WCRFT | Concraft |
| subst | 331570 | 85,21 | 86,25 | 87,36 | **88,29** |
| interp | 223542 | 99,63 | 99,97 | 99,97 | 99,97 |
| adj | 128703 | 76,53 | 81,10 | 81,56 | **82,52** |
| prep | 115818 | 97,04 | 97,28 | 97,54 | **98,05** |
| qub | 68079 | 92,98 | **93,82** | 92,91 | 92,92 |
| fin | 59458 | 98,64 | 98,70 | 98,81 | **98,94** |
| praet | 53326 | **90,90** | 88,96 | 89,80 | 89,69 |
| conj | 44840 | 95,17 | **95,41** | 94,61 | 93,96 |
| adv | 42750 | 95,31 | **95,59** | 95,29 | 94,77 |
| inf | 19213 | 98,91 | **99,20** | 99,09 | 99,14 |
| comp | 17842 | 97,26 | **97,29** | 96,84 | 96,88 |
| num | 16160 | 33,40 | 56,40 | **60,32** | 55,99 |

# Głosowanie (3)

**Uwzględnienie statystyk dotyczących dokładności w poszczególnych klasach gramatycznych.**

- głosowanie ważone dokładnością dla poszczególnych klas gramatycznych – każdy tager ma głos o wadze proporcjonalnej do jego własnej dokładności (najlepszy tager wygrywa w przypadku równowagi),
- wygrywa jeden tager, o najwyższej dokładnością dla danej klasy gramatycznej.

| Metoda | $Acc_{lower}$ | $Acc^{K}_{lower}$ | $Acc^{U}_{lower}$ |
|---|---|---|---|
| Głosowanie: dokł. ogólna | 92,0123% | 92,9128% | 62,8076% |
| Głosowanie: dokł. w ramach klasy | 92,0128% | 92,9564% | 61,4169% |
| Wygrywa najlepszy dla klasy | 89,9671% | 91,9206% | 26,5647% |

# Głosowanie (4)

## Uwzględnienie dokładności dla słów znanych i nieznanych

- oprócz dokładności dla klas gramatycznych, w wadze tagerów uwzględniamy ich dokładność w podziale na słowa znane i nieznane.

| Metoda | $Acc_{lower}$ | $Acc^{K}_{lower}$ | $Acc^{U}_{lower}$ |
|---|---|---|---|
| Głosowanie: dokł. ogólna | 92,0123% | 92,9128% | 62,8076% |
| Głosowanie: dokł. w ramach klasy + dokł. znane/nieznane | 92,0451% | 92,9513% | 62,6603% |

# Uwzględnianie kontekstu

# Uwzględnianie kontekstu

**Hipoteza: uwzględnienie informacji o kontekście tokenu może wpłynąć pozytywnie na dokładność wyboru właściwego tagera**

Założenia

- tworzymy nowy klasyfikator, który uczony jest na rezultatach działania poszczególnych tagerów na zbiorze treningowym,
- klasyfikator ma za zadanie wskazanie właściwego tagera w danym kontekście, a nie wskazanie właściwego tagu.

Wnioski wstępne

- metoda ta nie poprawi wyniku w przypadkach, gdy wszystkie tagery się mylą,
- nic nowego nie wniesie również w sytuacji, gdy wszystkie zwracają jednakowy tag.

# Nieco statystyk

**Pozostaje poza naszym zainteresowaniem:**

- wszystkie zwracają jednakowy tag: **84.83%**
  - wszystkie zwracają prawidłowy tag: **82,78%**
  - wszystkie się mylą: **2.11%**
- zwracają różne tagi, wszystkie błędne **2,07%**

**W tym obszarze możemy poprawić wynik (5,09% przypadków):**

- równowaga w głosowaniu: **2,71%**
- prawidłowy tag w mniejszości: **2,38%**

**Jako materiał treningowy możemy wykorzystać:**

- większość zwraca prawidłowy tag: **7,95%**

# Podział na klasy

**Istnieje wiele przypadków, w których więcej niż jeden tager zwróci prawidłowy wynik**

Dwie(?) możliwości sposobu określenia klas dla klasyfikatorów:

- zakodowanie każdej możliwej kombinacji tagerów jako etykiety klasy: 1, 2, 3, 4, 12, 13, 14, 23, 24, 123, 124, 234
- losowy wybór jednego z prawidłowych tagerów, aby nie zwiększać liczby kategorii: 1, 2, 3, 4
  - podczas uczenia klasyfikator widzi jedynie cztery klasy,
  - podczas ewaluacji traktujemy jako prawidłowy wybór każdego z tagerów, zwracających właściwy znacznik.

# Reprezentacja tokenów

## Jakie atrybuty wykorzystać przy uczeniu?

. . . o godzinie . . .

- znaczniki zwracane przez tagery,
  - $tag = subst : sg : loc : f$
- klasy gramatyczne i podstawowe atrybuty,
  - $pos = subst, ctag = sg$
- klasy gramatyczne i podstawowe atrybuty wcześniejszych tokenów (okno $= 3$),
  - $pos_{-1} = prep, ctag_{-1} = loc$
- pierwsza litera, pierwsze dwie litery formy ortograficznej,
  - $orth_1 = g, ctag_2 = go$

# Reprezentacja i wyniki

## Jakie atrybuty wykorzystać przy uczeniu (kont.)?

- ostatnia litera, ostatenie dwie litery formy ortograficznej,
  - $orth_1 = e, ctag_2 = ie$
- czy są wielkie litery w słowie?
  - $cap = false$,
- czy początek zdania?
  - $beg = false$.

## Wyniki na zbiorze testowym i odpowiadająca im dokładność znakowania dla pełnego korpusu

| Metoda | Acc | $Acc_{lower}$ |
|---|---|---|
| NaiveBayes | 70,34% | 91,95% |
| C4.5 | 71,58% | 92,11% |

# Podsumowanie

# Podsumowanie i dalsze prace

## Wnioski

- nawet prosty komitet głosujący zwiększa dokładność znakowania,
- wzrost dokładności jest podobny do różnicy pomiędzy najlepszym i najgorszym z testowanych tagerów!

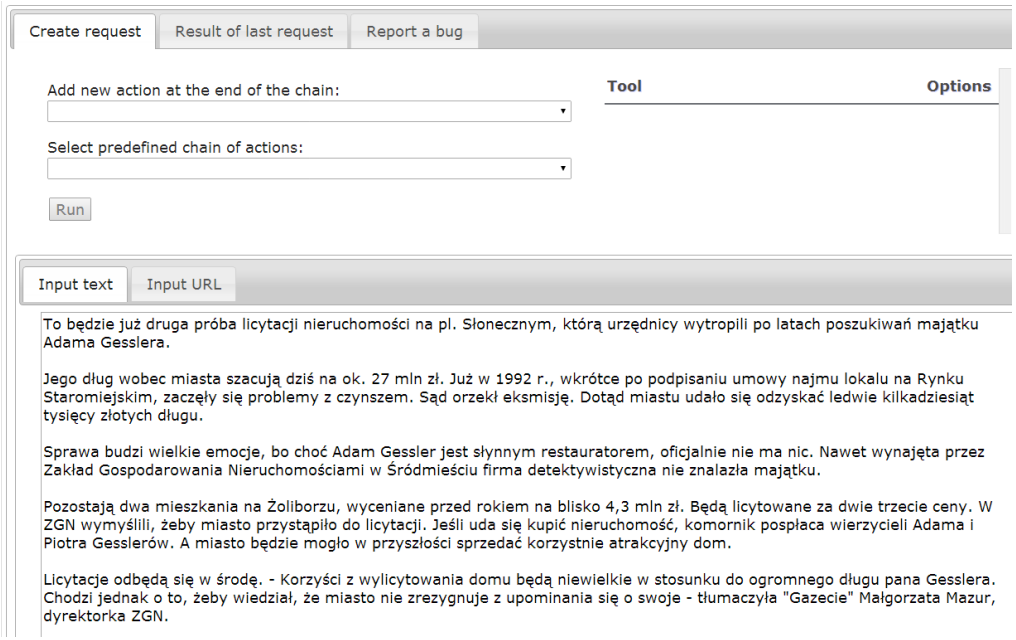
## Możliwe udoskonalenia

- uwzględnianie kontekstu:
  - zweryfikowanie większej liczby metod uczenia maszynowego i ich parametrów,
  - inne atrybuty tekstu;
- wykorzystanie większej liczby tagerów w głosowaniu:
  - niekoniecznie tagery o wyższej dokładności,
  - istotna jest różnorodność.

# Sposób udostępnienia - Multiservice?

## Udostępnienie planowane w formie usługi webowej

- brak konieczności instalowania indywidualnych tagerów,
- brak konieczności udostępniania wielu wyuczonych modeli,
- możliwość poprawiania mechanizmu w sposób transparentny dla użytkownika.



# Dziękuję za uwagę!

# Bibliografia I


Acedański, Szymon, 2010.
A morphosyntactic Brill tagger for inflectional languages.
In Advances in Natural Language Processing.

Brill, Eric and Jun Wu, 1998.
Classifier combination for improved lexical disambiguation.
In Proceedings of the 17th international conference on Computational linguistics - Volume 1, COLING '98. Stroudsburg, PA, USA: Association for Computational Linguistics.

# Bibliografia II

Śniatowski, Tomasz and Maciej Piasecki, 2012.
Combining Polish morphosyntactic taggers.
In Pascal Bouvry, Mieczysław A. Kłopotek, Franck Leprévost, Małgorzata Marciniak, Agnieszka Mykowiecka, and Henryk Rybiński (eds.), Security and Intelligent Information Systems, volume 7053 of LNCS. Springer-Verlag.

Radziszewski, Adam, 2013.
A tiered CRF tagger for Polish.
In R. Bembenik, Ł. Skonieczny, H. Rybiński, M. Kryszkiewicz, and M. Niezgódka (eds.), Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions. Springer Verlag.

# Bibliografia III



Radziszewski, Adam and Szymon Acedański, 2012.
Taggers gonna tag: an argument against evaluating disambiguation capacities of morphosyntactic taggers.
In Proceedings of TSD 2012, LNCS. Springer-Verlag.


Radziszewski, Adam and Tomasz Śniatowski, 2011a.
A Memory-Based Tagger for Polish.
In Proceedings of the LTC 2011.


van Halteren, Hans, Walter Daelemans, and Jakub Zavrel, 2001.
Improving accuracy in word class tagging through the combination of machine learning systems.
Comput. Linguist., 27(2):199–229.

# Bibliografia IV

Waszczuk, Jakub, 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). Mumbai, India.