Instrukcja korzystania z wyszukiwarki korpusu tekstów polskich z lat 1830-1918

wersja robocza

Spis treści

Wprowadzenie

Niniejszy dokument powstał w oparciu o Ściągawkę do Narodowego Korpusu Języka Polskiego, której autorem jest Adam Przepiórkowski i którą następnie poprawiali i rozszerzali Jakub Wilk i Aleksander Buczyński. *Ściągawka* stanowi instrukcję użytkowania wyszukiwarki Poliqarp z Narodowym Korpusem Języka Polskiego. Jej pełna wersja znajduje się w repozytorium wyszukiwarki Poliqarp.

Niniejszy dokument opisuje sposób użytkowania wyszukiwarki MTAS, niepowiązanej z Poliqarpem, ale wykorzystującej podobny język zapytań znany pod nazwą *Corpus Query Language* (CQL). Modyfikacje wprowadzone do pierwotnej wersji instrukcji uwzględniają różnice w języku zapytań oraz specyfikę korzystania z korpusu historycznego. Za zgodą wszystkich wyżej wymienionych autorów niniejsza wersja dokumentu zostaje udostępniona na zasadach licencji Creative Commons BY-SA.

1. Segmentacja

(Autor: Adam Przepiórkowski)

Znaczniki morfosyntaktyczne, tzw. tagi, przypisane są segmentom (tokenom, w przybliżeniu słowom). Segmenty nie są dłuższe niż słowa ortograficzne (słowa ‘od spacji do spacji’), ale w niektórych wypadkach segmenty mogą być krótsze niż takie słowa:

- Jako odrębne segmenty traktowane są formy aglutynacyjne leksemu być, a zatem następujące słowa reprezentują po dwa segmenty: *[łgał][eś]*, *[długo][śmy]*, *[tak][em]*.

- Za odrębne segmenty uznane są partykuły *by*, *-ż(e)* i *-li*, a zatem następujące słowa reprezentują po kilka segmentów: *[przyszedł][by]*, *[napisała][by][m]*, *[chodź][że]*, *[potrzebował][że][by][ś]*, *[znasz][li]*.
- Odrębnym segmentem jest poprzyimkowa nieakcentowana forma zaimka *-ń*: *[do][ń]*, *[ze][ń]*.
- Dzielone na segmenty są niektóre słowa zawierające łącznik, a mianowicie:
  — słowa typu *[polsko][-][niemiecki]*,
  — podwójne nazwiska, np. *[Kowalska][-][Nowakowska]*,

Nie są natomiast dzielone skrótowce zawierające łącznik sygnalizujący odmianę, np. *PRL-u*.

Dzielone na segmenty są także występujące na końcu zdania formy kończące się kropką, np. skróty typu *itd.*, *itp.*, liczby pisane cyframi w znaczeniu porządkowym i inicjały, np. *[itp][.]*, *[George] [W][.]* itp. Dzielenie form z kropką kończących zdanie jest uzasadnione podwójną rolą kropki w takiej pozycji: jest ona częścią formy i jednocześnie sygnalizuje koniec zdania. W wypadku, gdy takie formy nie występują na końcu zdania, są one uznawane za pojedyncze segmenty.

2. Zestaw znaczników morfosyntaktycznych

(Autor: Adam Przepiórkowski, Witold Kieraś)

Każdy znacznik morfosyntaktyczny jest ciągiem wartości rozdzielonych dwukropkami, np.: *subst:sg:nom:m1* dla segmentu *chłopiec*. Pierwsza wartość, np. *subst*, określa klasę gramatyczną (por. p. 2.2), następne zaś, np. *sg*, *nom* i *m1* wartości odpowiednich dla tej klasy kategorii gramatycznych (por. p 2.1).

2.1. Kategorie gramatyczne

Tabela 2 przedstawia repertuar kategorii gramatycznych używanych w korpusie tekstów polskich opublikowanych w latach 1830-1918. Repertuar kategorii oparty jest w znaczniej mierze na tagsecie stosowanym w analizatorze morfologicznym Morfeusz, uwzględnia jednocześnie kilka modyfikacji inspirowanych tagsetem Elektronicznego korpusu tekstów polskich z XVII i XVIII w. (do 1772 r.). Do tych modyfikacji należą:

- wyróżnienie (w oparciu o kryteria semantyczne) oddzielnych klas liczebników przymiotnikowych (*adjnum*) i przysłówkowych (*advnum*),
- oznaczenie użyć czasownika BYĆ w funkcji słowa posiłkowego czasu złożonych: przyszłego (*fut*) i zaprzeszłego (*plusq*),
- ograniczenie klasy przymiotników poprzyimkowych *adjp* wyłącznie form dopełniacza (np. *(z) wolna*) i celownika (np. *(po) polsku*) dawnej odmiany niezłożonej przymiotników oraz dodanie im wartości przypadka.

2.2. Klasy gramatyczne

Zasięg tradycyjnych części mowy, takich jak czasownik, rzeczownik, liczebnik czy zaimek, jest nieostry i przez to kontrowersyjny: czy tzw. odsłowniki, tj. formy typu *picie* i *palenie*, to czasowniki (posiadają kategorię aspektu, są regularnie powiązane z formami czasownikowymi typu *pić* i *palić*), czy też rzeczowniki (odmieniają się przez przypadek, posiadają słownikową kategorię rodzaju)?, czy *piąty* to liczebnik (na to wskazuje semantyka), czy też przymiotnik (na to wskazuje odmiana)?, czy *taki* to zaimek (semantyka), czy przymiotnik (odmiana)?

W korpusie tekstu z lat 1830-1918 klasy gramatyczne rozumiane są morfosyntaktycznie są one oparte na pojęciu fleksemu, będącym pojęciem węższym od terminu leksem.

Tabela 3.6 zawiera przybliżoną charakterystykę morfosyntaktyczną wszystkich klas fleksyjnych przyjmowanych w niniejszym tagsecie. Symbol $\oplus$ oznacza, że dla danej klasy fleksyjnej dana kategoria gramatyczna jest morfologiczna (fleksemy należące to tej klasy zwykle „odmieniają się" przez tę kategorię), zaś symbol $\odot$ oznacza, że dana kategoria jest słownikowa (wszystkie formy dowolnego fleksemu należącego do tej klasy mają tę samą wartość tej kategorii, choć mogą to być różne wartości dla różnych fleksemów, jak w wypadku rodzaju rzeczowników).

Tabela 3.6 zawiera informacje o formach podstawowych dla poszczególnych klas fleksyjnych, a także skróty nazw klas fleksyjnych używane w korpusie.

3. Język zapytań

(Autor: Adam Przepiórkowski, Jakub Wilk, Witold Kieraś)

Składnia zapytań w programie MTAS została oparta na języku zapytań o nazwie Corpus Query Language (CQL), wykorzystywanym w wielu innych tego typu programach, m.in. w programie Sketch Engine, ale też w znanym z NKJP Poliqarpie. Należy jednak zwrócić uwagę na drobne różnice, ponieważ mogą one wpływać na poprawność formułowanych zapytań. Niniejszy rozdział omawia składnię zapytań wyszukiwarki MTAS i ilustruje ją wieloma przykładami.

MTAS jest uniwersalną wyszukiwarką pozwalającą na przeszukiwanie korpusów zawierających wiele warstw anotacyjnych, np. warstwę morfosyntaktyczną, warstwę składniową, warstwę nazw własnych, warstwę sensu słów itp. Niniejsza instrukcja dotyczy przeszukiwania korpusu tekstów polskich publikowanych w latach 1830-1918, który zawiera jedynie dwie warstwy tekstowe (transliterowaną i transkrybowaną) oraz warstwę znakowania morfosyntaktycznego, dlatego instrukcja ogranicza się do zawartości korpusu i nie uwzględnia możliwości wyszukiwarki zastosowanej do innych korpusów. Nie należy jej zatem traktować jako ogólnej instrukcji użytkowania wyszukiwarki MTAS. Podstawowa dokumentacja wyszukiwarki znajduje się na jej stronie internetowej.

3.1. Zapytania o segmenty

Podstawową jednostką wyszukiwaną w korpusie jest segment. Segmenty w zapytaniach są ograniczone nawiasami kwadratowymi, wewnątrz których można określać konkretne cechy, które segment ma spełniać. W najprostszym przypadku jest to kształt tekstowy (napis). W wypadku korpusu historycznego kształt jest jednak określony w dwóch warstwach — transliterowanej (oryginalnej) i transkrybowanej (uwspółcześnionej). Obu warstwom odpowiadają odpowiednio atrybuty `translit` i `orth`. Jednocześnie w wypadku zapytań o kształt segmentów w warstwie uwspółcześnionej można pominąć nawiasy kwadratowe oraz nazwę atrybutu. Zatem poniższe zapytanie o dwa sąsiadujące ze sobą segmenty:

```
[orth="komisja"][orth="szkolna"]
```

można zadać również w prostszy sposób:

```
komisja szkolna
```

Analogicznie wyglądają zapytania o kształt segmentu w warstwie transliterowanej:

```
[translit="komisya"]
```

Warunki określające cechy segmentu można łączyć za pomocą operatora `&`. W szczególności można łączyć warunki dotyczące obu warstw tekstowych, np.:

```
[orth="komisja" & translit="komisya"]
```

Domyślnie rozróżniana jest kasztowość (wielkość) liter, a zatem poniższe dwa zapytania dadzą różne wyniki:

- `przyszedł`
- `Przyszedł`

W zapytaniach o segmenty mogą wystąpić standardowe wyrażenia regularne wykorzystujące następujące znaki specjalne: `?`, `*`, `+`, `.`, `,`, `|`, `,` , `[`, `]`, `(`, `)` oraz liczby naturalne pisane cyframi arabskimi, np. `0` czy `21`. Ponieważ formalny opis wyrażeń regularnych wykracza poza ramy niniejszej publikacji, ograniczymy się tutaj do kilku przykładów, które powinny pozwolić użytkownikowi na szybkie przyswojenie składni i znaczenia takich wyrażeń. W przykładach odwołujemy się zarówno do warstwy transliterowanej, jak i do transkrybowanej — wyrażeń regularnych można używać w obydwu, choć należy pamiętać, że przeszukując za ich pomocą korpus dwuwarstwowy, można uzyskać nieco inne wyniki w zależności od przeszukiwanej warstwy.

1. `[translit="(komisja|komisya)"]`
   znak `|` oznacza alternatywę dwóch wyrażeń (całość należy dodatkowo ująć w nawiasy okrągłe), a zatem zapytanie to może zostać użyte do znalezienia wszystkich wystąpień segmentów, które w warstwie transliteracyjnej mają postać *komisja* lub *komisya*,
2. `[translit="komis[yj]a"]`
   nawiasy kwadratowe oznaczają alternatywę znaków, a zatem zapytanie to może zostać użyte do znalezienia tych segmentów, których przedostatni znak to *y* lub *j*, poprzedzony ciągiem znaków postaci *komis* i po którym następuje znak *a*, tj. zapytanie to jest równoważne poprzedniemu,
3. `[translit="komm?isja"]`
   znak zapytania oznacza opcjonalność znaku (tutaj drugiego *m*) lub ujętego w nawiasy okrągłe wyrażenia bezpośrednio poprzedzającego znak `?`, a zatem w wyniku zadania tego zapytania znalezione zostaną segmenty *komisja* i *kommisja*,
4. `[orth="bez."]`
   kropka oznacza dowolny znak, a zatem wynikiem tego zapytania będą segmenty *beza*, *bezy*, *bezą* itp., ale nie *bez* czy *bezami*,
5. `[orth="bez.?"]`
   bez, *beza*, *bezy*, *bezą* itp., ale nie *bezami*,
6. `[orth=".z.z."]`
   segmenty pięcioznakowe, w których 2. i 4. znak to *z* (np. *czczą* i *rzezi*),
7. `[orth=".z.z..?"]`
   segmenty składające się z pięciu lub sześciu znaków, w których 2. i 4. znak to *z*, np. *czczą*, *rzezi* i *szczyt*,
8. `[orth="a*by"]`
   gwiazdka oznacza dowolną liczbę wystąpień znaku lub wyrażenia bezpośrednio przed nią, a zatem zapytanie to może posłużyć do znalezienia segmentów składających się z dowolnej liczby liter *a*, po których następuje ciąg *by*, np. *by* (zero wystąpień *a*), *aby*, *aaaaby* itp.,
9. `[orth="Ala.*"]`
   segmenty zaczynające się na *Ala*, np. *Ala* i *Alabama*,
10. `[orth=".*al+"]`
    plus ma działanie podobne do gwiazdki i oznacza dowolną większą od zera liczbę wystąpień znaku lub wyrażenia bezpośrednio przed nim, a zatem wynikiem tego zapytania będzie znalezienie segmentów kończących się na *al*, *all*, *alll* itd., ale nie na *a*, np. *dal*, *robal* i *Gall*,
11. `[orth="a{1,3}b.*"]`
    konstrukcja typu `n,m` oznacza od `n` do `m` wystąpień znaku lub wyrażenia bezpośrednio przed nią, a zatem zapytanie to pomoże znaleźć segmenty zaczynające się od ciągu od

1 do 3 liter a, po którym następuje litera b, a następnie dowolny ciąg znaków (por. `.*`), np. *aby*, *aaaby*, *absolutnie*,

12. `[orth=".*(la){3,}.*"]`
    konstrukcja typu `n,` oznacza co najmniej `n` wystąpień znaku lub ujętego w nawiasy okrągłe wyrażenia bezpośrednio przed nią, a zatem zapytanie to może posłużyć do znalezienia segmentów, w których ciąg *la* występuje przynajmniej 3 razy z rzędu, np. *tralalala*, *sialalala*,
13. `[orth="[bcćdfghjklłmnńprsśtwzźż]{4,}[aąeęioóuy]"]`
    segmenty składające się z co najmniej 4 liter spółgłoskowych i dokładnie jednej litery samogłoskowej, np. *źdźbła*, *drzwi* i *czczą*; wyrażenie `[bcćdfghjklłmnńprsśtwzźż]{4,}` oznacza co najmniej czterokrotne powtórzenie znaku pasującego do `[bcćdfghjklłmnńprsśtwzźż]`, tj. co najmniej cztery wystąpienia litery spółgłoskowej (niekoniecznie tej samej),
14. `[orth="([bcćdfghjklłmnńprsśtwzźż]{3}[aąeęioóuy]){2,}"]`
    segmenty składające się z co najmniej dwukrotnego powtórzenia wzorca CCCV, gdzie C to litera spółgłoskowa, a V to litera samogłoskowa, np. *wszystko*, *przykrzejszy* i *szlachta*; konstrukcja typu `n` oznacza dokładnie `n` wystąpień znaku lub ujętego w nawiasy okrągłe wyrażenia bezpośrednio przed nią,
15. `[orth="(pod|na|za)jecha.*"]`
    segmenty zaczynające się od *podjecha*, *najecha* i *zajecha*, np. *podjechał*, *zajechawszy*.

Specyfikacje segmentów podane powyżej muszą pasować do całych segmentów stąd konieczność umieszczenia po obu stronach ciągu `(la){3,}` w zapytaniu 13. o segmenty zawierające ciąg lalala wyrażenia .*, pasującego do dowolnego ciągu znaków.

3.2. Zapytania wyższego rzędu

Aby znaleźć wszystkie formy leksemu korpus, można użyć następującego zapytania:

```
[base="korpus"]
```

Atrybut `base` jest jednym z wielu możliwych atrybutów, jakie mogą pojawić się w zapytaniu. Wartością tego atrybutu powinna być specyfikacja formy podstawowej (hasłowej), a zatem zapytanie `[base="pisać"]` może być użyte do znalezienia form typu *pisać*, *piszę*, *pisała*, *piszcie*, *pisanie*, *pisano*, *pisane* itp.

Formy podstawowe zapisane są wyłącznie w postaci uwspółcześnionej, zatem zapytanie:

```
[base="komisja"]
```

zwróci wystąpienia różnych form fleksyjnych rzeczownika KOMISJA zapisane według różnych konwencji ortograficznych. Z kolei zapytanie:

```
[base="kommisja"]
```

nie zwróci żadnego wyniku, choć w korpusie istnieją formy fleksyjne postaci *kommisja*. Aby znaleźć wszystkie formy tego rzeczownika zapisane przed podwójne *m*, należy dołączyć dodatkowy warunek na segment opisujący jego postać w warstwie transliterowanej, np.:

```
[base="komisja" & translit=".*mm.*"]
```

Podobnie jak w wypadku atrybutów `translit`i `orth` wartościami atrybutu `base` mogą być wyrażenia regularne, np:

```
[base="komit[ae]t"]
```

znalezione zostaną wszystkie segmenty, których forma hasłowa ma postać KOMITET lub KOMITAT.

Jak widać to było na wcześniejszym przykładzie, zapytania o różne atrybuty segmentów można łączyć. Na przykład, aby znaleźć wszystkie wystąpienia segmentu *minę* rozumianego jako forma leksemu MINA (a nie na przykład leksemu MINĄĆ), można zadać następujące zapytanie:

```
[orth="minę" & base="mina"]
```

Podobne znaczenie ma następujące zapytanie o te wystąpienia segmentu *minę*, które nie są interpretowane jako formy leksemu MINĄĆ.

```
[orth="minę" & !base="minąć"]
```

W powyższych zapytaniach operator `&` spełnia rolę logicznej koniunkcji. Operatorem do niego dualnym jest operator `|`, spełniający rolę logicznej alternatywy. Oto kilka przykładów użycia tego operatora:

- `[base="on" | base="ja"]`
  wszystkie formy zaimków ON i JA, równoważne zapytaniu `[base="on|ja"]`,
- `[base="on" | orth="mnie" | orth="ciebie"]`
  wszystkie formy zaimka ON, a także segmenty *mnie* i *ciebie*,
- `[orth="pora" & !(base="por" | base="pora")]`
  segment *pora* nie będący ani formą leksemu POR, ani formą leksemu PORA.

Aby lepiej zrozumieć różnicę pomiędzy operatorami `&` i `|`, porównajmy następujące dwa zapytania:

```
[orth="minę" & base="mina"]
[orth="minę" | base="mina"]
```

W wyniku zadania pierwszego zapytania znalezione zostaną te segmenty, które są jednocześnie (koniunkcja) segmentem *minę* i formą leksemu MINA, a więc wyłącznie te wystąpienia segmentu *minę*, które są interpretowane jako formy leksemu mina. W wyniku zadania drugiego zapytania znalezione natomiast zostaną te segmenty, które są albo dowolnie interpretowanym segmentem *minę*, albo formą leksemu MINA (alternatywa), czyli wszystkie wystąpienia zarówno segmentu MINĘ, jak i segmentów MINA, MINY, MINAMI itp. interpretowanych jako formy leksemu mina.

Specyfikacje pozycji w korpusie, ujęte w nawiasy kwadratowe, mogą zawierać dowolną liczbę warunków typu `atrybut="wartość"` (na przykład `orth="nie"`) połączonych operatorami `!`, `&` i `|`, tak jak pokazują to powyższe przykłady. Możliwe jest także całkowite pominięcie jakichkolwiek warunków poniższe zapytanie mogłoby posłużyć do znalezienia wszystkich segmentów w korpusie.[1]

```
[]
```

Taka banalna specyfikacja pozycji w korpusie, pasująca do dowolnego segmentu, może posłużyć na przykład do znalezienia dwóch form oddzielonych od siebie dowolnymi dwoma segmentami, np.:

```
[orth="się"][][][base="bać"]
```

W wyniku tego zapytania zostaną znalezione ciągi takie jak *się mnie też bać* czy *się nie chcę bać*.

Dla wielu zastosowań ciekawsza byłaby możliwość zapytania na przykład o formy oddalone od siebie o najwyżej pięć pozycji. MTAS umożliwia zadawanie takich pytań, gdyż

[1] O ile nie wprowadzono w korpusie ograniczeń na liczbę wyników. W wypadku korpusu 1830-1918 takich ograniczeń nie ma.

pozwala na formułowanie wyrażeń regularnych także na poziomie pozycji korpusu. Na przykład zapytanie o formę leksemu BAĆ występującą dwie, trzy lub cztery pozycje dalej niż forma *się* może wyglądać następująco:

```
[orth="się"][]{2,4}[base="bać"]
```

W wyniku tego zapytania zostaną znalezione ciągi uzyskane w wyniku poprzedniego zapytania, a także na przykład ciąg *się pani niczego nie boi*.

Zapewne nieco bardziej precyzyjnym zapytaniem o różne wystąpienia form tzw. czasownika zwrotnego BAĆ SIĘ byłoby zapytanie o *się* w pewnej odległości przed formą leksemu BAĆ, ale bez znaku interpunkcyjnego pomiędzy tymi formami, lub bezpośrednio za taką formą, ewentualnie oddzielone od formy BAĆ zaimkiem osobowym:

```
[orth="się"][!orth="[.!?,:]"]{0,5}[base="bać"]
| [base="bać"][base="on|ja|ty|my|wy"]?[orth="się"]
```

3.3. Zapytania o znaczniki morfosyntaktyczne

Powyższe zapytanie można uprościć poprzez zastąpienie warunku `orth!="[.!?,:]"` bezpośrednim odwołaniem do „klasy gramatycznej” `interp`:

```
[orth="się"][!pos="interp"]{0,5}[base="bać"]
| [base="bać"][base="on|ja|ty|my|wy"]?[orth="się"]
```

Ogólniej, wartościami atrybutu `pos` (ang. *part of speech* ‘część mowy’) są skróty nazw klas gramatycznych omówionych w p. 2.2 (por. tabela 3.6). Na przykład zapytanie o sekwencję dwóch form rzeczownikowych rozpoczynających się na *a* może być sformułowane w sposób następujący:

```
[pos="subst" & orth="a.*"]{2}
```

Podobnie jak to miało miejsce w wypadku specyfikacji form obu warstw tekstowych i form hasłowych, także specyfikacje klas gramatycznych mogą zawierać wyrażenia regularne. Na przykład, zważywszy na to, że zaimki osobowe należą do klasy zaimków trzecioosobowych *ppron3* i do klasy zaimków nietrzecioosobowych *ppron12*, poniższe zapytania mogą posłużyć do znalezienia dowolnych form dowolnych zaimków osobowych:

```
[pos="ppron12" | pos="ppron3"]
[pos="ppron12|ppron3"]
[pos="ppron(12|3)"]
[pos="ppron[123]+"]
[pos="ppron.+"]
```

A zatem zapytanie o formy bać się może zostać jeszcze bardziej uproszczone do następującego zapytania:

```
[orth="się"][!pos="interp"]{0,5}[base="bać"]
| [base="bać"][pos="ppron.+"]?[orth="się"]
```

W zapytaniach można określać wartości nie tylko formy transliterowanej i transkrybowanej (za pomocą atrybutów `translit` i `orth`), formy hasłowej (za pomocą `base`) i klasy gramatycznej (za pomocą `pos`), ale także wartości poszczególnych kategorii gramatycznych, np. przypadka czy rodzaju. Służą do tego następujące atrybuty (por. p.2.1):

atrybut	kategoria	możliwe wartości
number	liczba	sg pl
case	przypadek	nom gen dat acc inst loc voc
gender	rodzaj	m1 m2 m3 f n
subgender	przyrodzaj	col ncol pt
person	osoba	pri sec ter
degree	stopień	pos comp sup
aspect	aspekt	imperf perf
negation	zanegowanie	aff neg
accentability	akcentowość	akc nakc
post-prepositionality	poprzyimkowość	npraep praep
agglutination	aglutynacyjność	agl nagl
vocalicity	wokaliczność	nwok wok
fullstoppedness	kropkowalność	pun npun

A zatem możliwe jest zadanie na przykład następujących zapytań:

1. `[number="sg"]`
   znalezione zostaną wszystkie formy w liczbie pojedynczej,
2. `[pos="subst" & number="sg"]`
   znalezione zostaną formy rzeczownikowe w liczbie pojedynczej,
3. `[pos="subst" & !gender="f"]`
   formy rzeczownikowe rodzaju męskiego lub nijakiego,
4. `[number="sg" & case="nom|acc" & gender="m[123]"]`
   pojedyncze mianownikowe lub biernikowe formy męskie.

O klasy gramatyczne i kategorie gramatyczne można także pytać łącznie, używając do tego atrybutu tag. Na przykład, aby znaleźć wszystkie rzeczowniki żeńskie w mianowniku o pojedynczej wartości liczby, można zadać następujące zapytanie:

```
[tag="subst:sg:nom:f"]
```

Wartości atrybutu `tag` mają postać $\texttt{kl:kat}_1\texttt{:kat}_2\texttt{:...:kat}_n$, gdzie `kl` to nazwa klasy gramatycznej, a $\texttt{kat}_i$ to wartości kategorii przysługujących tej klasie w kolejności, w jakiej zostały podane w tabeli 3.6.

Tak jak w wypadku innych atrybutów, specyfikacja atrybutu `tag` może być zadana wyrażeniem regularnym, np.:

```
[tag=".*:sg:(nom|acc):m[123].*"]
```

Ponieważ nazwy wartości poszczególnych kategorii są rozłączne, można również stosować zbiorczą kategorię `feat` (ang. *feature* 'cecha') w zastępstwie każdej innej. Ujednoznacznienie dokona się przez odpowiednią wartość. Dlatego następujące dwa zapytania zwrócą te same wyniki:

- `[pos="subst" & case="acc" & number="pl" & gender="f"]`
- `[pos="subst" & feat="acc" & feat="pl" & feat="f"]`

3.4. Graficzny konstruktor zapytań

Do tworzenia podstawowych zapytań o sekwencje segmentów można użyć prostego graficznego konstruktora. W oknie konstruktora można definiować warunki określające cechy kolejnych segmentów zapytania, np. część mowy, postać segmentu w obu warstwach tekstowych, formę hasłową, a także wartości wszystkich kategorii gramatycznych opisanych w tabeli 2. Poszczególne warunki w obrębie segmentu mogą być łączone operatorami *oraz* (koniunkcja) i *lub* (alternatywa). Po zdefiniowaniu wszystkich segmentów zapytania należy wcisnąć przycisk *Zapisz*, następnie określić dodatkowe parametry wyszukania, np.

ograniczenia za pomocą metadanych, i rozpocząć wyszukiwanie. Zbudowane za pomocą konstruktora zapytania pojawi się w pasku wyszukiwania, dzięki czemu można dodatkowo zweryfikować jego poprawność.

3.5. Ograniczenie zapytania do zdania lub akapitu

Teksty zawarte w korpusie zostały podzielone na zdania i akapity. Informację tę można wykorzystać w zapytaniach, na przykład ograniczając dopasowanie do jednego zdania.

Aby ograniczyć zasięg zapytania, należy dopisać do zapytania słowo kluczowe `within`, a po nim `<s/>` lub `<p/>`, w zależności od tego, czy zasięg ma być ograniczony do zdania (ang. *sentence*) czy do akapitu (ang. *paragraph*). Ilustruje to następujący przykład zapytania o zdania, w których forma *się* występuje za formą leksemu BYĆ, w odległości co najmniej jednego i nie więcej niż dziesięciu segmentów:

```
[base="bać"][!orth="się"]{1,10}[orth="się"] within <s/>
```

Dodatkowo można również na elementy `<s/>` i `<p/>` nałożyć pewne warunki dotyczące tego, czy zawierają segmenty innego typu. Przykładowo, za pomocą następującego zapytania można znaleźć wszystkie wystąpienia czasownika być BYĆ występującego w funkcji słowa posiłkowego czasu przyszłego złożonego ograniczone do zdań zawierających formę bezokolicznika:

```
[pos="fut"] within (<s/> containing [pos="inf"])
```

Wśród wyników będą oczywiście również takie zdania, w których czas przyszły został utworzony z formy pseudoimiesłowu, a bezokolicznik pełni zdaniu inną funkcję gramatyczną. Można też sformułować zapytanie odwrotnie — o zdania, w których forma pseudoimiesłowu w ogóle nie występuje:

```
[pos="fut"] within (<s/> !containing [pos="praet"])
```

Pełną listę słów kluczowych, które mogą się pojawić w zapytaniach wyszukiwarki MTAS, można znaleźć w jej dokumentacji, nie wszystkie jednak będą miały sens w kontekście korpusu XIX wieku i mogą być użyteczne w innych korpusach, w których uwzględniono znakowanie warstw innego typu (np. nazw własnych, nadrzędników składniowych, sensu słów itp.).

Oprócz znaczników odnoszących się do elementów struktury tekstu (np. `<s/>`) istnieją również znaczniki odnoszące się do ich początku i końca. W wypadku `<s/>` będą to odpowiednio: `<s>` i `</s>`. Ich dopasowaniem nie jest żaden segment, ale mogę być użyte w połączeniu z warunkami definiującymi inne segmenty, np. zapytanie:

```
<s> [pos="num"]
```

odnajdzie wszystkie wystąpienia liczebnika stojącego na początku zdania. Analogicznie zapytanie:

```
[pos="num"][pos="interp"]</s>
```

odnajdzie wszystkie wystąpienia ciągu składającego się z liczebnika i znaku interpunkcyjnego stojących na końcu zdania.

3.6. Ograniczenie zapytania za pomocą metadanych

Każda próbka znajdująca się w korpusie tekstów z lat 1830-1918 opatrzona została metryczką zawierającą informacje o tytule i autorze utworu, jego pochodzeniu, wydawcy itp. Pełna metryczka wraz z szerokim kontekstem jest dostępna dla każdego wyszukania pod tabelą wyników. Część z tych informacji można wykorzystać do ograniczenia zasięgu

zapytania, na przykład do tekstów danego autora lub tekstów powstałych w danym przedziale czasowym.

Z poziomu interfejsu webowego wyszukiwarki następujące pola metadanych mogą zostać wykorzystane do ograniczenia wyników:

- etykieta próbki (można podać pełną nazwę lub jej fragment),
- data wydania (roczna) (można podać dokładną datę, górne lub dolne ograniczenie oraz przedział czasowy np. `1850-1859`),
- autor (można podać dowolny fragment nazwiska, np. `Józef Ignacy Kraszewski`, `Józef Kraszewski`, `Kraszew`),
- tytuł (można podać dowolny fragment tytułu, np. `Klub nietoperzy`, `nietoperz klub`),
- miejsce wydania (można podać pełną nazwę lub fragment, np. `Paryż`, `Warszawa Kraków`),
- styl (można wybrać z listy pięciu stylów funkcjonalnych oraz ich wariantów wierszowanych).

Ponadto, można też wybrać wartość *Zaawansowane*, która pozwala definiować ograniczenia podobne jak powyższe, ale dla wszystkich pól obecnych w metryczce i z możliwością nakładania warunków obejmujących więcej niż jedno pole. Ograniczenia zaawansowane to sekwencja warunków postaci `atrybut:wartość`, w których `atrybut` jest nazwą z tabeli 1. Warunki można łączyć operatorami `AND` i `OR`. Każde z pól jest reprezentowane za pomocą jednego z trzech typów danych, które warunkują ich zachowanie w definiowanych ograniczeniu wyników o metadane:

- *string* — ciąg znaków, zwykle o ściśle określonej postaci; dopasowaniem takiego warunku jest pełny ciąg znaków (można używać znaku `*` oznaczającego dowolny ciąg znaków), np. dla pola *Etykieta* może to być `id:1840_4.1` lub `id:1850*`;
- *text* — lista słów; dopasowaniem warunku jest przynajmniej jedno słowo z listy, np. dla pola *Autor*: `author:Kraszewski`
- *int* — liczba całkowita; dopasowaniem warunku jest data roczna, np. dla pola *Data wydania (czterocyfrowy zapis liczbowy)* : `date_int:1890`; wartością atrybutu mogą być też zakresy liczbowe przedzielone słowem kluczowym `TO`.

Poniżej znajduje się kilka przykładów ograniczeń, które można zdefiniować w powyższy sposób:

- `author:Słowacki AND place:Paryż`
  ograniczenie wyników wyłącznie do utworów Słowackiego wydanych w Paryżu;
- `genre:popularnonaukowy AND date_int:1850`
  ograniczenie wyników do tekstów popularnonaukowych z roku 1850;
- `author:(Słowacki OR Krasiński) AND date_int:[1830 TO 1835]`
  ograniczenie wyników do utworów Słowackiego i Krasińskiego wydanych w latach 1830-1835.
- `publisher:Gebethner AND date_int:[* TO 1900]`
  ograniczenie wyników do tekstów wydanych oryginalnie przez wydawnictwo *Gebethner i Spółka* lub *Gebethner i Wolff* nie później niż w roku 1900.
- `text_origin:("e-Biblioteka Uniwersytetu Warszawskiego")`
  ograniczenie wyników do tekstów, których wersja źródłowa (skan) została zaczerpnięta z cyfrowej biblioteki Uniwersytetu Warszawskiego.

Pole	**Atrybut**	**Typ**
Etykieta	`id`	str
Autor	`author`	text
Tytuł	`title`	text
Data wydania (roczna, czasem również dzienna)	`date`	str
Data wydania (czterocyfrowy zapis liczbowy)	`date_int`	int
Miejsce wydania	`place`	text
Redaktor	`editor`	text
Wydawnictwo	`publisher`	text
Styl	`genre`	str
Źródło	`text_origin`	text

Tabela 1. Atrybuty pól metadanych w zaawansowanym ograniczeniu wyników

Liczba: (2 wartości)		
pojedyncza	*sg*	*oko*
mnoga	*pl*	*oczy*
Przypadek: (7 wartości)		
mianownik	*nom*	*woda*
dopełniacz	*gen*	*wody*
celownik	*dat*	*wodzie*
biernik	*acc*	*wodę*
narzędnik	*inst*	*wodą*
miejscownik	*loc*	*wodzie*
wołacz	*voc*	*wodo*
Rodzaj: (5 wartości)		
męski osobowy	*m1*	*papież*, *kto*, *wujostwo*
męski zwierzęcy	*m2*	*baranek*, *walc*, *babsztyl*
męski rzeczowy	*m3*	*stół*
żeński	*f*	*stuła*
nijaki	*n*	*dziecko*, *okno*, *co*, *skrzypce*, *spodnie*
Przyrodzaj: (3 wartości)		
przymnogi	*pt*	*wujostwo*, *skrzypce*, *spodnie*
zbiorowy	*col*	*dziecko*
niezbiorowy	*ncol*	*okno*
Osoba: (3 wartości)		
pierwsza	*pri*	*bredzę*
druga	*sec*	*bredzisz*
trzecia	*ter*	*bredzi*
Stopień: (3 wartości)		
równy	*pos*	*cudny*
wyższy	*com*	*cudniejszy*
najwyższy	*sup*	*najcudniejszy*
Aspekt: (2 wartości)		
niedokonany	*imperf*	*iść*
dokonany	*perf*	*zajść*
Zanegowanie: (2 wartości)		
niezanegowana	*aff*	*pisanie*, *czytanego*
zanegowana	*neg*	*niepisanie*, *nieczytanego*
Akcentowość: (3 wartości)		
akcentowana	*akc*	*niego*, *jego*, *tobie*
nieakcentowana	*nakc*	*go*, *-ń*, *ci*
zneutralizowana	*neut*	*one*, *im*, *je*
Poprzyimkowość: (2 wartości)		
poprzyimkowa	*praet*	*niego*, *-ń*
niepoprzyimkowa	*npraet*	*jego*, *go*
Aglutynacyjność: (2 wartości)		
nieaglutynacyjna	*nagl*	*niósł*
aglutynacyjna	*agl*	*niosł-*
Wokaliczność: (2 wartości)		
wokaliczna	*wok*	*-em*
niewokaliczna	*nwok*	*-m*
Kropkowalność: (2 wartości)		
z następującą kropką	*pun*	*tzn*
bez następującej kropki	*npun*	*wg*

Tabela 2. Kategorie gramatyczne

	liczba	przypadek	rodzaj	przyrodz.	osoba	stopień	aspekt	zaneg.	akcent.	poprzyim.	aglutyn.	wokal.	kopk.
rzeczownik	⊕	⊕	⊙	⊙									
rzeczownik deprecjatywny	⊕	⊕	⊙										
liczebnik główny	⊙	⊕	⊕										
liczebnik przymiotnikowy	⊕	⊕	⊕			⊕							
liczebnik przysłówkowy						⊕							
przymiotnik	⊕	⊕	⊕			⊕							
przymiotnik przyprzym.													
przymiotnik poprzyim.		⊕											
przysłówek						⊕							
zaimek nietrzecioosobowy	⊙	⊕		⊙					⊕				
zaimek trzecioosobowy	⊙	⊕	⊕	⊙					⊕	⊕			
zaimek SIEBIE		⊕											
forma nieprzeszła	⊕			⊕		⊙							
forma przyszła BYĆ	⊕			⊕		⊙							
forma BYĆ cz. przyszłego	⊕			⊕		⊙							
aglutynant BYĆ	⊕			⊕		⊙						⊕	
pseudoimiesłów	⊕		⊕			⊙					⊕		
forma BYĆ cz. zaprzeszłego	⊕		⊕			⊙					⊕		
rozkaźnik	⊕			⊕		⊙							
bezosobnik						⊙							
bezokolicznik						⊙							
im. przys. współczesny						⊙							
im. przys. uprzedni						⊙							
odsłownik	⊕	⊕	⊙			⊙	⊕						
im. przym. czynny	⊕	⊕	⊕			⊙	⊕						
im. przym. bierny	⊕	⊕	⊕			⊙	⊕						
winien	⊕		⊕			⊙							
predykatyw													
przyimek		⊙											
spójnik współrz.													
spójnik podrz.													
partykuła													
skrót												⊕	
człon wyrażenia													
wykrzyknik													
znak interpunkcyjny													
ciało obce													

Tabela 3. Klasy gramatyczne

fleksem	**skrót**	**forma podstawowa**	**przykład**
rzeczownik	*subst*	mianownik l. poj.	*doktor*
rzeczownik deprecjatywny	*depr*	mianownik l. poj. rzeczownika	*doktor*
liczebnik główny	*num*	mianownik rodz. m3	*pięć*, *dwa*
liczebnik przymiotnikowy	*adjnum*	mianownik l. poj. rodz. męskiego st. równego	*drugi*, *wtóry*
liczebnik przysłówkowy	*advnum*	jedyna forma fleksemu	*dwakroć*, *dwojako*
przymiotnik	*adj*	mianownik l. poj. rodzaju męskiego st. równego	*polski*
przymiotnik przyprzyim.	*adja*	mianownik l. poj. rodz. męskiego przymiotnika w st. równym	*polski*
przymiotnik poprzyim.	*adjp*	mianownik l. poj. rodz. męskiego przymiotnika w st. równym	*polski*
przysłówek	*adv*	forma stopnia równego	*dobrze*, *bardzo*
zaimek nietrzecioosobowy	*ppron12*	mianownik l. poj.	*ja*
zaimek trzecioosobowy	*ppron3*	mianownik l. poj.	*on*
zaimek SIEBIE	*siebie*	biernik	*siebie*
forma nieprzeszła	*fin*	bezokolicznik	*czytać*
forma przyszła być	*bedzie*	bezokolicznik	*być*
forma być cz. przyszłego	*fut*	bezokolicznik	*być*
aglutynant BYĆ	*aglt*	bezokolicznik	*być*
pseudoimiesłów	*praet*	bezokolicznik	*czytać*
forma BYĆ cz. zaprzeszłego	*plusq*	bezokolicznik	*być*
rozkaźnik	*impt*	bezokolicznik	*czytać*
bezosobnik	*imps*	bezokolicznik	*czytać*
bezokolicznik	*inf*	bezokolicznik	*czytać*
im. przys. współczesny	*pcon*	bezokolicznik	*czytać*
im. przys. uprzedni	*pant*	bezokolicznik	*czytać*
odsłownik	*ger*	bezokolicznik	*czytać*
im. przym. czynny	*pact*	bezokolicznik	*czytać*
im. przym. bierny	*ppas*	bezokolicznik	*czytać*
winien	*winien*	forma męska l. poj.	*winien*, *rad*
predykatyw	*pred*	jedyna forma fleksemu	*warto*
przyimek	*prep*	niewokaliczna forma fleksemu	*na*, *przez*, *w*
spójnik współrz.	*conj*	jedyna forma fleksemu	*oraz*
spójnik podrz.	*comp*	jedyna forma fleksemu	*że*
partykuła	*part*	jedyna forma fleksemu	*nie*, *-li*, *się*
skrót	*brev*	forma hasłowa rozwinięcia skrótu	*rok*, *i_tak_dalej*
człon wyrażenia	*frag*	jedyna forma fleksemu	*wskroś*, *dala*
wykrzyknik	*interj*	jedyna forma fleksemu	*laboga*, *pst*
znak interpunkcyjny	*interp*	jedyna forma fleksemu	*;*, *!*, *?*
ciało obce	*xxx*	jedyna forma fleksemu	*wsio*, *revolutionibus*

Tabela 4. Skróty nazw klas gramatycznych oraz ich formy hasłowe.