Learning to lemmatise Polish noun phrases

Adam Radziszewski
Institute of Informatics, Wrocław University of Technology
Wybrzeże Wyspiańskiego 27
Wrocław, Poland
adam.radziszewski@pwr.wroc.pl

Abstract

We present a novel approach to noun phrase lemmatisation where the main phase is cast as a tagging problem. The idea draws on the observation that the lemmatisation of almost all Polish noun phrases may be decomposed into transformation of singular words (tokens) that make up each phrase. We perform evaluation, which shows results similar to those obtained earlier by a rule-based system, while our approach allows to separate chunking from lemmatisation.

1 Introduction

Lemmatisation of word forms is the task of finding base forms (lemmas) for each token in running text. Typically, it is performed along POS tagging and is considered crucial for many NLP applications. Similar task may be defined for whole noun phrases (Degórski, 2011). By lemmatisation of noun phrases (NPs) we will understand assigning each NP a grammatically correct NP corresponding to the same phrase that could stand as a dictionary entry.

The task of NP lemmatisation is rarely considered, although it carries great practical value. For instance, any keyword extraction system that works for a morphologically rich language must deal with lemmatisation of NPs. This is because keywords are often longer phrases (Turney, 2000), while the user would be confused to see inflected forms as system output. Similar situation happens when attempting at terminology extraction from domain corpora: it is usually assumed that domain terms are subclass of NPs (Marciniak and Mykowiecka, 2013).

In (1) we give an example Polish noun phrase (‘*the main city of the municipality*’). Throughout the paper we assume the usage of the tagset of the National Corpus of Polish (Przepiórkowski, 2009), henceforth called NCP in short. The orthographic form (1a) appears in instrumental case, singular. Phrase lemma is given as (1b). Lemmatisation of this phrase consists in reverting case value of the main noun (*miasto*) as well as its adjective modifier (*główne*) to nominative (`nom`). Each form in the example is in singular number (`sg`), *miasto* has neuter gender (`n`), *gmina* is feminine (`f`).

(1) a. *głównym* *miastem* *gminy*
main city municipality
`inst:sg:n` `inst:sg:n` `gen:sg:f`

b. *główne* *miasto* *gminy*
main city municipality
`nom:sg:n` `nom:sg:n` `gen:sg:f`

According to the lemmatisation principles accompanying the NCP tagset, adjectives are lemmatised as masculine forms (*główny*), hence it is not sufficient to take word-level lemma nor the orthographic form to obtain phrase lemmatisation. Degórski (2011) discuses some similar cases. He also notes that this is not an easy task and lemma of a whole NP is rarely a concatenation of lemmas of phrase components. It is worth stressing that even the task of word-level lemmatisation is non-trivial for inflectional languages due to a large number of inflected forms and even larger number of syncretisms. According to Przepiórkowski (2007), “a typical Polish adjective may have 11 textually different forms (...) but as many as 70 different tags (2 numbers $\times$ 7 cases $\times$ 5 genders)”, which indicates the scale of the problem. What is more, several syntactic phenomena typical for Polish complicate NP lemmatisation further. E.g., adjectives may both precede and follow nouns they modify; many English prepositional phrases are realised in Polish using oblique case without any proposition (e.g., there is no standard Polish coun-

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 701–709,
Sofia, Bulgaria, August 4-9 2013.

terpart for the preposition *of* as genitive case is used for this purpose).

In this paper we present a novel approach to noun phrase lemmatisation where the main phase is cast as a tagging problem and tackled using a method devised for such problems, namely Conditional Random Fields (CRF).

2 Related works

NP lemmatisation received very little attention. This situation may be attributed to prevalence of works targeted at English, where the problem is next to trivial due to weak inflection in the language.

The only work that contains a complete description and evaluation of an approach to this task we were able to find is the work of Degórski (2011). The approach consists in incorporating phrase lemmatisation rules into a shallow grammar developed for Polish. This is implemented by extending the Spejd shallow parsing framework (Buczyński and Przepiórkowski, 2009) with a rule action that is able to generate phrase lemmas. Degórski assumes that lemma of each NP may be obtained by concatenating each token's orthographic form, lemma or 'half-lemmatised' form (e.g. grammatical case normalised to nominative, while leaving feminine gender). The other assumption is to neglect letter case: all phrases are converted to lower case and this is not penalised during evaluation. For development and evaluation, two subsets of NCP were chosen and manually annotated with NP lemmas: development set (112 phrases) and evaluation set (224 phrases). Degórski notes that the selection was not entirely random: two types of NPs were deliberately omitted, namely foreign names and "a few groups for which the proper lemmatisation seemed very unclear". The final evaluation was performed in two ways. First, it is shown that the output of the entire system intersects only with 58.5% of the test set. The high error rate is attributed to problems with identifying NP boundaries correctly (29.5% of test set was not recognised correctly with respect to phrase boundaries). The other experiment was to limit the evaluation to those NPs whose boundaries were recognised correctly by the grammar (70.5%). This resulted in 82.9% success rate.

The task of phrase lemmatisation bears a close resemblance to a more popular task, namely lemmatisation of named entities. Depending on the type of named entities considered, those two may be solved using similar or significantly different methodologies. One approach, which is especially suitable for person names, assumes that nominative forms may be found in the same source as the inflected forms. Hence, the main challenge is to define a similarity metric between named entities (Piskorski et al., 2009; Kocoń and Piasecki, 2012), which can be used to match different mentions of the same names. Other named entity types may be realised as arbitrary noun phrases. This calls for more robust lemmatisation strategies.

Piskorski (2005) handles the problem of lemmatisation of Polish named entities of various types by combining specialised gazetteers with lemmatisation rules added to a hand-written grammar. As he notes, organisation names are often built of noun phrases, hence it is important to understand their internal structure. Another interesting observation is that such organisation names are often structurally ambiguous, which is exemplified with the phrase (2a), being a string of items in genitive case ('*of the main library of the Higher School of Economics*'). Such cases are easier to solve when having access to a collocation dictionary — it may be inferred that there are two collocations here: *Biblioteka Główna* and *Wyższa Szkoła Handlowa*.

(2) a. *Biblioteki* *Głównej* *Wyższej*
library main higher
`gen:sg:f gen:sg:f gen:sg:f`
Szkoły *Handlowej*
school commercial
`gen:sg:f gen:sg:f`

b. *Biblioteka* *Główna* *Wyższej*
library main higher
`nom:sg:f nom:sg:f gen:sg:f`
Szkoły *Handlowej*
school commercial
`gen:sg:f gen:sg:f`

While the paper reports detailed figures on named entity recognition performance, the quality of lemmatisation is assessed only for all entity types collectively: "79.6 of the detected NEs were lemmatised correctly" (Piskorski, 2005).

3 Phrase lemmatisation as a tagging problem

The idea presented here is directly inspired by Degórski's observations. First, we will also assume

that lemma of any NP may be obtained by concatenating simple transformations of word forms that make up the phrase. As we will show in Sec. 4, this assumption is virtually always satisfied. We will argue that there is a small finite set of inflectional transformations that are sufficient to lemmatise nearly every Polish NP.

Consider example (1) again. Correct lemmatisation of the phrase may be obtained by applying a series of simple inflectional transformations to each of its words. The first two words need to be turned into nominative forms, the last one is already lemmatised. This is depicted in (3a). To show the real setting, this time we give full NCP tags and word-level lemmas assigned as a result of tagging. In the NCP tagset, the first part of each tag denotes grammatical class (`adj` stands for adjective, `subst` for noun). Adjectives are also specified for degree (`pos` — positive degree).

(3) a.

głównym		
główny		
`adj:sg:inst:n:pos`		
miastem	*gminy*	
miasto	gmina	
`subst:sg:inst:n`	`subst:sg:gen:f`	

b.

główne	*miasto*
`adj:sg:nom:n:pos`	`subst:sg:nom:n`
`cas=nom`	`cas=nom`

gminy
`subst:sg:gen:f`
=

Example (3b) consists of three rows: the lemmatised phrase, the desired tags (tags that would be attached to tokens of the lemmatised phrase) and the transformations needed to obtain lemma from orthographic forms. The notation `cas=nom` means that to obtain the desired form (e.g. *główne*) you need to find an entry in a morphological dictionary that bears the same word-level lemma as the inflected form (*główny*) and a tag that results from taking the tag of the inflected form (`adj:sg:inst:n:pos`) and setting the value of the tagset attribute `cas` (grammatical case) to the value `nom` (nominative). The transformation labelled = means that the inflected form is already equal to the desired part of the lemma, hence no transformation is needed.

A tagset note is in order. In the NCP tagset each tag may be decomposed into grammatical class and attribute values, where the choice of applicable attributes depends on the grammatical class. For instance, nouns are specified for number, gender and case. This assumption is important for our approach to be able to use simple tag transformations in the form *replace the value of attribute A with the new value V* (`A=V`). This is not a serious limitation, since the same assumption holds for most tagsets developed for inflectional languages, e.g., the whole MULTEXT-East family (Erjavec, 2012), Czech tagset (Jakubíček et al., 2011).

Our idea is simple: by expressing phrase lemmatisation in terms of word-level transformations we can reduce the task to tagging problem and apply well known Machine Learning techniques that have been devised for solving such problems (e.g. CRF). An important advantage is that this allows to rely not only on the information contained within the phrase to be lemmatised, but also on tokens belonging to its local neighbourhood.

Assuming that we have already trained a statistical model, we need to perform the following steps to obtain lemmatisation of a new text:

1. POS tagging,
2. NP chunking,
3. tagging with transformations by applying the trained model,
4. application of transformations to obtain NP lemmas (using a morphological dictionary to generate forms).

To train the statistical model, we need training data labelled with such transformations. Probably the most reliable way to obtain such data would be to let annotators manually encode a training corpus with such transformations. However, the task would be extremely tedious and the annotators would probably have to undergo special training (to be able to think in terms of transformations). We decided for a simpler solution. The annotators were given a simpler task of assigning each NP instance a lemma and a heuristic procedure was used to induce transformations by matching the manually annotated lemmas to phrases' orthographic forms using a morphological dictionary. The details of this procedure are given in the next section.

We decided to perform the experiments using the data from *Polish Corpus of Wrocław Univer-*

sity of Technology[1] (Broda et al., 2012). The corpus (abbreviated to *KPWr* from now on) contains manual shallow syntactic annotation which includes NP chunks and their syntactic heads. The main motivation to use this corpus was its very permissive licence (Creative Commons Attribution), which will not constrain any further use of the tools developed. What is more, it allowed us to release the data annotated manually with phrase lemmas and under the same licence[2].

One of the assumptions of KPWr annotation is that actual noun phrases and prepositional phrases are labelled collectively as NP chunks. To obtain real noun phrases, phrase-initial prepositions must be stripped off[3]. For practical reasons we decided to include automatic recognition of phrase-initial prepositions into our model: we introduced a special transformation for such cases (labelled `p`), having the interpretation that the token belongs to a phrase-initial preposition and should be discarded when generating phrase lemma. Prepositions are usually contained in single tokens. There are some cases of multi-word units which we treat as prepositions (*secondary prepositions*), e.g. *ze względu na* (*with respect to*). This solution allows to use our lemmatiser directly against chunker output to obtain NP lemmas from both NPs and PPs. For instance, the phrase *o przenoszeniu bakterii drogą płciową* (*about sexual transmission of bacteria*) should be lemmatised to *przenoszenie bakterii drogą płciową* (*sexual transmission of bacteria*).

4 Preparation of training data

First, simple lemmatisation guidelines were developed. The default strategy is to normalise the case to nominative and the number to singular. If the phrase was in fact prepositional, phrase-initial preposition should be removed first. If changing the number would alter semantics of the phrase, it should be left plural (e.g., *warunki* '*conditions*' as in *terms and conditions*). Some additional exceptions concern pronouns, fixed expressions and proper names. They were introduced to obtain lemmas that are practically most useful.

A subset of documents from KPWr corpus was drawn randomly. Each NP/PP belonging to this subset was annotated manually. Contrary to (Degórski, 2011), we made no exclusions, so the obtained set contains some foreign names and a number of cases which were hard to lemmatise manually. Among the latter there was one group we found particularly interesting. It consisted of items following the following pattern: NP in plural modified by another NP or PP in plural. For many cases it was hard to decide if both parts were to be reverted to singular, only the main one or perhaps both of them should be left in plural. We present two such cases in (4a) and (4b). For instance, (4b) could be lemmatised as *opis tytułu z Wikipedii* (*description of a Wikipedia title*), but it was not obvious if it was better than leaving the whole phrase as is.

(4) a. *obawy ze strony autorów*
'concerns on the part of the authors'
b. *opisy tytułów z Wikipedii*
'descriptions of the Wikipedia titles'

Altogether, the annotated documents contain 1669 phrases. We used the same implementation of the 2+1 model which was used to annotate morphosyntax in NCP (Przepiórkowski and Szałkiewicz, 2012): two annotators performed the task independently, after which their decisions were compared and the discrepancies were highlighted. The annotators were given a chance to rethink their decisions concerning the highlighted phrases. Both annotators were only told which phrases were lemmatised differently by the other party but they didn't know the other decision. The purpose of this stage was to correct obvious mistakes. Their output was finally compared, resulting in 94% phrases labelled identically (90% before reconsidering decisions). The remaining discrepancies were decided by a superannotator. The whole set was divided randomly into the development set (1105 NPs) and evaluation set (564 NPs).

The development set was enhanced with word-level transformations that were induced automatically in the following manner. The procedure assumes the usage of a morphological dictionary extracted from Morfeusz SGJP analyser[4] (Woliński,

[1] We used version 1.1 downloaded from `http://www.nlp.pwr.wroc.pl/kpwr`.

[2] The whole dataset described in this paper is available at `http://nlp.pwr.wroc.pl/en/static/kpwr-lemma`.

[3] Note that if we decided to use the data from NCP, we would still have to face this issue. Although an explicit distinctions is made between NPs and PPs, NPs are not annotated as separate chunks when belonging to a PP chunk (an assumption which is typical for shallow parsing).

[4] `morfeusz-SGJP-src-20110416` package

2006). The dictionary is stored as a set of (*orthographic form*, *word-level lemma*, *tag*). The procedure starts with tokenisation of the manually assigned lemma. Next, a heuristic identification of phrase-initial preposition is performed. The assumption is that, having cut the preposition, all the remaining tokens of the original inflected phrase must be matched 1:1 to corresponding tokens from the human-assigned lemma. If any match problem did occur, an error was reported and such a case was examined manually. The only problems encountered were due to proper names unknown to the dictionary and misspelled phrases (altogether about 10 cases). Those cases were dealt with manually. Also, all the extracted phrase-initial prepositions were examined and no controversy was found.

The input and output to the matching procedure is illustrated in Fig. 1. The core matching happens at token level. The task is to find a suitable transformation for the given inflected form from the original phrase, its tag and word-level lemma, but also given the desired form being part of human-assigned lemma. If the inflected form is identical to the desired human-assigned lemma, the '=' transformation is returned without any tag analysis. For other cases the morphological dictionary is required. For instance, the inflected form *tej* tagged as `adj:sg:loc:f:pos` should be matched to the human-assigned form *ta* (the row label H lem). The first subtask is to find all entries in the morphological dictionary with the orthographic form equal to human-assigned lemma (*ta*), the word-level lemma equal to the lemma assigned by the tagger (*ten*) and having a tag with the same grammatical class as the tagger has it (`adj`; we deliberately disallow transformations changing the grammatical class). The result is a set of entries with the given lemma and orthographic form, but with different tags attached. For the example considered, two tags may be obtained: `adj:sg:nom:f:pos` and `adj:sg:voc:f:pos` (the former is in nominative case, the latter — in vocative). Each of the obtained tags is compared to the tag attached to the inflected forms (`adj:sg:loc:f:pos`) and this way candidate transformations are generated (`cas=nom` and `cas=voc` here). The transformations are heuristically ranked. Most importantly, `cas=nom` is always preferred, then `nmb=sg` (enforcing singular number), then transforming the gender to different values, preferably to masculine inanimate (`gnd=m3`). The lowest possible ranking is given to a transformation enforcing case value other than nominative.

Original:	*przy*	*tej*	*drodze*
T tags:	`prep:`	`adj:`	`subst:`
	`loc`	`sg:loc:f:pos`	`sg:loc:f`
T lem:	*przy*	*ten*	*droga*
H lem:		*ta*	*droga*
Transf.:	`p`	`cas=nom`	`cas=nom`

Figure 1: Matching of an NP and its lemma. The first row shows the original inflected form. The next three present tagger output: tags (split into two rows) and lemmas. H lem stands for the lemma assigned by a human. Last row presents the transformations induced.

We are fully aware of limitations of this approach. This ranking was inspired only by intuition obtained from the lemmatisation guidelines and the transformations selected this way may be wrong in a number of cases. While many transformations may lead to obtaining the same lemma for a given form, many of them will still be accidental. Different syncretisms may apply to different lexemes, which can negatively impact the ability of the model to generalise from one phrase to other. On the other hand, manual inspection of some fragments suggest that the transformations inferred are rarely unjustified.

The frequencies of all transformations induced from the development set are given in Tab. 1. Note that the first five most frequent transformation make up 98.7% of all cases. These findings support our hypothesis that a small finite set of transformations is sufficient to express lemmatisation of nearly every Polish NP.

We have also tested an alternative variant of the matching procedure that included additional transformation '`lem`' with the meaning *take the word-level lemma assigned by the tagger as the correct lemmatisation*. This transformation could be induced after an unsuccessful attempt to induce the '=' transformation (i.e., if the correct human-assigned lemmatisation was not identical to orthographic form). This resulted in replacing a number of tag-level transformations (mostly `cas=nom`) with the simple '`lem`'. The advantage of this vari-

obtained from `http://sgjp.pl/morfeusz/dopobrania.html`. The package is available under 2-clause BSD licence.

=	2444	72%
cas=nom	434	13%
p	292	9%
nmb=sg	97	3%
cas=nom,nmb=sg	76	2%
gnd=m3	9	
cas=nom,gnd=m3,nmb=sg	7	
gnd=m3,nmb=sg	6	
acn,cas=nom	5	
acm=rec,cas=nom	3	
cas=gen	3	
cas=nom,gnd=m3	3	
cas=nom,gnd=m1	2	
gnd=f,nmb=sg	2	
cas=nom,gnd=f	1	
cas=nom,gnd=f,nmb=sg	1	
cas=nom,nmb=pl	1	
cas=nom,nmb=sg,gnd=m3	1	
Total	3387	100%

Table 1: Frequencies of transformations.

ant is that application of this transformation does not require resorting to the dictionary. The disadvantage is that it is likely to worsen the generalising power of the model.

5 CRF and features

The choice of CRF for sequence labelling was mainly influenced by its successful application to chunking of Polish (Radziszewski and Pawlaczek, 2012). The work describes a feature set proposed for this task, which includes word forms in a local window, values of grammatical class, gender, number and case, tests for agreement on number, gender and case, as well as simple tests for letter case.

We took this feature set as a starting point. Then we performed some experiments with feature generation and selection. For this purpose the development set was split into training and testing part. The most obvious, yet most successful change was to introduce features returning the chunk tag assigned to a token. As KPWr also contains information on the location of chunks' syntactic heads and this information is also output by the chunker, we could also use this in our features. Another improvement resulted from completely removing tests for grammatical gender and limiting the employed tests for number to the current token.

The final feature set includes the following items:

- the word forms (turned lower-case) of tokens occupying a local window $(-2, \ldots, +2)$,
- word form bigrams: $(-1, 0)$ and $(0, 1)$,
- chunk tags (IOB2 tags concatenated with Boolean value denoting whether the syntactic head is placed at the position), for a local window $(-1, 0, +1)$
- chunk tags (IOB2 tags only) for positions -2 and $+2$, and two chunk tag bigrams: $(-1, 0)$ and $(0, 1)$,
- grammatical class of tokens in the window $(-2, \ldots, +2)$,
- grammatical class for the focus token (0) concatenated with the last character of the word-form,
- values of grammatical case for tokens $(-2, -1, +1, +2)$,
- grammatical class of the focus token concatenated with its gender value,
- 2-letter prefix of the word form (lower-cased),
- tests for agreements and letter case as in (Radziszewski and Pawlaczek, 2012).

6 Evaluation

The performed evaluation assumed training of the CRF on the whole development set annotated with the induced transformations and then applying the trained model to tag the evaluation part with transformations. Transformations were then applied and the obtained phrase lemmas were compared to the reference annotation. This procedure includes the influence of deficiencies of the morphological dictionary. The version of KPWr used here was tagged automatically using the WCRFT tagger (Radziszewski, 2013), hence tagging errors are also included.

Degórski (2011) reports separate figures for the performance of the entire system (chunker + NP lemmatiser) on the whole test set and performance of the entire system limiting the test set only to those phrases that the system is able to chunk correctly (i.e., to output correct phrase boundaries). Such a choice is reasonable given that his system

is based on rules that intermingle chunking with lemmatisation. We cannot expect the system to lemmatise correctly those groups which it is unable to capture. Our approach assumes two-stage operation, where the chunker stage is partially independent from the lemmatisation. This is why we decided to report performance of the whole system on the whole test set, but also, performance of the lemmatisation module alone on the *whole test set*. This seems more appropriate, since the chunker may be improved or completely replaced independently, while discarding the phrases that are too hard to parse is likely to bias the evaluation of the lemmatisation stage (what is hard to chunk is probably also hard to lemmatise).

For the setting where chunker was used, we used the CRF-based chunker mentioned in the previous section (Radziszewski and Pawlaczek, 2012). The chunker has been trained on the entire KPWr except for the documents that belong to the evaluation set.

Degórski (2011) uses concatenation of word-level base forms assigned by the tagger as a baseline. Observation of the development set suggests that returning the original inflected NPs may be a better baseline. We tested both variants. As detection of phrase-initial prepositions is a part of our task formulation, we had to implement it in the baseline algorithms as well. Otherwise, the comparison would be unfair. We decided to implement both baseline algorithms using the same CRF model but trained on fabricated data. The training data for the 'take-orthographic-form' baseline was obtained by leaving the 'remove-phrase-initial-preposition' ('`p`') transformation and replacing all others with '='. Similarly, for the 'take-lemma' baseline, other transformations were substituted with '`lem`'.

The results of the full evaluation are presented in Tab. 2. The first conclusion is that the figures are disappointingly low, but comparable with the 58.5% success rate reported in (Degórski, 2011). The other observation is that the proposed solution significantly outperforms both baseline, out of which the 'take-orthographic-form' (*orth baseline*) performs slightly better. Also, it turns out that the variation of the matching procedure using the '`lem`' transformation (row labelled *CRF lem*) performs slightly worse than the procedure without this transformation (row *CRF nolem*). This supports the suspicion that relying on word-level lemmas may reduce the ability to generalise.

Algorithm	Prec.	Recall	F
CRF nolem	55.1%	56.9%	56.0%
CRF lem	53.7%	55.5%	54.6%
orth baseline	38.6%	39.9%	39.2%
lem baseline	36.2%	37.4%	36.8%

Table 2: Performance of NP lemmatisation including chunking errors.

Results corresponding to performance of the lemmatisation module alone are reported in Tab. 3. The test has been performed using chunk boundaries and locations of syntactic heads taken from the reference corpus. In this settings recall and precision have the same interpretation, hence we simply refer to the value as *accuracy* (percentage of chunks that were lemmatised correctly). The figures are considerably higher than those reported in Tab. 2, which shows the huge impact of chunking errors. It is worth noting that the best accuracy achieved is only slightly lower than that achieved by Degórski (82.9%), while our task is harder. As mentioned above, in Degórski's setting, the phrases that are too hard to parse are excluded from the test set. Those phrases are also likely to be hard cases for lemmatisation. The other important difference stems from phrase definitions used in both corpora; NPs in NCP are generally shorter than the chunks allowed in KPWr. Most notably, KPWr allows the inclusion of PP modifiers within NP chunks (Broda et al., 2012). It seems likely that the proposed algorithm would performed better when trained on data from NCP which assumes simpler NP definition. Note that the complex NP definition in KPWr also explains the huge gap between results of lemmatisation alone and lemmatisation including chunking errors.

Algorithm	Correct lemmas	Accuracy
CRF nolem	455 / 564	80.7%
CRF lem	444 / 564	78.7%
orth baseline	314 / 564	55.7%
lem baseline	290 / 564	51.4%

Table 3: Performance of NP lemmatisation alone.

We also checked the extent to which the entries unknown to the morphological dictionary could lower the performance of lemmatisation. It turned out that only 8 words couldn't be transformed during evaluation due to lack of the entries that

were sought in the morphological dictionary, out of which 5 were anyway handled correctly in the end by using the simple heuristic to output the '=' transformation when everything else fails.

A rudimentary analysis of lemmatiser output indicates that the most common error is the assignment of the orthographic form as phrase lemma where something else was expected. This seems to concern mostly many NPs that are left in plural, even simple ones (e.g. *audycje telewizyjne* '*TV programmes*'), but there are also some cases of personal pronouns left in oblique case (*was* '*you*-pl-accusative/genitive'). It seems that a part of these cases come from tagging errors (even if the correct transformation is obtained, the results of its application depend on the tag and lemma attached to the inflected form by the tagger). Not surprisingly, proper names are hard cases for the model (e.g. *Pod Napięciem* was lemmatised to *napięcie*, which would be correct weren't it a title).

7 Conclusions and further work

We presented a novel approach to lemmatisation of Polish noun phrases. The main advantage of this solution is that it allows to separate the lemmatisation phrase from the chunking phrase. Degórski's rule-based approach (Degórski, 2011) was also built on top of an existing parser but, as he notes, to improve the lemmatisation accuracy, the grammar underlying the parser should actually be rewritten with lemmatisation in mind. The other advantage of the approach presented here is that it is able to learn from a corpus containing manually assigned phrase lemmas. Extending existing chunk-annotated corpora with phrase lemmas corresponds to a relatively simple annotation task.

The performance figures obtained by our algorithm are comparable with that of Degórski's grammar, while the conditions under which our system was evaluated were arguably less favourable. To enable a better comparison it would be desirable to evaluate our approach against the phrases from NCP.

The main disadvantage of the approach lies in the data preparation stage. It requires some semi-manual work to obtain labelling with transformations, which is language- and tagset-dependent. A very interesting alternative has been suggested by an anonymous reviewer: instead of considering tag-level transformations that require an exhaustive morphological dictionary, it would be simpler to rely entirely on string-to-string transformations that map inflected forms to their expected counterparts. Such transformations may be expressed in terms of simple edit scripts, which has already been successfully applied to word-level lemmatisation of Polish and other languages (Chrupała et al., 2008). This way, the training data labelled with transformations could be obtained automatically. What is more, application of such transformations also does not depend on the dictionary. It is not obvious how this would affect the performance of the module and, hence, needs to be evaluated. We plan this as our further work.

Also, it would be worthwhile to evaluate the presented solution for other Slavic languages.

Acknowledgments

This work was financed by Innovative Economy Programme project POIG.01.01.02-14-013/09.

References

Robert Bembenik, Łukasz Skonieczny, Henryk Rybiński, Marzena Kryszkiewicz, and Marek Niezgódka, editors. 2013. *Intelligent Tools for Building a Scientific Information Platform*, volume 467 of *Studies in Computational Intelligence*. Springer Berlin Heidelberg.

Bartosz Broda, Michał Marcińczuk, Marek Maziarz, Adam Radziszewski, and Adam Wardyński. 2012. KPWr: Towards a free corpus of Polish. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of LREC'12*, Istanbul, Turkey. ELRA.

Aleksander Buczyński and Adam Przepiórkowski. 2009. Human language technology. challenges of the information society. chapter Spejd: A Shallow Processing and Morphological Disambiguation Tool, pages 131–141. Springer-Verlag, Berlin, Heidelberg.

Grzegorz Chrupała, Georgiana Dinu, and Josef van Genabith. 2008. Learning morphology with Morfette. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA).

Łukasz Degórski. 2011. Towards the lemmatisation of Polish nominal syntactic groups using a shallow

grammar. In Pascal Bouvry, Mieczysław A. Kłopotek, Franck Leprevost, Małgorzata Marciniak, Agnieszka Mykowiecka, and Henryk Rybiński, editors, *Security and Intelligent Information Systems: International Joint Conference, SIIS 2011, Warsaw, Poland, June 13-14, 2011, Revised Selected Papers*, volume 7053 of *Lecture Notes in Computer Science*, pages 370–378. Springer-Verlag.

Tomaž Erjavec. 2012. MULTEXT-East: morphosyntactic resources for Central and Eastern European languages. *Language Resources and Evaluation*, 46(1):131–142.

Miloš Jakubíček, Vojtěch Kovář, and Pavel Šmerk. 2011. Czech morphological tagset revisited. In *Proceedings of Recent Advances in Slavonic Natural Language Processing*, pages 29–42, Brno.

Jan Kocoń and Maciej Piasecki. 2012. Heterogeneous named entity similarity function. In Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, volume 7499 of *Lecture Notes in Computer Science*, pages 223–231. Springer Berlin Heidelberg.

Małgorzata Marciniak and Agnieszka Mykowiecka. 2013. Terminology extraction from domain texts in Polish. In Bembenik et al. (Bembenik et al., 2013), pages 171–185.

Jakub Piskorski, Karol Wieloch, and Marcin Sydow. 2009. On knowledge-poor methods for person name matching and lemmatization for highly inflectional languages. *Information Retrieval*, 12(3):275–299.

Jakub Piskorski. 2005. Named-entity recognition for Polish with SProUT. In Leonard Bolc, Zbigniew Michalewicz, and Toyoaki Nishida, editors, *Intelligent Media Technology for Communicative Intelligence*, volume 3490 of *Lecture Notes in Computer Science*, pages 122–133. Springer Berlin Heidelberg.

Adam Przepiórkowski. 2007. Slavic information extraction and partial parsing. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing*, pages 1–10, Praga, Czechy, June. Association for Computational Linguistics.

Adam Przepiórkowski. 2009. A comparison of two morphosyntactic tagsets of Polish. In Violetta Koseska-Toszewa, Ludmila Dimitrova, and Roman Roszko, editors, *Representing Semantics in Digital Lexicography: Proceedings of MONDILEX Fourth Open Workshop*, pages 138–144, Warszawa.

Adam Przepiórkowski and Łukasz Szałkiewicz. 2012. Anotacja morfoskładniowa. In Adam Przepiórkowski, Mirosław Bańko, Rafał L. Górski, and Barbara Lewandowska-Tomaszczyk, editors, *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Adam Radziszewski and Adam Pawlaczek. 2012. Large-scale experiments with NP chunking of Polish. In *Proceedings of the 15th International Conference on Text, Speech and Dialogue*, Brno, Czech Republic. Springer Verlag.

Adam Radziszewski. 2013. A tiered CRF tagger for Polish. In Bembenik et al. (Bembenik et al., 2013), pages 215–230.

Peter Turney. 2000. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2:303–336.

Marcin Woliński. 2006. Morfeusz — a practical tool for the morphological analysis of Polish. In Mieczysław A. Kłopotek, Sławomir T. Wierzchoń, and Krzysztof Trojanowski, editors, *Proceedings of IIPWM'06*, pages 511–520, Ustroń, Poland, June 19–22. Springer-Verlag, Berlin.