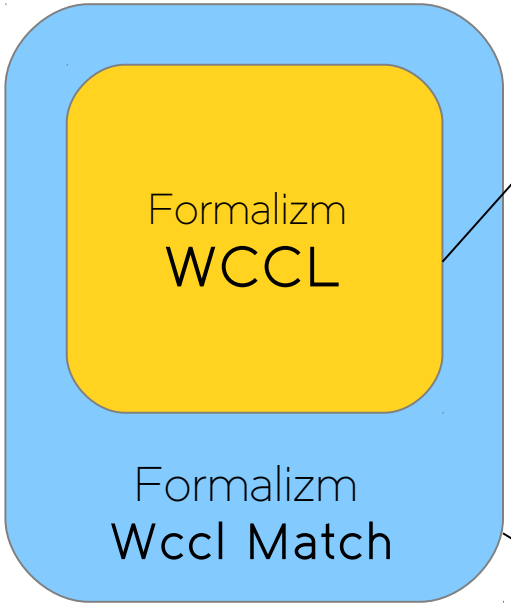
Zaawansowane narzędzie do analizy korpusu w oparciu o reguły

Michał Marcińczuk
Politechnika Wrocławska
Katedra Inteligencji Obliczeniowej
Grupa Naukowa G4.19
michal.marcinczuk@pwr.edu.pl
2015-04-13

WCCL i Wccl Match



WCCL (Wrocław Corpus Constraint Language) to formalizm pozwalający na pisanie **wyrażeń funkcyjnych wartościowanych na tekście oznakowanym morfo-syntaktycznie.** Takie wyrażenie mogą być użyte bezpośrednio jako cechy dla algorytmów maszynowego uczenia przy tworzeniu systemów przetwarzania języka naturalnego. Formalizm powstał z myślą o języku polskim.

Wccl Match to formalizm pozwalający na pisanie reguł dopasowania sekwencji tokenów i/lub anotacji w obrębie zdania. Dopasowanie pozwala na naniesienie nowych anotacji na wskazane fragmenty oraz usunięcie istniejących anotacji.

Składnia reguł Wccl Match

» sekcja match

```
match_rules (
  apply(
    match(
      is("person_nam"),
      in("w", base[0]),
      is("city_nam")
    ),
    cond(
      not(
        annsub(:1, "possessive")
      )
    ),
    actions(
      mark(M, "person_in_city")
    )
  )
)
```

Wewnątrz **match_rules** znajduje się lista reguł (**apply**). Reguły wykonywane są sekwencyjne w kolejności wystąpienia w match_rules.

Składnia reguł Wccl Match

» sekcja match

```
match_rules (
  apply(
    match(
      is("person_nam"),
      in("w", base[0]),
      is("city_nam")
    ),
    cond(
      not(
        annsub(:1, "possessive")
      )
    ),
    actions(
      mark(M, "person_in_city")
    )
  )
)
```

Sekcja **match** opisuje poszukiwaną sekwencję tokenów i/lub anotacji.

Operatory dopasowania

Operatory w sekcji match pozwalają na dopasowanie:

- Tokenu
  - po **wartościach morfo-syntaktycznych tokenu** (forma bazowa, przypadek, liczba, rodzaj, rodzaj, itd.)
  - po **anotacjach semantycznych** (test sprawdzający, czy token jest częścią anotacji, rozpoczyna lub kończy anotację określonego typu)
  - możliwość łączenia warunków przy pomocy **operatorów i/lub** oraz **negacji**.
- Anotacji
  - dopasowanie anotacji określonego typu (operator is)
- Wektora dopasowań
  - powtarzające się sekwencje tokenów/anotacji,
  - opcjonalne dopasowanie,
  - dopasowanie jednej z możliwych sekwencji (pierwsze lub najdłuższe).

Operatory dopasowania tokenów

- Warunki dla wartości atrybutu
  - Zgodność wartości (`equal`)
  - Zawieranie się zbiorów wartości (`in`)
  - Przecięcie zbiorów wartości (`inter`)
- Dopasowanie formy tekstowej/bazowej wyrażeniem regularnym (`regex`)
- Warunki sprawdzające oznakowanie anotacją
  - Token jest częścią anotacji (`isannpart`)
  - Token rozpoczyna anotację (`isannbeg`)
  - Token kończy anotację (`isannend`)

Atrybuty tokenów

- Forma tekstowa (`orth`)
- Forma bazowa (`base`)
- Atrybuty morfologiczne (tagset NKJP)
  - Klasa gramatyczna (`class`)
  - Liczba (`nmb`)
  - Przypadek (`cas`)
  - Rodzaj (`gnd`)
  - Osoba (`per`)
  - Stopień (`deg`)
  - Aspekt (`asp`)
  - Negacja (`ngt`)
  - Akomodacyjność (`acm`)
  - Akcentowość (`acn`)
  - Poprzyikmowość (`ppr`)
  - Agrultynacyjność (`agg`)
  - Wokaliczność (`vcl`)
  - Wymaganie kropki (`dot`)

http://nkjp.pl/poliqarp/help/ense2.html

Łączenie operatorów i negacja

- Koniunkcja operatorów (and)

  np., rzeczownik w mianowniku:

```
and(
  equal(class[0], subst),
  equal(cas[0], nom)
)
```

- Alternatywa operatorów (or)

  np., przymiotnik lub dowolny token w mianowniku:

```
or(
  equal(class[0], adj),
  equal(cas[0], nom)
)
```

- Negacja (not)

  np., dowolny token nie będący rzeczownikiem

```
not(equal(class[0], subst))
```

Operatory dopasowania wektorów

- Dopasowanie sekwencji tokenów o określonej formie tekstowej (`text`)
- Opcjonalne dopasowanie (`optional`)
- Wielokrotne dopasowanie (`repeat`)
- Dopasowanie najdłuższego wektora z listy możliwych (`longest`),
- Dopasowanie pierwszego pasującego wektora (`oneof`)

Składnia reguł Wccl Match

» sekcja cond

```
match_rules (
  apply(
    match(
      is("person_nam"),
      in("w", base[0]),
      is("city_nam")
    ),
    cond(
      not(
        annsub(:1, "possessive")
      )
    ),
    actions(
      mark(M, "person_in_city")
    )
  )
)
```

Sekcja **cond** zawiera dodatkowe warunki, jakie muszą spełnić elementy dopasowanej sekwencji.

Dodatkowe warunki

Operatory w sekcji `conds` pozwalają na:

- sprawdzenie wartości atrybutu tokenu (`in`, `inter`, `equal`),
- porównanie wartości atrybutów dwóch tokenów (`in`, `inter`, `equal`)
- sprawdzenie oznaczenia tokenu anotacją semantyczną określonej kategorii (`ann`, `annsub`),
- sprawdzenie zgodności przypadku, liczby i rodzaju sekwencji tokenów (`agr`, `agrpp`),
- sprawdzenie czy pozycja znajduje się w obrębie zdania (`inside`, `outside`),
- łączenia i negacji operatorów (`and`, `or`, `not`),
- konstrukcja warunkowa (`if`),
- szukanie tokenu względem określonej pozycji (`skip`).

Referencja do dopasowania

Referencja do dopasowanych elementów w sekcji `cond` i `actions`:

- symbol **`M`** reprezentuje cały wektor dopasowania,
- symbol **`:n`** reprezentuje n-ty element dopasowania,
  np. **`:2`** to element dopasowany przez drugi operator w `match`,
- dla dopasowania typu wektor możliwe jest odwołanie się do elementów zagnieżdżonych poprzez indeks wewnątrz wektora,
  np. `:2:1`
- wyłuskanie pozycji tokenu (dla operatorów wymagających pozycji tokenu):
  - `first(dopasowanie)` – pozycja pierwszego tokenu dopasowania, np. `first(M)`,
  - `last(dopasowanie)` – pozycja ostatniego tokenu dopasowania, np. `last(M)`,

Składnia reguł Wccl Match

» sekcja actions


```
match_rules (
  apply(
    match(
      is("person_nam"),
      in("w", base[0]),
      is("city_nam")
    ),
    cond(
      not(
        annsub(:1, "possessive")
      )
    ),
    actions(
      mark(M, "person_in_city")
    )
  )
)
```

Sekcja **actions** zawiera listę akcji, które zostaną wykonane na dopasowanej sekwencji.

Akcje

Operatory w sekcji actions pozwalają na:

- oznaczenie wskazanego fragmentu anotacją określonego typu,
- oznaczenie wskazanego fragmentu anotacją określonego typu z wymuszeniem nadpisania istniejącej anotacji,
- usunięcie dopasowanej anotacji,
- przypisanie wartości do tokenu.

Zastosowanie Wccl Match

1) Do wykrywania i kategoryzacji jednostek identyfikacyjnych.
2) Do wykrywania i kategoryzacji wyrażeń temporalnych.
3) W systemie odpowiedzi na pytania do kategoryzacji pytań i analizy ich struktury.
4) Do wykrywania relacji semantycznych między jednostkami identyfikacyjnymi.

Jednostki identyfikacyjne

```
apply(
  match(
    equal(base[0], "stolica"),
    is("dict_country_name"),
    equal(base[0], "być"),
    is("dict_city_name")
  ),
  actions(
    mark(:2,
      "nam_loc_gpe_country"),
    mark(:4,
      "nam_loc_gpe_city")
  )
)
```

- Reguła dopasowuje wyrażenie „stolicą [nazwa] jest [nazwa]" i znakuje drugi element jako nazwę państwa oraz czwarty element jako nazwę miasta.

Analiza pytań


```
apply(
  match(
    optional(in({prep}, class[0])) ,
    in(["ile"], base[0]) ,
    optional(
      and(
        in({subst}, class[0]) ,
        in({gen}, cas[0])
      )
  )),
  cond(
    if(skip(first(M), $L,
            inter(class[$L],
                  {interp, qub, adv, interj}),
                  -1),
       not(inter(class[$L],
                 {praet,fin,bedzie,imps,impt,
                  inf,pant,pcon,aglt,winien})),
       True
  )),
  actions(
    mark(M, M, M:2, "qphrase"),
    setprop(M:2, "question", "yes")
  )
)
```

Opcjonalny przyimek.

Opcjonalny rzeczownik w dopełniaczu..

Sprawdzenie, czy przed dopasowaniem znajduje się token nie będący znakiem interpunkcyjnym, kublikiem, przysłówkiem bądź wykrzyknikiem.

Jeżeli jest to sprawdzamy jego klasę gramatyczną. Dla określonych klas zwracamy False, a wpp True.

Jeżeli nie ma, to jest to początek zdania więc zwracamy True.

Narzędzia wykorzystujące język reguł Wccl Match

Wccl Match Tester

Narzędzie do testowania reguł Wccl Match na znakowanym korpusie.

Wymagania:

- Otagowany i anotowany korpus w formacie ccl,
- Lista kategorii anotacji.

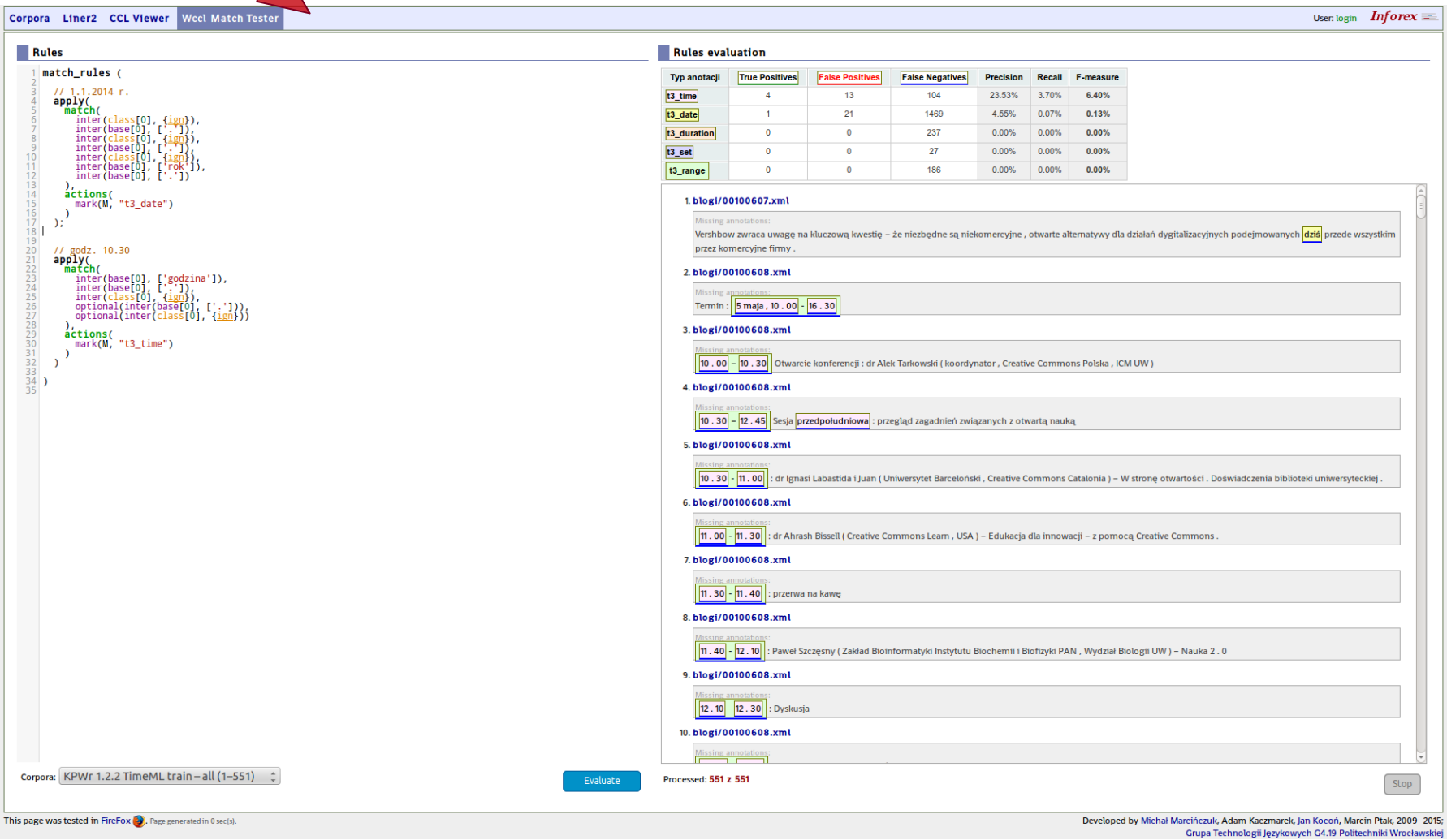
Funkcje:

- Kolorowanie składni reguł Wccl Match,
- Statystyki liczone w czasie rzeczywistym:
  - Liczba rozpoznanych, nierozpoznanych i błędnie rozpoznanych anotacji dla określonych kategorii.
  - Precyzja, kompletność i średnia harmoniczna.
- Podgląd rozpoznanych i nierozpoznanych anotacji oraz anotacji pomocniczych.
- Możliwość filtrowania anotacji.

Demo:

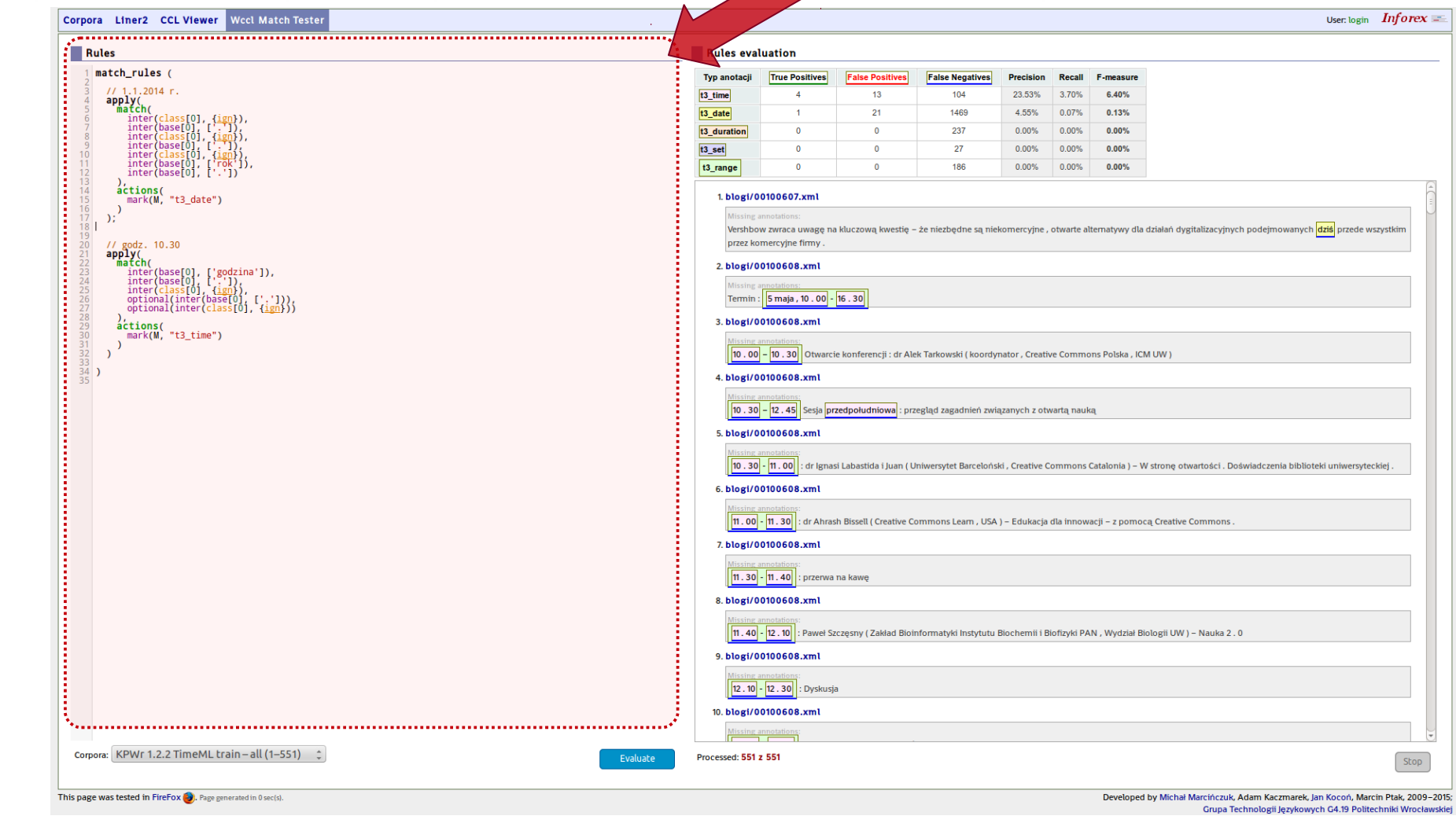
Korpus KPWr 1.2.2 znakowany wyrażeniami temporalnymi.

Wccl Match Tester



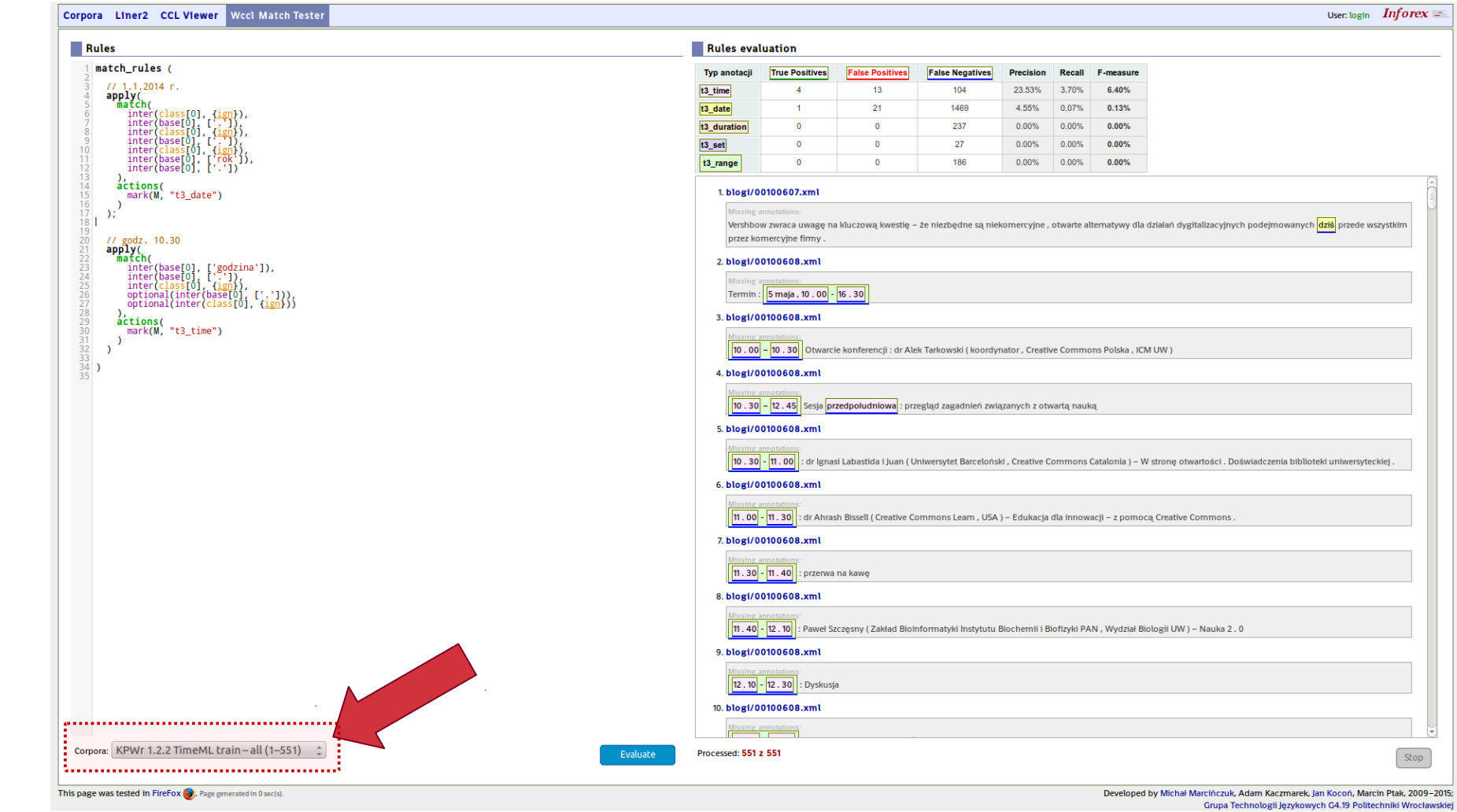
Wccl Match Tester

» pole do edycji reguł



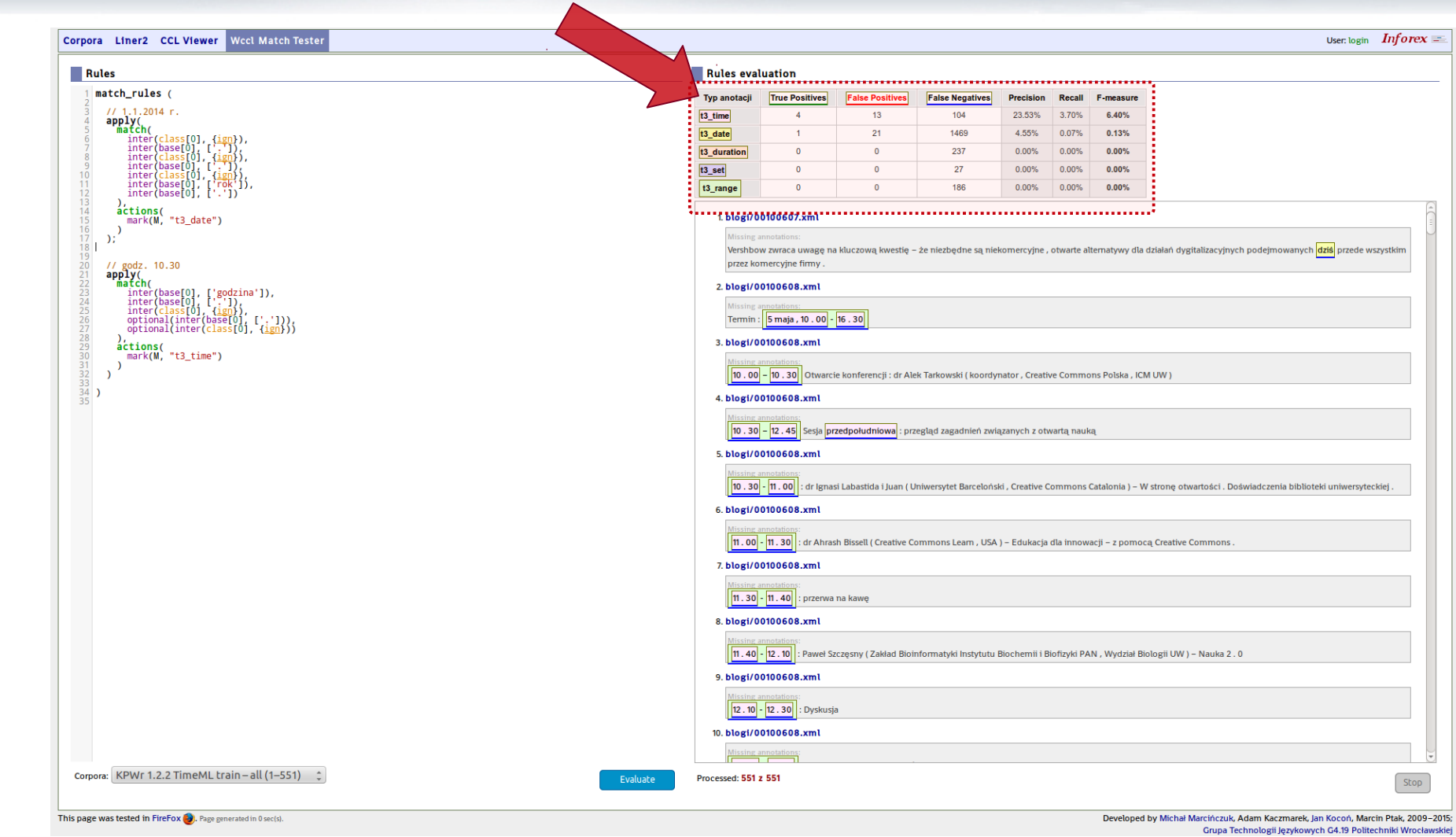
Wccl Match Tester

» wybór korpusu



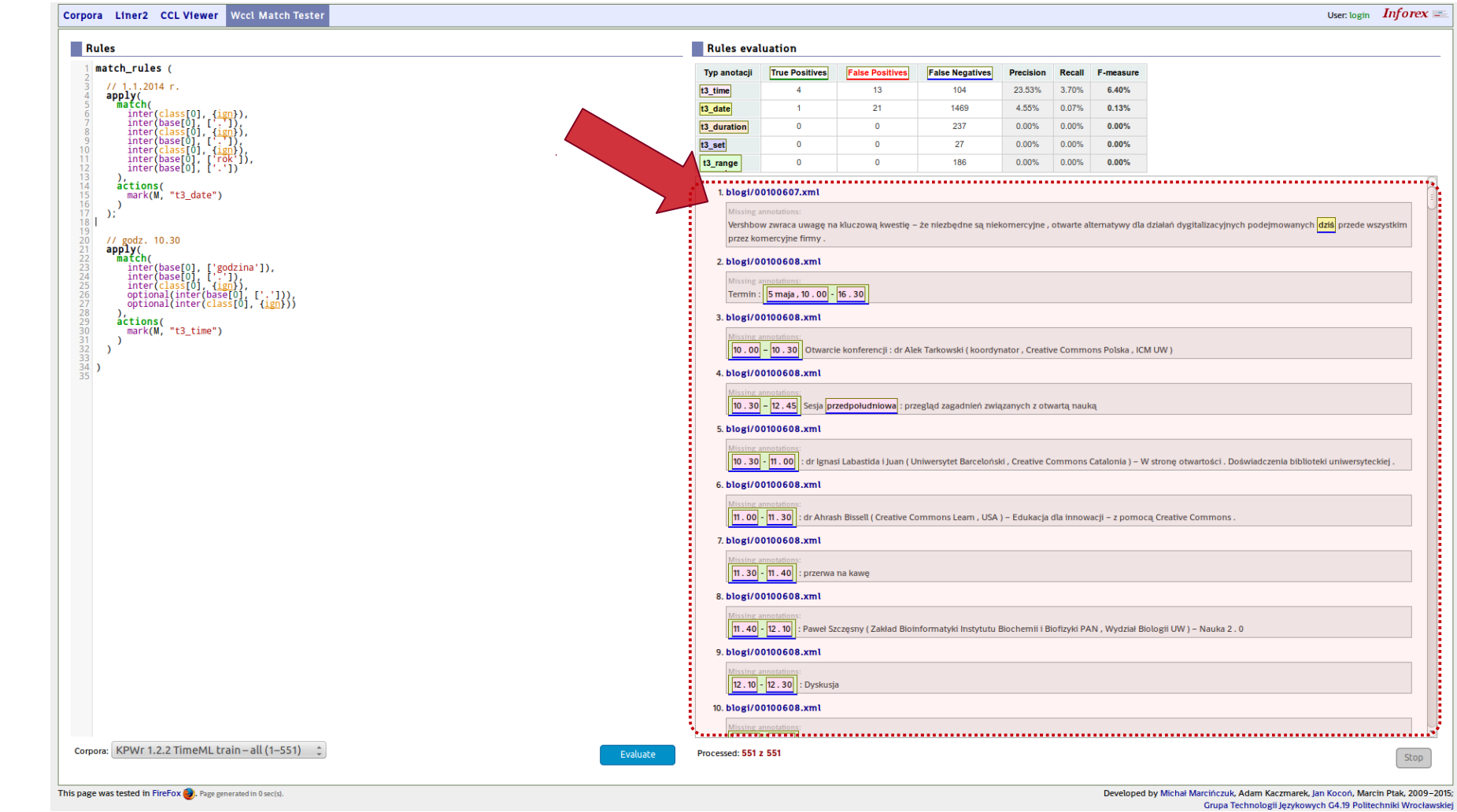
Wccl Match Tester

» ocena reguł



Wccl Match Tester

» podgląd anotacji



Wccl Match (Corpus)

Narzędzie do wyszukiwania fraz przy użyciu reguł Wccl Match.

Wymagania:

Korpus zaimportowany z DSpace.

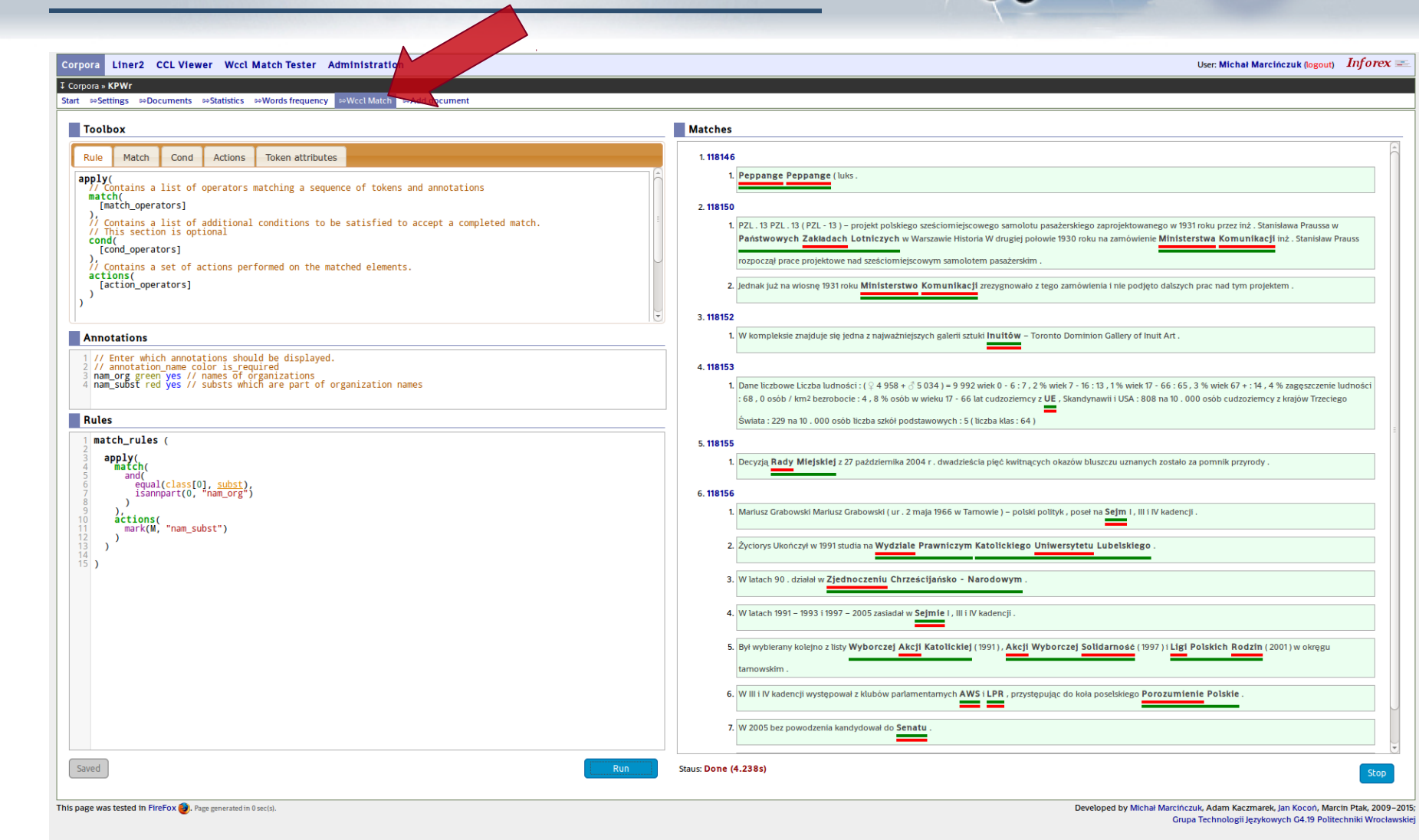
Funkcje:

- Przybornik z operatorami Wccl Match,
- Kolorowanie składni reguł Wccl Match,
- Opis anotacji do wyświetlenia.
- Podgląd rozpoznanych anotacji.

Demo:

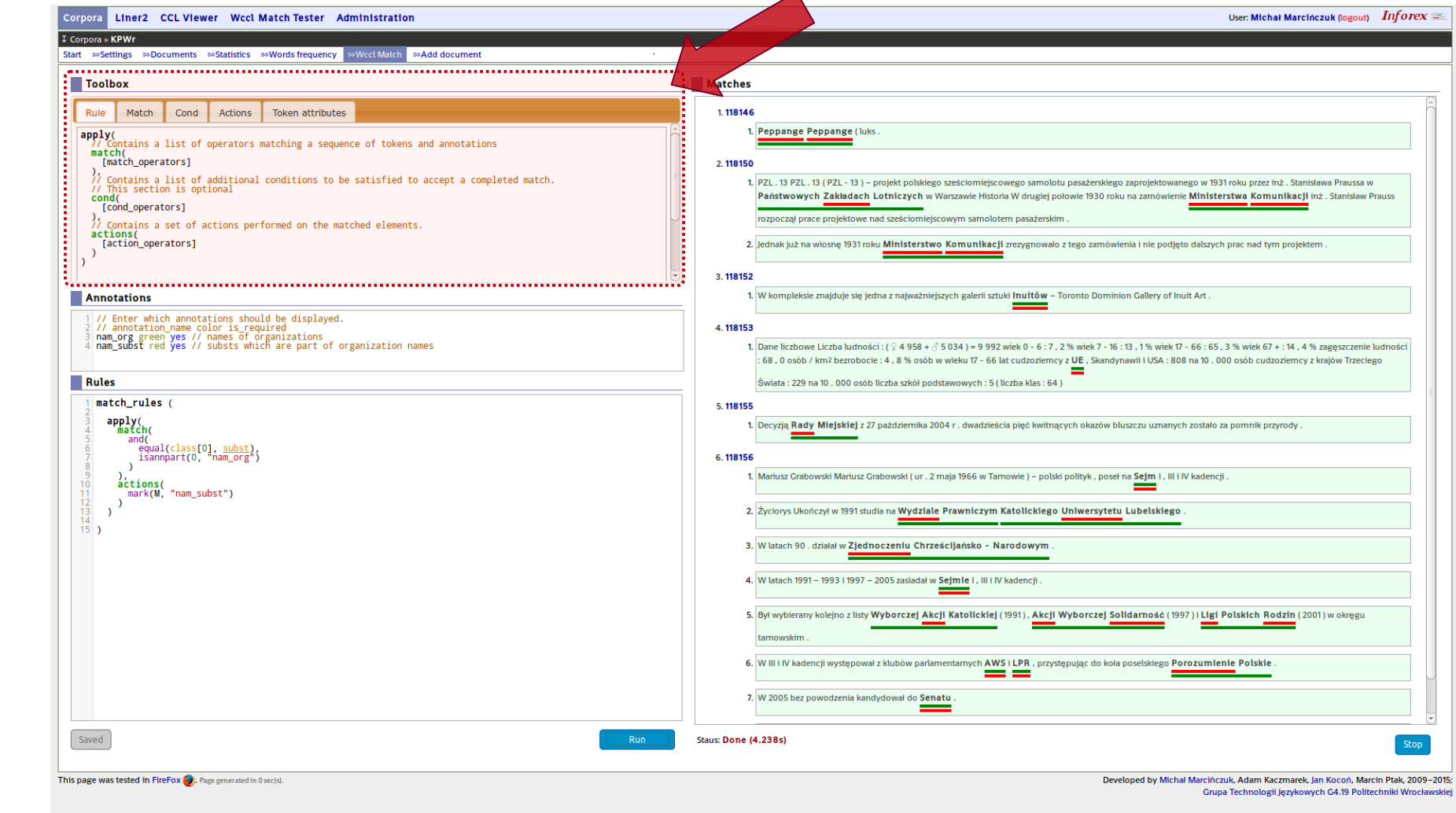
Korpus zaimportowany z DSpace.

Wccl Match (Corpus)

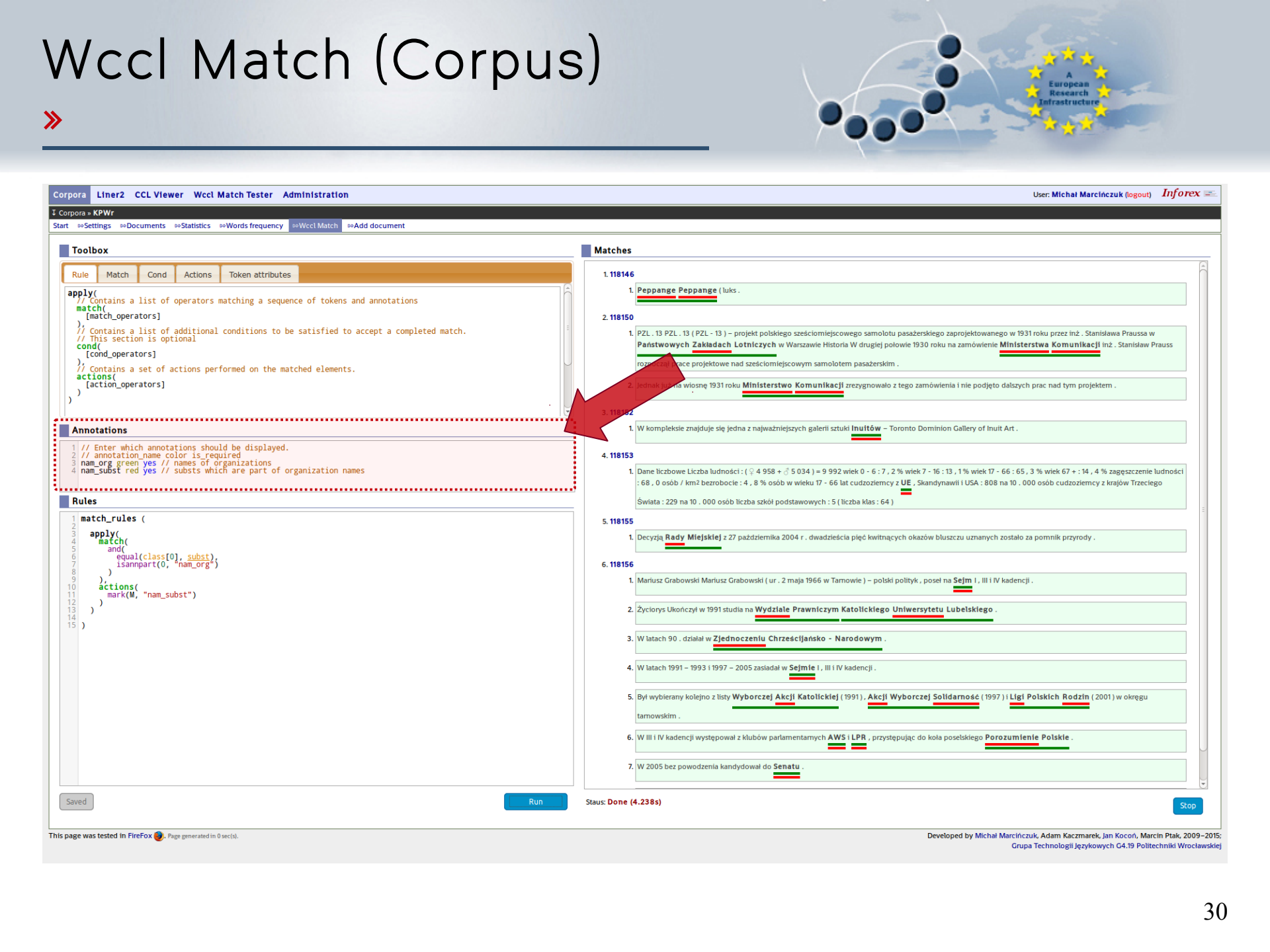


Wccl Match (Corpus)

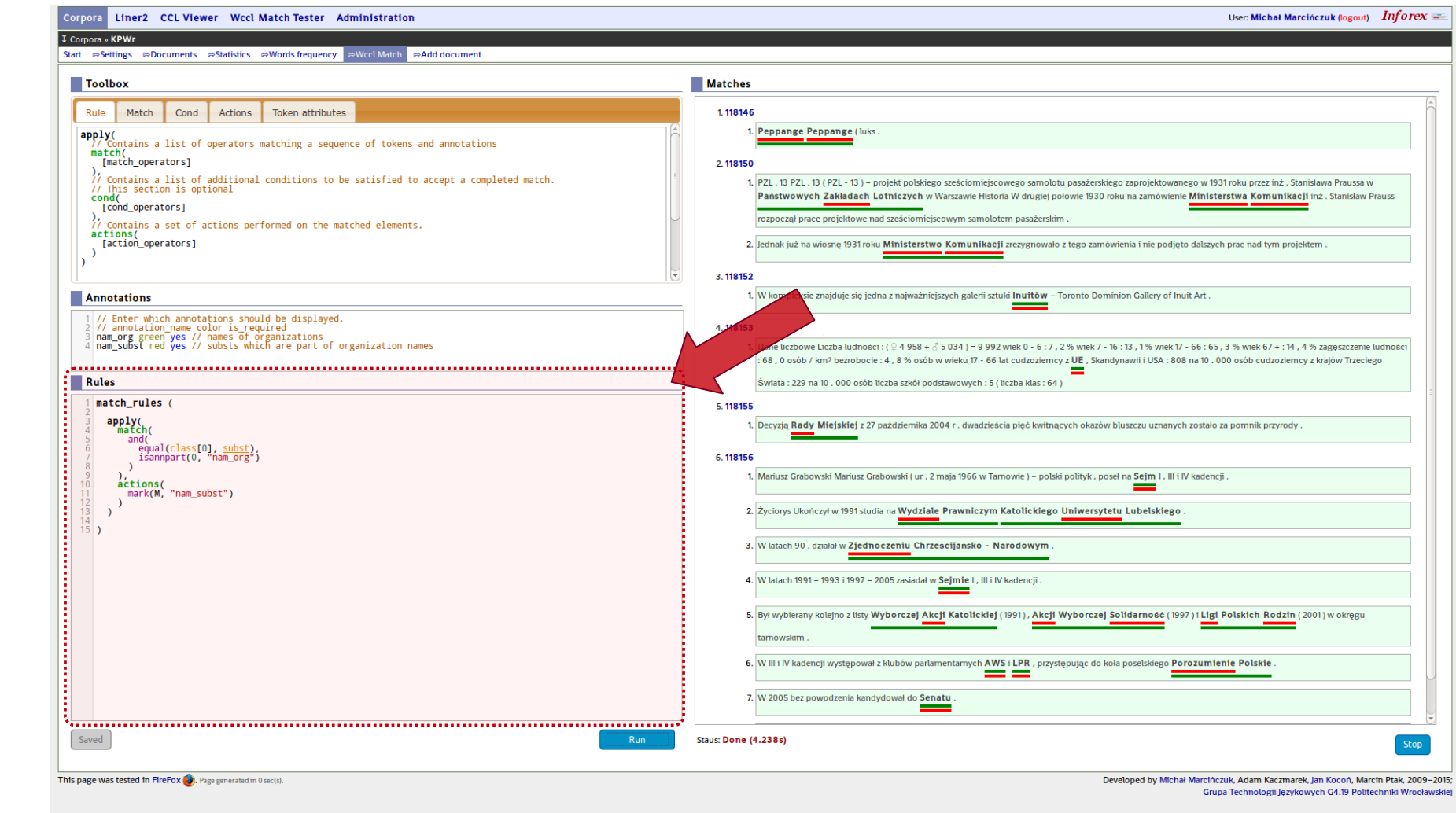
» Przybornik



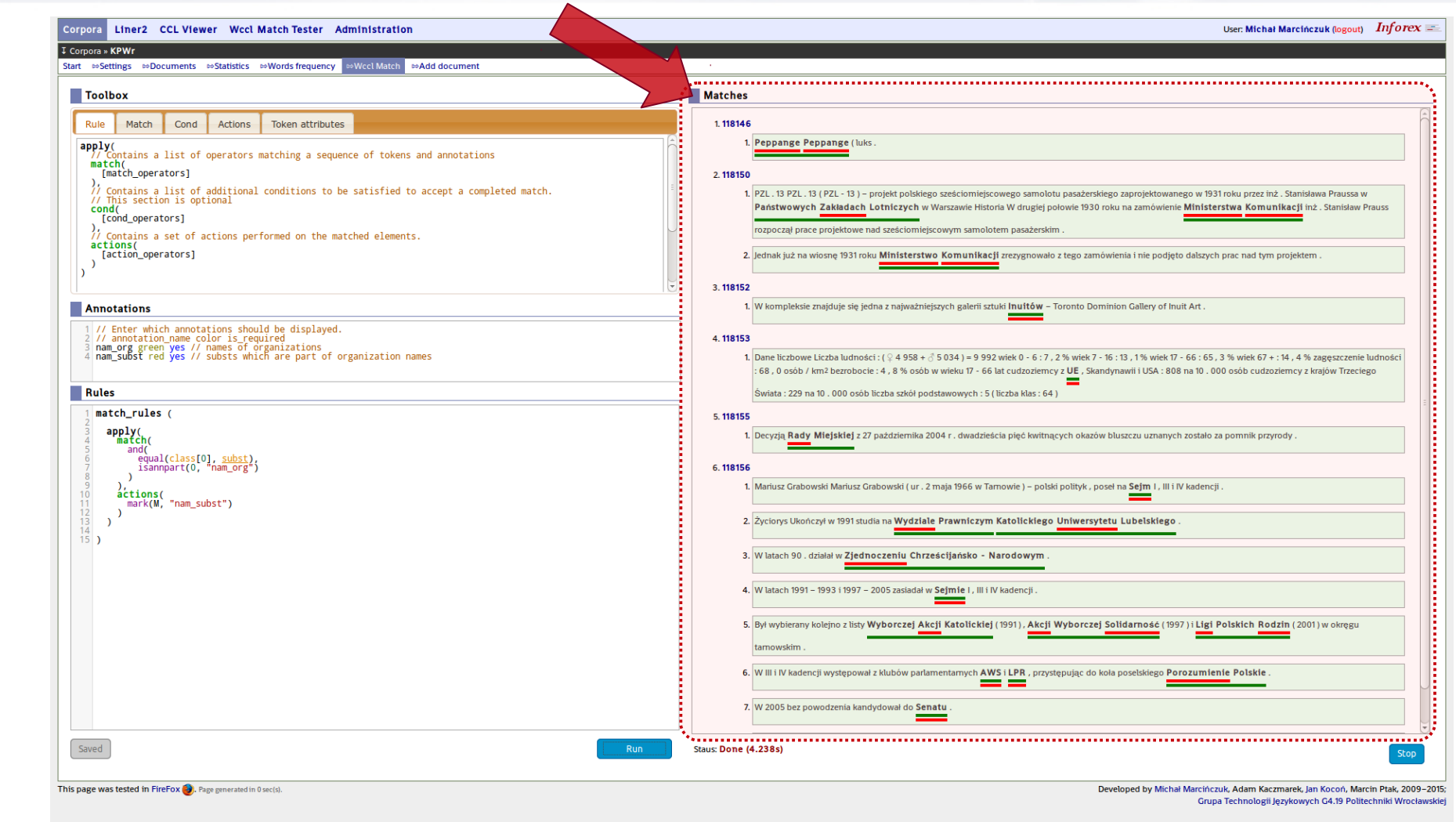
Wccl Match (Corpus)



Wccl Match (Corpus)



Wccl Match (Corpus)



Pytania i odpowiedzi

Dziękuję bardzo za uwagę