Lemmatization of Multi-Word Entity Names for Polish Language Using Rules Automatically Generated Based on the Corpus Analysis

Jacek Małyszko*, Witold Abramowicz*, Agata Filipowska*, Tomasz Wagner*

*Poznan Univeristy of Economics and Business, al. Niepodległości 10, 61-614 Poznań, Poland
jacek.malyszko@kie.ue.poznan.pl

Abstract

The article concerns automatic lemmatization of Mutli-Word Units for highly inflective languages. We present an approach, where the lemmatization is conducted using rules generated solely based on a corpus analysis. Conducted experiments revealed, that the accuracy of automatic lemmatization of MWUs for the Polish language according to the developed approach may reach up to 82%.

1. Introduction

Multi-Word Units (MWUs), or Multi-Word Expressions (MWEs), are "idiosyncratic interpretations that cross word boundaries (or spaces)" (Sag et al., 2002). Although they consist of many words (graphical units), for some application-dependent reasons they should be listed, described and processed as a single unit at some level of linguistic analysis (Handl, 2013; Savary, 2008). One type of MWUs are multi-word entity names. MWUs pose a serious difficulty in many Natural Language Processing tasks (Sag et al., 2002). One such difficulty is morphological analysis of such expressions, especially for languages with rich morphology, such as Slavic languages.

An example of task, which becomes difficult when dealing with MWUs, is their lemmatization. This is due to the fact, that the lemma of a MWU may contain words, which are not lemmas themselves (Savary, 2008). Let's analyze a Polish multi-word entity name *Organizacji Narodów Zjednoczonych* (United Nations in genitive case). If we lemmatize each word separately and concatenate received lemmas, we obtain the following phrase: *Organizacja Naród Zjednoczyć*, which is an incorrect expression according to the grammar of Polish language (correct lemma is *Organizacja Narodów Zjednoczonych*). Therefore, trying to obtain the lemma of the phrase simply by performing lemmatization of each word separately, would result in generation of a grammatically incorrect phrase.

In this paper, we analyze a problem of lemmatization of multi-word entity names for Polish language. As will be discussed in the Related Work section, a number of approaches exist towards this issue. It is commonly acknowledged that, to ensure high accuracy of the achieved results, the inflection of a phrase should be analyzed at a lexical rather than grammatical level. This usually requires a significant amount of manual work. Still, we have not found any evaluation on what accuracy can be obtained for highly inflective languages, like Polish, when the lemmatization is based only on grammatical rules, which ignore lexical information. We believe, that in some cases, such approach may be sufficient and much less labour intensive, especially when the inflection rules are automatically extracted from a corpus. Thus, the goal of this paper is to analyze what accuracy may be achieved for Polish language using only grammar-based inflection rules automatically extracted from a corpus.

The structure of this article is the following. First, in section 2., we describe the problem of MWU lemmatization in greater detail. Next, in section 3. a brief analysis of related work is presented. In sections 4. and 5. we first present a developed approach towards grammar-based MWU lemmatization and next analyze the obtained results of performed experiments. The article is concluded with a short summary.

2. Description of the encountered problem

We encountered the problem of multi-word names lemmatization for Polish language during our work on a search engine for legislative acts of Greater Poland Regional Assembly and Greater Poland Executive Board. We wanted to tag acts with names of entities, which were mentioned in the titles of acts. In many cases, some multi-word names were mentioned in the titles, usually in an inflected form. The entity names could be easily extracted, because each word in these names started with a capital letter. Having these names, we wanted to present users with tags representing these entities; if a user would click such tag, he would be presented with a list of all acts, in which this entity was mentioned in the title.

Two problems resulting from the inflection of multi-word entity names arise here:

1. linking differently inflected forms of the same names together,
2. presenting the users with lemmatized forms of the entity names.

The first problem can be solved using some text normalization techniques and string similarity mea-

inflected form $POS_{p',infl}$	**lemma** $POS_{p,lemma}$	**in English**
Zakładu Leczenia Uzależnień subst:sg:gen:m3, subst:sg:gen:n, subst:pl:gen:n	Zakład Leczenia Uzależnień subst:sg:nom:m3,subst:sg:gen:n, subst:pl:gen:n	Substance Abuse Treatment Facility
Regionalną Strategią Innowacji adj:sg:acc:f:pos, subst:sg:inst:f, subst:sg:gen:f	Regionalna Strategia Innowacji adj:sg:nom:f:pos, subst:sg:nom:f, subst:sg:gen:f	Regional Innovation Strategy
Międzynarodowych Targach Poznańskich adj:pl:gen:m3:pos, subst:pl:loc:m3, adj:pl:gen:m3:pos	Międzynarodowe Targi Poznańskie adj:sg:nom:n:pos, subst:pl:nom:m3, adj:sg:nom:n:pos	Poznań International Fair

Table 1: Exemplary three-words long MWUs. In each of them, a different number of words must be inflected to produce a lemma from the inflected form. Below the phrases, their POS tags (using NKJP tagset) are presented.

sures, such as Levenshtein distance. Still, the other one poses a greater challenge, because, as was discussed, simple lemmatization of each constituent separately will usually result in a grammatically incorrect phrase and not the lemma of the MWU. There are three main types of decisions, which must be made to correctly generate a lemma for a given MWU:

1. which words from the MWU should be inflected; in different MWUs a different number of words is being inflected. For example, having a three words long phrase, in some cases its inflection may require that only one word must be inflected, while in other cases two or even all three word must be inflected (see table 1)
2. if a given word is to be inflected, in the next step we must determine which form of a given word should be chosen, e.g. grammatical case, number and gender must be determined
3. for some languages, inflection may change the order of constituents in the MWU (Stankovic et al., 2011; Savary, 2008); still, for Polish language, this is generally not the case and we will skip this type of decisions in our work.

3. Related work

Inflection of Multi Word Units is a well-established problem in Natural Language Processing (Handl, 2013). Among others, it is often encountered when developing electronic dictionaries. Lemmatization of a phrase is one of the most important steps in this task (Stankovic et al., 2011). A list of all inflected forms of a phrase, together with their inflectional description, is called an inflectional paradigm (Savary, 2008) and generation of such paradigm was the goal of a number of previous research.

A basic requirement, which has to be met to enable automatic inflection of MWUs, is acquisition of a comprehensive inflection module or an inflectional dictionary for single words, which are constituents of MWUs (Stankovic et al., 2011). For Polish language, PoliMorf, an open morphological dictionary for Polish may be used for this purpose (Woliński et al., 2012). Still, lemmatization of single words is much more difficult when proper names are concerned, for example person names (Piskorski et al., 2007).

It is generally acknowledged, that a high accuracy of automatic inflection of MWUs may be achieved only when a lexical information is taken into account. That means, that inflection rules must be assigned on a per-phrase basis by the lexicon engineer, which is a labour intensive task (Handl, 2013). A survey of such lexical approaches to the inflection of MWUs was published in (Savary, 2008).

An exemplary lexicalized approach towards inflection of MWUs was *Multiflex*, proposed in paper (Savary, 2005). In this approach, to each phrase a so-called inflection graph is assigned, which is used to describe the inflectional behavior of a given MWU. The inflection graph is directed and acyclic and each node in it represents a single, possibly inflected, constituent. Each path in such graph corresponds to one or more inflected forms of a whole MWU. There may be many nodes corresponding to a single word in one graph and in each node there is information whether a given constituent should be inflected and, if so, how. A set of restrictions can be put on constituents, for example ensuring the agreement between specific attributes of several constituents, e.g. a grammatical case.

4. Proposed approach

In our work, we decided to try to automatically retrieve a list of lemmatization rules based on a corpus analysis. The quality of such rules will be worse than of those prepared by the expert. Still, the accuracy of lemma identification performed this way may be sufficient for some tasks and it is much less labour intensive. Also, we did not find any evaluation on how such approach may work for morphology-rich languages like Polish and we hope to fill this gap with the method described below.

4.1. Available corpus and data preparation

As was stated in section 2., we were processing legislative acts of Greater Poland Regional Assembly and Greater Poland Executive Board. In the corpus, there were in total 5172 documents. From titles of these acts, using regular expressions, we extracted 3932 multi-word units, in which there were 942 unique phrases. The acts were well formatted and in most cases, phrases from the titles, in which several consecutive words were capitalized, were entity names (we

extracted only MWUs at least three words long). The extracted entity names in many cases were inflected, but some of them were in their base form.

For each phrase, at the beginning we were determining if it is a lemma or some inflected form. We did that using a simple heuristic: if the first word of the MWU was in nominative case, we considered the phrase to be in its base form. Otherwise, the phrase was classified as inflected. For that purpose, we were using WCRFT (Radziszewski, 2013), a morphosyntactic tagger for Polish language. We found, that such approach allowed us to identify MWUs in lemma forms with accuracy above 95%.

Identification of MWUs, which already are lemmas, immediately gave us two benefits. Firstly, obviously,we did not have to process lemmas anymore. Moreover, having a lemma of a phrase, we could search trough all extracted MWUs to find inflected forms of the same phrase. Thus, we would identify other phrases, for which we know their lemma.

To identify other MWUs, which are inflected forms of a given lemma, we were generating simplified forms of phrases, where as simplified form of a phrase we understand a form, where all words from that phrase were lemmatized separately and then concatenated. For lemmatization of single words, we used Hunspell tool [1]. An example of such simplified form was already given in the Introduction; for phrases *Organizacja Narodów Zjednoczonych* and *Organizacji Narodów Zjednoczonych*, the simplified form is *Organizacja Naród Zjednoczyć*. If both phrases had the same simplified form (as is the case in the presented example), we assumed, that they differ only because of the inflection. Thus, we could identify, that a lemma for a phrase *Organizacji Narodów Zjednoczonych* is *Organizacja Narodów Zjednoczonych* (because the first word of the latter phrase is in nominative case). We will refer to such identified pairs of phrases as (*lemma, inflected form*) pairs.

Analyzing phrases from titles of acts we found 67 such (*lemma, inflected form*) pairs. To find additional pairs, we searched trough whole documents (not only the titles) to find phrases with the same simplified form as some of MWUs extracted from the titles. We found in total 634 different (*lemma, inflected form*) pairs. Still, for 433 MWUs we did not find any corresponding lemma. For these MWUs, their lemmas had to be generated automatically.

4.2. Generation of lemmatization rules

As was stated, after some data preparation steps, we were identifying (*lemma, inflected form*) pairs in the corpus. For each phrase, we also had POS tags sequences, generated using WCRFT tagger. Thus, by analyzing tags sequences in such pairs we could identify, how POS tags sequences tend to change, when a phrase with a certain tag sequence is lemmatized. We will denote POS tags sequence for a phrase p for its inflected form as $POS_{p,infl}$, and for its base form as $POS_{p,lemma}$. Having such pairs of POS tags sequences, we were automatically generating four types of lemmatization rules, which are described below.

Each rule consists of two sides: a Left Hand Side (LHS) and a Right Hand Side (RHS), separated from each other with $\rightarrow$ sign. Each side of the rule is a sequence of tags. LSH was used to match a given phrase to a specific rule; that is, having an inflected phrase p' and its POS tags sequence $POS_{p',infl}$, we were comparing it with LHSs of all rules to find a match. If a match was found, the matched rule was applied to p', that is the constituents of the phrase were inflected as was stated on the RHS the rule.

Complete rules. In this type of rules, we take POS tags sequences from lemma - inflected form pairs and consider those as lemmatization rules as shown on equation 1. Examples of such lemmatization rules are presented in table 1. In each row of this table, below phrases, there are POS tags sequences $POS_{p,infl}$ in the first and $POS_{p,lemma}$ in the second column. Using such rules, for each phrase p', for which we do not know its lemma, we retrieve its POS tags sequence $POS_{p',infl}$ and we search through all complete rules for a rule, in which $POS_{p',infl}$ was equal to its LHS. We were assuming, that in such case, if we inflect the words in the MWU according to the RHS of the rule, we will receive a correct lemma for that phrase.

$$POS_{p,infl} \rightarrow POS_{p,lemma} \tag{1}$$

An example of application of this type of rule is the following. Lets assume, that we have the following inflected MWU: p' =*Miejskim Programem Rewitalizacji* (Urban Renewal Programme in genitive). Its POS tags sequence $POS_{p',infl}$ is exactly the same as for phrase *Regionalną Strategią Innowacji* in table 1. A rule generated based on the second row in table 1 would therefore have a LHS matching to $POS_{p',infl}$. Thus, the lemma for p' is generated based on the RHS of the rule, that is using tags from the lemma column of the same row in table 1. For example, first word of p' (*Miejskim*) should be inflected to `adj:sg:nom:f:pos`. By inflecting all words from p' according to RHS, we receive phrase *Miejski Program Rewitalizacji*, which is a correct lemma for p'.

Partial Rules. Partial rules differ from complete rules in that, having a phrase p', for which we want to get its lemma, we go trough all $(POS_{p,infl}, POS_{p,lemma})$ pairs and we try to find the longest match between subsequences of $POS_{p',infl}$ and $POS_{p,infl}$, where such subsequences always start from the beginning of the sequence. If we denote subsequence starting at tag with index t_1 and ending at t_2 as $POS_{p,infl}[t_1, t_2]$, we look for the pair, in which $POS_{p,infl}[1, t_2] = POS_{p',infl}[1, t_2]$ and t_2 has the highest value. Then, we create a RHS of the rule as a concatenation of two sequences: subsequence

[1] http://hunspell.sourceforge.net/

of $POS_{p,lemma}$ ending at t_2 and subsequence of the $POS_{p',infl}$, starting at index t_2 + 1 and reaching the end of the sequence, as shown on equation 2. Please note, that $POS_{p,infl}$ and $POS_{p',infl}$ may have a different lenght (that is, the phrase being inflected may have a different number of words comparing to the phrase, which was used to generate the rule).

$$POS_{p,infl}[1,t_2] \rightarrow POS_{p,lemma}[1,t_2] + \\ + POS_{p',infl}[t_2+1,...] \quad (2)$$

Such rules are based on the fact, that in Polish language, when we inflect MWUs, in many cases some number of words at the end of the unit remain unchanged. This was shown in table 1, where in the first row two final word, and in second row one final word, remained unchanged. Thus we assume, that in many cases we may skip the analysis of some number of POS tags at the end of the sequence and still get the proper lemma. On the other hand, it is unlikely, that inflection of the MWU will change some words at the end, without affecting the ones at the beginning.

Caseless Complete Rules. In this type of rules, having a $POS_{p',infl}$, that is a sequence of tags for phrase p', for which we wanted to generate a lemma, we were analyzing lemma - inflected form pairs in search for a pair $(POS_{p,infl}, POS_{p,lemma})$, in which for $POS_{p,infl}$ all tags were the same as in $POS_{p',infl}$ apart from the grammatical case. We assume here, that grammatical cases of words in these phrases are different only because phrases p' and p were used in the text in a different case and if they would be used in the same case, then $POS_{p,infl}$ and $POS_{p',infl}$ would be identical. The described type of rules is therefore identical to Complete Rules apart from the fact, that we ignore the information about the grammatical case on the LHS of the rule.

Caseless Partial Rules. This type of rules is a variant of Partial rules, in which information about grammatical cases on the LHS of the rule is ignored. For each $POS_{p',infl}$, that is a sequence of tags for phrase p', for which we wanted to generate a lemma, we were analyzing $(POS_{p,infl}, POS_{p,lemma})$ pairs in search of the longest match between subsequences of $POS_{p',infl}$ and $POS_{p,infl}$, while in both sequences we were ignoring information about the grammatical case.

4.3. Generation of lemmas

Having some phrase in an inflected form, we were obtaining its lemma in the following manner. First, we were searching trough all lemmas found in the corpus to check, if the lemma of that phrase was found somewhere in the corpus. If the lemma was not found, we were applying rules described in the previous section in a cascade manner, in the same order as they were described above. Such order was set to ensure that rules, which we assumed would produce better results, were applied before the less reliable ones. If we found basis to apply rule of a certain type, we were generating the lemma for the phrase using a selected rule and we were ignoring rules of the subsequent types. In some cases, perhaps the analyzed phrase could match LHSs of two different rules of the same type; in such situation, we were choosing the rule to be applied randomly.

If we decided, that a certain rule should be applied, based on its RHS we knew, how words in the phrase should be inflected. For the inflection of single words, we used PoliMorf (Woliński et al., 2012) dictionary.

5. Evaluation

We performed an experiment, in which we wanted to assess what accuracy of lemma identification may be achieved for the described approach. In the experiment, we were identifying lemmas for all MWUs identified as being inflected, according to the procedure described in subsection 4.3. Using the described approach, we were trying to identify lemmas for 1067 inflected MWUs extracted from the corpus.

The evaluation of accuracy of lemmatization was performed manually. A human annotator (a native speaker of Polish language) was presented with pairs, each consisting of an inflected phrase and a lemma generated (or identified) for that phrase. The annotator was to assign to each pair two annotations:

- annotation stating if the lemma for a given phrase is correct,
- annotation stating whether the phrase is processable; by processable we understand phrases which:
  - are correctly extracted, i.e. span across the whole entity name; incorrectly extracted phrases are for example phrases missing some words from the entity name (for example the first or the last word),
  - contain only words, that may be inflected using the available dictionary; many phrases may contain non-Polish words or some proper names, which are impossible to be correctly lemmatized without appropriate dictionaries; we decided to annotate such phrases as unprocessable.

The results of the annotation are presented in table 2. There are four columns with statistics in the table. In column "accuracy for all phrases" we put accuracy for all phrases, regardless whether they were annotated as processable or not. In column "accuracy for processable phrases" we did not take into account phrases annotated as unprocessable. In the third column, we put information about the percentage of MWUs lemmatized using a given rule type, which were annotated as processable. Finally, in the last column, there is information about how many

rule type	accuracy for all phrases	accuracy for processable ph.	% of processable MWUs	# of phrases lemmatized
lemma in corpus	.9328	.9407	99.16	634
complete	.8182	.8421	86.36	22
partial	.6852	.9167	66.67	150
caseless complete	.6153	.875	61.54	26
caseless partial	.5472	.6545	51.89	146
not lemmatized	.0	.0	58.43	89
total	**.7573**	**.8214**	**83,54**	**1063**

Table 2: Accuracy of automatic lemmatization of MWUs using different lemmatization rules and a percentage of correctly extracted phrases among all that phrases were lemmatized using a given lemmatization rule type

phrases were lemmatized using a given lemmatization rule type.

The total accuracy of the proposed approach, when only processable phrases are concerned, was above 82%. When taking all phrases into account (also those incorrectly extracted ones or MWUs containing words, which we were not able to inflect) the result was around 76%. For most of the inflected phrases (634 out of 1063), using the proposed approach, the lemma could be found in the corpus. In such case, more than 94% of lemmas were assigned correctly.

For the remaining inflected MWUs, lemmas had to be generated automatically using rules described in subsection 4.2. The accuracy of lemma generation for phrases annotated as processable was generally between 84% up to 92%, except for caseless partial rules, which performed much worse than the other types of rules. To some extent, this is probably due to the fact, that rules of this type were executed only when no other rule could lemmatize a given phrase. Because of that, the rules of this type were dealing with the most difficult MWUs. For 89 phrases, we were not able to generate the lemma using the developed approach at all (none of the generated rules was matching POS tags sequences of these phrases).

6. Summary

In this paper, we presented an approach towards automatic lemmatization of Multi-Word Units for Polish language and an evaluation of lemmatization accuracy, which may be obtained using the proposed approach. The presented method utilizes rules automatically generated based on the corpus analysis. Conducted experiments revealed, that the accuracy of automatic lemmatization of MWUs for the Polish language may reach up to 82%. We believe, that such results prove, that the automatic lemmatization of MWUs may be used for some tasks. When high accuracy is a crucial factor, the proposed method may be followed by an additional step of verification by a human expert. In such case, the amount of manual work by the expert would be highly reduced comparing to situation, when he would have to assign lemmas to all phrases without any aid of a computer system.

7. References

Handl, Johannes, 2013. Computational inflection of contiguous multi-word units with jslim. *Conferencje on Intelligent Information Systems 2013*:113–126.

Piskorski, Jakub, Marcin Sydow, and Anna Kupść, 2007. Lemmatization of polish person names. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing: Information Extraction and Enabling Technologies*, ACL '07. Stroudsburg, PA, USA: Association for Computational Linguistics.

Radziszewski, Adam, 2013. A tiered CRF tagger for Polish. In R. Bembenik, Ł. Skonieczny, H. Rybiński, M. Kryszkiewicz, and M. Niezgódka (eds.), *Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions*. Springer Verlag, page to appear.

Sag, Ivan A, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger, 2002. Multiword expressions: A pain in the neck for nlp. In *Computational Linguistics and Intelligent Text Processing*. Springer, pages 1–15.

Savary, Agata, 2005. A formalism for the computational morphology of multi-word units. *ARCHIVES OF CONTROL SCIENCE*, 15(3):437.

Savary, Agata, 2008. Computational inflection of multi-word units, a contrastive study of lexical approaches. *Linguistic Issues in Language Technology*, (1-2):1–53.

Stankovic, Ranka, Ivan Obradovic, Cvetana Krstev, and Duško Vitas, 2011. Production of morphological dictionaries of multi-word units using a multipurpose tool. In *Proceedings of the Computational Linguistics-Applications Conference, October 17–19, 2011, Jachranka, Poland*. Polish Information Processing Society.

Woliński, Marcin, Marcin Miłkowski, Maciej Ogrodniczuk, and Adam Przepiórkowski, 2012. Polimorf: a (not so) new open morphological dictionary for polish. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey: European Language Resources Association (ELRA).