

Instrukcja korzystania z wyszukiwarki korpusowej MTAS w Korpusie Narodowym Języka Polskiego

Spis treści

Wprowadzenie	1
1. Segmentacja	1
2. Zestaw znaczników morfosyntaktycznych	2
2.1. Kategorie gramatyczne	2
2.2. Klasy gramatyczne	3
3. Język zapytań	3
3.1. Zapytania o segmenty	3
3.2. Zapytania z innymi atrybutami	5
3.3. Zapytania o znaczniki morfosyntaktyczne	7
3.4. Interpretacje spoza słownika	8
3.5. Graficzny konstruktor zapytań	8
3.6. Ograniczenie zapytania do zdania lub akapitu	9
3.7. Warstwa składniowa	10
3.8. Warstwa jednostek nazewniczych	11
3.9. Ograniczenie zapytania za pomocą metadanych	13

Wprowadzenie

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

Niniejszy dokument został przygotowany przez Witolda Kierasia i opisuje sposób użytkownika 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 w tagsecie stosowanym w wersji Narodowej Korpusu Języka Polskiego dostępnej w wyszukiwarce MTAS oraz w aplikacji Korpusomat. Za zgodą wszystkich wyżej wymienionych autorów niniejsza wersja dokumentu zostaje udostępniona na zasadach licencji Creative Commons BY-SA.

1. Segmentacja

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: [tgał][eś], [długo][śmy], [tak][em].

- Za odrębne segmenty uznane są partykuły *by*, *-ź(e)* i *-li*, a zatem następujące słowa reprezentują po kilka segmentów: *[przyszedeł][by]*, *[napisała][by][m]*, *[chodź][że]*, *[potrzebowat][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. Dzielone 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.

Poniżej znajduje się przykładowe zdanie i jego segmentacja zgodna z opisanymi w tej części zasadami.

Pojechalibyśmy z Janem M. Rokitą i Janem Nowakiem-Jeziorańskim na sesję polsko-amerykańską, gdyby nas zaprosił George W. Byłaby to nasza już 2. doń podróż od czasów PRL-u, a może i 3., czy nawet 4.

[Pojechali][by][śmy][z][Janem][M.][Rokitą][i][Janem]
 [Nowakiem][-][Jeziorańskim][na][sesję]
 [polsko][-][amerykańską][,][gdyby][nas][zaprosił][George]
 [W][.] [Była][by][to][nasza][już][2.][do][ń][podróż][od]
 [czasów][PRL-u][,][a][może][i][3.][,][czy][nawet][4][.]

2. Zestaw znaczników morfosyntaktycznych

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 1 przedstawia repertuar kategorii gramatycznych używanych w znakowaniu tekstów w NKJP. Repertuar kategorii pochodzi z tagsetu analizatora morfologicznego Morfeusz, który jest podobny do tagsetu NKJP, ale zawiera też kilka modyfikacji. Najważniejszą cechą tagsetu Morfeusza w stosunku do tagsetu NKJP jest wprowadzenie obok kategorii rodzaju opcjonalnej tzw. kategorii przyrodzaju o wartościach *pt*, *co1* i *nco1* — przysługują one wyłącznie klasom *subst* i *num*. W wypadku *subst* informują o łączliwości danego rzeczownika rodzaju nijakiego z liczebnikami głównymi (*nco1*, np. *okno*) lub zbiorowymi (*co1*, np. *dziecko*) oraz o braku w paradygmacie rzeczownika form liczby pojedynczej (*pt*, np. *skrzydce*). W wypadku klasy *num* wartość kategorii przyrodzaju informuje o tym, że dana forma jest zbiorowa (*co1*, np. *dwoje*) lub niezbirowa (*nco1*, np. *dwa*). Konsekwencją wprowadzenia kategorii przyrodzaju w klasie *num* było usunięcie z tagsetu klasę *numco1*. Inne różnice to m.in.:

- ograniczenie klasy przymiotników poprzyimkowych *adjp* wyłącznie do form dopełniacza (np. (z) *wolna*) i celownika (np. (po) *polsku*) dawnej odmiany niezłożonej przymiotników oraz dodanie im wartości przypadku;
- przemianowanie klas *qub* i *burk* odpowiednio na *part* i *frag*.

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 NKJP klasy gramatyczne rozumiane są morfosyntaktycznie, są one oparte na pojęciu fleksemu, będącym pojęciem węższym od terminu leksem.

Tabela 2 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 do tej klasy zwykle „odmieniają się” przez tę kategorię), zaś symbol \ominus 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 zawiera informacje o formach podstawowych dla poszczególnych klas fleksyjnych, a także skróty nazw klas fleksyjnych używane w korpusie.

3. Język zapytań

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 jednej z dwu dotychczasowych wyszukiwarek NKJP, czyli w 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 korpusów tekstów polskich w postaci zindeksowanej w NKJP, w którym znajdują się aktualnie trzy warstwy znakowania: warstwa morfosyntaktyczna, warstwa składniowa i warstwa jednostek nazewniczych. Z tego powodu instrukcja języka zapytań ogranicza się tylko do tych warstw i nie uwzględnia możliwości wyszukiwarki zastosowanej do innych korpusów. Nie należy jej zatem traktować jako ogólnej instrukcji użytkownika wyszukiwarki MTAS, choć jej aktualna wersja jest niemal tożsama z instrukcją języka zapytań w aplikacji Korpusomat, która korzysta z tej samej wyszukiwarki i buduje korpusy z takimi samymi warstwami znakowania. Podstawowa ogólna 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). Do zapytań o tę postać ortograficzną segmentu służy atrybut `orth`, można też jednak ograniczyć się do wpisania w oknie wyszukiwarki poszukiwanego słowa (lub słów). Zatem poniższe zapytanie o dwa sąsiadujące ze sobą segmenty:

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

można zadać również w prostszy sposób:

```
komisja szkolna
```

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 instrukcji, ograniczymy się tutaj do kilku przykładów, które powinny pozwolić użytkownikowi na szybkie przyswojenie składni i znaczenia takich wyrażeń.

1. `[orth="(Ala|Ela)"]`
znak `|` oznacza alternatywę dwóch wyrażeń (całość należy dodatkowo ująć w nawiasy okrągłe), a zatem zapytanie to może zostać użyte do znalezienia wszystkich wystąpień segmentów *Ala* lub *Ela*,
2. `[orth="[AE]la"]`
nawiasy kwadratowe oznaczają alternatywę znaków, a zatem zapytanie to może zostać użyte do znalezienia tych segmentów, których pierwszy znak to *A* lub *E*, po którym następuje ciąg znaków postaci *la*, tj. zapytanie to jest równoważne poprzedniemu,
3. `[orth="beza?"]`
znak zapytania oznacza opcjonalność znaku (tutaj ostatniego *a*) lub ujętego w nawiasy okrągłe wyrażenia bezpośrednio poprzedzającego znak `?`, a zatem w wyniku zadania tego zapytania znalezione zostaną segmenty *bez* i *beza*,
4. `[orth="bez. "]`
kropka oznacza dowolny znak, a zatem wynikiem tego zapytania będą segmenty *beza*, *bezy*, *bezq* itp., ale nie *bez* czy *bezami*,
5. `[orth="bez. ?"]`
bez, *beza*, *bezy*, *bezq* itp., ale nie *bezami*,
6. `[orth=".z.z. "]`
segmenty pięciznakowe, w których 2. i 4. znak to *z* (np. *czczq* i *rzezi*),
7. `[orth=".z.z. .?"]`
segmenty składające się z pięciu lub sześciu znaków, w których 2. i 4. znak to *z*, np. *czczq*, *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=".*(1a){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="[bcćdfghjklłmnńprśstwzżź]{4,}[aąęiioóuy]"]

segmenty składające się z co najmniej 4 liter spółgłoskowych i dokładnie jednej litery samogłoskowej, np. *źdźbła*, *drzwi* i *czcza*; wyrażenie [bcćdfghjklłmnńprśstwzżź]{4,} oznacza co najmniej czterokrotne powtórzenie znaku pasującego do [bcćdfghjklłmnńprśstwzżź], tj. co najmniej cztery wystąpienia litery spółgłoskowej (niekoniecznie tej samej),

14. [orth="([bcćdfghjklłmnńprśstwzżź]{3}[aąęiioóuy]){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 (1a){3,} w zapytaniu 12. o segmenty zawierające ciąg *lalala* wyrażenia *.**, pasującego do dowolnego ciągu znaków.

3.2. Zapytania z innymi atrybutami

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*, *pisala*, *piszcie*, *pisanie*, *pisano*, *pisane* itp.

Podobnie jak w wypadku atrybutu *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*.

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.

[]

Taka „pusta” 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 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 2). 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 nietercjoosobowych *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 określić nie tylko postać ortograficzną segmentu (za pomocą atrybutu `orth`), formę hasłową (za pomocą `base`) i klasę gramatyczną (za pomocą `pos`), ale także wartości poszczególnych kategorii gramatycznych, np. przypadku 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	kropkwalność	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"]
znajdzone 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 $kl:kat_1:kat_2:\dots:kat_n$, gdzie kl to nazwa klasy gramatycznej, a kat_i to wartości kategorii przysługujących tej klasie w kolejności, w jakiej zostały podane w tabeli 2.

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. Ujednoliczenie 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. Interpretacje spoza słownika

Interpretacje fleksyjne w znakowaniu morfosyntaktycznym NKJP pochodzą z analizatora Morfeusz 2 i tagera Concraft 2 — analizator zwraca wszystkie możliwe interpretacje dla danego słowa, a tager wybiera najbardziej prawdopodobną ze względu na swój model statystyczny. Interpretacje Morfeusza pochodzą ze *Słownika gramatycznego języka polskiego* (SGJP). Jeśli danego słowa nie da się w żaden sposób zinterpretować jako formy wyrazowej leksemu zanotowanego w SGJP, to Morfeusz nie zwraca żadnej interpretacji. Wówczas tager „zgadyuje” znacznik morfosyntaktyczny, czyli wybiera taki, który zgodnie z jego modelem jest najbardziej prawdopodobny. Skuteczność zgadywania jest w oczywisty sposób dużo niższa niż skuteczność wybierania spośród gotowych interpretacji z Morfeusza, dlatego użytkownik może uznać za przydatną możliwość sterowania tym parametrem w swoich wyszukaniach, np. w wypadku słownictwa najnowszego, nienotowanego w słownikach. Segmenty, którym Morfeusz nie przypisał żadnej interpretacji, mają dodatkowy parametr postaci [ign="true"]. Poniższe przykładowe zapytanie odnajdzie w korpusie wszystkie słowa, które zaczynają się od „tofu” i nie są znane Morfeuszowi:

```
[orth="tofu.*" & ign="true"]
```

Analogicznie można usunąć z wyszukiwania interpretacje zgadywane, np.:

```
[pos="subst" & !ign="true"]
```

3.5. 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 1. 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 wyszukiwania, 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.

3.6. Ograniczenie zapytania do zdania lub akapitu

Jednostkami organizacji tekstu w NKJP są zdania i akapity. Podział ten 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* w czasie przyszłym złożonym ograniczone do zdań zawierających formę bezokolicznika:

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

Intencją takiego zapytania jest odnalezienie (w przybliżeniu) wszystkich wystąpień konstrukcji czasu przyszłego złożonego, w których pojawia się bezokolicznik. Wśród wyników będą oczywiście również takie zdania, w których czas przyszły został utworzony z formy pseudoimiesłowu, a bezokolicznik pełni w zdaniu inną funkcję gramatyczną. Można też sformułować zapytanie odwrotnie — zapytać 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 sensowne zastosowanie.

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 mogą 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.

3.7. Warstwa składniowa

W NKJP zideksowano również informację składniową pochodzącą z parsera zależnościowego COMBO. Każdy tekst w korpusie został automatycznie podzielony na wypowiedzenia, które z kolei zostały poddane pełnej analizie składniowej w aparacie zależnościowym według zasad przyjętych w Polskim Banku Drzew Zależnościowych. Przykład takiej analizy znajduje się na rysunku 1.

MTAS nie jest wyszukiwarką struktur składniowych, nie pozwala zatem na indeksowanie i przeszukiwanie pełnych rozbiorów zdań. Jednak na poziomie każdego segmentu w NKJP zindeksowano informację o jego bezpośrednim nadrzędniku składniowym (tzn. jego formie hasłowej i klasie fleksyjnej) oraz o typie relacji zależności łączącej oba te elementy w wypowiedzeniu. Ponadto zindeksowano również ich położenie względem siebie w wypowiedzeniu: kolejność w porządku linearnym oraz odległość (liczoną w segmentach). Pozwala to na łatwe wyszukanie w korpusie prostszych konstrukcji składniowych oraz analitycznych nieciągłych form fleksyjnych.

W warstwie znakowania składniowego dostępne są następujące atrybuty:

- `deprel` — typ zależności, jaką dany segment jest związany ze swoim bezpośrednim nadrzędnikiem składniowym; wartością tego atrybutu może być jeden z 28 typów zależności przewidzianych w Polskim Banku Drzew Zależnościowych,
- `head.pos` — klasa fleksyjna bezpośredniego nadrzędnika segmentu (tabela 2),
- `head.base` — forma hasłowa bezpośredniego nadrzędnika segmentu,
- `head.distance` — odległość bezpośredniego nadrzędnika segmentu,
- `head.position` — położenie (lewo- lub prawostronne) bezpośredniego nadrzędnika względem segmentu w porządku linearnym wypowiedzenia.

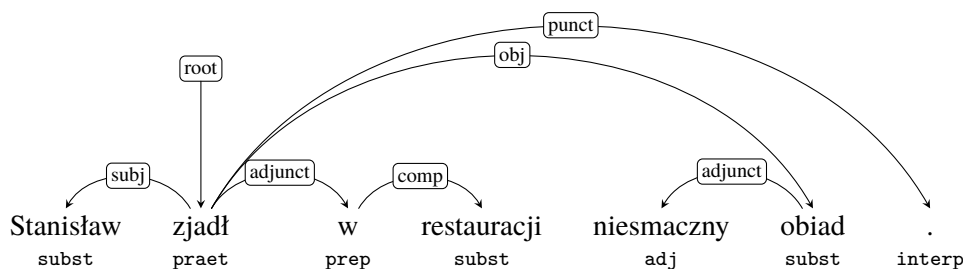
Dzięki rozszerzeniu języka zapytań o powyższe atrybuty można np. łatwo znaleźć wszystkie rzeczowniki użyte w funkcji dopełnienia bliższego konkretnego czasownika:

```
[pos="subst" & deprel="obj" & head.base="kupić"]
```

Możliwe jest również odwrotne wyszukanie odpowiadające na pytanie, przy jakich czasownikach w roli dopełnienia występuje w korpusie konkretny rzeczownik:

```
[deprel="obj.*" & head.pos="(fin|praet|ppas|pact|ger|impt|imps)"  
& base="betel"]
```

Należy jednak zwrócić uwagę, że w powyższym przykładzie wynikiem zapytania będą wystąpienia rzeczownika `BETEL`, nadrzędne względem nich formy czasownikowe (finitywne i niefinitywne) będą się zaś znajdowały w lewym lub prawym kontekście wyników wyróżnione pismem pogrubionym. Wartością atrybutu `deprel` jest wyrażenie regularne, do którego dopasowują się dwa możliwe typy relacji zależności: *obj* i *obj_th* opisane w dokumentacji Polskiego Banku Drzew Zależnościowych.



Rysunek 1. Rozbiór składniowy przykładowego zdania.

Podobne wyszukanie możliwe jest również w wypadku wymagań czasownika innych niż nominalne. Na przykład za pomocą zapytania:

```
[deprel="comp" & head.pos="(fin|praet|imps|impt|ppas|pact)"  
& base="o" & case="loc"]
```

można znaleźć czasowniki wymagające frazy przyimkowej miejscownikowej z przyimkiem o.

Dzięki atrybutowi kodującemu lewo- i prawostronną pozycję nadrzędnika względem segmentu można znaleźć przykłady niekanonicznego szyku zdania, np. podmiotu po orzeczeniu:

```
[deprel="subj" & head.position="left"]
```

lub dopełnienia bliższego przed orzeczeniem:

```
[deprel="obj" & head.position="right"]
```

Podobnie w wypadku innych konstrukcji — brak określenia pozycji nadrzędnika w zapytaniu:

```
[pos="adj" & deprel="adjunct" & head.base="zupa"]
```

zwróci wszystkie przymiotnikowe określenia rzeczownika ZUPA. Dodanie parametru pozycji pozwoli ograniczyć wyszukanie do określeń lewostronnych (np. *gorąca zupa*) lub prawostronnych (np. *zupa pomidorowa*).

Częściowa anotacja składniowa pozwala na odnalezienie elementów wypowiedzenia połączonych ze sobą bezpośrednią relacją zależności bez względu na to, czy sąsiadują one ze sobą w porządku linearnym, czy też są przedzielone innymi elementami wypowiedzenia. Atrybut odległości pozwala np. na ograniczenie wyników tylko do takich przypadków, w których elementy nie sąsiadują ze sobą:

```
[deprel="obj" & head.pos="praet" & !head.distance="1"]
```

Powyższe przykładowe zapytanie wyszuka dopełnienia bliższe orzeczenia w czasie przeszłym, które są oddzielone od tego orzeczenia co najmniej jednym elementem.

Jeszcze jednym praktycznym przykładem wykorzystania anotacji składniowej jest możliwość wyszukania analitycznych form fleksyjnych, których poszczególne fleksy nie są oznaczane w warstwie morfosyntaktycznej jako elementy takiej formy. Dotyczy to np. form czasu przyszłego niedokonanego (utworzonych z formami bezokolicznika lub pseudoimiesłowu lub w obu wariantach):

```
[pos="bedzie" & deprel="aux" & head.pos="(inf|praet)"]
```

czy analitycznych form stopnia wyższego i najwyższego przymiotników:

```
[deprel="adjunct" & base="bardzo" & degree="(com|sup)"  
& head.pos="adj"]
```

Podobnie w wypadku konstrukcji biernej:

```
[base="(być|zostać)" & deprel="aux" & head.pos="ppas"]
```

3.8. Warstwa jednostek nazewniczych

Korpus zawiera również warstwę znakowania jednostek nazewniczych (ang. *named entities*). Są to jednostki tekstowe jedno- lub wielowyrazowe nazywające osoby, miejsca, instytucje czy momenty czasowe. Automatyczna klasyfikacja takich jednostek tekstowych pochodzi z programu *Liner2*, który określa początek i koniec danej jednostki nazewniczej

oraz przydziela jej odpowiednią etykietę. Liner2 opiera się na wzorcowej anotacji jednostek nazewniczych przygotowanej w ramach projektu NKJP, której szczegóły zostały opisane w rozdziale *Anotacja jednostek nazewniczych* (str. 129-167) książki *Narodowy Korpus Języka Polskiego*. Niniejsza instrukcja ogranicza się jedynie do opisu sposobu korzystania z tej klasyfikacji w wyszukiwarce.

Jednostki nazewnicze, podobnie jak opisane wyżej zdania i akapity, przekraczają granicę segmentu, więc można się do nich odnosić w zapytaniach korpusowych tak samo jak do zdań, za pomocą znacznika `<ne />`. Obowiązują również te same zasady dotyczące znaku ukośnika wewnątrz znacznika:

- `<ne>` oznacza początek ciągu opisanego jako jednostka nazewnicza,
- `</ne>` oznacza koniec ciągu opisanego jako jednostka nazewnicza.

Najprostsze możliwe zapytanie tego typu ma postać:

```
<ne />
```

i zwróci wszystkie jednostki nazewnicze wszystkich typów odnalezione w korpusie. Wyszukiwanie można ograniczyć do konkretnego typu nazw np. nazw miejsc:

```
<ne="placeName" />
```

Ta kategoria jednostek ma swoją dodatkową podkategorię klasyfikującą rodzaje miejsc: regiony, kraje, miejscowości itp. Następujące zapytanie ograniczy wyniki jedynie do nazw krajów:

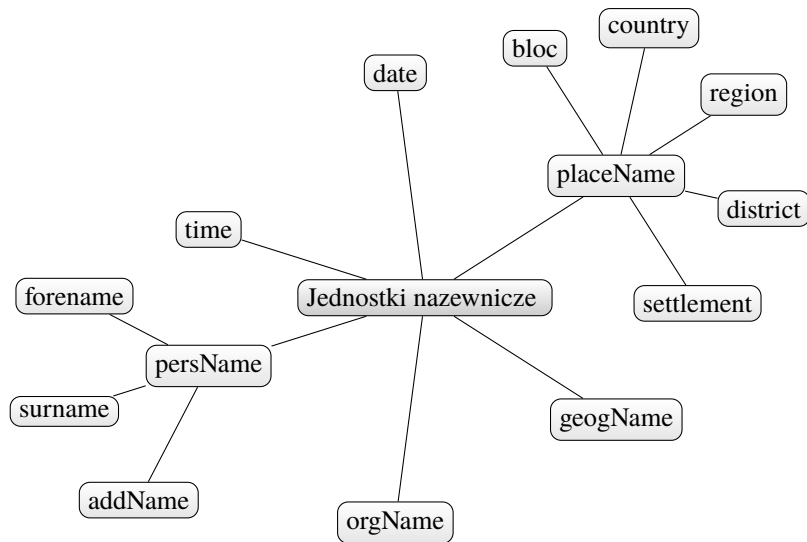
```
<ne="placeName.country" />
```

Pełny repertuar wartości klasyfikacji jednostek nazewniczych to:

- `persName` (nazwy osób) z podtypami: `forename` (imiona), `surname` (nazwiska) i `addName` (pseudonimy, przydomki itp.),
- `orgName` (nazwy organizacji),
- `geogName` (nazwy geograficzne),
- `placeName` (nazwy miejsc czy też tzw. nazwy geopolityczne) z podtypami: `district` (jednostki administracyjne miast, np. *Mokotów*), `settlement` (miasta, wioski, osady, np. *Warszawa*), `region` (jednostki administracyjne większe niż miasto, np. *województwo mazowieckie*), `country` (państwa, kraje, wspólnoty, kolonie, np. *Polska*, *Gujana Francuska*), `bloc` (organizacje polityczne obejmujące co najmniej dwa państwa, np. *Unia Europejska*, *Grupa Wyszehradzka*),
- wyrażenia czasowe: `date` (daty kalendarzowe, np. 13 sierpnia 2018 r.) oraz `time` (określenia czasu w postaci godzin, minut i sekund, np. *ósma wieczorem*).

Podobnie jak w wypadku zdań i akapitów, zapytania o jednostki nazewnicze można łączyć z cechami ortograficznymi i morfosyntaktycznymi segmentów, z których są one zbudowane lub klasyfikacją nazewniczą ich elementów składowych. Oto kilka przykładów takich zapytań:

- `[pos="conj" & base="i"] within <ne="orgName" />`
— wszystkie nazwy organizacji zawierające spójnik *i*, np. *Krajowa Rada Radiofonii i Telewizji* czy *Instytut Meteorologii i Gospodarki Wodnej*,
- `<ne="persName" /> !containing <ne="persName.forename" />`
— wszystkie jednostki nazywające osoby, których składową nie jest imię,
- `<ne="geogName" /> [pos="conj"] <ne="geogName" />`
— wystąpienia dwóch nazw geograficznych połączonych spójnikiem współrzędnym, np. *Europa Zachodnia lub Skandynawia*.
- `[orth="A.*"][orth="M.*"] fullyalignedwith <ne="persName" />`



Rysunek 2. Hierarchia typów jednostek nazewniczych w NKJP

— dwa kolejne segmenty, z których pierwszy zaczyna się od *A*, drugi zaś od *M* i które w całości w tekście występują jako nazwa osoby, np. *Adam Michnik*, *Antoni Macierewicz*.

3.9. Ograniczenie zapytania za pomocą metadanych

W NKJP zindeksowano również dostępne w plikach źródłowych korpusu metadane w postaci oddzielnych pól. Pole metadanych można użyć następnie do ograniczenia zasięgu zapytań w wyszukiwaniach korpusowych. Służy do tego przycisk METADANE, pod którym można zdefiniować takie ograniczenia. Można nałożyć wiele ograniczeń jednocześnie, dodając je za pomocą przycisku DODAJ OGRANICZENIE.

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

Tabela 1. Kategorie gramatyczne

	liczba	przypadek	rodzaj	przyrodz.	osoba	stopień	aspekt	zaneg.	akcent.	poprzyim.	akomod.	aglutyn.	wokal.	kropk.
rzeczownik	⊕	⊕	⊖	⊖										
rzeczownik deprecjatywny	⊖	⊕	⊖											
liczebnik główny	⊖	⊕	⊕								⊕			
przymiotnik	⊕	⊕	⊕			⊕								
przymiotnik przyprzym.														
przymiotnik poprzyim.		⊕												
przysłówek						⊕								
zaimek nietrzecioosobowy	⊖	⊕	⊕		⊖				⊕					
zaimek trzecioosobowy	⊕	⊕	⊕		⊖				⊕	⊕				
zaimek SIEBIE		⊕												
forma nieprzeszła	⊕				⊕		⊖							
forma przyszła BYĆ	⊕				⊕		⊖							
aglutynant BYĆ	⊕				⊕		⊖						⊕	
pseudoomiesłów	⊕		⊕				⊖					⊕		
rozkaznik	⊕				⊕		⊖							
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 2. Klasy gramatyczne

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

Tabela 3. Skróty nazw klas gramatycznych oraz ich formy hasłowe.