



Polish Morphological Guesser Based on a Statistical *A Tergo* Index

Maciej Piasecki and Adam Radziszewski

Institute of Applied Informatics, Wrocław University of Technology, Poland
Wybrzeże Wyspiańskiego 27, Wrocław, Poland,
maciej.piasecki,adam.radziszewski@pwr.wroc.pl,

Abstract. We present a direct method of construction of a morpho-syntactic guesser for Polish, which is a program producing morpho-syntactic descriptions for word forms unknown to the morphological analyser. The core of the method is the construction of a statistical *a tergo* index, in which pseudo-suffixes (endings) extracted by a statistical tree define morpho-syntactic properties of corresponding word forms. The secondary aim was to investigate to what extent it is possible to develop the morphological analyses exclusively on the basis of endings. Experiments in the extraction of a guesser for a domain of texts are also presented. The method can be applied to any other inflectional language with only minor technical changes.

Keywords: morphological guesser, Polish, automatic extraction, corpus linguistics, statistical a tergo index

1 Introduction

Morfeusz [1], which is one of the most advanced morphological analysers for Polish, recognises about 1 700 000 word forms. However, in the case of a small domain corpus of medical texts *KorMedIIS* [2], it covers only about 75% tokens occurring there. This due to the following reasons: there are many misspelled words and domain-specific terms. Tokens from both of these categories are not worth to be included in a general tool like *Morfeusz*, but assigning them to one single class of *unknown words* blur the description for language tools of subsequent levels of processing e.g. decreases the accuracy of OCR correction done with the help of a tagger [3].

In the case of English, the initial approaches to guessing a morpho-syntactic tag for an unknown word solely on the basis of its ending (trailing) and also some leading segments, e.g. [4] were next exchanged by extraction of rules expressing patterns of a morphological construction of word forms, e.g. [5–7], or on the basis of *Inductive Logic Programming* [8].

Methods based on extraction of rules, as long as they do not use probabilistic criteria inside the rules (statistics is always present in the algorithms of extraction), are in their expressive power equivalent to finite state automaton

construction proposed in [9], where several heuristic rules of automaton reduction are proposed in order to get a generalised description of the characteristic features of word form construction.

However, the rule-based approaches originate from the works done for English which displays limited inflection. In the case of an English word form ending one can say mostly a little about its morpho-syntactic properties. The situation is completely different in the case of an inflectional language like Polish. The morphological analyser SAM-95 [10] has been constructed on the basis of a linguistic index *a tergo* of Polish verbs of Tokarski [11] (recently published). Tokarski's index consists of manually defined suffixes of verbs which identifies particular morphological descriptions. On this basis one can assign a description to an unknown verb. SAM-95 covering all parts of speech is able to do this for any word, but it assigns tags to unknown word forms too eagerly.

Our main idea is to correct this over-generation by extracting a kind of statistical index *a tergo* from a large corpus, and as a consequence to take into account only the tags supported by the data during guessing. Moreover, we can use the collected statistics concerning pseudo-suffixes (elements of the statistical index *a tergo*) and assign tags to them. This work also originates from the need to construct a guesser for a narrow domain by means of limited man power.

It is hard to find works on morphological guessers for Polish, especially based on a statistical index *a tergo*. A Polish guesser based on the automatic extraction of endings was presented in [12], but endings of the maximum fixed size were used and the coverage was limited to a subset of word forms covered by the extracted endings. Concerning similar inflectional languages, [13] deals with a different problem of the lemmatisation combined with the morpho-syntactic disambiguation. In [14] suffixes of a definite end are used and there is no information on the accuracy in relation to tags; only the coverage of unknown words is given. Hungarian, the target language of [7], according to its agglutinative character is significantly different from Polish.

The goal of this work was to construct a robust morphological guesser for Polish on the basis of a statistical extraction of an index similar to the index *a tergo* of Tokarski [11]. The guesser should be easily portable to different domains of text by using as simple and as direct methods as possible. Moreover, we wanted to investigate experimentally to what extent it is possible to reconstruct a morphological description of a word form exclusively on the basis of a statistically extracted index *a tergo*.

2 Construction of Pseudo-suffix Tree

The main idea is to identify endings of word forms associated with particular morpho-syntactic descriptions. As a model of such descriptions we assume the set of tags of the IPI PAN Corpus (IPIC) [15]—the largest corpus of Polish, containing about 254 millions of words in the version 2.0. The model was implemented in the morphological analyser *Morfeusz* [1], which was also employed

in the training phase¹. In IPIC format, a word form is assigned a set of pairs consisting of morphological *base form* and a *tag* expressing morpho-syntactic properties.

The construction of guesser for a domain of texts is performed in three steps:

1. Construction of a raw *a tergo tree*
2. Pruning orphaned branches (introduces generalisation)
3. Assigning morphological information to tree nodes

To gain higher degree of data generalisation, additional pruning techniques can be applied after the last step. Some experiments with different heuristics and parameter values have been carried out, yet no significant improvement in the overall results could be observed.

2.1 Construction of a raw tree

Word forms needed for the training process are acquired along with their frequencies from the IPIC corpus. These forms are passed through *Morfeusz*. Unrecognised forms are immediately removed from the training set.

In IPIC a notion of *grammatical class* is introduced [15] for more fine grained division than into parts of speech. There 32 grammatical classes defined in the IPIC format. The analyses for the extracted word forms are filtered according to the predefined list of ‘open’ grammatical classes. The list includes (the mnemonics are taken from the IPIC format): **subst** (noun), **depr** (depreciative form of noun), **adj** (adjective), **adjp** (post-prepositional adjective), **adv** (adverb), **fin** (non-past form of verb), **praet** (“1-participle”, ‘past form’), **impt** (imperative form), **imps** (impersonal form), **inf** (infinitive), **pcon** (contemporary adverbial participle), **pant** (anterior adverbial participle), **ger** (gerund), **pact** (active adjectival participle), and **ppas** (passive adjectival participle). The **adja** (ad-adjectival adjective) was omitted, as word forms of this class are recognised by *Morfeusz* only in context, not on the basis of a single word form alone. We assume full description of other grammatical classes by *Morfeusz* or a lexicon.

The training process leads to the construction of an *a tergo tree*, i.e. a tree-shaped automaton of reversed word forms whose edges are marked with letters. Fig. 1 shows an *a tergo tree* built with these forms:

mężczyzna ($men_{case=nom,num=sg}$), *mężczyzny* ($men_{case=gen,num=sg}$),
zna ($know_{tense=pres,per=1st,num=sg}$), *oka* ($eye_{case=gen,num=sg}$), *kózka* ($young\ female\ goat_{case=nom,num=sg}$). The ‘!’ marks a node where a form ends without further branching.

Every word form is reversed and along with its frequency passed to the tree. This way the branches are successively built, resulting in a tree-like representation of the filtered training data. To enable generalisation in the guessing process we need to introduce pruning.

¹ We would like to thank dr. Marcin Woliński and dr. Adam Przepiórkowski for their kind acceptance for our experiments done on IPIC with the help of *Morfeusz*.

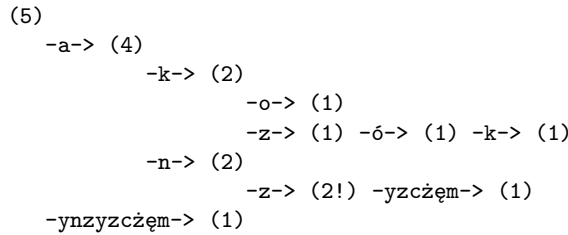


Fig. 1. An example *a tergo* tree.

2.2 Pruning orphaned branches

The described method enables generalisation of the classification process without precision loss on training data. In inflectional languages it is the word ending that conveys most of the morphological information. Thus, we can abstract from particular prefixes when they seem irrelevant to determine the morphological analysis.

This is achieved by a recursive tree traversal. The leaves are pruned as long as they belong to non-branching paths (i.e. a leaf is subjected to pruning when it is the only child of its father). This operation corresponds to removing prefixes of stored word forms where the prefixes are not needed to distinguish between different analyses.

2.3 Assigning morphological information to tree nodes

The training set is employed again in order to assign morphological analyses (tags) to the nodes of the pruned tree. Those nodes are found by following the paths corresponding to consecutive letters of the inverted form (this process is identical with the guessing stage and will be described in the next section). The set of tags assigned to processed form (coming from Morfeusz) is being added to the node. Every tag is associated with its frequency; it is calculated by summing the frequencies of the processed forms having this tag.

We also need to reconstruct the base form generation rule using the encountered form. We attach such a rule to every tag in the node. It consists of the encountered form ending and the length of base form ending. The following example explains the heuristics:

Encountered form: *przepustowościach* ($capacity_{case=loc,num=pl}$)
 Base form: *przepustowość* ($capacity_{case=nom,num=sg}$)
 Base form ending: *ć*
 Encountered form ending: *ciach* (5 letters)
 Procedure to get the base form: substitute the last 5 letters from the encountered form with “*ć*”

The first experiments produced relatively low accuracy of generated base forms. The analysis of the data showed that most of the incorrectly restored

base forms belonged to negated participles. This was due to the rules used in IPIC: the negated forms of participles beginning with “*nie*” (*not*) are lemmatised to base forms without the prefix. The negation is included as a value **neg** of the attribute **negation** in the tag. For instance the form *niemyty* (*not cleaned*) has base form of *myć* (*to clean*). This approach leads to some problems as basing on the mere ending we cannot tell if the form is negated or not.

The adopted solution uses a heuristic, which tries to convert all negated forms to their affirmative equivalents. If the attribute **negation** of a tag is set to **neg** and the form begins with “*nie*”, this prefix is stripped off and the value of **negation** is set to **aff** (affirmative) before storing the tags in a tree node. The correct value of negation is evaluated by a similar heuristic when guessing. By using this approach we can avoid the unnecessary division of some data into affirmative and negative entries which is no longer meaningful to the guesser (as the value of the negation attribute needs to be restored anyway).

3 The Algorithm of Guessing

The form being guessed is inverted first. Then we follow the branches of the tree corresponding to consecutive letters of the inverted form until the tree path or the inverted form ends. In both cases a node is reached. If this node has tags attached, these tags are treated as analyses of the form. If not, the tags of its ancestor are taken instead.

We need to reconstruct base forms as well; this is achieved by using the rules attached to every tag. Let us assume that we have been guessing the form “*niezwykłościach*”. We have obtained the tags from the tree and one of them contains the ending “*ć*” and the length of “*ciach*”, i.e. 5. Then:

Encountered form: *niezwykłościach* (*singularity_{case=loc,num=pl}*)
 Length of encountered form ending: 5 (i.e. 5 letters to strip off)
 Base form ending: *ć*
 Reconstructed base form: “*niezwykłość*” + “*ć*” = “*niezwykłośćć*”

If the analysed form begins with “*nie*” its tags are checked. The tags containing the attribute **negation** set to **aff** are subjected to negation restoration heuristics: their **negation** is set to **neg** and the “*nie*” prefix is stripped off from the reconstructed base form.

4 Experiments

The experiments were performed on a list of word forms recognised by *Morfeusz* and collected from two corpora: IPIC [15] — 504 320 word forms of the appropriate grammatical classes (see Sec. 2) and KorMedIIS [16] — 26 070. Documents in KorMedIIS are short medical electronic documents — descriptions and contain a limited vocabulary repetitively used across different documents. IPIC is the largest corpus of Polish containing various genres. The word forms were stored together with their frequencies in a corpus.

The test presented in Table 1 was performed following the *ten-fold scheme*, i.e. the list of word forms were randomly divided into 10 parts, nine were used in training, with one kept for tests, and this procedure was repeated ten times. Word forms from the testing part are unknown to the guesser but known to *Morfeusz*. Precision, recall and F-measure were calculated on the level of tags/base forms in relation to tags/base forms assigned to word forms in the test set, i.e.:

$$P = \frac{T_G \cap T_T}{T_G}, R = \frac{T_G \cap T_T}{T_T}, F = \frac{2PR}{P + R} \quad (1)$$

where T_G is the set of tags/base forms returned by the guesser for all word forms in the test part altogether, T_T — the set of tags/base forms collected from word forms in the training parts.

All the results presented in Table 1 are calculated as an average from ten tests, including the reported numbers (i.e. average numbers) of tested word forms (*Frm.*) and non-recognised (*Non-rec.*).

For some applications of guesser only the morpho-syntactic description is important (e.g. language model on the level of tag trigrams); for others only base forms are necessary (e.g. calculation of semantic similarity of documents). We have performed three kinds of tests:

- only morpho-syntactic tags: grammatical class (Part of Speech) plus morpho-syntactic categories, regardless of the base form (the label *Tags* in Table 1),
- only base forms, regardless of the tag (*Bases* in Table 1),
- and tags together with base forms (*Tags+Bases*), i.e. a test of full descriptions in the IPIC format.

During tests across different test parts, the results are stable and the differences lie in the range of $\pm 2\%$

In the row *all* of Table 1, the overall results calculated for all word forms are presented. The result for tags is much lower than reported for English, but the number of tags being recognised is more than ten times larger. It is worth to emphasize that our intention was to analyse only the endings of words (with one exception described in Sec. 2.3) and to investigate the ability of the mere endings to differentiate among word form morpho-syntactic properties. The higher value of recall means that the guesser generates too many descriptions per tested word, as it finds the closest match instead of the exact one in those cases in which there is no exact match in the tree. The result is no doubt better than a baseline of a random choice (1642 possible choices) and manual assessment of the guesser is quite positive as the errors concern mostly grammatical categories other than the basic set: case, gender, number and person. The assignment of grammatical classes to analysed word forms seems to produce a significantly smaller number of errors than the general result for tags. The number of word forms which were not recognised is very small (about 0.6% of all).

The result is also lower than the precision of 91.5% reported in [12], but it is very hard to compare our approach with [12] as there: only words started with a lower case letter were analysed (in our approach, all word forms are converted

Table 1. Average results of the guesser in ten-fold test: *Tags* — only morpho-syntactic description evaluated, *Bases* — only base form evaluated, *Tags+Bases* — full IPIC tags evaluated, *Non-rec.* — the average number of non recognised word forms, *Frm.* — the average number of word forms across test folds.

PoS	Tags [%]			Bases [%]			Tags+Bases [%]			Non-rec.	Frm.
	R	P	F	R	P	F	R	P	F		
<i>all</i>	81.75	70.32	75.61	88.93	81.49	85.05	79.77	68.61	73.77	282.5	50697.0
<i>subst</i>	74.96	58.21	65.53	83.61	70.72	76.63	73.87	57.37	64.58	1070.7	20497.3
<i>depr</i>	52.34	34.88	41.82	51.12	34.07	40.85	51.12	34.07	40.85	106.3	224.7
<i>adj</i>	88.63	82.61	85.51	87.28	79.87	83.41	87.7	81.74	84.61	1042.8	10594.6
<i>adjp</i>	93.26	80.47	86.13	93.26	80.47	86.13	93.26	80.47	86.13	2.2	32.4
<i>adv</i>	74.64	64.52	69.19	77.69	67.53	72.24	73.04	63.14	67.72	53.0	401.2
<i>fin</i>	69.49	55.01	61.40	89.65	83.44	86.43	66.03	52.27	58.34	217.6	3576.3
<i>praet</i>	71.54	56.39	63.07	96.43	95.94	96.19	69.8	55.02	61.54	58.9	4373.9
<i>impt</i>	69.41	53.69	60.53	85.68	77.65	81.46	63.43	49.07	55.32	52.3	765.2
<i>imps</i>	72.70	58.56	64.86	97.77	97.59	97.68	71.46	57.55	63.75	4.6	598.8
<i>inf</i>	68.66	55.58	61.42	98.74	98.94	98.84	68.62	55.55	61.39	13.0	1150.3
<i>pcon</i>	97.43	97.13	97.27	89.37	90.15	89.76	87.93	87.67	87.80	1.3	451.0
<i>pant</i>	98.00	97.93	97.96	96.53	97.03	96.77	96.11	96.04	96.07	2.2	161.5
<i>ger</i>	73.27	61.58	66.92	94.32	94.00	94.16	71.66	60.23	65.45	70.8	3371.3
<i>pact</i>	95.68	95.58	95.63	88.80	89.52	89.16	86.33	86.23	86.28	16.8	2777.0
<i>ppas</i>	75.69	61.89	68.10	96.25	95.41	95.83	74.87	61.22	67.36	144.8	5642.5

to lower case first), the precision for words starting with an upper case letter was only 43.6%, the precision was calculated only for 71% of unknown words recognised by the guesser of [12], and finally, the tagset used in [12] consists of only about 400 tags (four times less than the KIPI tagset).

The result of the base form generation is much better than of tag guessing and it is balanced between precision and recall. It is hard to compare it with other approaches, as it is hard to find data in the literature. This problem is very specific for inflective languages. The error rate of tag plus base form recognition is only slightly higher than the error rate for tags alone. It could be expected, as the wrong choice of tag implicates the wrong identification of the cut point.

In order to analyse the information conveyed by endings in different grammatical classes, we calculated the results for the classes, as well, see Table 1. The rows are named with IPIC class mnemonics described in Sec. 2.1. The numbers of non-recognised word forms are higher in the results of classes, as only tags of given case were taken into account as the expected outcome of recognition.

The worst result was achieved for depreciative forms (*depr*); also generation of base forms works poorly. Probably the number of learning cases is too small and they share endings with nouns too often to be properly distinguished. A similar problem can be noticed in the gerund class (*ger*). The smallest error rate is produced for: post-prepositional adjectives (*adjp*), and adverbial participles (*pcon*, *pant*) — these classes have a limited set of possible endings.

Table 2. Statistics: the length of an ending (*Len.*) needed for the recognition of a base form; all data are given in percentage [%] of endings of the given class.

Len.	subst	depr	adj	adjp	adv	fin	praet	impt	imps	inf	pcon	pant	ger	pact	ppas
1	0,41							0,02							
2	3,80		2,39		0,80	8,19		0,29							
3	7,39	1,16	7,05	0,01	13,91	2,61	4,95	33,78		10,23	0,77				
4	15,18	5,79	9,40	4,67	14,59	24,69	13,05	34,14	0,60	7,07	7,68			0,41	0,82
5	19,22	17,48	13,92	13,90	15,39	14,80	12,30	21,22	7,51	9,07	15,98	0,13	2,47	1,35	3,47
6	17,84	18,05	14,91	79,37	14,61	18,51	19,17	4,39	8,83	17,93	13,30	1,80	7,51	6,42	6,94
7	12,60	19,97	14,12	1,59	11,29	11,63	18,62	3,26	20,68	20,21	13,00	8,98	11,73	8,57	10,35
8	7,64	24,89	11,45	0,29	12,45	12,37	13,85	1,53	18,02	17,02	16,29	11,22	12,90	14,43	15,26
9	6,34	7,89	10,02	0,10	7,51	4,03	9,15	0,87	18,25	7,73	14,71	47,17	16,46	11,02	18,69
10	4,56	1,42	6,42		5,05	2,45	5,70	0,37	15,52	7,22	11,58	15,61	20,86	19,18	15,30
11	2,26	1,53	5,15		2,57	0,59	1,99	0,10	7,11	1,71	3,90	7,83	11,50	11,34	14,28
12	1,63	0,27	2,43	0,07	1,06	0,10	0,98	0,03	2,31	1,48	2,64	2,97	10,22	12,94	6,86
13	0,43	1,53	1,54		0,11	0,02	0,18		0,84	0,27	0,12	3,96	3,14	6,41	3,55
14	0,43		0,74		0,62		0,05		0,25	0,05	0,01	0,31	2,17	4,90	3,01
15	0,14	0,01	0,35		0,02		0,01		0,06	0,02	0,01	0,01	0,89	1,81	0,92
16	0,03		0,08						0,01				0,14	1,17	0,39
17	0,09		0,04						0,01				0,02	0,03	0,09
18	0,01													0,01	0,05
19			0,01											0,01	0,01

In Table 2 a histogram of ending lengths is presented. The length of an ending is taken from the tree as the length of a tree path leading to the first node in which a decision can be unambiguously made. The data are collected from the tree built for all word forms collected from both corpora, i.e. they are generated during learning on full data (any form could be perfectly recognised) and they describe the regularity observed in mapping: morpho-syntactic description—word form. It is hard to find any correlation between results of recognition and the length histogram. For some more specific classes, clear cut points can be observed, e.g. post-prepositional adjectives (*adjp*), anterior adverbial participle (*pant*) and imperative form (*impt*). In general, the automatically generated endings are much longer than reported in linguistic literature, but here the automatic endings are by definition unambiguous in the set of 504 320 word forms.

Some preliminary experiments on the application of the guesser in the improvement of handwriting OCR showed a significant positive impact. It is worth to notice, that the result for a domain corpus is much higher, especially the precision is improved. It means that a guesser constructed for the given domain generates descriptions better matching the vocabulary of given domain.

In the next experiment we have tested the influence of the learning set size on the results of guesser. In Fig. 2, one can notice that the result is increasing with the increase in the size — linearly in logarithmic scale. We expected this, as there were only 504 320 different word forms used in the experiment in relation to about 1 700 thousands of word forms known to *Morfeusz*.

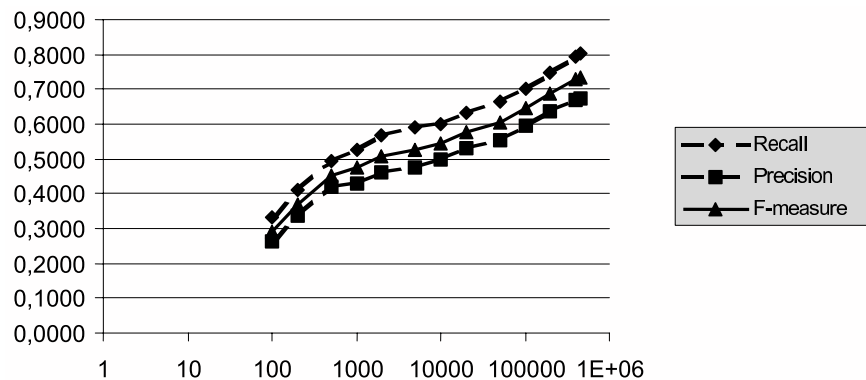


Fig. 2. Incremental test of the guesser: results in relation to the size of learning set (the horizontal axis).

5 Conclusions

The achieved results are much lower than reported for French [9] or English [5], but there are 1642 tags used in the IPIC, i.e. more than ten times more than in an English corpus. It is worth to emphasize that the coverage of the guesser is very good especially in comparison to [14], it leaves unrecognised only about 0.6% of words. The precision is lower than the recall, yet this means a little over-generation of tags. The good result for the very large general corpus of Polish—IPIC, was achieved on the basis of less than 30% of theoretically possible word forms according to *Morfeusz*. The result for a small domain corpus—KorMedIIS is even higher, probably due to capturing some patterns of inflection specific for the given corpus. Statistical information kept in guesser and the tree of pseudo-suffixes facilitate defining different strategies of generalisation by pruning. These positive results were achieved by application of a straightforward method, especially simple in the case of generating the base forms. The method can be applied to any other inflectional language with only minor technical changes concerning the input format of the learning data.

The necessity of the introduction of a heuristic for negative participles shows the limitation of the statistical *a tergo* index. In order to improve the result, one needs to add some mechanism sensitive for the leading segments of word forms. The created tree could be also transformed into an automaton by the application of a solution proposed in [9].

Acknowledgement. This work was financed by the Ministry of Education and Science project No 3 T11E 005 28.

References

1. Woliński M.: *Morfeusz — a practical tool for the morphological analysis of Polish*, [18].
2. Piasecki M., Godlewski G., Pejcz J.: *Corpus of medical texts and tools*, In: Proceedings of Medical Informatics and Technologies 2006, Silesian University of Technology (2006) 281–286.
3. Godlewski G., Piasecki M.: *Application of syntactic properties to three-level recognition of Polish hand-written medical texts*, In Brailsford D. F., ed.: Proceedings of 2006 ACM Symposium on Document Engineering, ACM (2006) 115–121.
4. Brill E.: *Some advances in transformation-based part of speech tagging*, In: Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94). (1994).
5. Mikheev A.: *Automatic rule induction for unknown-word guessing*, Computational Linguistics **23**(3) (1997) 405–423.
6. Schone P., Jurafsky D.: *Knowledge-free induction of inflectional morphologies*, In: Proceedings of the North American chapter of the Association for Computational Linguistics (NAACL-2001). (2001).
7. Novák A., Nagy V., Oravecz C.: *Combining symbolic and statistical methods in morphological analysis and unknown word guessing*, In: Proceedings of LREC'04. (2004).
8. Kazakov D.: *Achievements and prospects of learning word morphology with inductive logic programming*, In: Learning Language in Logic. LNCS, Springer (2004) 89–111.
9. Daciuk J.: *Treatment of unknown words*, In: Proceedings of Workshop on Implementing Automata WIA'99, Potsdam, Germany, 1999. Volume 2214 of LNCS., Springer Verlag (2001) 71–80.
10. Szafran K.: *SAM-96 — the morphological analyser for Polish*, In Narin'yani, A., ed.: Proceedings of International Workshop DIALOGUE'97 Computational Linguistics and its Applications, Yasnaya Polyana, Russia, June, 10–15, 1997, Moskwa (1997) 304–308.
11. Tokarski J.: *Schematyczny indeks a tergo polskich form wyrazowych*, Warszawa (2002).
12. Rudolf M.: *Metody automatycznej analizy korpusu tekstów polskich*, Uniwersytet Warszawski, Wydz. Polonistyki (2004).
13. Džeroski S., Erjavec T.: *Learning to lemmatise slovene words*, In Cussens J., Džeroski S., eds.: Proceedings of LLL'99. Volume 1925 of LNAI., Springer (2000) 69–88.
14. Hlaváčová J.: *Morphological guesser of Czech words*, In: Proceedings of Text, Speech, and Dialogue 2001. Volume 2166 of LNAI., Springer (2001) 70–75.
15. Przepiórkowski A.: *The IPI PAN Corpus Preliminary Version*, Institute of Computer Science PAS (2004).
16. Godlewski G., Piasecki M., Pejcz J.: *Corpus of medical texts and tools*, Silesian University of Technology (2006) 273–280.
17. Sojka P., Kopecek I., Pala K., eds.: *Proceedings of the Text, Speech and Dialog 2006 Conference*, Lecture Notes in Artificial Intelligence, Springer (2006).
18. Kłopotek M. A., Wierzchoń S. T., Trojanowski K., eds.: *Intelligent Information Processing and Web Mining—Proceedings of the International IIS: IIPWM'06 Conference held in Zakopane, Poland, June, 2006*. Advances in Soft Computing. Springer, Berlin (2006).