Three attempts in PolEval 2017 Sentiment Analysis Task

Michal Ptaszynski[*], Fumito Masui[*], Arkadiusz Janz[†], Jan Kocoń[†], Maciej Piasecki[†], Monika Zaśko-Zielińska[‡], Pawel Dybala[§]

[*]Kitami Institute of Technology
Koen-cho 165, Kitami, Japan
{ptaszynski,f-masui}@cs.kitami-it.ac.jp

[†]Wroclaw University of Science and Technology
Wybrzeze Wyspianskiego 27, Wroclaw, Poland
{maciej.piasecki,arkadiusz.janz}@pwr.wroc.pl

[‡]University of Wrocław
Plac Nankiera 15, Wroclaw, Poland
monika.zasko-zielinska@uwr.edu.pl

[§]Jagiellonian University
Gronostajowa 3, Krakow, Poland
pawel.dybala@uj.edu.pl

Abstract

In this paper we present our attempts in the PolEval 2017 Sentiment Analysis Task. The task is not only one of the first challenges in sentiment analysis focused on Polish language, but also represents a novel approach to sentiment analysis, namely, predicting the sentiment not of a sentence, or a document, but of a word or a phrase within the context of a sentence. This makes typical approaches, such as sentiment dictionaries, inapplicable or greatly limited, and forces the development of novel approaches taking advantage of the whole sentence context. The presented and evaluated approaches include a multiclass machine learning approach, a context-based dictionary lookup approach and a Jaccard index-based method with a bag of features.

1. Introduction

Sentiment Analysis (SA) has one of the most popular research fields in recent years. It is focused on detecting whether some contents, such as a sentence, a document, or a tweet, etc., represent positive or negative attitude of the speaker/writer towards the topic of the contents.

Grounds for the field were laid by such research as General Inquirer (Stone et al., 1966), introducing a set of procedures for computer-based general content analysis, or Affective Reasoner (Elliot,), a simplistic keyword-based Sentiment Analysis system applying an affect lexicon (e.g., “happy", or “sad") with intensity modifiers (e.g., “extremely", “somewhat"). Works such as the one by (Turney, 2002) or (Pang et al., 2002) defined the field and directed the main stream on the task of automatic discrimination whether an input is of positive or negative attitude.

Till present day a number of open access datasets for studying SA have been released, including the Amazon product data[1] developed at University of California San Diego (McAuley et al., 2015; He and McAuley, 2016), multiple Customer Review Datasets[2] developed at University of Illinois at Chicago (Hu and Liu, 2004), OpinRank Dataset with reviews from TripAdvisor and Edmunds[3] (Ganesan and Zhai, 2012), the MPQA Opinion Corpus[4] (Wiebe et al., 2005), the Large Movie Review Dataset[5] developed at Stanford University (Maas et al., 2011), or the UMICH SI650 - Sentiment Classification Dataset[6].

Moreover, there have also been SA contests and tasks organized to promote the development of methods for sentiment analysis optimized to certain kinds of datasets, such as the SemEval Sentiment Analysis Tasks[7] [8] [9] [10], or the WASSA-2017 Shared Task on Emotion Intensity[11].

However, a great majority of the available datasets has been collected only in English language, and there has been little done in Sentiment Analysis for other, less-resourced languages, such as Polish. So far, Sentiment Analysis-related resources for Polish include sentiment annotations included in plWordNet 3.0 – Słowosieć 3.0[12] (Zaśko-Zielińska et al., 2015a), Polish sentiment dictionary “Slownik Wydzwieku"[13] (Wawer, 2012), HateSpeech corpus[14], containing two thousand documents manually annotated for various types of offensive language, or an SVM classifier of Polish sentiment[15], trained on Bag-of-Words model (Bartusiak and Kajdanowicz, 2015).

Among these, the present PolEval 2017 dataset and the task it proposes is unique in several ways. Firstly, the task is one of the first challenges in sentiment analysis focused only on Polish language. It also represents a novel approach to sentiment analysis, namely, predicting the sentiment not of a sentence, or a document, but of a word or a phrase within the context of a sentence, which makes typical approaches, such as sentiment dictionaries, inapplicable or greatly limited, since the lack of the word in the dictionary immediately makes the prediction impossible. This forces the development of novel approaches taking advantage of the whole sentence context.

Below we explain the dataset in more detail and present the methods we developed for sentiment analysis for the dataset provided within the task.

2. PolEval 2017 Dataset

The dataset provided in the PolEval Sentiment Analysis Task[16] contains two training sets and one test set.

The training sets contain:

[1]http://jmcauley.ucsd.edu/data/amazon/
[2]https://www.cs.uic.edu/ liub/FBS/sentiment-analysis.html
[3]http://kavita-ganesan.com/entity-ranking-data
[4]http://mpqa.cs.pitt.edu/corpora/mpqa_corpus/
[5]http://ai.stanford.edu/ amaas/data/sentiment/
[6]https://www.kaggle.com/c/si650winter11
[7]https://www.cs.york.ac.uk/semeval-2013/task2/
[8]http://alt.qcri.org/semeval2014/task9/
[9]http://alt.qcri.org/semeval2015/task10/
[10]http://alt.qcri.org/semeval2016/task5/
[11]https://www.aclweb.org/portal/content/wassa-2017-shared-task-emotion-intensity
[12]http://plwordnet.pwr.wroc.pl/wordnet/
[13]http://zil.ipipan.waw.pl/SlownikWydzwieku
[14]http://zil.ipipan.waw.pl/HateSpeech
[15]https://github.com/riomus/polish-sentiment
[16]http://poleval.pl/index.php/tasks/

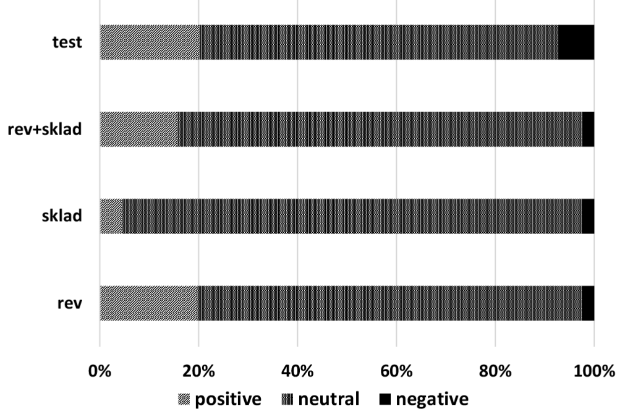

Figure 1: Percentage of sentiment labels for each part of the PolEval 2017 dataset.

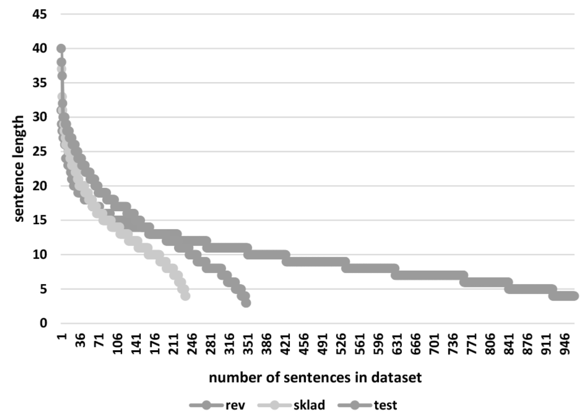

Figure 2: Percentage of sentiment labels for each part of the PolEval 2017 dataset.

- 235 Sentences from the Skladnica[17], a Polish treebank manually annotated for sentiment and dependency information (later called here: SKLAD).
- 965 Reviews of perfumes and clothes with manually annotated for sentiment and automatically annotated dependency information (later called here: REV).

The TEST set contains 350 sentences. Sentences from all datasets are annotated for tokens, POS and dependency information. Additionally, training data was annotated for sentiment. The test set originally did not contain sentiment annotations, which were provided later for result verification. For each sentence the sentiment labels correspond to the sentiment of a whole phrase being a sub-part of dependency tree. As stated in the task description "the goal of the task is to provide the correct sentiment for each subtree (phrase)." An example of annotations provided in the datasets was represented in Table 1. A general overview of the dataset was represented in Table 2. The percentage of labels for each part of the dataset was shown in Figure 1.

As it can be noticed in Figure 1 and in Table 2, the labels were not balanced. Most of the labels were neutral, meaning they did not contain any sentiment-related connotations. Second in number was the positive label, while the negative labels were the most sparse. Due to the imbalance in the number of labels, and sparseness of non-neural labels the conclusions that could be drawn from the results of systems developed for this dataset are limited and should be taken carefully. It could also make optimization of such systems problematic, as the lowest Accuracy achieved with the simplest baseline method for the development of such system is set comparatively high. For example, a simplistic system classifying all of the labels in test dataset as neutral would still obtain 72.64% of Accuracy.

Analysis the length of sentences in the datasets provided additional insights. Although REV dataset contained the highest number of sentences, these were usually the shortest with average sentence length of 9.86 words. The SKLAD dataset contained more longer sentences with average sentence length of 14.26 words per sentence. The TEST dataset contained the most of long sentences with average sentence length of 14.46 words per sentence. When compared with the percentage of labels in sentences, these statistics confirm the insights of previous corpus related studies (Ptaszynski et al., 2014). For example, sentences of more emotional character (product reviews) tend to be shorter than neutral sentences or various general sentences (Skladnica). On the other hand, SKLAD dataset contained longer sentences most probably due to the aim of studying dependency relations, which are better revealed in longer, more sophisticated constructions.

Despite the differences in quality of the datasets, in the methods developed for this research we always applied both training sets together (REV+SKLAD) due to the generally small size of the provided training data.

[17] http://zil.ipipan.waw.pl/Składnica

Table 1: One example of annotations provided in the PolEval 2017 dataset. Sentiment-related annotations in **bold**.

Tokens	Według	mnie	to	**zapach**	**godny**	**polecenia**	.
POS	adjunct	comp	pred	**subj**	**adjunct**	**adjunct**	punct
Depend.	3	1	0	**3**	**6**	**4**	3
Sentiment	0	0	0	**1**	**1**	**1**	0
(English)	According_to	me	this_is_a	**smell**	**worth**	**recommending**	.

3. Multiclass Nearest Neighbours Condominium

The Multiclass Nearest Neighbours Condominium or **MuNNCo** was the first method we developed for the PolEval 2017 Sentiment Analysis task. In the method, at first all available annotations are integrated (in the order: tokens, POS, dependency, sentiment), thus making one sentence element contain all types of information. An example of integration is represented in Table 3. Next, all unique elements were extracted separately from the training and test corpus, and each of such unique elements containing integrated information was considered as one separate class in machine learning. This provided 7729 unique classes. Next, each unique class was assigned all sentences it appeared in. Each assigned sentence contained all annotations except sentiment. Next, a Bag-of-Words (BOW) lan-

Table 2: General overview of the PolEval 2017 dataset.

	rev	(%)	sklad	(%)	rev + sklad	(%)	test	(%)
# all tokens	9510		3351		12861		5047	
# unique tokens	5361		2667		7729		3656	
lexical density	0.564		0.796		0.601		0.724	
# sentences	965		235		1200		350	
class 1	1859	19.55%	151	4.51%	2010	15.63%	1016	20.13%
0	7427	78.10%	3116	92.99%	10543	81.98%	3666	72.64%
-1	224	2.36%	84	2.51%	308	2.39%	365	7.23%

class	Sentences containing class
zapach_subj_3_1	Ten_adjunct_2 zapach_subj_3 uwiódł_conjunct_5 mnie_obj_3 i_coord_0 uzależnił_conjunct_5 !_punct_5 Wedlug_adjunct_3 mnie_comp_1 to_pred_0 zapach_subj_3 godny_adjunct_6 polecenia_adjunct_4 ._punct_3
	⋮
zapach_subj_7_-1	Jak_pre_coord_7 dla_adjunct_7 mnie_comp_2 Chanel_pd_7 No_adjunct_6 5_adjunct_4 to_pred_0 przereklamowany_adjunct_9 zapach_subj_7 ._punct_7
⋮	⋮

Figure 3: Example of dictionary entry.

guage model was build for all classes, and further applied to a machine learning classifier. This way the model could classify new input without knowing the correct answer, and depending on the classifier, could predict the closest class within the range of its possibilities. This time due to time constraints we decided to apply the k-Nearest Neighbour (kNN) classifier. The kNN takes as input k-closest training samples with assigned classes and classifies input sample to a class by a majority vote. Here, we used k=1 setting in which the input sample is assigned to the class of the first nearest neighbour.

Finally, after the classification, in the evaluation, to check the performance of the method, we verified whether the sentiment value of the predicted class was correct. We did not look at whether the whole class was correct, since the goal of the task was to predict only the sentiment value of the word or phrase.

3.1. Results

Multiclass classification with several thousand classes, typically performed on one-vs-all approach, also implemented here, usually tends to have high level of confusability between classes (Gupta et al., 2014). Therefore we did not expect high results. As expected, the results were equal to the “all 0" scenario described in section 2., and reached 72.64%. When analysed in detail, although the labels were predicted in various manners, the reduction to only sentiment labels revealed that all predicted labels were in fact “0." This was due to the fact that the 0-sentiment labels were in the most frequent group within the corpus, while 1 and -1 appeared rarely. Since, the most frequent labels also appeared in the largest number of sentences, the highest prediction confidence was calculated for those most frequent labels. It is possible that the method could be improved in a number of ways, which we plan to apply in the near future. The possible improvements include:

- Clustering classes not by all annotations, but only tokens, or tokens with POS, etc.;
- Applying in classification only tokens, or tokens with POS, etc.;
- Applying different classifier;
- Applying dataset balancing techniques (oversampling, etc.);

4. Normalized Contextual Levenshtein Similarity

The second developed method was Normalized Contextual Levenshtein Similarity, or **NoCoLeS**. In the method we applied the same dictionary as in MuNNCo, but used a different approach. NoCoLeS applied a basic dictionary lookup method to scan through the new input sentence. The part of dictionary entry containing linguistic annotations (token_pos_dependency) was used for lookup. However, there could be a case when there are two dictionary entries with the same linguistic annotations, but with different sentiment values due to the surrounding context. To deal with such situations we used the sentences included for each dictionary entry. Each such available sentence was compared to the input sentence by applying a Linear Space Refinement algorithm (Ukkonen, 1985; Myers, 1986) to calculate sentence similarity[18], which was in turn based on Levenshtein distance (Levenshtein, 1966). The sentence with the highest similarity score was selected as the context closest to input and the sentiment value of the dictionary entry containing this sentence was selected as the predicted sentiment value.

Table 3: Class-based accuracies for the NoCoLeS method.

	class-based result		
	positive	**neutral**	**negative**
Accuracy	7.09%	97.98%	1.10%

4.1. Results

The results of the NoCoLeS method were also not much higher above the “all 0" scenario, and reached 72.68%. We also summarized the results per each sentiment class (positive, negative, neutral). The results were represented in Table 3. For non-neutral classes the Accuracy did not surpass 10%. The low result could be explained by the fact that only 658 dictionary entries from the training method appeared in the test dataset, which contained 3741 unique entries (17.6%).

To calculate statistical significance of results we used Chi square statistics due to the fact that data comprised of three classes. The two-tailed P value was less than 0.0001, which by conventional criteria, is considered an extremely statistically significant difference. Numbers of expected and observed classes applied in calculation of statistical significance of results were represented in Table 5.

Although the results were not high, the method was more successful than the machine learning approach and could be improved further. Below we list up the possible ways of improvement:

- Dictionary lookup search in input could be performed not only on the fully annotated entries (token_ POS_dependency), but also on less annotated (token_pos, or token_dependency), or in case of a new word not present in the dictionary, even on partial linguistic annotations (pos_dependency, or even pos, or dependency alone).
- Sentence similarity could be calculated not for sentences with all annotations, but as above for partial annotations.
- At this time sentence similarity is calculated for each letter of the entry. This way Levenshtein distance could be different for changing between various types of annotations (subj > obj). Therefore, the distance

[18]http://search.cpan.org/~mlehmann/String-Similarity-1.04/

could be calculated not for each letter, but for for each type of annotation (e.g., one Levenshtein point per type of annotation).

- At this time all linguistic annotations (token, pos, dependency) are considered equal, meaning that their weight is equal. It could be possible that some type of information is more important for sentiment classification. The weights (e.g., 1, 2, 3) could be selected in a preliminary experiment on training data with the prediction performed on each of annotation separately. The type of annotation with the highest results when used alone would have higher weight.

5. Jaccard Index on Big Bag of Features

Finally, we present yet another method for an instance-based recognition, but this time taking into a account a broader, amorphous context representation. In this method, every subtree in a syntactic dependency tree is represented by a bag of features, where the features are combining the words, their polarities and dependency relations linking the words in this subtree. The bag for a sub-tree consists of features generated for the root node of this sub-tree, and features generated for every child linked to this root node. To reduce a negative impact of word variability, we replaced the words with their lemmas and Parts of Speech, using WCRFT[19] (Radziszewski, 2013), a tagger for Polish.

To compute a sentiment for every sub-tree in a syntactic dependency tree, we start the computation from the leaves, using depth-first procedure. For each part of a tree we generate the features according to proposed feature-templates:

- $Lemma\_PoS_{root}$: $Polarity_{root}$, sentiment value for a root node, represented by a lemma and assigned Part of Speech,
- $Lemma\_PoS_{child}$: $Polarity_{child}$, polarity of a child linked to the root node, represented by a lemma and assigned Part of Speech,
- $Lemma\_PoS_{child}$: $Deprel_{root:child}$, dependency relation linking the child of a given lemma and Part of Speech,
- $Lemma\_PoS_{child}$: $Label_{child}$, label computed for a sub-tree linked to the child of a given lemma and Part of Speech.

In case were our tagger could not determine a correct base form and Part of Speech for a word, we simply use this word in original form. Then we determine a sentiment polarity for a node using the polarity acquired from the training set. If the word did not appear in the training set, we acquire a sentiment from plWordNet 3.1 emo[20] (Zaśko-Zielińska et al., 2015b), a very large thesaurus of Polish (a lexical semantic network) partially described with emotive annotations. plWordNet includes ≈190,500 lemmas described by ≈282,500 *lexical units*[21] grouped into ≈217,000 *synsets* and described by a lexico-semantic relations (e.g. hypernymy, meronymy, cause, gradation etc.). Synsets include lexical units sharing sets of relations and because of this considered to be synonymous, see (Maziarz et al., 2013). Almost 65,000 selected noun and adjective lexical units have been annotated by:

[19] `http://ws.clarin-pl.eu/tager.shtml`

[20] `http://plwordnet.pwr.wroc.pl/wordnet/`

[21] A lexical unit is a triple: a lemma, Part of Speech and a sense id.

Table 4: Class-based accuracies for the Jaccard Index-based method.

	class-based result		
	positive	**neutral**	**negative**
Accuracy	34.12%	91.16%	10.11%

- *sentiment polarity*, 5 grades scale: strong & weak vs negative & positive, plus neutral, lexical units that express both positive and negative sentiment polarity and are described as *ambiguous* (amb)
- *basic emotions* (Plutchik, 1980): joy, trust, fear, surprise, sadness, disgust, anger, anticipation,
- and *fundamental human values*: *użyteczność* 'utility', *dobro drugiego człowieka* 'another's good', *prawda* 'truth', *wiedza* 'knowledge', *piękno* 'beauty', *szczęście* 'happiness' (all of them positive), *nieużyteczność* 'futility', *krzywda* 'harm', *niewiedza* 'ignorance', *błąd* 'error', *brzydota* 'ugliness', *nieszczęście* 'misfortune' (all negative) (Puzynina, 1992)

In order to use emotive annotations from plWordNet 3.1 emo, we needed first to perform Word Sense Disambiguation and identify senses for words appearing in texts. The disambiguation was performed using WoSeDon (Kędzia et al., 2015), a weakly supervised word sense disambiguation system for Polish, based on personalized Page Rank algorithm (Agirre et al., 2014) (word-2-word configuration) run on plWordNet.

Then we generated features for higher parts of syntactic dependency tree, including a sentiment value computed for lower parts of the tree. To compute a final label for a sub-tree, we refer to training set again. Given the bag-of-features generated for a sub-tree with a root node, represented by a lemma and Part of Speech, we look for the sub-trees in the training set, linked to the same lemma and Part of Speech. Then we compute Jaccard Index between their bags of features, and select a label of a sub-tree, with the greatest similarity value.

We also evaluated another variation of this method, with an extended similarity context, taking into account features generated not only for individual sub-trees, but also for full sentences. This approach resulted with a slightly higher accuracy, in comparison to the previous one with a narrow similarity context.

5.1. Results

The resulted accuracy for the initial version of the method reached 73.7%. The accuracy for the improved method reached 73.9%. When it comes to the class based-results, although the accuracy for the neutral class was slightly lower then for the NoCoLeS method (91.16%), the accuracies for positive and negative classes were much higher, reaching 34.12% and 10.11%, respectively. Class-based results for the Jaccard Index-based method were represented in Table 4.

As for statistical significance, the two-tailed P value was less than 0.0001. By conventional criteria, such difference is considered to be extremely statistically significant. Numbers of expected and observed classes applied in calculation of statistical significance of results were represented in Table 5.

Table 5: Numbers of expected and observed classes used in calculation of Chi squared test for statistical significance.

Category	Expected	Observed	
		NoCoLeS	Jaccard
positive	1016	144	665
neutral	3666	4878	4274
negative	365	25	108

6. Conclusions

We presented our initial attempts in the sentiment analysis task for Polish language performed on a novel PolEval 2017 Sentiment Analysis dataset. The task assumed predicting the sentiment of previously unknown phrases within the context of sentences, thus making typical approaches, like keyword spotting or simple dictionary lookup in sentiment dictionaries, greatly limited. Among the three presented approaches, a multiclass machine learning approach (MuNNCO) achieved the lowest scores (72.64%). A context-based dictionary lookup approach (NoCoLeS) achieved better, though similarly low scores (72.68%). Finally, a Jaccard index-based method with a bag of features achieved the highest scores (73.9%). There is a room for improvement for all three methods, which we plan to investigate in the near future. These include applying classifiers other than proposed, or a stepwise matching on either full or limited label set.

7. References

Agirre, Eneko, Oier Lopez de Lacalle, and Aitor Soroa, 2014. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–84.

Bartusiak, Roman and Tomasz Kajdanowicz, 2015. Sentiment analysis based on collaborative data for polish language. In *International Conference on Cooperative Design, Visualization and Engineering*. Springer.

Elliot, C. The affective reasoner: A process model of emotions in a multi-agent system. 1992. *Northwestern University Institute for the Learning Sciences: Northwestern, IL*, 48.

Ganesan, Kavita and ChengXiang Zhai, 2012. Opinion-based entity ranking. *Information retrieval*, 15(2):116–150.

Gupta, Maya R., Samy Bengio, and Jason Weston, 2014. Training highly multi-class linear classifiers. *Journal Machine Learning Research (JMLR)*:1461–1492.

He, Ruining and Julian McAuley, 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee.

Hu, Minqing and Bing Liu, 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

Kędzia, Paweł, Maciej Piasecki, and Marlena Orlińska, 2015. Word sense disambiguation based on large scale Polish CLARIN heterogeneous lexical resources. *Cognitive Studies / Études cognitives*, (15):269–292.

Levenshtein, Vladimir I, 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10.

Maas, Andrew L., Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts, 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA: Association for Computational Linguistics.

Maziarz, Marek, Maciej Piasecki, and Stanisław Szpakowicz, 2013. The chicken-and-egg problem in wordnet design: synonymy, synsets and constitutive relations. *Language Resources and Evaluation*, 47(3):769–796.

McAuley, Julian, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel, 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM.

Myers, Eugene W, 1986. An o (nd) difference algorithm and its variations. *Algorithmica*, 1(1):251–266.

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan, 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics.

Plutchik, Robert, 1980. *EMOTION: A Psychoevolutionary Synthesis*. Harper & Row.

Ptaszynski, Michal, Rafal Rzepka, Kenji Araki, and Yoshio Momouchi, 2014. Automatically annotating a five-billion-word corpus of japanese blogs for sentiment and affect analysis. *Computer Speech & Language*, 28(1):38–55.

Puzynina, Jadwiga, 1992. *Język wartości [The language of values]*. Scientific Publishers PWN.

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

Stone, Philip J, Dexter C Dunphy, Marshall S Smith, Daniel M Ogilvie, et al., 1966. The general inquirer: a computer approach to content analysis.

Turney, Peter D, 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics.

Ukkonen, Esko, 1985. Algorithms for approximate string matching. *Information and control*, 64(1-3):100–118.

Wawer, Aleksander, 2012. Extracting emotive patterns for languages with rich morphology. *International Journal of Computational Linguistics and Applications*, 3(1):11–24.

Wiebe, Janyce, Theresa Wilson, and Claire Cardie, 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Zaśko-Zielińska, Monika, Maciej Piasecki, and Stan Szpakowicz, 2015a. A large wordnet-based sentiment lexicon for polish. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*.

Zaśko-Zielińska, Monika, Maciej Piasecki, and Stan Szpakowicz, 2015b. A large wordnet-based sentiment lexicon for Polish. In R. Mitkov, G. Angelova, and K. Boncheva (eds.), *Proc. of the Conf. Recent Advances in Natural Language Processing – RANLP'2015*.