Extraction and Recognition of Polish Multiword Expressions using Wikipedia and Finite-State Automata

Paweł Chrząszcz
Computational Linguistics Department, Jagiellonian University
Gołębia 24, 31-007 Kraków, Poland
p.chrzaszcz@uj.edu.pl

Abstract

Linguistic resources for Polish are often missing multiword expressions (MWEs) – idioms, compound nouns and other expressions which have their own distinct meaning as a whole. This paper describes an effort to extract and recognize nominal MWEs in Polish text using Wikipedia, inflection dictionaries and finite-state automata. Wikipedia is used as a lexicon of MWEs and as a corpus annotated with links to articles. Incoming links for each article are used to determine the inflection pattern of the headword – this approach helps eliminate invalid inflected forms. The goal is to recognize known MWEs as well as to find more expressions sharing similar grammatical structure and occurring in similar context.

1 Introduction

Natural language processing often involves feature extraction from text. Extracted features include statistical measures and morphosyntatic tags – the latter are especially important for inflecting languages like Polish. For example, analyzing the word "psem" in the sentence "Wyszedłem z psem na spacer" (*I went for a walk with my dog*) results in recognition of the lemma "pies" (*dog*) and grammatical features: *masculine animate non-personal noun, instrumental case*. To obtain such information, one could use the Polish Inflection Dictionary SFJP (Lubaszewski et al., 2001) with the CLP library (Gajęcki, 2009), Morfeusz (Woliński, 2006) or Morfologik[1]. For recognition of rare words and feature disambiguation these tools can be augmented with statistical taggers using e.g. SVM, HMM or CRF classifiers. Their current accuracy for Polish reaches 90% (Waszczuk, 2012; Pohl and Ziółko, 2013).

Syntactic features are often insufficient. For example, when searching for sentences about animals, we would not find the sentence "Wyszedłem z psem na spacer" (*I went for a walk with my dog*) as the relation between the words *animal* and *dog* is semantic. Processing text semantics is a difficult task, so we often resort to manually crafted taxonomies based on paradigmatic relations like synonymy and hyponymy. Examples of such resources include WordNet (Fellbaum, 1998) and ontologies like CYC (Matuszek et al., 2006). They usually lack syntagmatic relations, which depend on the semantic roles in the particular utterance – this issue has been addressed in projects like FrameNet (Ruppenhofer et al., 2006). Unfortunately most of such resources are incomplete for English and simply not available for Polish[2].

The resources mentioned above are missing multiword expressions (MWE) which consist of multiple tokens that have their own, distinct meaning, e.g. terms ("tlenek węgla" – *carbon oxide*), idioms ("panna młoda" – *bride*), proper names ("Polski Związek Wędkarski" – *Polish Fishing Association*, "Lech Wałęsa"). Their own meaning, which cannot be inferred from their constituents, is the root cause for including them in syntactic and semantic resources for Polish. Their syntactic features can be extracted from their occurrences in corpora – their inflected forms may be used to build inflection patterns. Semantic features are more difficult

[1] Stemming library including precompiled dictionaries, https://github.com/morfologik/morfologik-stemming

[2] Except WordNet, for which there is Polish equivalent (Maziarz et al., 2012).

Proceedings of the 12th Workshop on Multiword Expressions, pages 96–106,
Berlin, Germany, August 7-12, 2016.

to extract – one could start with assigning simple semantic labels to Wikipedia headwords, like "city" for "Bielsko-Biała" (Chrząszcz, 2012).

2 Problem analysis

Simplest methods for MWE recognition use statistical measures and yield rather poor results (Ramisch et al., 2008; Zhang et al., 2006; Pecina, 2008; Ramisch et al., 2010). To increase result quality, MWE lexicons and tagged corpora are needed (Constant and Sigogne, 2011; Constant et al., 2012). The main issue with Polish is the lack of such resources – the main motivation for this work is to fill in this gap. The work is exploratory as there are no previous attempts to solve the general problem of recognition and extraction of MWEs from Polish text. One of the main assumptions of this work is to avoid the need to create lexical resources or rules by hand and use automatic methods instead – manual refinements or other improvements including e.g. supervised learning could be applied later. The results of this work should become the baseline for more advanced solutions in the future as well as provide linguistic resources (dictionaries) with MWEs.

Semantic resources such as WordNet can often be replaced with Wikipedia – although its content often lacks the quality and formal structure provided by ontologies and WordNet, its large and diverse data collection seems enough to make up for these issues. Wikipedia content can be used in many ways, e.g. to extract words and MWEs (from page titles), semantic labels describing meaning (from article content), semantic relations between concepts (from redirections, links and categories) and as an annotated corpus to train statistical algorithms. It has been successfully used for named entity (NE) recognition (NER), e.g. the category of the entity can be inferred from the definition itself (Kazama and Torisawa, 2007) and links between articles can be considered tags marking NE occurrences in text (Mihalcea and Csomai, 2007; Nothman et al., 2009). There is also some evidence that e.g. semantic relatedness for word pairs can be computed more accurately using Wikipedia than with WordNet or other resources (Gabrilovich and Markovitch, 2007). MWE recognition and extraction using Wikipedia is less common, but there are some attempts of classifying Wikipedia headwords using e.g. manual rules (Bekavac and Tadic, 2008) or cross-lingual correspondence asymmetries in interwiki links (Attia et al., 2010). Vincze et al. tagged 50 articles of the English Wikipedia to create a corpus with marked MWE occurrences and used a CRF classifier to recognize MWEs and NEs in text with F-measure (F_1) of 63% (Vincze et al., 2011). These examples are enough to let us consider Wikipedia as the primary linguistic resource for MWE recognition and extraction. Together with an inflection dictionary it can be used to extract Polish MWEs using various methods. This work focuses on design and implementation of such methods. However, the first step is to formulate the definition of a Polish MWE that would narrow down the scope of the problem.

3 Definition of a Nominal MWE

The most widely used definition of an MWE is the one by Sag et al.: "idiosyncratic interpretations that cross word boundaries (or spaces)" (Sag et al., 2002). The authors distinguish four different categories of MWEs for which we could find Polish equivalents:

1. Fixed expressions – they have a fixed meaning and structure and are uninflected, e.g.: "ad hoc", "mimo wszystko" (*regardless*), "ani mru-mru" (*not a dicky bird*).
2. Semi-fixed expressions – they are mostly nominal expressions that have a fixed meaning and are inflected. Examples include "panna młoda" (*bride*, literally: *young maiden*), "biały kruk" (*rarity*, literally: *white crow*). Verbal idioms like "mówić trzy po trzy" (*to speak nonsense*) as well as proper names also belong to this category.
3. Syntactically-flexible expressions – they also have a fixed meaning, but their syntactic structure is loose, allowing changes like inserting new tokens or changing their order. They are often verbal templates that can be filled with nouns to make complete sentences, e.g. "działać jak płachta na byka" (*to irritate sb.*, literally *to be like a red rag to a bull*), "gotów na czyjeś każde skinienie" (*to be at one's beck and call*).
4. Institutionalized phrases – their meaning and syntactic structure can be inferred from the in-

Table 1: Examples of nominal MWEs that are the concern of this research. Inflected tokens are underlined.

Category	Examples
Personal names	Józef Piłsudski, Szymon z Wilkowa (*Simon from Wilków*)
Other proper names	Lazurowa Grota (*Azure Cave*), Polski Związek Wędkarski (*Polish Fishing Association*)
Expressions including names	rzeka Carron (*River Carron*), jezioro Michigan (*Lake Michigan*), premier Polski (*Prime Minister of Poland*)
Common words, semantically non-decomposable	panna młoda (*bride*), świnka morska (*guinea pig*), czarna dziura (*black hole*)
Common words, semantically partially decomposable	chlorek sodu (*sodium chloride*), baza wojskowa (*military base*), lampa naftowa (*kerosene lamp*), zaimek względny (*relative pronoun*)

dividual tokens. The complete expression can be considered an MWE only because of its frequent use. Examples include "czyste powietrze" (*clean air*), "dookoła świata" (*around the world*), "ciężka praca" (*hard labour*).

A decision was made to choose only the **second** category from the list above, further limited to the **nominal** expressions. The main motivation for these restrictions is that this category is the most well-defined one and vast majority of MWEs used in Polish text are nominal. What is more, this limitation helps avoid issues with classifying the word as an MWE (Pecina, 2008) as well as non-continuous expressions (Graliński et al., 2010; Kurc et al., 2012). As a consequence, Polish multiword expressions can be defined in this paper as inflected nominal expressions that have a fixed meaning which is not fully decomposable and have a well-defined, strict inflection pattern. An MWE is thus a sequence of tokens (words, numbers and punctuation marks), which fall into two main categories:

- **Inflected tokens** build the main part of the MWE. They can be nouns, adjectives, numerals or adjectival participles. Their case and number have to agree with the corresponding features of the whole expression. In the base form all inflected tokens are nominative and singular (except *pluralia tantum*). Inflected tokens need not have the same gender, e.g. "kobieta kot" (*cat-woman*), but they cannot change gender through inflection.
- **Uninflected tokens** are all the remaining tokens that remain fixed when the whole expression is inflected, e.g. words, numbers, punctuation marks or other segments (e.g. "K2").

Examples of such MWEs are presented in tab. 1.

4 A system for MWE processing

After defining Polish nominal MWEs, the next goal was to develop a system for automatic extraction and recognition of such expressions. The architecture of the implemented system is shown in fig. 1. The first step is the extraction of data from Polish Wikipedia[3]. To do this, Wikimedia dumps[4] were used. Extracted data included article content, redirections, links between pages, templates and page categories. The Wiktionary[5] was also considered

[3] http://pl.wikipedia.org
[4] http://dumps.wikimedia.org
[5] http://pl.wiktionary.org

as a potential data source, but it turned out that the number of MWEs it contained was very low – only 1118 (Wikipedia dump contained about 973 thousand MWEs).

It was decided that all the extracted MWEs should contain at least one inflected token that would be recognized by Polish dictionaries. The main morphosyntactic resource used for token recognition and grammatical feature extraction was the Polish Inflection Dictionary SFJP (Lubaszewski et al., 2001) with the CLP library. Its content was extended with other Polish resources: Morfeusz (Woliński, 2006) and Morfologik. SFJP is a dictionary where each entry has its unique identifier and a vector of forms while the latter two dictionaries use a completely different data format (morphosyntatic tags), so the data needed to be merged using a new format – the resulting dictionary was called CLPM. The content of this dictionary was stored using LMDB[6] – a database optimized for the lowest possible read time. The following example presents the result (**dictionary tag**) returned for the token "wola" found in text:

$$\{(\texttt{ADA-wola}, \{1\}),\\ (\texttt{AEA-wole}, \{2, 8, 11, 14\}),\\ (\texttt{CC-woli}, \{15, 21\})\}$$

The result is ambiguous. There are three possible recognized lexemes:

- `ADA-wola` – feminine noun "wola" (*will*), singular nominative (1),
- `AEA-wole` – neuter noun "wole" (*craw*), singular genitive (2) or plural: nominative, accusative or vocative (8, 11, 14),
- `CC-woli` – adjective "woli" (*bovine*), plural feminine, nominative or vocative (15, 21).

These ambiguities could be limited by using statistical or rule-based taggers or parsers, but this would introduce a significant error rate – approximately 10% for Polish (Pohl and Ziółko, 2013). It is worth noting that the dictionary tag format presented above is less verbose and repetitive than the morphosyntactic tag format used by Morfeusz and Morfologik. It can also distinguish between fixed and inflected grammatical categories. The main downside is that it is slightly less human-readable.

[6]Symas Lightning Memory-Mapped Database, `http://symas.com/mdb`

4.1 DM Method

DM (Dictionary Matching) is the first proposed method that uses the set of Wikipedia headwords as a lexicon of MWEs. It can be considered both a baseline with which better algorithms could be compared and a building block for compound methods. The main issue with using such a lexicon is that we have no knowledge of the inflection pattern of the headwords – tokens can be inflected or not, have ambiguous form etc. For each headword we create a **dictionary pattern** that includes all the possible variants for each token. For example, while processing the headword "Droga wojewódzka nr 485" (*Provincial road no. 485*) several ambiguities are encountered:

- The token "Droga" (*Road*) can be capitalized or not as all Wikipedia headwords are capitalized and the token itself is a common word.
- The token "Droga" (*Road*) can be inflected or not. Similarly, the token "wojewódzka" (*provincial*) can be inflected or not. The only thing we know is that at least one of these tokens has to be inflected for the expression to be a nominal MWE.
- The token "Droga" (*Road*) can actually also be a feminine adjective meaning *expensive*.

A simple textual format was used to store all possible ambiguous variants for each token (fig. 1, transition 1a). As there could be multiple ambiguities for a single sequence of input tokens and the number of possible variants grows exponentially with the number of ambiguities, it was decided that instead of a flat lexicon with all possible forms, a finite state machine would be used (fig. 1, transition 1b). As the machine outputs the recognized dictionary patterns in each state, it can be defined formally as a **Moore machine**. For this approach to work in case of continuous text, a separate machine has to be started for each token – each instance thus recognizes all possible MWEs starting at that token.

When a sequence of input tokens successfully matches a pattern, the expression is stored in a database with its lemma and disambiguated syntactic features. As an example let us consider the sentence "Rozpoczął się remont drogi wojewódzkiej nr 485." (*Renovation of the provincial road no. 485*

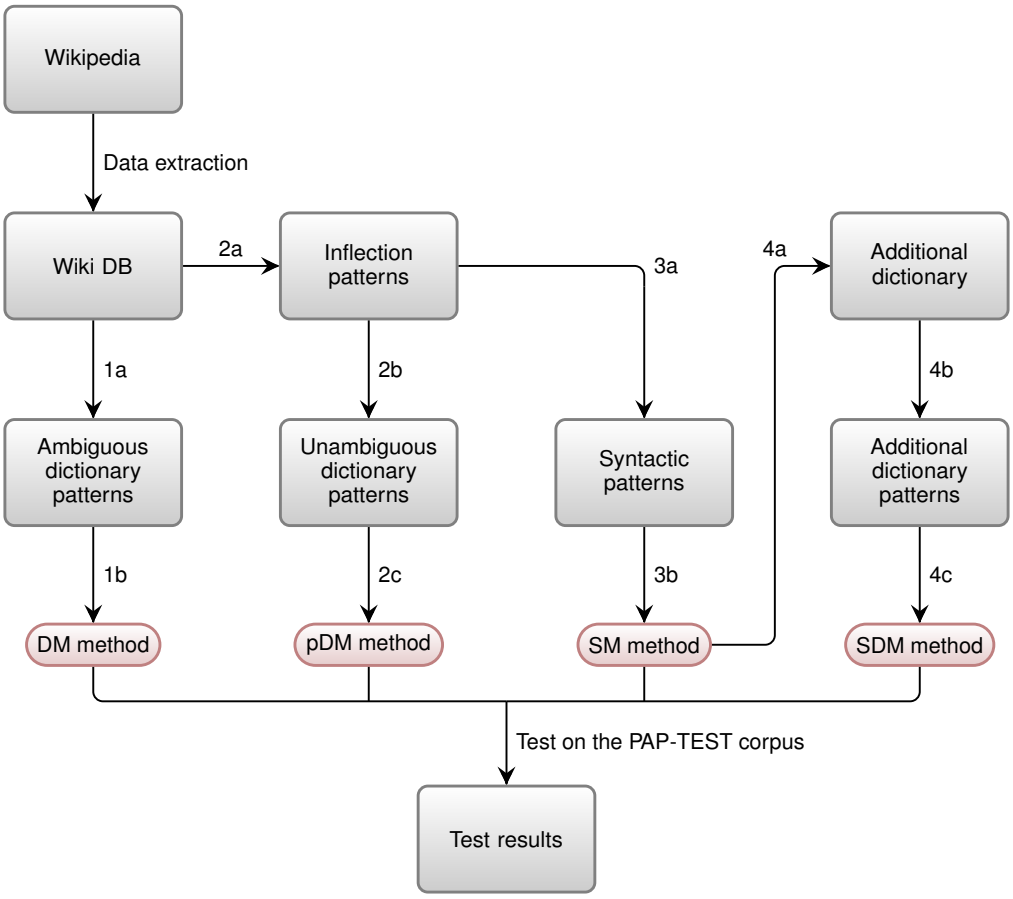

Figure 1: Architecture of the Polish MWE recognition and extraction system.

has started). The sequence "drogi wojewódzkiej nr 485" matches the pattern described above and the whole expression is in the genitive case[7]. The first word is also lowercased. This allows us to not only recognize the MWE, but also disambiguate the pattern and store the disambiguated version in a dictionary of extracted MWEs. Of course this is not always possible – for example the sentence "Droga wojewódzka nr 485 rozpoczyna się w Gdańsku." (*Provincial road no. 485 starts in Gdańsk*) does not allow such disambiguation. Multiple patterns can overlap and the algorithm offers a few different strategies of choosing the best non-contradictory subset of such patterns.

4.2 pDM method

After analysis of the DM method performance it became obvious that there was a need for prior disambiguation of the dictionary patterns. The first attempt to solve this was to use a heuristic disambiguation, but it was limited by the simple finite-state logic it used. To make the method open and not limited by any handcrafted rules, a new approach was chosen. For a given article, it uses incoming links to learn the **inflection pattern** of the headword (fig. 1, transition 2a). For example, the link "czarnej dziury" (genitive case) leads to the headword "Czarna dziura" (*black hole*). This allows us to identify the inflected tokens and determine if the first token is lowercased. For entries that have little or no incoming links, we could either use the original DM method or skip them completely. Another issue is poor quality of the links – some of them are mislabeled, contain incorrect inflected forms or differ from the entry (e.g. are abbreviated or contain additional tokens). This issue is the main reason for designing a quite complex algorithm that determines the inflection pattern for a given Wikipedia headword in the following steps:

1. A statistics of the incoming links is created.

[7] Although individual tokens have ambiguous grammatical form, matching them against the dictionary pattern allows to disambiguate it.

Table 2: Elements of the syntactic pattern with context for the link "centralnej **czarnej dziury**."

Pattern element	Content	Description
left context	`cc16,cc17,cc20`	The label '`cc`' means 'adjective' (as the word "centralnej" (*central*) is an adjective), while the numbers 16, 17 and 20 denote the possible cases (genitive, dative or locative) together with the feminine gender.
expression	`*cc15 *ad1`	The MWE "czarna dziura" (*black hole*) consists of two inflected tokens, marked with asterisks. The first one is a feminine singular (form number 15) adjective ('`cc`' label) while the second one is a nominative singular (form number 1) feminine noun ('`ad`' label). Note: this is the pattern of the MWE in its **base** form.
right context	`_p`	The full stop following the expression is a punctuation mark ('`p`' label) without a preceding space ('`_`' prefix).
grammatical form	$\{2\}$	The MWE occurs in the singular genitive form.

2. For each link in the statistics all the possible inflection patterns are generated.
3. An attempt is made to determine if the first token should be capitalized.
4. The largest set of links that have non-contradictory inflection patterns is found.
5. The inflection pattern for the discovered set is saved to the database.

For the entries for which inflection patterns were successfully determined, new unambiguous dictionary patterns are created. They are then used to construct a Moore machine like for the DM method (fig. 1, transitions 2b and 2c). This variant is called **pDM**.

4.3 SM method

The methods of MWE extraction described so far focus on recognition of the Wikipedia entries and extract some new syntactic information. To overcome this limitation, we need to introduce rules or patterns that would allow extraction of new, unknown expressions. Such patterns and rules are often handcrafted (Bekavac and Tadic, 2008; Woźniak, 2011; Buczyński and Przepiórkowski, 2009; Piskorski et al., 2004; Ramisch et al., 2010). However, it turns out that a lot can be achieved using only the existing inflection patterns that we have already created for the pDM method – we could use them to extract new MWEs that have similar grammatical structure. For example, expressions such as "tlenek węgla" (*carbon oxide*), "siarczan miedzi" (*copper sulfate*) or "wodorotlenek sodu" (*sodium hydroxide*) consist of an inflected masculine nominative noun followed by an uninflected genitive noun. Moreover, the pattern can include the context in which such expressions occur[8], e.g. the mentioned MWEs occur in similar expressions like "... reakcja **siarczanu miedzi** z ..." (... *reaction of copper sulfate with* ...). This observation was the motivation to create a new algorithm that would use the inflection patterns and contexts extracted from links to create **syntactic patterns** describing the syntactic structure of the MWEs as well as the contexts in which they occurred (fig. 1, transition 3a). Different levels of pattern granularity were examined and the final decision was to store the following information:

- For each token of the expression: part of speech, inflection flag (inflected/uninflected), grammatical number and gender for inflected tokens and the case for uninflected ones.
- The context is limited to one token before and after the MWE. The information stored for each token of the context includes token type (word, number, punctuation mark), part

[8] Farahmand and Martins also noticed and utilized this fact (Farahmand and Martins, 2014).

Table 3: Examples of syntactic patterns with context created for a few MWEs. There are two unique pattern identifiers: $cpid$ identifies the pattern with its context while pid identifies the pattern without the context. Form statistics consists of pairs (F, N) where F is a set of grammatical forms in the CLPM format (it has more than one element if the form is ambiguous) and N is the number of occurrences of the MWEs with form set F. A vertical line "|" indicates a sentence boundary while "g" indicates a preposition. The last MWE is a *plurale tantum*.

MWE	$cpid$	pid	Pattern with context	Form statistics
ślad macierzy	1	1	`|*ac1 ad2,ad3,ad6,ad7,ad9 cc37`	$(\{1,4\},1)$
cząstka elementarna	2	2	`ac1,ac4 *ad1 *cc15 g`	$(\{2,3,6\},3),(\{9\},8)$
łódź podwodna	2	2	`ac1,ac4 *ad1 *cc15 g`	$(\{9\},1)$
łódź podwodna	3	2	`ac1,ac4,ad9 *ad1 *cc15 g`	$(\{9\},7)$
wojny syryjskie	4	3	`ac1,ac4 *ad8 *cc36 g`	$(\{9\},1)$

of speech, case and for pronouns – the word itself.

For example, the link "centralnej **czarnej dziury**." would result in the pattern `cc16,cc17,cc20 *cc15 *ad1_p`. This example is shown in detail in table 2.

The patterns are saved with their grammatical forms (case and number) in which they occurred in text – this results in a large database of pattern statistics. The next step is to create an automaton similar to the one used for the DM and pDM methods (fig. 1, transition 3b), which is used to recognize expressions matching the patterns and to extract their syntactic features. The resulting method is called **SM** (*Syntactic Matching*). Contrary to pDM, its results are highly ambiguous as each expression could match multiple patterns and yield multiple overlapping results. Choosing the right one requires introducing a function that would assign a quality measure to each result. We decided to use a quantitative measure rs (result score) which sums the numbers of occurrences of the recognized patterns in given forms in the original set of Wikipedia links.

Example. Let us consider the following Wikipedia headwords: "Ślad macierzy" (*matrix trace*), "Cząstka elementarna" (*elementary particle*), "Łódź podwodna" (*submarine*) and "Wojny syryjskie" (*Syrian Wars*). Let us also limit the occurrences of these MWEs to the ones listed in table 3. The table shows that three patterns are created. The second pattern has two different context patterns, hence the four different values of $cpid$. It is also worth noting that the set of forms (F) can have multiple elements in case of ambiguous forms. Such sets cannot be split in the statistics. The patterns from tab. 3 can be used to create the Moore machine shown in fig. 2. This FSM can be then used to recognize MWEs in the following sentence: "Rozwój chmur kłębiastych i lokalnych burz." (*Development of cumulus clouds and local storms*). Table 4 shows the recognized MWE candidates with corresponding values of $cpid$. These results should be now converted into MWEs – this means changing their form to the base one, identifying inflected tokens and the IDs of the tokens in CLPM. As the example is very simple, it turns out that each result yields exactly one MWE candidate and all of them are overlapping. This means that we need to calculate their rs scores. The highest score (16) is achieved by the MWE "chmura kłębiasta" (*cumulus cloud*). This is because the pattern with $cpid$'s 2 and 3 (table 3) has $8 + 1 + 7 = 16$ occurrences for the form sets which intersect $F = \{9\}$. As the remaining candidates (meaning *cumulus clouds* and *cloud development*, respectively) have lower scores (1), they are discarded.

To improve MWE candidate selection, supervised learning was also considered and tested. The training set contained 4000 manually annotated MWE candidates: about 1500 positive and 2500 negative samples. This set was used to train binary classifiers including LDA, SVM with different kernels, Maximum Entropy model, decision trees and finally AdaBoost, which performed best. However, the initial results were only marginally better (within 1%) than the ones given by the rs measure described above. This research is still ongoing.

4.4 SDM method

The results of applying the SM method to a text corpus can be converted to a dictionary format (fig. 1, transition 4a) – this way we would create an additional dictionary resource that could increase the

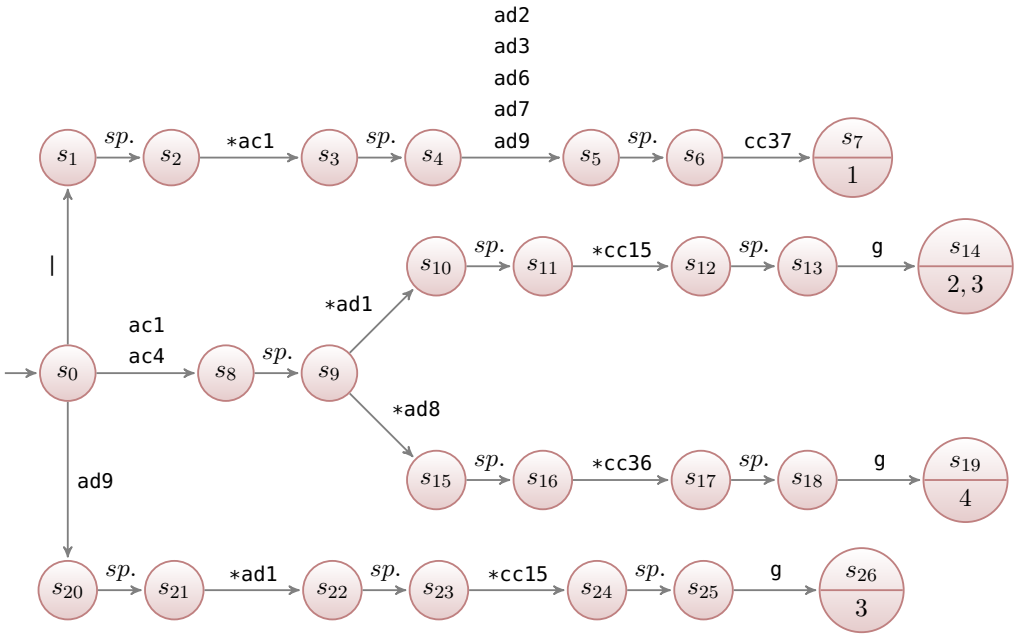

Figure 2: State machine recognizing the patterns from tab. 3. Multiple transitions between the same pair of states are denoted with a single arrow and aligned vertically. The symbol *sp.* means a space. Numbers below the state symbol are *cpid* values of recognized patterns.

Table 4: Results of MWE recognition using the FSM from fig. 2 in the sentence "Rozwój chmur kłębiastych i lokalnych burz.".

cpid	**Path in the FSM**	**Forms** (F)	**Token sequence**	**MWE (base form)**	rs
1	`| *ac1 ad9 cc37`	1, 4	Rozwój chmur	rozwój chmur	1
2, 3	`ac1,ac4 *ad1 *cc15 g`	9	chmur kłębiastych	chmura kłębiasta	16
4	`ac1,ac4 *ad8 *cc36 g`	9	chmur kłębiastych	chmury kłębiaste	1

possibilities of the pDM method. Two text corpora were used for this operation:

- PAP-TRAIN – Polish Press Agency (PAP) releases, 3.6 million tokens.
- WIKI – contents of all Wikipedia articles, 202.7 million tokens.

The resulting dictionary was filtered and disambiguated to increase its quality. There is a trade-off between size and quality of the resulting dictionary – the values depend on the threshold rs measure described above. For example, if the target is a dictionary with one million expressions, it would contain about 75% correct MWEs[9]. The remaining steps are similar as for pDM: dictionary patterns are created, followed by the automaton (fig. 1, transitions 4b and 4c). The resulting method is called **SDM**.

[9]Tested on a sample of 2000 entries.

5 Tests

The described methods were tested on a random sample of 100 PAP press releases, in which MWEs were manually annotated by two annotators[10]. The test corpus, which contains 572 tagged MWEs, is called PAP-TEST[11]. For each MWE its location was marked and all inflected tokens were also indicated. The test itself consists in choosing one or more methods (**DM**, **pDM**, **SM** and **SDM**) with their optimal parameters[12] and re-tagging the PAP-TEST corpus automatically. The resulting automatically tagged corpus, denoted PAP-WW, was then compared with PAP-TEST. As a result, four sets of expressions are determined:

- T_i – correct MWEs present in both corpora

[10]Disagreements between annotators were discussed and resolved.

[11]Its content is excluded from PAP-TRAIN.

[12]Two-fold cross validation was performed for parameter optimization.

Table 5: Results of the MWE recognition and extraction tests. The best result in each column is **highlighted**.

Method	Recognition test			Extraction test		
	P_{rec}	R_{rec}	F_{rec}	P_{ext}	R_{ext}	F_{ext}
DM	80.97	42.54	55.78	58.71	30.85	40.44
pDM	**90.12**	38.64	54.09	**86.96**	37.29	52.19
SM	50.46	64.75	56.72	47.82	61.36	53.75
SDM	62.83	64.75	63.77	60.86	62.71	61.77
pDM + SDM + SM	72.27	**70.14**	**71.19**	69.23	**67.19**	**68.19**

with correctly identified inflected tokens.

- T_d – correct MWEs present in both corpora with incorrectly identified inflected tokens.
- F_n – missing MWEs (false negatives, present only in PAP-TEST).
- F_p – incorrect MWEs (false positives, present only in PAP-WW).

Two types of test were performed: the **recognition test** considers T_d elements as correct while the **extraction test** considers them as incorrect. For each test **precision** (P) and **recall** R values are calculated using the following formulas:

$$P_{rec} = \frac{|T_i \cup T_d|}{|T_i \cup T_d \cup F_p|} \qquad R_{rec} = \frac{|T_i \cup T_d|}{|T_i \cup T_d \cup F_n|}$$

$$P_{ext} = \frac{|T_i|}{|T_i \cup T_d \cup F_p|} \qquad R_{ext} = \frac{|T_i|}{|T_i \cup T_d \cup F_n|}$$

For both methods **F-measure** is also calculated: $F_1 = \frac{2PR}{P+R}$, denoted F_{rec} and F_{ext} respectively.

5.1 Test results

The results are shown in table 5. The pDM method is the most precise as it extracts only Wikipedia headwords that have been additionally filtered when creating inflection patterns. The most noticeable difference to DM is in the P_{ext} value. The SM method does not have high precision, but its recall is enough to build a dictionary which enables SDM to reach high results. The last row shows a combined method that merges the results of the three methods: pDM, SDM and SM. The methods are prioritized respectively – this ensures that results of methods with higher recall are preferred. Although the combined method yields good results, there is still a quite large number of errors, whose reasons mostly fall into the following categories:

- Long and complicated expressions, e.g. long school name "V Liceum Ogólnokształcące im. Augusta Witkowskiego" consisting of the short name "V Liceum Ogólnokształcące" and the patron name "August Witkowski", which were recognized separately – this means one false negative and two false positives.
- Missing foreign words (including names) in CLPM, e.g. "Sampras" in "Pete Sampras".
- Spelling and typographical errors like "W.Brytania" (*Great Britain*, missing space), "Białego Domy" (*the White House*, the grammatical form of the tokens does not match).
- Expressions which are not considered MWEs e.g. dates like "stycznia 1921" (January 1921), "grudniu 1981" (December 1981).

To sum up, the results are positive and reflect the quality of the method in a real-word scenario. There are possibilities of future improvement.

6 Conclusions

The results show that it is possible to recognize and extract Polish MWEs using an inflection dictionary and Wikipedia without the need for manually crafted rules or training sets. It is also possible to create a dictionary of Polish MWEs from the results of the extraction process. The main future goal is to clean the resulting dictionary using both manual effort and machine learning algorithms. However, initial research shows that this will be a difficult problem as even a training set of 4000 positive/negative MWE examples used to train various classifiers including AdaBoost was not enough to give improvement in F_{ext} larger than 1%. This research is still ongoing.

References

Mohammed Attia, Lamia Tounsi, Pavel Pecina, Josef van Genabith, and Antonio Toral. 2010. Automatic extraction of arabic multiword expressions. In *23rd International Conference on Computational Linguistics: Proceedings of the Workshop on Multiword Expressions: From Theory to Applications (MWE)*, pages 19–27. Association for Computational Linguistic.

Božo Bekavac and Marko Tadic. 2008. A generic method for multi word extraction from wikipedia. In *30th International Conference on Information Technology Interfaces (ITI)*, pages 663–668. IEEE.

Aleksander Buczyński and Adam Przepiórkowski. 2009. Spejd: A shallow processing and morphological disambiguation tool. In *Human Language Technology. Challenges of the Information Society*, pages 131–141. Springer.

Paweł Chrząszcz. 2012. Enrichment of inflection dictionaries: automatic extraction of semantic labels from encyclopedic definitions. In *Proceedings of the 9th International Workshop on Natural Language Processing and Cognitive Science (NLPCS, ICEIS)*, pages 106–119. SciTePress.

Matthieu Constant and Anthony Sigogne. 2011. Mwu-aware part-of-speech tagging with a crf model and lexical resources. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pages 49–56. Association for Computational Linguistics.

Matthieu Constant, Anthony Sigogne, and Patrick Watrin. 2012. Discriminative strategies to integrate multiword expression recognition and parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers – Volume 1*, pages 204–212. Association for Computational Linguistics.

Meghdad Farahmand and Ronaldo Martins. 2014. A supervised model for extraction of multiword expressions based on statistical context features. In *Proceedings of the 10th Workshop on Multiword Expressions (MWE, EACL)*, pages 10–16. Association for Computational Linguistics.

Christiane Fellbaum. 1998. *WordNet: an electronic lexical database*. MIT Press.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence (IJCAI)*, volume 7, pages 1606–1611. Morgan Kaufmann Publishers Inc.

Marek Gajęcki. 2009. Słownik fleksyjny jako biblioteka języka c. In Wiesław Lubaszewski, editor, *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu*. AGH Press, Kraków.

Filip Graliński, Agata Savary, Monika Czerepowicka, and Filip Makowiecki. 2010. Computational lexicography of multi-word units: how efficient can it be? In *Proceedings of the Workshop on Multiword Expressions: from Theory to Applications (MWE)*, pages 1–9. Association for Computational Linguistics.

Jun'ichi Kazama and Kentaro Torisawa. 2007. Exploiting wikipedia as external knowledge for named entity recognition. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 698–707. ACL.

Roman Kurc, Maciej Piasecki, and Bartosz Broda. 2012. Constraint based description of polish multiword expressions. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*, pages 2408–2413. European Language Resources Association.

Wiesław Lubaszewski, H. Wróbel, M. Gajęcki, B. Moskal, A. Orzechowska, P. Pietras, P. Pisarek, and T. Rokicka. 2001. *Słownik Fleksyjny Języka Polskiego*. Computational Linguistics Group, Department of Computer Science, AGH UST and Department of Computational Linguistics, Jagiellonian University, Kraków.

Cynthia Matuszek, John Cabral, Michael J. Witbrock, and John DeOliveira. 2006. An introduction to the syntax and content of cyc. In *AAAI Spring Symposium: Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, pages 44–49.

Marek Maziarz, Maciej Piasecki, and Stanisław Szpakowicz. 2012. Approaching plWordNet 2.0. In *Proceedings of the 6th Global Wordnet Conference*. Global WordNet Association.

Rada Mihalcea and Andras Csomai. 2007. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the 16th Conference on Information and Knowledge Management (CIKM)*, pages 233–242. Association for Computing Machinery.

Joel Nothman, Tara Murphy, and James R Curran. 2009. Analysing wikipedia and gold-standard corpora for ner training. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 612–620. Association for Computational Linguistics.

Pavel Pecina. 2008. A machine learning approach to multiword expression extraction. In *Proceedings of the LREC Workshop – Towards a Shared Task for Multiword Expressions (MWE)*, pages 54–61. European Language Resources Association.

Jakub Piskorski, Peter Homola, Małgorzata Marciniak, Agnieszka Mykowiecka, Adam Przepiórkowski, and Marcin Woliński. 2004. Information extraction for polish using the sprout platform. In *Intelligent Information Processing and Web Mining*, volume 25 of *Advances in Soft Computing*, pages 227–236. Springer Berlin Heidelberg.

Aleksander Pohl and Bartosz Ziółko. 2013. A comparison of polish taggers in the application for automatic speech recognition. In *Proceedings of the 6th Language and Technology Conference (LTC)*, pages 294–298.

Carlos Ramisch, Paulo Schreiner, Marco Idiart, and Aline Villavicencio. 2008. An evaluation of methods for the extraction of multiword expressions. In *Proceedings of the LREC Workshop – Towards a Shared Task for Multiword Expressions (MWE)*, pages 50–53. European Language Resources Association.

Carlos Ramisch, Aline Villavicencio, and Christian Boitet. 2010. Mwetoolkit: a framework for multiword expression identification. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*, pages 662–669. European Language Resources Association.

Josef Ruppenhofer, Michael Ellsworth, Miriam R.L. Petruck, Christopher R. Johnson, and Jan Scheffczyk. 2006. *FrameNet II: Extended theory and practice*. International Computer Science Institute, Berkeley, CA.

Ivan A. Sag, 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*, volume 2276 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin Heidelberg.

Veronika Vincze, István Nagy, and Gábor Berend. 2011. Multiword expressions and named entities in wikipedia articles. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 289–295.

Jakub Waszczuk. 2012. Harnessing the crf complexity with domain-specific constraints. the case of morphosyntactic tagging of a highly inflected language. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 2789–2804.

Marcin Woliński. 2006. Morfeusz — a practical tool for the morphological analysis of polish. *Advances in Soft Computing*, 26(6):503–512.

Michał Woźniak. 2011. Automatic extraction of multiword lexical units from polish text. In *5th Language and Technology Conference (LTC)*.

Yi Zhang, Valia Kordoni, Aline Villavicencio, and Marco Idiart. 2006. Automated multiword expression prediction for grammar engineering. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 36–44. Association for Computational Linguistics.