Categorization of Persons Based on Their Mentions in Polish News Texts

Submitted: 20th October 2018; accepted: 2nd June 2020

Maciej Pachocki, Anna Wróblewska

DOI: 10.14313/JAMRIS/2-2020/19

Abstract:

Our goal described in this paper was to design, implement and test a method of categorization of mentions of persons in Polish news texts. We gathered and classified the input data in order to measure the accuracy of the method. Train and test data were constructed by using lists of persons collected from YAGO knowledge base and Polish Wikipedia. During tests the efficiency of categorization depending on different representations of a person was studied. Experiments were executed on our and a chosen solution from literature. The results are shown and discussed in the paper.

Keywords: *fined-grained named entity classification, text classification, categorization of persons*

1. Introduction

The problem of categorizing persons considered in this article concerns finding additional information about individuals detected in text written in the Polish language. Apart from the basic knowledge that a given entity in text is a person further classification predominantly lets us obtain the name of the profession which is pursued by a given person. It results from the fact that most often in text people are mentioned or described in terms of their work and less often with respect to their beliefs, age or interests.

The categorization of persons enables the creation of a taxonomy of people sharing the same profession. This can be applied in different domains. In information retrieval task persons following the same profession can be suggested (e.g. in Web search engines). Furthermore a taxonomy of individuals may be employed in a question answering system.

In comparison to the recognition of general entities (persons, places, organisations, etc.) the categorization of people is a narrower area of study and at the same time more difficult due to higher number of possible categories and fewer semantic differences appearing between them. Mistakes made by a computer classifier or even by a human will more often be an assignment of an unsuitable profession for an individual rather than the recognition of a person as a place or organisation or other named entity. Words occurring on the left and right side of the given entity very often indicate her general category (person, place or organisation). However after detecting a person in text these words may be not sufficient to define her/his occupation.

Most of the analyzed methods applied to categorizing persons used supervised machine learning ([1], [2], [3], [4], [5]). The authors of mentioned papers focused on feature selection and data set generation. Other considered methods employed an algorithm which measured similarity between an entity's representation and possible categories ([6], [7]). Subsequently the considered entity was categorized to the most similar category in a taxonomy.

The objective of this work was to design and implement an application which would categorize persons previously recognized in texts written in Polish language. The program takes as input a text containing tags which indicate occurrences of persons. Result of the application is assigning each occurring person to one of the possible categories. The number of used categories was restricted to ten. These categories were: "clergymen", "painters", "musicians", "journalists", "sportsmen", "politicians", "lawyers", "actors", "doctors" and "poets".

In the next section we describe our method. Subsequently we show the results (section 3), discuss them (section 4) and draw conclusions (section 5).

2. The Categorization Method

Subsection 2.1 describes a procedure used to determine possible categories of persons. The developed categorization method uses supervised machine learning. The next subsections present the realization of the fundamental stages that must be followed according to this approach.

2.1. Set of Possible Categories

In order to determine the main possible categories of persons an algorithm has been developed which takes as input lists of persons from YAGO knowledge base ([8]) and a corpus and automatically detects most often appearing classes of persons. The devised algorithm consists of the following steps:

1) Download lists of persons from YAGO knowledge base which has more than k persons (k = 10 000).
2) Search the set of documents in order to find the number of occurrences of persons belonging to each list.
3) Select categories in which the number of occurrences of persons is bigger than l (l = 200). Subsequently delete categories which are not leafs in the created hierarchy and do not concern certain profession.

YAGO knowledge base makes available a very large hierarchy of persons consisting of categories downloaded from taxonomy of WordNet and category system of English Wikipedia. To cut it down in the first step of the algorithm the taxonomy was restricted to categories which has more than 10,000 persons. Consequently 105 categories were obtained. The second step of the algorithm was conducted on a subcorpus of National Corpus of Polish[1] which has approximately 1 million of words. In the third step the minimum number of occurrences of persons was set to 200. After deleting categories which are not leafs in the created hierarchy and do not concern a certain profession 10 categories were left. Six categories from YAGO are consistent with the final set of categories. Category "football player" was replaced with more general "sportsman". Classes "minister", "president" and "artist" were deleted. The first two were removed because they were very similar to "politician" category and the third one was too general. Newly added classes were: "doctor", "musician" and "painter".

2.2. Input Data set

A set of documents was constructed from texts published on popular Polish news portals. Ten thousand documents were gathered. The method of creating a data set used in categorization is shown in Figure 1. The depicted process consists of:

- acquiring lists of persons grouped according to their profession - it is based on gathering list of persons for each possible category; in our method lists were downloaded from the YAGO knowledge base and Polish Wikipedia;
- recognizing persons in text - it is based on applying a program whose task is to process a set of documents and tag places where persons occur in text; this kind of application was made available by *Findwise* company[2];
- adding to persons tags denoting their category - it is based on comparing entities tagged as persons in texts with individuals appearing in lists of persons grouped by categories. As a result the data set used in categorization is created with marked places of occurring politicians, actors, etc.

Figure 1 also shows the processing of an example sentence in the developed method. The sentence in our set of documents may be: "A Postgame interview was conducted with the captain of Polish team Jakub Błaszczykowski" (number 1 on the illustration). After recognizing the persons in this sentence the place of a person's occurrence is marked using the appropriate tags (number 2 on the illustration). In the stage of adding tags denoting a profession Jakub Błaszczykowski gets the category "sportsman" because he was found on the list with sportsmen (number 3 on the illustration). Thereby he can be included in the data set for categorization.

2.3. Input Data Representation

Considering the number of contexts which can be included in a person's representation it can be named a single-context or multi-context representation. A single-context representation consists of a single context-sentence in which a given person appeared. Multi-context representation of a given person comprises several contexts (sentences) in which the individual appeared. The connection of several contexts of a given person can be restricted to a single document or the whole set of documents. In the first case such a representation was named a document multi-context representation while in the second case the name is a corpus multi-context representation. Three representations of a person mentioned above are shown in fig. 2.

The possibility of categorization according to an accepted person's representation requires an adequate procedure of splitting persons' occurrences from data set used in classification. For a single-context representation single sentences are processed independently. Obtaining all of the contexts in which a given person appeared (whether in any document or the whole corpus) involves the creation of lists whose entries are groups of occurrences. Subsequently training and test set are constructed from these groups.

Feature extraction and classification depend on a selected person's representation. For a single-context representation feature extraction and classification is realized for each person's occurrence separately. On the other hand for a multi-context representation the construction of a feature vector and categorization task is performed simultaneously for all occurrences of a given person.

Polish language is an inflected language. Therefore the main part of the preprocessing of text was obtaining base forms of words which were retrieved using WCRFT - morpho-syntactic tagger for Polish language [9]. At the end of processing stop words were removed. List of stop words for Polish language was acquired from Wikipedia[3].

2.4. Implemented Features

Features from the literature. The approach presented in [5] was implemented in order to compare it with our solution. In the paper the categorization task was carried out for English texts. The used features were context, cluster-based, entity-related and class-specific ones. The similarity between these features and ours will be defined in the next sections of the article. Measures connected with micro-averaging and macro-averaging achieved for our input data set which contained Polish news texts were much smaller. The results obtained for different input data sets presented in [5] are shown in tab. 1.

Tab. 1. Results of method [5] obtained for different input data sets

Input data set	Micro-F1	Macro-F1
English texts	79.60	76.50
Polish texts	48.05	48.52

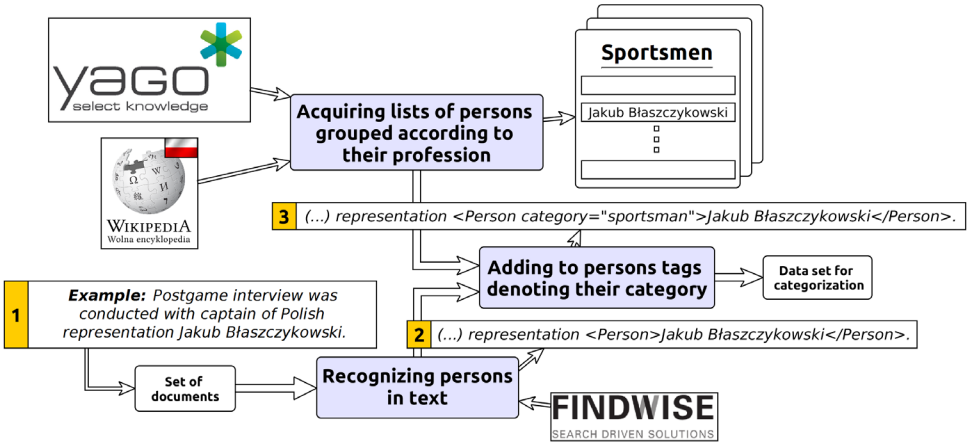

Fig. 1. The method of creating a data set used in the categorization procedure

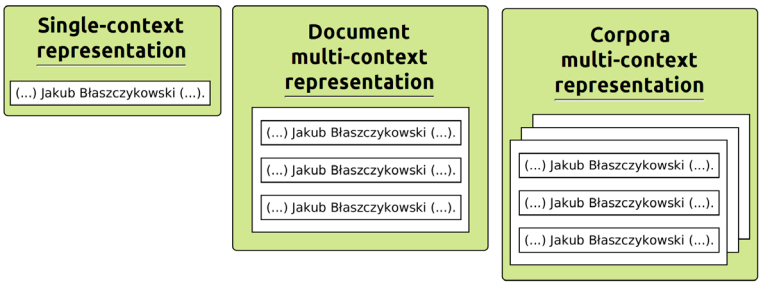

Fig. 2. Representations of a person that are considered during tests

In the following subsections features used in our categorization method will be described.

Context features. Context features use words surrounding person's occurrence from left and right side together with their parts of speech. Our features are created only from content words which in Polish language are verbs, nouns, adjectives, numerals, adverbs and pronouns. Context features from [5] solution did not filter words in terms of their part of speech.

Features of words co-occurring with a category. Next features are features of words co-occurring with a category. To determine them a training set was used to create sets of words which appear only with persons' occurrences from one category. For the surroundings of a given person's occurrence the number of words that belong to each word set is counted. A feature takes the value 1 for a category whose word set contained the highest number of surrounding words. Similar features were class-specific features from [5] but they also included in word sets words that appeared often enough with persons from one category more than the others (threshold 0.8).

Synonym features. The third type of features are synonym features. Words creating context features in different occurrences of persons may be more or less semantically similar. For two synonyms it is not visible while comparing text strings. Therefore identifiers of sets of synonyms were collected for context words using Polish Wordnet([10]). These features are similar to cluster-based features from [5] which in contrast to our implementation used groups of synonyms created using Brown algorithm and TDT5 corpora [11].

Category synonyms features. The last type of devised features are category synonyms features. For each possible category a separate file with her synonyms was created. Synonyms were downloaded from an online dictionary of Polish synonyms[4]. Queries concerned the masculine forms of professions. From these words feminine forms were created and added manually. A feature takes the value 1 for a category when any of the words surrounding a person's occurrence belonged to her/his synonym set.

The number of words taken into consideration from left and right side of person's occurrence was determined separately for features of words co-occurring with a category and category synonyms features. On the other hand the width of the context window for context and synonym features was the same.

An example. Figure 3 presents an example sentence with a person's occurrence and our extracted features. On the top of the illustration the original sentence in Polish language and translated into English are presented. The rectangles on the right side of figure show extracted features translated into English.

For the example presented in fig. 3 the context features are the words surrounding the person's occurrence together with their parts of speech. In this case

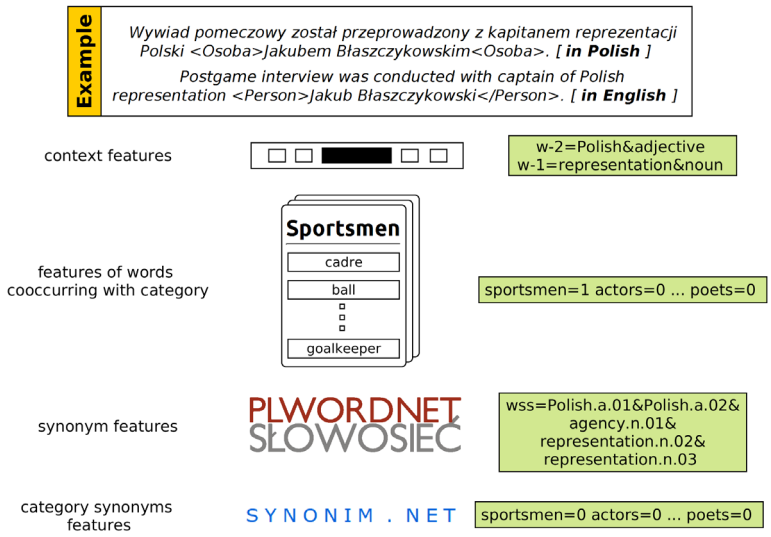

Fig. 3. Our features extracted from an example sentence with a person's occurrence

the words "Polish" (adjective) and "team" (noun) are included. For features of words co-occurring with a category only the feature connected with sportsmen category took the value 1 (sportsmen=1 in figure 3) because the biggest number of words surrounding the considered person was found in a word set connected with sportsmen. Some of the words creating a word set for sportsmen are presented in fig. 3 (these are team, ball and goalkeeper). Synonym features consist of identifiers returned by Polish Wordnet for words "Polish" and "team". Thus the words belonging to the same synonym set were given common feature values. All of the category synonyms features took the value 0 because in the surroundings of person's occurrence did not appear a synonym from any category. The sample words creating a synonyms set for the sportsmen category were: athlete, runner or player.

2.5. Building a Classifier

Building and testing a classifier was based on a 2-fold cross-validation in which every person's occurrence was used for building a model and testing it. As quality measures of the whole classification task micro-averaging and macro-averaging techniques were adopted. Micro-averaging is based on summing the correct classifications of persons from each category and gives us an idea about the overall performance. After computing a confusion table T_i for each from k possible categories the following measures are calculated:

$$Micro\text{-}P = \frac{\sum_{i=1}^{k} TP_{T_i}}{\sum_{i=1}^{k}(TP_{T_i} + FP_{T_i})} \tag{1}$$

$$Micro\text{-}R = \frac{\sum_{i=1}^{k} TP_{T_i}}{\sum_{i=1}^{k}(TP_{T_i} + FN_{T_i})} \tag{2}$$

$$Micro\text{-}F_1 = \frac{2\, Micro\text{-}P\, Micro\text{-}R}{Micro\text{-}P + Micro\text{-}R} \tag{3}$$

A complementary technique to micro-averaging is macro-averaging which calculates an average from the results obtained for each category. In this case single classifications have smaller influence on measures being calculated while precision and recall computed for the whole categories have bigger importance. For each category basing on her confusion table precision (P), recall (R) and F_1 score is computed. Subsequently the following measures are calculated:

$$Macro\text{-}P = \frac{\sum_{i=1}^{k} P}{k} \tag{4}$$

$$Macro\text{-}R = \frac{\sum_{i=1}^{k} R}{k} \tag{5}$$

$$Macro\text{-}F_1 = \frac{2\, Macro\text{-}P\, Macro\text{-}R}{Macro\text{-}P + Macro\text{-}R} \tag{6}$$

3. Results

3.1. Experiments With Different Persons Lists

In our method of creating a data set used in the categorization procedure (Fig. 1) different persons' lists were used. Before merging lists downloaded from YAGO knowledge base with lists from Polish Wikipedia certain tests were executed. Fig. 4 presents a comparison of number of found persons in a set of documents using different persons' lists.

The merged lists in each category let us acquire more unique persons than using both types of lists separately. A significant part of found persons in YAGO lists and merged lists were politicians.

Figure 5 presents a comparison of number of found persons' occurrences in a set of documents using different persons' lists. The number of persons' occurrences varies greatly for the studied lists and simple growth trend is not visible like for found unique persons. in fig. 5 politicians were not mentioned because the number of found persons' occurrences from this category is much bigger than in other categories. For politicians the following numbers were achieved: 16,336 persons' occurence (lists from YAGO), 4,627

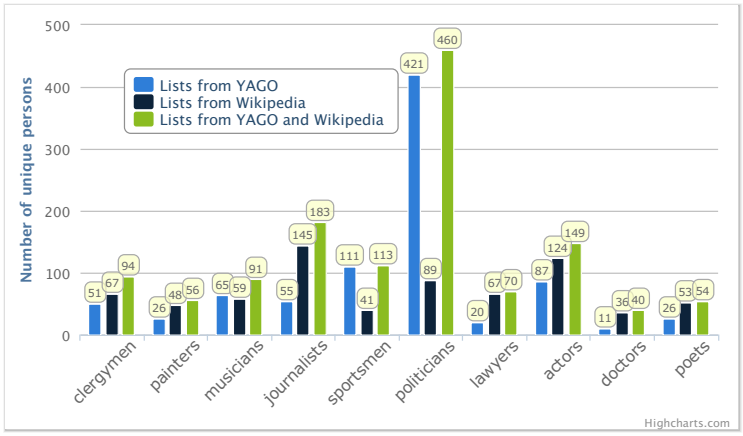

Fig. 4. The number of unique persons found in a set of documents using different persons' lists

(lists from Wikipedia), 16,161 (merged lists). All persons and their occurrences in Figures 4 and 5 were counted only if they were found exactly in one list. After merging the lists from YAGO and Wikipedia the number of persons in every list raised but in the same overlapping of persons in categories increased. The effect of this phenomenon is visible in fig. 5 where in some categories the number of persons' occurrences decreased after merging lists.

3.2. Categorization Results

Tests were carried out with different classifiers. A maximum Entropy classifier available in OpenNLP Maxent library[5] was used and 5 classifiers from Weka machine learning software ([12]). These were NaiveBayes, C4.5, SMO (sequential minimal optimization), RandomTree and BayesNet. The Maxent classifier was used in order to have the same setup as in [5]. The five algorithms from Weka library were chosen basing on the achieved results gathered from initial tests. Most of the applicable classifiers in Weka library[6] were examined. Algorithms with the top five results with using the default parameters were selected. Experiments were performed with all person's representations presented in section 2.3. Tests were conducted on our and chosen solution from literature [5] with default parameters of classifiers. Table 2 presents results for different settings whereas table 3 shows the best outcomes achieved according to tested method and representation of a person.

In both tested solutions the best results were achieved for the multi-context document representation. The highest values of measures for both implementations were about 5-7% bigger than in the tests with single-context representation. The worst results were computed for the corpus multi-context representation. The best outcome for our method was about 3% better than for the solution implemented from the literature and it was attained with a Maximum Entropy classifier. The differences in Micro-F_1 and Macro-F_1 for the single-context and document multi-context representation were small for all used classifiers. However the spread of these measures for corpus multi-context representation was very large which means that the classification in the categories was very unequal.

Table 4 shows the results in each category for best classification outcome (Micro-F_1=51.36%, Macro-F_1=51.34%).

4. Discussion

Tests were carried out according to the devised person's representations described in section 2.3. Although all sentences of a given person (whether in a document or a corpus) were taken into consideration it did not ensure high categorization results. In the final stage of the study one hundred of misclassification cases were examined and 42% of them were subjectively assessed by ourselves as not possible to classify by a human. In the analyzed cases all sentences with a given person in the document were took into account in assigning a category (multi-context document representation). Further improvement of our categorization method seems viable with information about the topic of the document. Additional features indicating a topic could be determined based on all sentences in a given text. Assuming that the content of a document is often related with profession of persons mentioned in it, it can be employed to classify individuals who are not surrounded with words that indicate their profession.

5. Conclusion

During experiments we studied the efficiency of categorization depending on the adopted representation of person. The use of grouped persons' occurrences brought different results. For a document multi-context representation a significant growth of the calculated measures can be noticed in comparison with a single-context representation. On the other hand the use of corpus multi-context representation did not improve the classification measures.

Tests were conducted on our and a chosen solution from the literature with use of six classifiers (Maxent, NaiveBayes, C4.5, SMO, RandomTree i BayesNet). Better results were achieved using our categorization method in comparison with a solution from the literature that was redesigned and adopted to the Po-

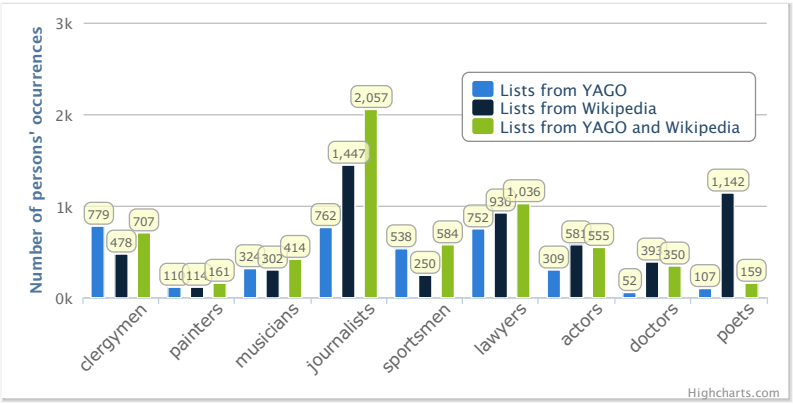

Fig. 5. The number of persons' occurrences found in a set of documents using different persons' lists

Tab. 2. Results achieved for different settings

Method	Representation	Classifier	Micro-F_1	Macro-F_1
literature	single-context	Maxent	**42.39**	**42.88**
		NaiveBayes	36.95	34.84
		C4.5	29.75	27.31
		SMO	34.13	33.50
		RandomTree	24.33	22.72
		BayesNet	36.91	35.64
our	single-context	Maxent	**43.88**	**43.74**
		NaiveBayes	42.09	42.39
		C4.5	41.00	40.35
		SMO	43.36	42.61
		RandomTree	34.22	32.77
		BayesNet	39.65	41.31
literature	multi-context document	Maxent	**48.05**	**48.52**
		NaiveBayes	41.77	39.23
		C4.5	36.86	33.54
		SMO	31.19	28.56
		RandomTree	21.28	20.07
		BayesNet	39.94	37.86
our	multi-context document	Maxent	**51.36**	**51.34**
		NaiveBayes	45.64	43.04
		C4.5	42.71	42.09
		SMO	46.14	45.56
		RandomTree	29.81	28.29
		BayesNet	44.15	43.37
literature	multi-context corpus	Maxent	18.77	31.41
		NaiveBayes	48.33	34.07
		C4.5	39.79	23.08
		SMO	**48.34**	**35.30**
		RandomTree	27.56	14.01
		BayesNet	15.22	6.99
our	multi-context corpus	Maxent	17.02	27.96
		NaiveBayes	45.27	32.00
		C4.5	47.64	34.91
		SMO	29.34	30.25
		RandomTree	15.25	17.24
		BayesNet	**48.13**	**36.30**

lish language. The best outcomes were attained with a Maximum Entropy classifier.

Notes

[1] http://nkjp.pl/
[2] http://findwise.com
[3] http://pl.wikipedia.org/wiki/Wikipedia:Stopwords
[4] Online dictionary of Polish synonyms, http://synonim.net
[5] OpenNLP Maxent classifier, http://maxent.sourceforge.

Tab. 3. Best results achieved for different methods and representations of a person

Method	Representation	Classifier	Micro-F_1 (%)	Macro-F_1 (%)
our	single-context	Maxent	43.88	43.74
literature	single-context	Maxent	42.39	42.88
our	multi-context document	Maxent	51.36	51.34
literature	multi-context document	Maxent	48.05	48.52
our	multi-context corpus	BayesNet	49.78	37.55
literature	multi-context corpus	SMO	48.34	35.30

Tab. 4. Precision, recall and F_1 measure for each category in both iterations of 2-fold cross-validation

Category	Precision	Recall	F_1
clergymen	67.35	84.77	75.06
painters	70.27	32.1	44.07
musicians	41.67	36.23	38.76
journalists	53.17	51.69	52.42
sportsmen	50.24	36.3	42.15
politicians	37.7	42.6	40
lawyers	44.43	57.4	50.09
actors	41.87	37.05	39.31
doctors	77.57	48.82	59.93
poets	67.5	33.75	45

Category	Precision	Recall	F_1
clergymen	68.64	74.37	71.39
painters	93.33	17.5	29.47
musicians	61.07	38.65	47.34
journalists	66.22	68.04	67.12
sportsmen	68.09	54.79	60.72
politicians	43.79	64.2	52.07
lawyers	50.57	52.15	51.35
actors	62.03	53.07	57.2
doctors	59.87	52.22	55.79
poets	55.32	32.91	41.27

net/about.html

[6]Weka classifiers, http://wiki.pentaho.com/display/DATAMINING/Classifiers

AUTHORS

Maciej Pachocki – Warsaw University of Technology, Faculty of Mathematics and Information Science, ul. Koszykowa 75, Warsaw, Poland.

Anna Wróblewska* – Warsaw University of Technology, Faculty of Mathematics and Information Science, ul. Koszykowa 75, Warsaw, Poland, e-mail: a.wroblewska@mini.pw.edu.pl, www: www.ii.pw.edu.pl/~awroblew.

*Corresponding author

REFERENCES

[1] M. Fleischman and E. Hovy, "Fine Grained Classification of Named Entities". In: *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002.

[2] V. Ganti, A. C. König, and R. Vernica, "Entity categorization over large document collections". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2008, 274–282, 10.1145/1401890.1401927.

[3] C. Giuliano, "Fine-Grained Classification of Named Entities Exploiting Latent Semantic Kernels". In: *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, Boulder, Colorado, 2009, 201–209.

[4] A. Ekbal, E. Sourjikova, A. Frank, and S. P. Ponzetto, "Assessing the Challenge of Fine-Grained Named Entity Recognition and Classification". In: *Proceedings of the 2010 Named Entities Workshop*, Uppsala, Sweden, 2010, 93–101.

[5] W. Li, J. Li, Y. Tian, and Z. Sui, "Fine-Grained Classification of Named Entities by Fusing Multi-Features". In: *Proceedings of COLING 2012: Posters*, Mumbai, India, 2012, 693–702.

[6] E. Alfonseca and S. Manandhar, "An Unsupervised Method for General Named Entity Recognition and Automated Concept Discovery". In: *Proceedings of the 1 st International Conference on General WordNet*, Mysore, India, 2002, 34–43.

[7] P. Cimiano and J. Völker, "Towards large-scale, open-domain and ontology-based named entity classification". In: *Proceedings of the International Conference on Recent Advances in Natural Language Processing RANLP'05*, 2005, 166–172.

[8] F. M. Suchanek, G. Kasneci, and G. Weikum, "YAGO: a core of semantic knowledge". In: *Proceedings of the 16th international conference on World Wide Web*, New York, NY, USA, 2007, 697–706, 10.1145/1242572.1242667.

[9] A. Radziszewski. "A Tiered CRF Tagger for Polish". In: R. Bembenik, L. Skonieczny, H. Rybinski, M. Kryszkiewicz, and M. Niezgodka, eds., *Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions*, Studies in Computational Intelligence, 215–230. Springer, Berlin, Heidelberg, 2013.

[10] M. Maziarz, M. Piasecki, and S. Szpakowicz, "Approaching plWordNet 2.0". In: C. Fellbaum and P. Vossen, eds., *Proceedings of 6th International Global Wordnet Conference*, Matsue, Japan, 2012, 189–196, Book: http://www.globalwordnet.org/gwa/proceedings/gwc2012.pdf.

[11] P. F. Brown, V. J. Della Pietra, P. V. deSouza, J. C. Lai, and R. L. Mercer, "Class-Based *n*-gram Models of Natural Language", *Computational Linguistics*, vol. 18, no. 4, 1992, 467–480.

[12] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update", *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, 2009, 10–18, 10.1145/1656274.1656278.