Morfeusz Reloaded

Marcin Woliński
Institute of Computer Science, Polish Academy of Sciences
Warszawa, Poland
wolinski@ipipan.waw.pl

Abstract

The paper presents recent developments in Morfeusz – a morphological analyser for Polish. The program, being already a fundamental resource for processing Polish, has been reimplemented with some important changes in the tagset, some new options, added information on proper names, and ability to perform simple prefix derivation.

The present version of Morfeusz (including its dictionaries) is made available under the very liberal 2-clause BSD license. The program can be downloaded from `http://sgjp.pl/morfeusz/`.

Keywords: morfological analysis and generation, Polish, finite state automata

1. Introduction

Morfeusz is a morphological analyser for Polish. During its over 10 years long history it has established its position as a basic resource for morphological processing of Polish. Morfeusz was used for the annotation of The IPI PAN Corpus of Polish (Przepiórkowski, 2004) and National Corpus of Polish, NKJP, (Przepiórkowski et al., 2011), it serves as a basis for several taggers: (Dębowski, 2004), TaKIPI (Piasecki, 2007), PANTERA (Acedański, 2010), WMBT (Radziszewski and Śniatowski, 2011), Concraft-pl (Waszczuk, 2012). Moreover, Morfeusz was integrated with several parsing tools (Spejd (Przepiórkowski, 2008), Świgra (Woliński, 2004), SproUT (Piskorski et al., 2004)) as well as with the multiword expression toolkit Multiflex (Savary, 2005). During these years we have accumulated some experience and have identified its drawbacks. Now the time has come for an overhaul of the program.

First version of Morfeusz (Woliński, 2006) was based on approximated description of Polish inflection. Soon the data was replaced with inflectional data coming from the *Grammatical dictionary of Polish*, SGJP (Saloni et al., 2007), which is much richer and more precise. At some point this version was released as an open source program. Then the data of SGJP was merged with the community developed dictionary of Polish (`sjp.pl`), resulting in the largest freely available inflectional dictionary of Polish – Polimorf (Woliński et al., 2012). Currently Morfeusz is available in both SGJP and Polimorf flavours.

The new version of the program described here includes not only an analyser but also a compatible generator. Morfeusz is getting less tightly bound to its dictionary. The present version provides an infrastructure for including domain dictionaries or replacing the basic one completely. We are also working on some optimisations of finite automata used in the program.

2. Segmentation, Morphological Analysis

As for segmentation (or tokenization), we assume that segments cannot contain blanks so each segment is contained within a word. However, we allow for words consisting of several segments. A simple example is words containing punctuation characters that have to be interpreted separately (we have a separate tag `interp` for punctuation).

The next level of complication involves some productive mechanisms in the language which introduce myriads of words of very low textual frequency. Polish adjectives have the ability to form compounds like *zielono-niebieski* meaning ‘partly green and partly blue’ and *zielononiebieski* meaning ‘having a color between green and blue’. This works not only for colours: ‘a box made of wood and metal’ can be *drewniano-metalowe pudełko* and ‘a Polish-Czech-Hungarian summit’ is *szczyt polsko-czesko-węgierski*. Including such lexemes in the dictionary does not make much sense, since the mechanism is very regular and the meaning of a compound can be determined from its components. We have decided to split such formations into several segments. Unfortunately the hyphen is not an obvious segment boundary in Polish, since it is used in inflection of acronyms, e.g., *PRL-u* (genitive of PRL, the acronym for ‘People’s Republic of Poland’).

These facts lead to the conclusion that proper segmentation for Polish has to be dictionary-based.

We assume that an inflectional dictionary consists of entries describing some abstract units of the language. We call these units *lexemes*. A lexeme can be considered to be a set of other abstract units — namely *grammatical forms*. Lexemes gather sets of forms which have similar relation to the reality (e.g., all denote the same physical object) and differ in some regular manner. The differences between forms are described with values of grammatical categories attributed to them. Forms are represented in texts by segments.

For identifying the lexemes we will use *lemmas* (*base forms*), which traditionally have the shape of one of the forms belonging to the lexeme but should be in fact considered arbitrary unique identifiers (see also Section 4.2.).

By *morphological analysis* we will understand the interpretation of segments as grammatical forms. Technically that means assignment of a lemma and a tag. The lemma identifies a lexeme and the tag contains values of grammatical categories specifying the form.

In case of ambiguity, the result of morphological analysis includes all possible interpretations. We do not pay attention to the context that a word occurs in. In this setting, morphological *tagging* consists of morphological analysis and contextual disambiguation.

Figure 1 presents an example of morphological analysis.

0	1	*Mam*	MAMA [MOTHER] MAMIĆ [TO BEGUILE] MIEĆ [TO HAVE]	subst:pl:gen:f impt:sg:sec:imperf fin:sg:pri:imperf
1	2	*próbkę*	PRÓBKA [SAMPLE]	subst:sg:acc:f
2	3	*analizy*	ANALIZA [ANALYSIS]	subst:sg:gen:f subst:pl:nom.acc.voc:f
3	4	*morfologicznej*	MORFOLOGICZNY [MORPHOLOGICAL]	adj:sg:gen.dat.loc:f:pos
4	5	.	.	interp

Figure 1: Morphological interpretations for the text *Mam próbkę analizy morfologicznej.* ('I have a sample of morphological analysis.')

Each row of the table includes one morphological interpretation, the lines separate groups of interpretations for respective segments. The input text was segmented into tokens (in particular the full stop was separated from the word *morfologicznej*). Corresponding lemmas were provided in the third column. The last column presents tags describing the values of grammatical categories of particular forms.

The word *mam* has three interpretations: the genitive plural form of the noun MAMA, the imperative of the verb MAMIĆ and the present tense form of the verb MIEĆ. The word *analizy* was unambiguously associated with the lemma ANALIZA but with two possible tags representing singular and plural form in different grammatical cases.

The tags are positional. The first position defines the part of speech (more precisely: the flexeme, see below), the following ones stand for the values of grammatical categories of each class. For instance, the tag subst stands for a noun, it is followed by the values of the number, case and gender. The tags are usually abbreviated forms of Latin value names.

3. The Tagset

The tagset of Morfeusz was originally developed for the IPI PAN Corpus of Polish (Przepiórkowski and Woliński, 2003b; Przepiórkowski, 2003; Woliński, 2003; Woliński and Przepiórkowski, 2001). The main criteria for delimiting grammatical classes (parts of speech) in the tagset were morphological (how a given lexeme inflects; e.g., nouns inflect for case and number, but not gender) and morphosyntactic (in which categories forms agree with other forms; e.g., nouns agree in gender with adjectives and verbs).

The tagset is based on the notion of a *flexeme* – a morphosyntactically homogeneous set of forms belonging to the same lexeme (Przepiórkowski and Woliński, 2003a; Bień, 1991; Saloni, 1974). For example, past forms of a verb constitute a flexeme separate from, e.g., present tense forms, since the former inflects for gender while the latter does not. Deverbal nouns (gerunds) and adjectives (participles) form separate flexemes as clearly different from finite forms of verbs (for one, they do not inflect for person). However, Morfeusz assigns the infinitive as a lemma to gerunds and participles, which means that we (somewhat implicitly) treat these flexemes as parts of a broad verbal lexeme. This makes sense for further processing the text since valence and semantics of these forms are in a regular relation with those of finite forms of verbs. Thus, a lexeme can be considered to be a set of flexemes which are sets of forms.

In total there are 13 different verbal flexemes. If they are to be considered collectively as verbal forms, the processing system has to maintain their list. The relation is purely deterministic.

We call the tagset used in Morfeusz morphosyntactic since some attributes contained in the tags are not of inflectional nature. For example we provide information on gender for nouns, although Polish nouns do not inflect for gender. Nonetheless, gender is included in the tags as an important attribute of nominal lexemes describing their syntactic features.

The tagset uses a very detailed system of 9 genders (Przepiórkowski and Woliński, 2003c) based on the works of Saloni (1976). This system was reduced to 5 genders for the NKJP (Przepiórkowski, 2009), which simplifies automatic tagging. Morfeusz, however, uses the more detailed classification, since projecting it to 5 genders is trivial but the opposite transformation is not.

The most controversial feature of the Morfeusz and NKJP tagsets concerns movable inflections. In Polish, endings of past tense forms of verbs can be detached from the verb form under some conditions. This is illustrated with the following examples:

(1) *Nie wiedziałem, że to czytaliście.*
Not known-I that it read-you
'I didn't know that you have read this.'

(2) *Nie wiedziałem, żeście to czytali.*
Not known-I that/aux-you it read
'I didn't know that you have read this.'

The construction in the second example is probably more common in less formal texts, but with some complementizers (mainly used in the conditional) the detachment is obligatory:

(3) **Przyszedłbym, gdyby to czytaliście.*
Would-have-come-I if it read-you

(4) *Przyszedłbym, gdybyście to czytali.*
Would-have-come-I if/aux-you it read
'I would have come if you read this.'

This means that movable inflections have to be accounted for in a linguistically adequate tagset of Polish. In the Morfeusz tagset it was decided to describe these inflections as separate, including the form in example (1). The same was true of the particle *by* forming conditional, which means forms like *przyszedł·by·m* are reported by Morfeusz as three tokens. This decision simplified the tagset: the tags for past forms report only the number and gender, while the person (and the number again) is marked on the movable inflections. The word *widziałem* is analysed as:

0	1	*widział*	WIDZIEĆ	praet:sg:m1.m2.m3:imperf
1	2	*em*	BYĆ	aglt:sg:pri:imperf:wok

Moreover, there are no tags for conditional mood, only the particle *by* marks the conditional. We report only the form and not the function.

We hoped this system would get accepted by the community. It has obviously been adopted by NKJP.[1] But from the people building simpler, less linguistically oriented processing chains we received a stream of complaints. While they acknowledged the merits of the theoretical model, it caused them too much trouble since even in simple circumstances it requires the processing system to consider multi-token units.

For the new Morfeusz we have decided to provide an alternative. The old system stays in place and we will probably use it. However, an option is provided to analyse past and conditional forms as entities. This means that we have to add a series of tags with gender and person combined for the past tense and for the conditional:

widziałem	praet:sg:m1.m2.m3:pri:imperf
widziałeś	praet:sg:m1.m2.m3:sec:imperf
widział	praet:sg:m1.m2.m3:ter:imperf
widziałam	praet:sg:f:pri:imperf
...	...
widziałbym	cond:sg:m1.m2.m3:pri:imperf
widziałbyś	cond:sg:m1.m2.m3:sec:imperf
widziałby	cond:sg:m1.m2.m3:ter:imperf
widziałbym	cond:sg:f:pri:imperf
...	...

It is worth noting that even with these tags one has to take analytical forms into account when determining the tense and mood of Polish verbs (analytical future involving an "infinitive" or "past": *Będę to czytać/czytał.* and analytical conditional: *Ja bym tego nie czytał.*).

4. Programming Interface of Morfeusz

The analyser is provided as a dynamic link/shared library which can be easily incorporated into programs. We provide compiled versions for 32 and 64 bit versions of Linux, Mac OS X, and Windows. On other systems it should be relatively easy to compile Morfeusz from sources. For demonstrative purposes we provide a simple command line client for the library and a graphical interface.

Morfeusz is written in C++. The new version provides an object-oriented API. However, it is well known, that C++ libraries are not portable across compilers. For that reason we provide as well a pure C interface that does not have this problem. The distribution contains as well bindings for using Morfeusz in programs written in Java, Python, Perl (these are generated with SWIG), and SWI-Prolog.[2] The new interface is thread-safe, separate instances of the analyser object can be used in parallel threads of a program.

Nowadays most systems seem to use Unicode for representing text. Unfortunately, Unix systems prefer encoding it as UTF-8, while Java and Windows use UTF-16. Moreover, some pieces of Windows (e.g., the console) still use code pages. To accommodate this situation Morfeusz can process text encoded in UTF-8, UTF-16, ISO8859-2, CP852, and CP1250.

The behaviour of the library can be controlled with several options provided by the API.

4.1. Analysis

In analysis mode Morfeusz takes a fragment of text (e.g., a line or a sentence) and returns a list of morphological interpretations of its words.

Due to the assumed rules of segmentation, it is possible to obtain an ambiguous segmentation in the results of morphological analysis. For that reason, we find it convenient to represent the results as a directed acyclic graph of interpretations (cf. Fig. 2). This idea was utilised and proved useful in the parser *Świgra* (Woliński, 2004; Woliński, 2005). A similar representation is used by Obrębski (2002).

Nodes in the graph represent positions in the text (between the segments) while edges represent possible segment interpretations. The edges are labelled with triples consisting of a segment, a lemma, and a tag.

Technically, the DAG of interpretations is represented in the results of Morfeusz as a list:

0	1	*Co*	CO	subst:sg:nom.acc:n2
1	2	*ś*	BYĆ	aglt:sg:sec:imperf:nwok
0	2	*Coś*	COŚ	subst:sg:nom.acc:n2
2	3	*zrobił*	ZROBIĆ	praet:sg:m1.m2.m3:perf
3	4	*?*	?	interp

The numbers represent the nodes of the DAG. The third column lists segments, the fourth one lemmas, and the fifth one tags. A tag consists of values separated with colons. The first value denotes the flexeme (e.g., subst for a noun), the rest contains values of grammatical categories (e.g., sg for singular number). Some tags are presented in a compact form where multiple possible values of a category are joined in one tag with dots (e.g., m1.m2.m3 for three subgenders of the masculine gender).

The interpretations are generated in no particular order. In particular, the order is not based on frequency of forms.

The new version of Morfeusz adds two new pieces of information to its interpretations: classification of proper names and stylistic labels.

The classification is rather simplified: proper names are classified as geographical names, organisations, persons and other. The class of persons is subdivided into: first

[1] OK, it caused problems even there: each time the size of the corpus in tokens was to be reported, it was necessary to explain how tokens of NKJP corresponded to words.

[2] We prefer the library+bindings architecture to implementing the analyser within some particular NLP toolkit, since that way we are not binding users to that toolkit.

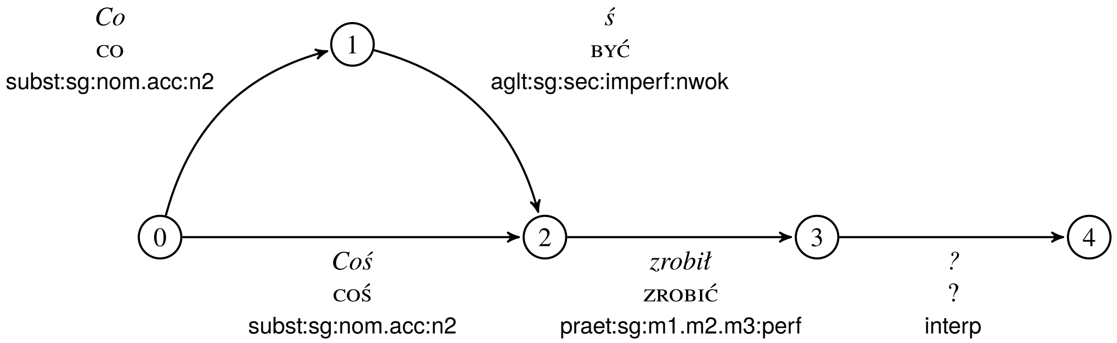

Figure 2: Morphological interpretations for the sentence *Coś zrobił?* with ambiguous segmentation. The sentence can be read as 'What (*co*) have you done (*-ś zrobił*)?' or 'Did he do (*zrobił*) anything (*coś*)?'.

names, last names, pseudonyms, and patronyms. Appropriate labels have been added for the whole scope of Polimorf dictionary.

Obviously, this information is not part of morphological tagging. Nonetheless, it was included since it can be useful even when semantic processing is not done, e.g., for parsing names.

The labels include, e.g., 'archaism', 'colloquialism', 'coarse/vulgarism', as well as those signalling terminology of specific domains (e.g., 'chemical'). The labels allow to filter out some interpretations when the domain of the text analysed is known. For example, when processing hospital documentation one can safely ignore archaic words and colloquialisms. It is probably also safe to ignore vocative forms of nouns (which can be homonymous with nominatives) and imperatives of verbs. Both make sense for limiting homonymy in Polish.

4.2. Generation

The generating module of Morfeusz has two flavours. The first takes a lemma and generates the full paradigm of the given lexeme (i.e. forms of all flexemes comprising the lexeme). The second takes a lemma and a tag and generates only forms matching that tag (there can be more than one). In both cases the program returns structures closely resembling results of analysis (including stylistic labels and proper name information).

In case of homonymy we use lemmas containing disambiguating elements. For example, in the case of the lexeme PIEC which in Polish can be a verb ('to bake') or a noun ('an oven') the lemmas have the form PIEC:V and PIEC:S, respectively. If there is more than one lexeme of the same grammatical class arbitrary numbers are used. For instance, the dictionary contains lexemes ZAMEK:S1 ('a castle', with genitive *zamku*) and ZAMEK:S2 ('a lock', with genitive *zamka*). In this example the difference in inflection is what forces us to introduce two separate lexemes. Even if a word has several clearly separate meanings we will consider it a single lexeme if the meanings share the whole inflectional paradigm (as in the case of the noun PARA 'a couple' or 'a vapour').

Since we lemmatise deverbal flexemes to the infinitive, we are free from some cases of systematic homonymy in Polish. For example, the gerund *mieszkanie* derived from the verb MIESZKAĆ ('to live/inhabit') is homonymous with a noun ('a flat'). However, the gerund is a part of the verbal lexeme with the lemma MIESZKAĆ and the lemma MIESZKANIE points unambiguously to the noun. The same goes for homonymy between adjectival participles and regular adjectives.

The SGJP dictionary contains only about 10,000 lemmas with a disambiguator.

The use of arbitrary numbers is a bit unfortunate. But the analysing module always generates lemmas that can be fed back to the generator. Thus, if we analyse the word *zamka*, we will learn that the corresponding lemma is ZAMEK:S2. Moreover, to ease this situation the generating module accepts lemmas without the disambiguating part and generates forms of all matching lexemes in response (so a call with lemma `piec` will result in both verbal and nominal forms generated).

5. Dictionaries

Previous versions of Morfeusz used to be tightly coupled with a compiled-in dictionary. In the present version we want to be able to adapt the dictionary to particular needs.

The tool described in our previous paper (Woliński et al., 2012) allows to work simultaneously on several dictionaries. It is used for development of both dictionaries distributed with Morfeusz. But it can be used as well to develop domain dictionaries. Such need arises when processing, e.g., hospital documentation as some of medical terminology is too specific to be included in a general dictionary. Moreover, hospital documentation uses specific set of abbreviations which should not be considered when processing general text.

The tool allows to export a list of forms from an arbitrary set of dictionaries contained in the system. The dictionary compiling tool of Morfeusz turns such lists into a binary representation used by the Morfeusz library. Obviously lists of forms can be also prepared by other means and merged with those in Morfeusz distribution or replace them completely.

5.1. Precompiled Dictionaries

Two inflectional dictionaries are included in the program's distribution available at `http://sgjp.pl/`

	SGJP	Polimorf
lexemes	264166	315055
forms	4037250	3844535

Table 1: Sizes of precompiled dictionaries of Morfeusz

`morfeusz/`. The SGJP dictionary contains data from the second edition of the *Grammatical dictionary of Polish* (Saloni et al., 2012). The Polimorf dictionary is a merger of SGJP with Morfologik dictionary based on community developed `sjp.pl` dictionary (Woliński et al., 2012). Table 1 presents the sizes of these dictionaries.

5.2. Compiling Dictionaries

The core dictionary of Morfeusz maps segments to sets of possible interpretations. The dictionary is represented as a minimal deterministic finite state automaton with the transitions labelled with consecutive letters of the words and the accepting states labelled with interpretations. The automaton is generated with a variant of the algorithm presented by Daciuk et al. (2000).

Figure 3 presents the form of a dictionary that is fed to the dictionary compiler. Each row contains one grammatical form. Five columns are separated with tabulation (U+0008). Their content is as follows:

1	segment
2	lemma (including disambiguator if necessary)
3	tag
4	proper name/common classification
5	stylistic label(s) (optional)

The list used by Morfeusz contains all the inflected forms of lexemes including the special adjectival and numeral forms used in compounding. It includes as well special segments that cannot appear by themselves but can be combined with another segment to form a complete word (see below).

6. Segment Joining (Compounding)

As explained above, Morfeusz treats some orthographic words as consisting of several tokens interpreted separately. This mechanism is used, for example, to analyse compound adjectival forms like *biało-czerwony* ‘red and white’.

In the new version we have also taken into account less common adjectival compounds without a hyphen (*ciemnoczerwony* ‘dark red’) and compounds including numeral element (*dwurzędowy* ‘[having] two rows’, *drugorzędowy* ‘[belonging to the] second row’). Also the rules for attaching movable inflections were extended to some inflecting lexemes, mainly pronouns (*myśmy* ‘we/1per.pl.aux’).

The compounding mechanism is also used to guess lexemes unknown to Morfeusz that can be derived with a list of frequently used Polish prefixes. The list of prefixes is kept in the dictionary, which means users can adjust the list depending on the domain/genre of texts.

To describe allowed combinations of segments, each segment in the dictionary is associated with a *segment type*. These are defined in an additional file that is used by the dictionary builder together with the list of forms. Segment types can be associated with specific tags or with specific forms of specific lexemes. The latter take precedence as exceptions. Another section of the file defines possible combinations of segments in terms of regular expressions over segment types.

This mechanism allows us to experiment with segmentation rules without recompiling the program. This is useful since, e.g., the possibility of agglutinative formations occurring in the text depends on the genre of the text. The pre-compiled dictionaries of Morfeusz contain several variants of the rule set that can be selected with options at run-time.

7. Summary

Morfeusz together with its SGJP and Polimorf dictionaries is available under the very liberal 2-clause BSD license. This makes it accessible both for scientific and commercial uses. Mofeusz’s model of inflection has a linguistically sound base. The changes in the present version make the program more attractive for simplified practical solutions.

8. References

Acedański, S. (2010). A morphosyntactic brill tagger with lexical rules for inflectional languages. In *Advances in Natural Language Processing: Proceedings of the 7th International Conference on Natural Language Processing, IceTAL 2010, Reykjavík, Iceland*, pages 3–14. Springer-Verlag.

Bień, J. S. (1991). *Koncepcja słownikowej informacji morfologicznej i jej komputerowej weryfikacji.* Rozprawy Uniwersytetu Warszawskiego. Wydawnictwa Uniwersytetu Warszawskiego.

Daciuk, J., Mihov, S., Watson, B., and Watson, R. (2000). Incremental construction of minimal acyclic finite state automata. *Computational Linguistics*, 26(1):3–16, April.

Dębowski, Ł. (2004). Trigram morphosyntactic tagger for Polish. In Kłopotek, M. A., Wierzchoń, S. T., and Trojanowski, K., editors, *Intelligent Information Processing and Web Mining. Proceedings of the International IIS:IIPWM’04 Conference held in Zakopane, Poland, May 17-20, 2004*, pages 409–413. Springer.

Obrębski, T. (2002). *Automatyczna analiza składniowa języka polskiego z wykorzystaniem gramatyki zależnościowej.* Rozprawa doktorska, Instytut Podstaw Informatyki PAN, Warszawa.

Piasecki, M. (2007). Polish tagger TaKIPI: Rule based construction and optimisation. *Task Quarterly*, 11(1–2):151–167.

Piskorski, J., Homola, P., Marciniak, M., Mykowiecka, A., Przepiórkowski, A., and Woliński, M. (2004). Information extraction for Polish using the SProUT platform. In Kłopotek, M., Wierzchoń, S., and Trojanowski, K., editors, *Intelligent Information Processing and Web Mining*, Advances in Soft Computing, pages 227–236. Springer.

Przepiórkowski, A. and Woliński, M. (2003a). A flexemic tagset for Polish. In *Proceedings of the Workshop on Morphological Processing of Slavic Languages, EACL 2003*, pages 33–40.

Przepiórkowski, A. and Woliński, M. (2003b). A morphosyntactic tagset for Polish. In Kosta, P., Błaszczak, J., Frasek, J., Geist, L., and Żygis, M., editors, *Investigations into Formal Slavic Linguistics (Contributions of the*

```
Gdańsk     Gdańsk  subst:sg:acc:m3  geograficzna
Gdańsk     Gdańsk  subst:sg:nom:m3  geograficzna
Gdańska    Gdańsk  subst:sg:gen:m3  geograficzna
Gdański    Gdańsk  subst:pl:nom:m3  geograficzna
Gdańskiem  Gdańsk  subst:sg:inst:m3 geograficzna
funkcja    funkcja subst:sg:nom:f   pospolita
funkcjach  funkcja subst:pl:loc:f   pospolita
funkcjami  funkcja subst:pl:inst:f  pospolita
funkcje    funkcja subst:pl:acc:f   pospolita
funkcje    funkcja subst:pl:nom:f   pospolita
funkcje    funkcja subst:pl:voc:f   pospolita     rzad.
funkcji    funkcja subst:pl:gen:f   pospolita
funkcji    funkcja subst:sg:gen:f   pospolita
funkcjo    funkcja subst:sg:voc:f   pospolita     rzad.
funkcjom   funkcja subst:pl:dat:f   pospolita
funkcyj    funkcja subst:pl:gen:f   pospolita     arch.
```

Figure 3: Textual form of an inflectional dictionary suitable for Morfeusz

Fourth European Conference on Formal Description on Slavic Languages), pages 349–362.

Przepiórkowski, A. and Woliński, M. (2003c). The unbearable lightness of tagging: A case study in morphosyntactic tagging of Polish. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC-03), EACL 2003*, pages 109–116.

Przepiórkowski, A., Bańko, M., Górski, R. L., and Lewandowska-Tomaszczyk, B., editors. (2011). *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Przepiórkowski, A. (2003). Składniowe uwarunkowania znakowania morfosyntaktycznego w korpusie IPI PAN. *Polonica*, XXII–XXIII:57–76.

Przepiórkowski, A. (2004). *Korpus IPI PAN. Wersja wstępna*. Instytut Podstaw Informatyki PAN, Warszawa.

Przepiórkowski, A. (2008). *Powierzchniowe przetwarzanie języka polskiego*. Exit, Warsaw.

Przepiórkowski, A. (2009). A comparison of two morphosyntactic tagsets of Polish. In Koseska-Toszewa, V., Dimitrova, L., and Roszko, R., editors, *Representing Semantics in Digital Lexicography: Proceedings of MONDILEX Fourth Open Workshop*, pages 138–144, Warsaw.

Radziszewski, A. and Śniatowski, T. (2011). A memory-based tagger for Polish. In Vetulani, Z., editor, *Proceedings of the 5th Language & Technology Conference*, pages 556–560, Poznań.

Saloni, Z., Gruszczyński, W., Woliński, M., and Wołosz, R. (2007). *Słownik gramatyczny języka polskiego*. Wiedza Powszechna, Warszawa.

Saloni, Z., Woliński, M., Wołosz, R., Gruszczyński, W., and Skowrońska, D. (2012). *Słownik gramatyczny języka polskiego*. Warszawa, 2 edition.

Saloni, Z. (1974). Klasyfikacja gramatyczna leksemów polskich. *Język Polski*, LIV:z.1, 3–13, z.2, 93–101.

Saloni, Z. (1976). Kategoria rodzaju we współczesnym języku polskim. In *Kategorie gramatyczne grup imiennych we współczesnym języku polskim*, pages 41–75. Ossolineum, Wrocław.

Savary, A. (2005). MULTIFLEX. User's Manual and Technical Documentation. Version 1.0. Technical Report 285, LI-François Rabelais University of Tours, France.

Waszczuk, J. (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)*, pages 2789–2804, Mumbai, India.

Woliński, M. and Przepiórkowski, A. (2001). Projekt anotacji morfosynktaktycznej korpusu języka polskiego. Prace IPI PAN 938, Instytut Podstaw Informatyki Polskiej Akademii Nauk.

Woliński, M., Miłkowski, M., Ogrodniczuk, M., Przepiórkowski, A., and Szałkiewicz, Ł. (2012). PoliMorf: a (not so) new open morphological dictionary for Polish. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012*, pages 860–864, Istanbul, Turkey. ELRA.

Woliński, M. (2003). System znaczników morfosyntaktycznych w korpusie IPI PAN. *Polonica*, XXII–XXIII:39–55.

Woliński, M. (2004). *Komputerowa weryfikacja gramatyki Świdzińskiego*. Ph.D. thesis, Instytut Podstaw Informatyki PAN, Warszawa, December.

Woliński, M. (2005). An efficient implementation of a large grammar of Polish. In Vetulani, Z., editor, *Human Language Technologies as a Challenge for Computer Science and Linguistics. 2nd Language & Technology Conference April 21–23, 2005*, pages 343–347, Poznań.

Woliński, M. (2006). Morfeusz — a practical tool for the morphological analysis of Polish. In Kłopotek, M., Wierzchoń, S., and Trojanowski, K., editors, *Intelligent Information Processing and Web Mining, IIS:IIPWM'06 Proceedings*, pages 503–512. Springer.