Mining Class Association Rules for Word Sense Disambiguation

Łukasz Kobyliński

Institute of Computer Science, Polish Academy of Sciences,
ul. Ordona 21, 01-237 Warszawa, Poland,
lkobylinski@ipipan.waw.pl

Abstract. In this paper we propose an approach to the task of Word Sense Disambiguation problem that uses Class Association Rules to create an effective and human-understandable rule-based classifier. We present the accuracy of classification of selected polysemous words on an evaluation corpus using the proposed method and compare it to other known approaches. We discuss the advantages and weaknesses of a classifier based on association rules and present ideas for future work on the idea.

1 Introduction

The task of Word Sense Disambiguation (WSD) consists of correlating a given instance of a polysemous word, used in a particular context (sentence, paragraph, etc.), with one of known senses of this word. It is a problem we face every day communicating, as every natural language seems to contain some lexical ambiguity as its characteristic feature. Typical examples of English language words that may convey multiple senses are "bank" (having a meaning related to geographical feature or a financial institution) and "pen" (a place or an instrument for writing). It is thus necessary to resolve such ambiguities each time they appear in spoken or written text to be able to comprehend the text as a whole.

Automatic WSD is an important problem, for which an accurate solution would greatly simplify implementations of other tasks related to computational linguistics, such as machine translation. Whether it can be solved completely is an open question, having in mind that even humans vary in their decisions about the sense of a particular word in context. From a computational point of view this problem translates to the problem of classification: assigning one of known senses (classes) to each of the polysemous words appearing in a text fragment (instances).

The aim of our contribution is twofold: to present the results of a supervised learning approach to the task of WSD, evaluated on a Polish language corpus constrained to one specific domain and to propose a novel method of word sense classification, based on mining Class Association Rules (CARs). The motivation for the latter approach is creating a classifier that may be understood and modified by a human, which is not possible using the classical best-performing machine learning methods (neural networks, Bayes approaches, SVM, etc.).

In the following chapters we first briefly describe work done previously in the field of Word Sense Disambiguation (Chapter 2). Next, we discuss the corpus used to assess the accuracy of the proposed method, which was created by manually annotating Polish language texts (Chapter 3). In Chapter 4 we describe the approach used to represent the context of disambiguated words in the form of a feature vector. In Chapter 5 we present the idea of using Class Association Rules in the task of WSD for classifying word senses. Finally, we show the results of experiments conducted using the proposed method and compared with other, known approaches (Chapter 6) and conclude with a summary of the contribution and ideas for future work (Chapter 7).

2 Word Sense Disambiguation

The idea of performing WSD automatically seems to have emerged in the late 1940s, when also the work on machine translation began. Many approaches have been proposed since then, including AI-based methods (as a part of larger systems intended for full language understanding), knowledge-based methods (using such language resources as thesauri and machine-readable dictionaries to compare the context of a particular word with definitions of each of the senses) and corpus-based methods (learning on the samples provided by an annotated text corpus) [1]. The Lesk's algorithm [2] is a particularly notable approach to WSD, which prompted evolution of knowledge-based methods and to which corpus-based methods are compared still today. In this algorithm, a list of words from each sense definition from the dictionary is created. Disambiguation is accomplished by selecting the sense, for which the overlap between the word list and the words in disambiguated context is the largest.

Recently, machine learning methods have been used extensively for the task of WSD and these may be further divided into supervised, semi-supervised and unsupervised approaches. Supervised learning methods require a text corpus, annotated with information about the correct sense of each or some of the appearing words. Sense annotation consists of associating a sense label (taken from a sense dictionary) with each instance of a polysemous word in the running text. Methods of this type are first trained on a learning corpus, manually annotated by linguists and then evaluated on another corpus, by automatically assigning annotations for ambiguous words. As reported by [3] these methods usually achieve the best results, compared with semi- and unsupervised approaches. Examples of algorithms used include Naive Bayes, kNN and SVM.

Semi-supervised methods usually require only a small "bootstrap sample" of annotations and large corpus of unannotated data. For example in [4] an approach is presented, where co-training and self-training paradigms are used for WSD, attempting to increase the small amount of available training data and tag new, previously unlabeled samples from a dataset.

Finally, unsupervised methods, which use external knowledge sources, such as WordNet or Wikipedia and unsupervised learning approaches, can be used in situations where very little or no training data in the form of annotated corpus

is available. In [5] the authors present a graph-based approach, where WordNet is used as a lexical knowledge base containing hierarchical information about relationships between ambiguous words and other elements of the language.

In the context of Polish language there is very little work done in the field of automatic WSD. One of the first results of WSD for Polish language texts has been presented in [6], where supervised learning methods have been trained and evaluated on a small corpus of 1500 annotated examples, taken from a dictionary of 13 polysemous words. Some experimental results have also been presented in [7], where the classifier comparison environment used also in this contribution has been introduced.

Rule based approaches have been already used in the task of WSD and promising results of experiments have been reported. For example, the performance of several rule-based classifiers (J48, PART, decision table) has been compared in [8]. The authors show that rule-based methods may achieve better results than purely statistical approaches, such as Naive Bayes. The idea of mining association rules in a corpus annotated with word senses has been presented in [9], but for finding correlations between annotations done by different linguists and not for sense classification itself.

3 Evaluation Corpus

We have created a sense-annotated corpus of Polish language texts from the domain of economy. The evaluation corpus has been composed of resources coming from: 1 million subcorpus of the National Corpus of Polish [10], with morphosyntactic annotation and a collection of stock market reports in Polish, collected from the Internet. Details of the corpus may be found in Table 1.

Table 1. Statistics of the evaluation corpus.

Corpus	Number of segments	Number of annotated segments
Subcorpus of NCP	87 816	3 821
Stock market reports	282 366	18 719
Overall	370 182	22 540

We have automatically selected a subcorpus from the National Corpus of Polish by choosing the fragments, which had the greatest number of occurrences of words related to the domain of economy. The words have been collected by hand-picking 5100 multi-word economy-related dictionary entries, names of institutions and agencies, as well as stock names from the Warsaw Stock Exchange. While the resources from NCP subcorpus have already been human-annotated morphosyntactically, the market reports have been not. Therefore, we have used the TaKIPI tagger [11] to add the annotation automatically.

To enable the task of annotating the corpus with sense tags, we have created a dictionary of polysemous lexemes. We have gathered 52 polysemous words from the domain of economy (in Polish) and associated them with a list of possible senses. For each word the senses have been grouped into a few broader senses, to lower the granularity of the dictionary. The experience with word sense disambiguation seems to tell us [12] that most automated methods fail with high granularity of senses and it is not needed in real applications. For example, for the word "rynek" the dictionaries offer no less than 14 different definitions. We have combined these 14 senses into 5 broader senses, which are more intuitive, easier to grasp by human annotators and should result in better classification accuracy using automated methods. The dictionary has been created by a linguist and edited using a simple web-based application, to enable easy synchronization of the definitions between the linguists during the annotation phase. Table 2 presents the words included in the resulting dictionary and Figure 1 shows the histogram of the number of senses per each lexeme. There is an average of 3.62 senses per lexeme in the dictionary.

Table 2. Lexemes in the sense dictionary.

idx	lexeme	idx	lexeme	idx	lexeme	idx	lexeme
1	agent[n]	14	koszt[n]	27	punkt[n]	40	ubezpieczenie[n]
2	akcja[n]	15	linia[n]	28	rachunek[n]	41	udział[n]
3	baza[n]	16	ochrona[n]	29	rynek[n]	42	umowa[n]
4	cena[n]	17	opcja[n]	30	rząd[n]	43	unia[n]
5	dochód[n]	18	pieniądz[n]	31	sąd[n]	44	wartość[n]
6	efekt[n]	19	podatek[n]	32	siła[n]	45	warunek[n]
7	firma[n]	20	podstawa[n]	33	spółka[n]	46	zasada[n]
8	fundusz[n]	21	polityka[n]	34	stan[n]	47	zmiana[n]
9	gospodarka[n]	22	pomoc[n]	35	stopa[n]	48	zysk[n]
10	granica[n]	23	postępowanie[n]	36	stopień[n]	49	czarny[a]
11	inwestycja[n]	24	praca[n]	37	system[n]	50	specjalny[a]
12	jednostka[n]	25	prawo[n]	38	środek[n]	51	wolny[a]
13	kontrola[n]	26	projekt[n]	39	świadczenie[n]	52	złoty[a]

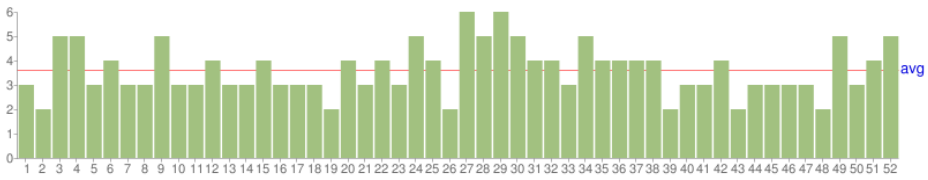

Fig. 1. Histogram of the number of senses per each lexeme in the dictionary. Numbers on the horizontal axis reflect the index of a lexeme from Table 2.

Semantic annotation of the final corpus has been performed by an average number of four linguists. Fragments of the texts (usually paragraphs) have been selected at random from the corpus and assigned to the annotators. Each fragment has been assigned to any of two annotators at the same time. One of the linguists had been assigned the role of a "super-annotator", who has the final decision about a particular annotation in case of a disagreement of two annotators working on a fragment. He or she also had a general overview of the work already done and may have reviewed the statistics of individual annotators' work. The annotation has been performed using a multi-user, web-based application developed for that purpose. The resulting distribution of instances of each of the senses in the annotated corpus is presented on Figure 2.

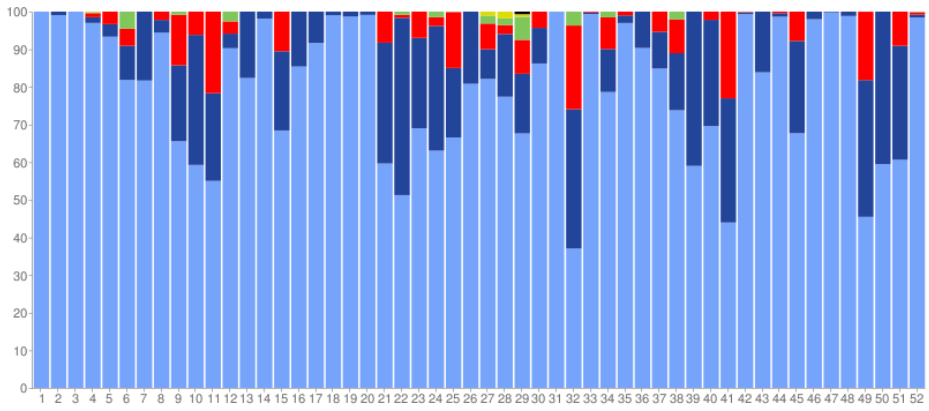

Fig. 2. Percentage of occurrences of each of the senses (per lexeme from the dictionary) in the evaluation corpus. Sense occurrences are sorted in descending order and colors indicate particular senses of the lexemes (e.g. bottom bar – light blue – most frequent sense of a particular lexeme, dark blue – second most frequent, and so on).

It may be noted that some words from the dictionary were not found in the corpus at all (and have been ignored in the evaluation), while the distribution of senses of other words turned out to be highly skewed towards one or two most frequent meanings. This type of distribution is an example of Zipf's Law, which states that frequency of an object is inversely proportional to its rank in the frequency table.

4 Feature Representation

As we are treating the WSD task as a classification problem, we have to be able to represent the textual data (disambiguated words in context) in the form of a fixed-length feature vector. We have chosen to modify for our needs and use the feature generators implemented in the WSD Development Environment [7].

Thematic Feature Generator (TFG) Existence of a word in a window around the disambiguated lexeme with window size: 5–25 and lemmatization: on/off.

Structural Feature Generator 1 (SFG1) Existence of a word on a particular position in a small window relative to the disambiguated lexeme with window size: 1–5 and lemmatization: on/off.

Structural Feature Generator 2 (SFG2) Existence of part-of-speech on a particular position in a small window relative to the disambiguated lexeme with window size: 1–5 and tagset: full or simplified.

Keyword Feature Generator (KFG) Grammatical form of the disambiguated lexeme with tagset: full or simplified.

Examples of feature vectors created by the generators described above are presented on Figure 4.

TFG

płacić	cena	złotówka	moralność	kilogram	przetwarzać
1	0	1	0	1	1

SFG1

obniżyć-2	obniżyć-1	siebie-1	surowiec+1	praca+1
1	0	1	1	0

SFG2

praet-2	subst-1	adj-1	subst+1
1	0	0	1

KFG

subst	sg	pl	dat	acc
1	0	1	0	1

Fig. 3. Examples of feature vectors.

5 Class Association Rules

Association rule mining has been proposed in [13], originally as a method for market basket analysis. This knowledge representation method focuses on showing frequent co-occurrences of attribute values in data. During the last two decades the work on association rules has bloomed, as the technique proved to efficiently provide interesting insights into very large collections of data. Some interesting applications of association rules to real-world problems include: mining medical data to predict heart diseases ([14]), text document categorization ([15]) and image classification ([16]).

Definition Let's assume the database $\mathcal{D}$ contains data described by binary attributes $I = \{I_1, I_2, \ldots, I_m\}$. We call the set I the itemspace. Database $\mathcal{D}$ is a set of transactions, $\mathcal{D} = \{T_1, T_2, \ldots, T_n\}$ and each transaction T is a set of items

(an *itemset*) from the itemspace, $T \subseteq I$. Association rules have the form of an implication over two itemsets, X and Y, where $X, Y \in I$ and $X \cap Y = \emptyset$:

$$R : X \to Y \tag{1}$$

Itemset X is called the rule's *body*, while itemset Y is called the rule's *head*. A rule of the form shown above indicates, that the occurrence of items in the set X often implicates the occurrence of items in the set Y. The strength of this implication may be measured by two basic parameters: support and confidence. The support of a set of items A is determined by the number of transactions in D, which contain A:

$$\text{supp}(A) = |\mathcal{D}_A| \tag{2}$$

A relative support value, calculated in relation to the size of the database, may also be used:

$$\text{supp}_r(A) = \frac{|\mathcal{D}_A|}{|\mathcal{D}|} \tag{3}$$

The relative support of a rule is defined as the support of its body and head, which is the union of itemsets X and Y, divided by the size of the database:

$$\text{supp}_r(X \to Y) = \frac{\text{supp}(X \cup Y)}{|\mathcal{D}|} = \frac{|\mathcal{D}_{X \cup Y}|}{|\mathcal{D}|} \tag{4}$$

The confidence of a rule is a conditional probability that a transaction containing the rule's body also contains its head.

$$\text{conf}(X \to Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}X} = \frac{|\mathcal{D}_{X \cup Y}|}{|\mathcal{D}_X|} \tag{5}$$

We say that an itemset A is *frequent* in database $\mathcal{D}$, when its support in $\mathcal{D}$ is greater than a certain threshold, called *minimum support*, $\text{supp}(A) > minSup$. Similarly, we say that a rule R is *strong* in database $\mathcal{D}$ if its support and confidence are greater than minimum rule support and confidence, $\text{supp}(X \to Y) > minSup$ and $\text{conf}(X \to Y) > minConf$.

Use in classification Association rules used for classification, frequently referred to as Class Association Rules (CARs), are rules constrained to have a class label in its head. Having $I = \{I_1, I_2, \ldots, I_m\}$ (the set of items) and $C = \{c_1, c_2, \ldots, c_k\}$ (the set of class labels), $X \subset I$, $c \in C$, CAR is rule of the following form:

$$CAR : X \to c \tag{6}$$

The first method of building a classifier based on a set of mined association rules, named CBA, has been introduced in [17]. The process is divided into two parts: rule generation (CBA-RG) and building the classifier (CBA-CB). During the rule generation step frequent itemsets (having support greater than a specified *minsup* value) are being found in the data, using the Apriori algorithm [18] to avoid searching the entire feature space. Apriori principle tells us

that no superset of an infrequent itemset can be frequent. The difference in the approach to finding general frequent itemsets for building association rules and the CBA-RG algorithm consists in considering also the category label as an item in the formed itemsets. Next, rules are created from the itemsets, which have a confidence higher than a set minimum value *minconf*.

In the second step of the process the generated rules are sorted according to a precedence relation. This relation is defined as follows:

$$\begin{aligned} r_i \prec r_j \Leftrightarrow\ & [\mathrm{conf}(r_i) > \mathrm{conf}(r_j)] \vee \\ & [\mathrm{conf}(r_i) = \mathrm{conf}(r_j) \wedge \mathrm{sup}(r_i) > \mathrm{sup}(r_j)] \vee \\ & [\mathrm{conf}(r_i) = \mathrm{conf}(r_j) \wedge \mathrm{sup}(r_i) = \mathrm{sup}(r_j) \wedge \\ & r_i \text{ generated earlier than } r_j] \end{aligned} \tag{7}$$

Next, for each of the rules in the sorted order all matching examples from the training set are found and number of correct classifications is noted. Rules, which classify at least one example correctly are added to the final classifier and the matching examples are removed from memory. This step is iterated until no data is available in the current memory.

6 Experimental Results

We have adapted the framework described in [7] to carry out a series of classification experiments using a selection of supervised learning methods and text feature representation approaches. Specifically, we have added the ability to use a CARs-based classifier to be able to compare its effectiveness against other well-known methods.

Each experiment has been conducted using the ten-fold cross-validation methodology to be able to use the evaluation corpus as a source for both training and testing data. At first, we have calculated the Most Frequent Sense (MFS) minimum classification accuracy baseline to be able to relate the achieved results to the characteristics of the available corpus. We have also noted the Inter-Annotator Agreement (ITA) value, which reflects the percentage of annotations, for which two annotators provided the same sense labels and no conflict resolution was necessary. This value is frequently described as a good candidate for an upper bound of classification accuracy, as we cannot expect that the system trained on annotated data will perform better than human linguists, who provided the annotation. Abovementioned statistics are presented in Table 3 and for each of the lexemes from the dictionary on Figure 4.

We have performed experiments of classification of the entire evaluation corpus using both the classical NaiveBayes approach (which proved to perform best among others we have tried: J48, SVM, RandomForest) and the method based on mining Class Association Rules. Classifiers have been built individually for each of the disambiguated words and in each case an attribute selection method has been employed to limit the size of feature vectors to less than 400 attributes.

Table 3. Most Frequent Sense classification baseline, Inter-Annotator Agreement and classification accuracy for individual corpora.

corpus	MFS (%)	ITA (%)	NaiveBayes (%)	CARs (%)
NCP subcorpus	77.65	91.97	87.67	84.14
market reports	94.31	96.82	98.86	97.26
overall	91.06	95.99	96.87	94.28

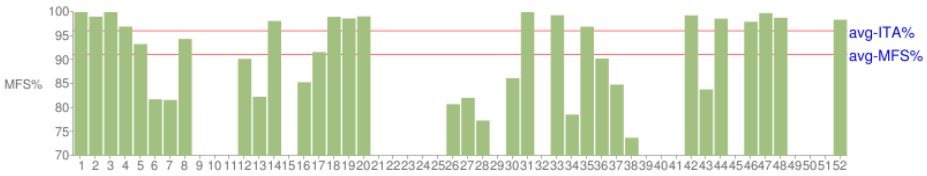

Fig. 4. Most Frequent Sense for each of the disambiguated words and Inter-Annotator Agreement for the entire corpus.

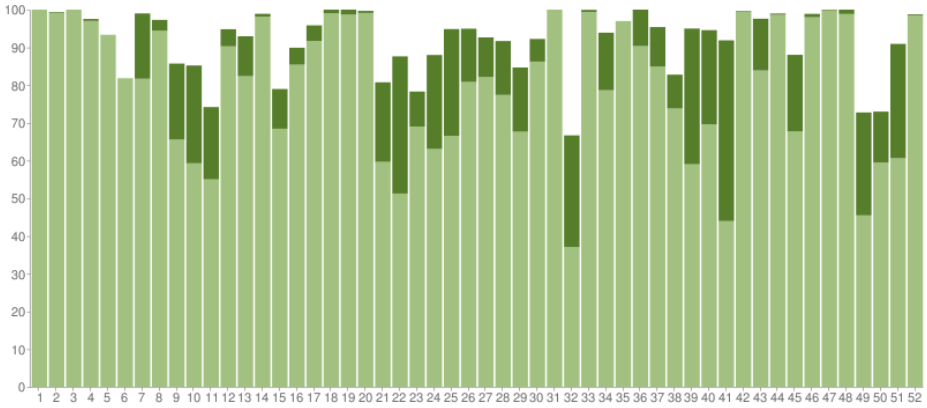

Fig. 5. Accuracy (%) of classification using the NaiveBayes method. Bottom bar (light green): MFS baseline, top bar (dark green): improvement over MFS.

The accuracy of classification using the NaiveBayes method has been presented on Figure 5.

Figure 6 shows the results of classification using the CARs method. As may be seen from the overall accuracy results, shown in Table 3, the rule-based method is slightly less accurate, than the NaiveBayes approach. However, the classifier built using the CBA algorithm may be interpreted by a human expert and a potentially interesting knowledge can be extracted from it, which is not the case for the NaiveBayes method.

As an example, below we present a rule generated by the CBA algorithm. Left-hand-side of the rule consists of attributes generated by particular feature generators. Here, the KFG generator provided an attribute pl_KFG (equal

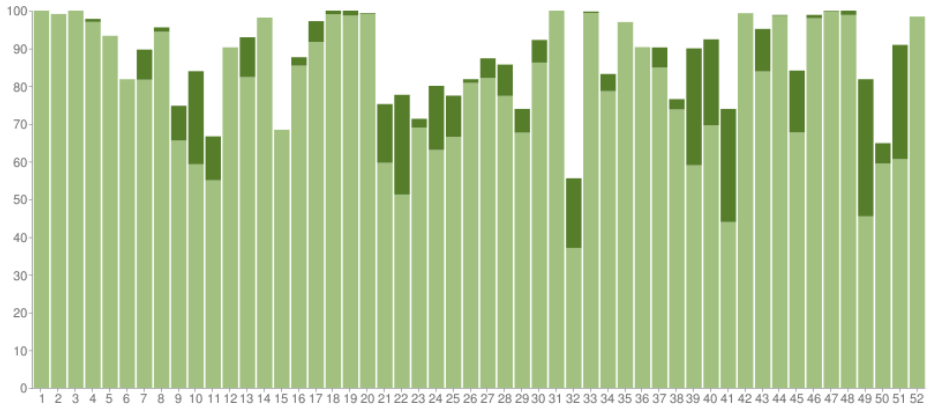

Fig. 6. Accuracy (%) of classification using the CARs method. Bottom bar (light green): MFS baseline, top bar (dark green): improvement over MFS.

to 0), which indicates that the disambiguated word has a singular form. Similarly, noun-1_SFG2=1 noun+1_SFG2=0 attributes indicate that a noun should appear one place before the disambiguated word and no noun one place after the disambiguated word, for the rule to hold. If the rule holds, the selected sense is praca.2.

pl_KFG=0 pos+1_SFG2=0 noun-1_SFG2=1 noun+1_SFG2=0 →
→ SENSE=praca.2 [conf:0.93]

7 Conclusions and Future Work

In this paper we have presented an application of Class Association Rules to the problem of Word Sense Disambiguation of Polish language texts from the domain of economy. We have created a hand-annotated corpus of economy-related textual resources, containing ambiguous lexemes, and used it to train a CARs-based classifier, using the CBA algorithm. Using the standard ten-fold cross-validation methodology we have evaluated the accuracy of the proposed approach and compared it with a well-known NaiveBayes algorithm. Achieved results, while showing the rule-based method to be less accurate than a purely statistical approach are encouraging, because for the cost of slightly lower accuracy we get a classifier that is understandable by human experts and may potentially be manually edited and enhanced.

It remains for future work to test the effectiveness and accuracy of other algorithms for building CARs-based classifiers and also increasing the number of features used to represent the disambiguated word in context.

References

1. Ide, N., Véronis, J.: Word sense disambiguation: The state of the art. Computational Linguistics **24**(1) (1998) 1–40
2. Lesk, M.: Automated sense disambiguation using machine-readable dictionaries: How to tell a pine cone from an ice cream cone. In: Proceedings of the 1986 SIGDOC Conference, Toronto, Canada (June 1986)
3. Pradhan, S., Loper, E., Dligach, D., M.Palmer: Semeval-2007 task-17: English lexical sample srl and all words. In: Proceedings of SemEval-2007. (2007)
4. Mihalcea, R.: Co-training and self-training for word sense disambiguation. In: CoNLL-2004, Poznań, Poland (November 2004)
5. Agirre, E., Soroa, A.: Personalizing pagerank for word sense disambiguation. In: Proceedings of the 12th conference of the European chapter of the Association for Computational Linguistics (EACL-2009). (2009)
6. Baś, D., Broda, B., Piasecki, M.: Towards word sense disambiguation of Polish. In: Proceedings of the International Multiconference on Computer Science and Information Technology. (2008) 73–78
7. Młodzki, R., Przepiórkowski, A.: The WSD development environment. In: Proceedings of the 4th Language and Technology Conference. (2009)
8. Paliouras, G., Karkaletsis, V., Androutsopoulos, I., Spyropoulos, C.D.: Learning rules for large-vocabulary word sense disambiguation: a comparison of various classifiers. In Christodoulakis, D., ed.: Proceedings of the 2nd International Conference on Natural Language Processing (NLP 2000). Volume 1835 of Lecture Notes in Artificial Intelligence., Springer (2000) 383–394
9. Passonneau, R.J., Salleb-Aoussi, A., Bhardwaj, V., Ide, N.: Word sense annotation of polysemous words by multiple annotators. [19]
10. Przepiórkowski, A., Górski, R.L., Łaziński, M., Pęzik, P.: Recent developments in the National Corpus of Polish. [19]
11. Piasecki, M.: Polish tagger TaKIPI: Rule based construction and optimisation. Task Quarterly **11**(1–2) (2007) 151–167
12. Agirre, E., Edmonds, P., eds.: Word Sense Disambiguation: Algorithms and Applications. Springer (2006)
13. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, Washington, D.C., USA (May 1993) 207–216
14. Ordonez, C., Omiecinski, E., Braal, L.d., Santana, C.A., Ezquerra, N., Taboada, J.A., Cooke, D., Krawczynska, E., Garcia, E.V.: Mining constrained association rules to predict heart disease. In: Proceedings of the 2001 IEEE International Conference on Data Mining. ICDM '01, Washington, DC, USA, IEEE Computer Society (2001) 433–440
15. Antonie, M.L., Zaïane, O.R.: Text document categorization by term association. In: Proceedings of the 2002 IEEE International Conference on Data Mining. ICDM '02, Washington, DC, USA, IEEE Computer Society (2002) 19–
16. Kobyliński, Ł., Walczak, K.: Class association rules with occurrence count in image classification. TASK Quarterly **11**(1–2) (2007) 35–45
17. Liu, B., Hsu, W., Ma, Y.: Integrating classification and association rule mining. In: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, New York, USA (August 27–31 1998) 80–86
18. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proceedings of 20th Interntaional Conference on Very Large Data Bases, Santiago, Chile (September 1994) 487–499

19. Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., Tapias, D., eds.: Proceedings of the Seventh International Conference on Language Resources and Evaluation, LREC 2010. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., Tapias, D., eds.: Proceedings of the Seventh International Conference on Language Resources and Evaluation, LREC 2010, Valletta, Malta, ELRA, European Language Resources Association (ELRA) (May 2010)