Morfeusz — a Practical Tool for the Morphological Analysis of Polish

Marcin Woliński

Institute of Computer Science, Polish Academy of Sciences,
ul. Ordona 21, 01-237 Warsaw, Poland

Abstract. This paper describes a morphological analyser for Polish. Its features include a large dictionary, a carefully designed tagset, presentation of results as a DAG of interpretations, high efficiency, and free availability for non-commercial use and scientific research.

Introduction

The topic of this paper is a morphological analyser for Polish developed by Zygmunt Saloni and Marcin Woliński. To be more precise, Saloni is the author of linguistic data used in the analyser (cf. section 2), while Woliński is responsible for the programming part.

The key factor that triggered development of Morfeusz was the availability of the second edition of Tokarski's book [19] (prepared by Saloni) without many omissions and mistakes of the first edition.[1] Another factor was the necessity of a more subtle analysis of Tokarski's data than that performed by Szafran's SAM [15], the first morphological analyser based on Tokarski's data.

The authors have decided to make the program available free of charge for non-commercial use and scientific research. The program can be downloaded from the Internet address `http://nlp.ipipan.waw.pl/~wolinski/morfeusz/`. An on-line demo of the program is also available.

Although there exist several morphological analysers for Polish (cf. [5]), so far only SAM has been available for free, so we feel that Morfeusz fills an important gap on the market. And, indeed, the program, whose development started in 2000, has already been used in several projects, including annotation of the IPI PAN Corpus, two taggers for Polish by Łukasz Dębowski [3] and Maciej Piasecki [7], a DCG parser Świgra [21], a TRALE parser by Adam Przepiórkowski, an information extraction system [8], and some student projects.

1 The task of morphological analysis

Given a text (being a sequence of characters and blanks) it is relatively easy to conceive the notion of an orthographic *word* — a maximal sequence of

[1] About 1000 lemmatisation rules were improved.

characters not including any blanks or punctuation. Unfortunately this rather technical notion is not suitable as the unit considered in morphology (at least for Polish). In some cases (see section 5) it is reasonable to interpret some parts of a word, which we call *segments* (or *tokens*).

A dictionary consists of entries describing some abstract units. We call these units *lexemes*. A lexeme can be considered 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.

We need some means of identifying lexemes. For that 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 some unique identifiers.

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 any attention to the context that a word occurs in. According to this view, morphological *tagging* consists of morphological analysis and contextual disambiguation.

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. Gender is included in the tags because it is an important attribute of nominal lexemes describing their syntactic features.

2 Tokarski's description of Polish inflection

It seems that Jan Tokarski was the first Polish linguist who started to build a computational description of Polish inflection. We find remarks on "teaching inflection to a computer" already in his 1951 book on conjugation [17]. In this book, he presents an if-then-else approach (*if the last letter of the word in question is* y *then depending on the preceding letter consider the following cases...*). About ten years later he switched to a data driven approach and started to couple endings of inflected forms with endings of base forms (strictly speaking, these are not inflectional morphemes, nor strings of morphemes, rather just strings of letters which change with inflection).

This idea took its final shape in the book [19]. Tokarski has not finished this work himself. The book was prepared by Zygmunt Saloni on the basis of author's hand-written notes and its first edition appeared in 1993 after Tokarski's death.

The book provides information on virtually all possible endings of Polish words, and how to lemmatise them. Typical lemmatisation rules have the following form:

-kście	*mIV LV*	-kst	kontekście, tekście, mikście	(6)
-kście	*żIV D*	-ksta	sekście	

The first row states that a word ending with *-kście* can be a form of a lexeme with the base form ending with *-kst*. In such a case, the lexeme is a masculine noun of Tokarski's inflectional group[2] *mIV* and the form in question is singular locative or vocative (LV). The rest of the row consists of examples of words, which can be analyzed according to it.

According to [19, p. 8] "the algorithm of automatic morphological analysis of a Polish text can proceed as follows:

1. the machine cuts some string $a_{i+1}, ..., a_n$ from the word $a_1, ..., a_n$ and finds a matching row in [Tokarski's] index,
2. the machine reads the grammatical characteristic from the second field of the row, and the string of the lemma $b_{i+1}, ..., b_m$ — from the third,
3. the word $a_1, ..., a_i, b_{i+1}, ..., b_m$ is searched for in the list of admissible lemmas and, if found, the word $a_1, ..., a_n$ is considered to represent a form of the same lexeme as the lemma."

The first attempt to use Tokarski's data for morphological analysis was a work of Krzysztof Szafran, who helped Saloni to prepare the first edition of the book. During his first experiments, a comprehensive list of lemmas was not available, which led to massive overgeneration of interpretations. Fortunately, the list of all lemmas in Doroszewski's dictionary of Polish [4] became available thanks to the work of Robert Wołosz.[3] In SAM analyser Szafran used a version of this list enriched with identifiers of Tokarski's inflectional groups.

3 The inflectional dictionary of Morfeusz

Although the results of SAM are much better than these of Tokarski's index used without a dictionary, there is still much room for improvement.

First, Tokarski's rules can be divided into two categories: general and specific. The general rules apply to forms of numerous lexemes, while the specific ones are meant to be used only for few forms listed in the example column. This information is ignored by SAM.

[2] Tokarski's groups provide only approximate information on the type of inflection. They are not precise inflectional patterns.

[3] The list can be downloaded from the Internet: `ftp://ftp.mimuw.edu.pl/pub/users/polszczyzna/a_terDor/`.

Second, about 43,000 forms are given explicitly in the example column. For these forms, other candidate words with the same morphosyntactic characteristics should be ignored (cases when variants are possible are marked in Tokarski's data).

In Morfeusz, Tokarski's index is not used directly. Instead a dictionary of all possible forms is generated and then compacted (as described in section 4). Thus Morfeusz has currently no capability of guessing unknown forms.

The starting point for generating the dictionary is the list of lemmas of Doroszewski's dictionary with identifiers of inflectional groups attached (some new entries were added to the list and some archaic ones were pruned). For each element of this list, all matching rows of Tokarski's index are considered. A row matches if the lemma in question ends with an ending specified in the third field of the row and the identifier of the inflectional group is equal to the one given in the second column. After the matching rows are gathered, the information on grammatical features is decoded — in some cases a row describes multiple forms. And then, for each combination of features, the best candidate word is chosen. If a word is given explicitly as an example for the row, it is considered the best candidate. The next rank is given to words generated by the general rows. And only if no other candidate is at hand, we use a word generated by a specific row but not listed as an example.

Actually the procedure is more complicated because of so called generalised rows, which compactly describe groups of forms analysed in a regular way. For example, all plural locative forms of Polish nouns can be derived from the respective dative form:

.ach *S lL* ⇒ .om *S lD* domach, drzwiach, polach, latach (70000)

Unfortunately Tokarski's description is not sufficiently selective for some forms. For example, consider the singular genitive of masculine nouns, which can end with *-a* or *-u* on purely lexical basis. The index contains the following two rows for nouns with the lemma ending with *-om*:

oma	*mIV G*	om	kondoma, oszołoma?, gnoma, astronoma, anatoma (10)
omu	*mIV G*	om	idiomu, poziomu, slalomu, przełomu, symptomu (90)

When generating the dictionary there is no way of checking what is the correct genitive for a given noun (e.g., *atom*, *agronom*), so Morfeusz accepts both candidate words (e.g., **atoma* and *atomu*, *agronoma* and **agronomu*).

The above procedure is used only for non-verbal lexemes, since for verbs we have at hand much more precise description developed by Saloni [13]. That book presents the inflection of about 12000 Polish verbs, but in fact the data covers about 27000 verbal lexemes. Saloni's inflectional patterns include all verbal forms, as well as regular derivatives such as gerunds and adjectival participles (cf. [14]).

As a result of the processing, we get a list of all possible grammatical forms:

```
...
kontekstu 1 subst:sg:gen:m3
konteksty 1 subst:pl:nom.acc:m3
kontekstów 2 subst:pl:gen:m3
kontekście 4st subst:sg:loc.voc:m3
kontem 2o subst:sg:inst:n1.n2
...
```

(The lemmas in the list are provided implicitly. E.g., `4st` above means: to get the lemma for *kontekście* strip the last 4 letters and replace them with *st.*)

In the current version, the dictionary consists of about 115,000 lexemes, and 4,750,000 forms[4], which provides for recognising about 1,700,000 different Polish words.

4 The representation of the linguistic data in Morfeusz

Apart from building a suitable inflectional dictionary the construction of a morphological analyser can be seen as an exercise in domain-specific data compression. The task is to find a compact representation of the dictionary that would provide for fast access.

As said above, the core dictionary of Morfeusz maps words 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.* [2].

The key trick that provides for an acceptable size of the automaton is that final states do not include lemmas. Instead they contain instructions how to make the lemma from a given word. These instructions take the form: 'replace the given number of characters from the end of the word with a given string'. Since the instructions tend to be the same for analogous forms of various lexemes, the minimal automaton is smaller. If the full lemmas were put in the accepting states, the automaton would be very close to an uncompressed trie.

We should note, however, that not every aspect of Polish inflection can be modelled conveniently with a single automaton.

First, inflection in Polish affects not only word endings. Polish gerunds and some participles inflect for negation by prepending the letters *nie*, and

[4] This number should not be taken too seriously, since it heavily depends on the assumed tagset. A reasonable tagset could be presented for which this number would be twice as large or twice smaller.

the superlative degree of adjectives is formed by prepending the letters *naj* to the comparative degree. Including such forms directly in the dictionary would lead to an unnecessary growth of the automaton. The states of the automaton used for representing comparative case could not be reused for superlative case, since the morphosyntactic description in the respective final states would be different.

Second, there are some productive mechanisms in the language which allow for introducing myriads of words of very low textual frequency. E.g., it is possible to join some adjectival forms. Consider adjectives *zielony* (green) and *niebieski* (blue). Then *zielono-niebieski* means 'partly green and partly blue', while *zielononiebieski* means '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*. Introducing such lexemes would significantly increase the size of the dictionary, so the better solution is to split such words into multiple forms.

For these reasons we introduced a level of processing which describes acceptable ways of joining strings recognised by the core dictionary. This process is again conveniently represented with a finite state automaton. Each string in the core dictionary is given a label which we call a segment type. The segment types work as the input alphabet for the segment-joining automaton. If $\langle adja \rangle$ is the special ad-adjectival form of an adjective, and $\langle adjf \rangle$ is any 'regular' adjectival form, then the analyser should accept any forms matching the following regular expressions:

$$\langle adja \rangle^+ \langle adjf \rangle$$
$$(\langle adja \rangle \texttt{-})^+ \langle adjf \rangle$$

Similarly, if $\langle comp \rangle$ is a comparative form then $\texttt{naj}\langle comp \rangle$ is a superlative form. Note that these forms require different processing when it comes to lemmas. For superlative degree, the lemma is the same as for the comparative form, but the tag has to be changed accordingly. The compound adjectival forms are treated in our approach as sequences, the hyphen being a separate segment.

This mechanism provides also nice means for recognising strings of digits as numbers and for analysing words including agglutinative forms ('floating inflections') mentioned in the next section.

5 The IPI PAN Tagset

The morphological codes in Tokarski's index are very concise and rather inconvenient to deal with. Morfeusz replaces them with a carefully designed tagset — the IPI PAN Tagset.

The IPI PAN tagset [11,9,20] was developed by Marcin Woliński and Adam Przepiórkowski for the annotation of the IPI PAN Corpus of Polish.[5] 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).

One of the aims of the IPI PAN tagset was to define grammatical classes which are homogeneous with respect to inflection. However, a traditional verb lexeme contains forms of very different morphosyntactic properties: present tense forms have the inflectional categories of person and number, past tense forms have gender as well, and the impersonal *-no/-to* form is finite but does not have any inflectional categories. To overcome that problem we have decided to apply in the tagset the notion of *flexeme* proposed by Bień [1]. A flexeme is a morphosyntactically homogeneous set of forms belonging to the same lexeme (for a more detailed discussion see [10]). Thus a lexeme is a set of flexemes which are sets of forms.

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. This happens in the case of Polish 'floating inflections', which can be reasonably treated as weak forms of the verb *być* 'to be' (cf. [12]). We treat words expressing past tense of verbs as built of two segments. For example, *czytałem* is analysed as *czytał*, which is past form of the verb, and *em*, which is a floating inflection. Similarly, *czytałbym* is split into three segments: *czytał*, *by*, and *m*. Some adjectival formations mentioned in the previous section are split as well. There are, however, words containing a hyphen which are treated as one segment, e.g., *ping-pong* or *PRL-u*, which is an inflectional form of an acronym.

We have assumed the following grammatical classes: noun, adjective, ad-adjectival adjective (special form mentioned in section 4), post-prepositional adjective (form that is required after some prepositions, e.g. *[po] polsku* 'in Polish'), adverb, numeral, personal pronoun, non-past verb (present tense for imperfect and future for prefect verbs), auxiliary future of *być*, *l*-participle (past tense), agglutinative ('floating inflection'), imperative, infinitive, impersonal *-no/-to* form, adverbial contemporary and anterior participles, gerund, adjectival active and passive participles, *winien*-like verb, predicative, preposition, conjunction, particle-adverb.

A more detailed presentation of the tagset was given in the articles mentioned at the beginning of this section.

[5] See `http://korpus.pl`.

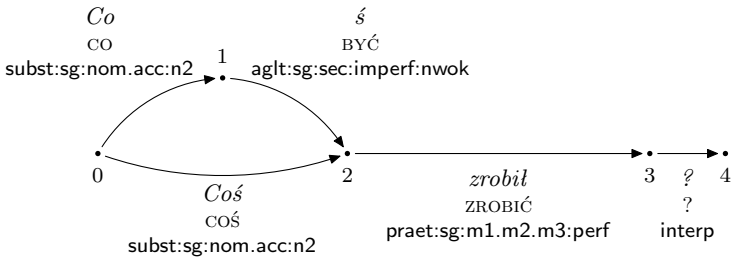

Fig. 1. Morphological interpretations for the sentence *Coś zrobił?*

6 The representation of the results of an analysis

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 (DAG, cf. Fig. 1). 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.

This idea was utilised and proved useful in Świgra parser [21,22]. A similar representation is used by Obrębski [6].

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 lemmas, and the fifth tags. A tag consists of values separated with colons. The first value denotes the grammatical class (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., n1.n2 for two possible neuter genders).

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

7 The Morfeusz library

The analyser is provided as a library which can be easily incorporated into programs. The library is provided as Linux shared object file (`.so`) and MS

Windows dynamic link library (`.dll`). The programming interface consists mainly of one function that takes as an argument a piece of text and returns a list of interpretations.

Morfeusz is written in C++ but the programming interface is in C, for portability between compilers. Some glue/interface code has been prepared by the authors that enables the use of Morfeusz in programs written in Perl and SWI Prolog. A Java module by Dawid Weiss is available separately. Morfeusz has also been interfaced with SICStus Prolog, SProUT information extraction system, and TRALE grammar.

Summary and outlook

Morfeusz recognises 96.6% of running words and 87.0% of word types of the corpus of Frequency Dictionary of Polish (about 500,000 words). For the IPI PAN Corpus (version 1.0 of the 'source' sub-corpus, almost 85 millions of words) the respective numbers are: 95.7% of words and 69% of word types.

The current version of Morfeusz's dictionary contains virtually no proper names. Doroszewski's dictionary is somewhat outdated, so some new Polish words are not recognised by Morfeusz. Another problem is overgeneration of forms, mentioned in section 3. We currently work on these issues.

An important planned extension of the program is to implement morphological generation and guessing of the forms of unknown lexemes.

Some technical improvements in the program are also planned. These include Unicode awareness and more options as to the form of results generated.

References

1. Janusz Stanisław Bień. *Koncepcja słownikowej informacji morfologicznej i jej komputerowej weryfikacji.* Rozprawy Uniwersytetu Warszawskiego. Wydawnictwa Uniwersytetu Warszawskiego, 1991.
2. Jan Daciuk, Stoyan Mihov, Bruce Watson, and Richard Watson. Incremental construction of minimal acyclic finite state automata. *Computational Linguistics*, 26(1):3–16, April 2000.
3. Łukasz Dębowski. Trigram morphosyntactic tagger for Polish. In Mieczysław A. Kłopotek, Sławomir T. Wierzchoń, and Krzysztof Trojanowski, 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, 2004.
4. Witold Doroszewski, editor. *Słownik języka polskiego PAN.* Wiedza Powszechna – PWN, 1958–1969.
5. Elżbieta Hajnicz and Anna Kupść. Przegląd analizatorów morfologicznych dla języka polskiego. Prace IPI PAN 937, Instytut Podstaw Informatyki Polskiej Akademii Nauk, 2001.
6. Tomasz Obrębski. *Automatyczna analiza składniowa języka polskiego z wykorzystaniem gramatyki zależnościowej.* PhD thesis, Instytut Podstaw Informatyki PAN, Warszawa, April 2002.

7. Maciej Piasecki and Grzegorz Godlewski. Reductionistic, tree and rule based tagger for Polish. In this volume, 2006.
8. Jakub Piskorski, Peter Homola, Małgorzata Marciniak, Agnieszka Mykowiecka, Adam Przepiórkowski, and Marcin Woliński. Information extraction for Polish using the SProUT platform. In Mieczysław Kłopotek, Sławomir Wierzchoń, and Krzysztof Trojanowski, editors, *Intelligent Information Processing and Web Mining*, Advances in Soft Computing, pages 227–236. Springer, 2004.
9. Adam Przepiórkowski. Składniowe uwarunkowania znakowania morfosyntaktycznego w korpusie IPI PAN. *Polonica*, XXII–XXIII:57–76, 2003.
10. Adam Przepiórkowski and Marcin Woliński. A flexemic tagset for Polish. In *Proceedings of the Workshop on Morphological Processing of Slavic Languages, EACL 2003*, pages 33–40, 2003.
11. Adam Przepiórkowski and Marcin Woliński. A morphosyntactic tagset for Polish. In Peter Kosta, Joanna Błaszczak, Jens Frasek, Ljudmila Geist, and Marzena Żygis, editors, *Investigations into Formal Slavic Linguistics (Contributions of the Fourth European Conference on Formal Description on Slavic Languages)*, pages 349–362, 2003.
12. Adam Przepiórkowski and Marcin Woliński. 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, 2003.
13. Zygmunt Saloni. *Czasownik polski. Odmiana, słownik.* Wiedza Powszechna, Warszawa, 2001.
14. Zygmunt Saloni and Marcin Woliński. A computerized description of Polish conjugation. In Peter Kosta, Joanna Błaszczak, Jens Frasek, Ljudmila Geist, and Marzena Żygis, editors, *Investigations into Formal Slavic Linguistics (Contributions of the Fourth European Conference on Formal Description on Slavic Languages)*, pages 373–384, 2003.
15. Krzysztof Szafran. Analizator morfologiczny SAM-95: opis użytkowy. TR 96-05 (226), Instytut Informatyki Uniwersytetu Warszawskiego, Warszawa, 1996.
16. Jan Tokarski. Fleksja polska, jej opis w świetle mechanizacji w urządzeniu przekładowym. *Poradnik Językowy*, 1961: z. 3 s. 97–112, z. 8 s. 343–353; 1962: z. 4 s. 145–158; 1963: z. 1 s. 2–21, z. 2 s. 55–77, z. 5/6 s. 173–184, z. 9 s. 360–378; 1964: z. 4 s. 135–152, z. 5 s. 185–196, z. 6 s. 241–261.
17. Jan Tokarski. *Czasowniki polskie. Formy, typy, wyjątki. Słownik.* Warszawa, 1951.
18. Jan Tokarski. Dialog: człowiek – maszyna cyfrowa. *Prace Filologiczne*, XXIII:183–185, 1972.
19. Jan Tokarski. *Schematyczny indeks a tergo polskich form wyrazowych*, red. Zygmunt Saloni. Wydawnictwo Naukowe PWN, Warszawa, second edition, 2002.
20. Marcin Woliński. System znaczników morfosyntaktycznych w korpusie IPI PAN. *Polonica*, XXII–XXIII:39–55, 2003.
21. Marcin Woliński. *Komputerowa weryfikacja gramatyki Świdzińskiego.* PhD thesis, Instytut Podstaw Informatyki PAN, Warszawa, December 2004.
22. Marcin Woliński. An efficient implementation of a large grammar of Polish. In Zygmunt Vetulani, editor, *Human Language Technologies as a Challenge for Computer Science and Linguistics. 2nd Language & Technology Conference April 21–23, 2005*, pages 343–347, Poznań, 2005.