J. Hlaváčová (Ed.): ITAT 2017 Proceedings, pp. 176–180
CEUR Workshop Proceedings Vol. 1885, ISSN 1613-0073, © 2017 T. Hercig, P. Krejzl, B. Hourová, J. Steinberger, L. Lenc

# Detecting Stance in Czech News Commentaries

Tomáš Hercig[1,2], Peter Krejzl[1], Barbora Hourová[1], Josef Steinberger[1], Ladislav Lenc[1,2]

[1] Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic
[2] NTIS—New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Technická 8, 306 14 Plzeň, Czech Republic
nlp.kiv.zcu.cz
{tigi,krejzl,hourova,steinberger,llenc}@kiv.zcu.cz

*Abstract:* This paper describes our system created to detect stance in online discussions. The goal is to identify whether the author of a comment is in favor of the given target or against. We created an extended corpus of Czech news comments and evaluated a support vector machines classifier, a maximum entropy classifier, and a convolutional neural network.

*Keywords:* Stance Detection, Opinion Mining, Sentiment Analysis

## 1 Introduction

Stance detection has been defined as automatically determining from text whether the author is in favor of the given target entity (person, movement, topic, proposition, etc.), against it, or whether neither inference is likely.

Stance detection can be viewed as a subtask of opinion mining, similar to sentiment analysis. In sentiment analysis, systems determine whether a piece of text is positive, negative, or neutral. However, in stance detection, systems predict author's favorability towards a given target, which may not even be explicitly mentioned in the text. Moreover, the text may express positive opinion about an entity contained in the text, but one can also infer that the author is against the defined target (an entity or a topic). It has been found difficult to infer stance towards a target of interest from tweets that express opinion towards another entity[10].

There are many applications which could benefit from the automatic stance detection, including information retrieval, textual entailment, or text summarization, in particular opinion summarization.

We created an extended corpus for stance detection for Czech and evaluate standard top-performing models on this dataset and report the results.

The rest of this paper is organized as follows. We summarise the releated work in Section 2). The creation of the used corpus is covered by Section 3. Our approach is described in Section 4. The convolutional neural network architecture is depicted in Section 5. Evaluation and results discussion is in Section 6 and future work is proposed in Section 7.

## 2 Related Work

The SemEval-2016 task **Detecting Stance in Tweets**[1] [10] had two subtasks: supervised and weakly supervised stance identification.

The goal of both subtasks was to classify tweets into three classes (*In favor*, *Against*, and *Neither*). The performance was measured by the macro-averaged F1-score of two classes (*In favor* and *Against*). This evaluation measure does not disregard the *Neither* class, because falsely labelling the *Neither* class as *In favor* or *Against* still affects the scores. We use the same evaluation metric ($F1\_2$), accuracy, and the F1-score of all classes ($F1\_3$).

The supervised task (subtask A) tested stance towards five targets: *Atheism*, *Climate Change is a Real Concern*, *Feminist Movement*, *Hillary Clinton*, and *Legalization of Abortion*. Participants were provided with 2814 labeled training tweets for the five targets.

A detailed distribution of stances for each target is given in Table 1. The distribution is not uniform and there is always a preference towards a certain stance (e.g., 63% tweets about *Atheism* are labeled as *Against*). The distribution reflects the real-world scenario, in which a majority of people tend to take a similar stance. It also depends on the source of the data. For example, in the case of *Legalization of Abortion*, we can assume that the distribution will be significantly different in religious communities than in atheistic communities.

For the weakly supervised task (subtask B), there were no labeled training data but participants could use a large number of tweets related to the single target: *Donald Trump*.

The best results for subtask A were achieved by an advanced baseline using SVM classifier with unigrams, bigrams, and trigrams along with character n-grams (2, 3, 4, and 5-gram) as features.

Wei et al. [12] present the best result for subtask B and close second team in subtask A of the SemEval stance detection task. They used a convolutional neural network (CNN) designed according to Kim [4]. It utilizes the same kernel widths and numbers of filters as proposed by Kim. Pre-trained word2vec embeddings are used for initialization of the embedding layer. The main difference from

[1] http://alt.qcri.org/semeval2016/task6/

Table 1: Statistics of the SemEval-2016 task corpora in terms of the number of tweets and stance labels.

| Target Entity | Total | *In favor* | *Against* | *Neither* |
|---|---|---|---|---|
| Atheism | 733 | 124 (17%) | 464 (63%) | 145 (20%) |
| Climate Change is Concern | 564 | 335 (59%) | 26 (5%) | 203 (36%) |
| Feminist Movement | 949 | 268 (28%) | 511 (54%) | 170 (18%) |
| Hillary Clinton | 934 | 157 (17%) | 533 (57%) | 244 (26%) |
| Legalization of Abortion | 883 | 151 (17%) | 523 (59%) | 209 (24%) |
| All | 4,063 | 1,035 (25%) | 2,057 (51%) | 971 (24%) |

Table 2: Statistics of the Czech corpora in terms of the number of news comments and stance labels.

| Target Entity | Total | *In favor* | *Against* | *Neither* |
|---|---|---|---|---|
| "Miloš Zeman" – Czech president | 2,638 | 691 (26%) | 1,263 (48%) | 684 (26%) |
| "Smoking Ban in Restaurants" – Gold | 1,388 | 272 (20%) | 485 (35%) | 631 (45%) |
| "Smoking Ban in Restaurants" – All | 2,785 | 744 (27%) | 1,280 (46%) | 761 (27%) |

Kim's network is the used voting scheme. During each training epoch, several iterations are selected to predict the test set. At the end of each epoch, the majority voting scheme is applied to determine the label for each sentence. This is done over a specified number of epochs and finally the same voting is applied to the results of each epoch. The train and test data are separated according to the stance targets.

The initial research on Czech data has been done in [7]. They collected 1,460 comments from a Czech news server[2] related to two topics – Czech president – *"Miloš Zeman"* (181 *In favor*, 165 *Against*, and 301 *Neither*) and *"Smoking Ban in Restaurants"* (168 *In favor*, 252 *Against*, and 393 *Neither*).

The results with maximum entropy classifier were *"Miloš Zeman"* $F1\_2$[3] = 0.435, $F1\_3$[4] = 0.52 and *"Smoking Ban in Restaurants"* $F1\_2$[3] = 0.456, $F1\_3$[4] = 0.54.

## 3 Dataset

We extended the dataset from [7], nearly quadrupling its size. The detailed annotation procedure was described in master thesis [3] in Czech. The whole corpus was annotated by three native speakers. The distribution of stances for each target is given in Table 2.

The target entity "Miloš Zeman" part of the dataset was annotated by one annotator and then 302 comments were also labeled by a second annotator to measure inter-annotator agreement. The target entity "Smoking Ban in Restaurants" part of the dataset was independently annotated by two annotators. To resolve conflicts a third annotator was used and then the majority voting scheme was applied to the gold label selection. The inter-annotator agreement (Cohen's $\kappa$) was calculated between two annotators on 2,203 comments. The final $\kappa$ is 0.579 for "Miloš Zeman" (2,638 comments) and 0.423 for "Smoking Ban in Restaurants" (2,785 comments).

The inter-annotator agreement for the target "Smoking Ban in Restaurants" was quite low, thus we selected a subset of the "Smoking Ban in Restaurants" part of dataset, where the original two annotators assigned the same label as the gold dataset (1,388 comments).

The corpus is available for research purposes at `http://nlp.kiv.zcu.cz/research/sentiment#stance`.

## 4 The Approach Overview

We evaluate common supervised classifiers, namely maximum entropy classifier and support vector machines (SVM) classifiers from Brainy[6]. We also experimented with top-performing models for sentiment analysis and stance detection in particular convolutional neural network. The models were trained separately for each target entity.

### 4.1 Preprocessing

The same preprocessing has been done for all datasets. We use UDPipe [11] with Czech Universal Dependencies 1.2 models for tokenization, POS tagging and lemmatization. Stemming has been done by the HPS stemmer [2]. Preliminary experiments have shown that lower-casing the data achieves slightly better results, thus all the experiments are performed with lower-cased data.

### 4.2 Features

We selected features commonly used in similar natural language processing tasks e.g. sentiment analysis. The following baseline features were used:

[2] `www.idnes.cz`
[3] F1 – (*In favor*/*Against*)
[4] F1 – (*In favor*/*Against*/*Neither*)

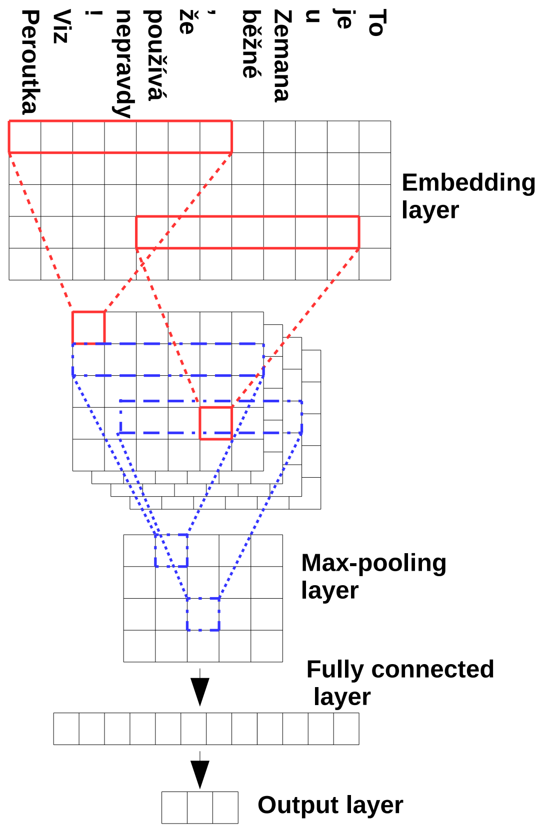


Figure 1: Neural network architecture.

**Character *n*-gram** – Separate binary feature for each character *n*-gram in the text. We do it separately for different orders $n \in \{3,5,7\}$.[5]

**Bag of words** – Word occurrences in the text.

**Bag of adverbs** – Bag of adverbs from the text.

**Bag of adjectives** – Bag of adjectives from the text.

**Negative emoticons** – We used a list of negative emoticons[6] specific to the news commentaries source. The feature captures the presence of an emoticon within the text.

**Word shape** – We assign words into one of 24 classes[7] similar to the function specified in [1].

We experimented with additional features such as *n*-grams, text length, etc. but using these features did not lead to better results. Bag of words, adjectives and adverbs use the word lemma or stem. We report results for various feature combinations and perform an ablation study of the best feature set.

[5]Note that words e.g. emoticon ":-)" would be separated by spaces during tokenization resulting in ": - )".

[6] `":-(", ";-(", ":-/", "8-o", ";-€", ";-O", "Rv"`

[7]We use edu.stanford.nlp.process.WordShapeClassifier [9] with the WORDSHAPECHRIS1 setting.

## 5 Convolutional Neural Network

The architecture of the proposed CNN is depicted in Figure 1. We use similar architecture to the one proposed in [8]. The input layer of the network receives a sequence of word indices from a dictionary. The input vector must be of a fixed length. We solve this issue by padding the input sequence to the maximum text length occurring in the train data denoted $M$. A special "PADDING" token is used for this purpose. The embedding layer maps the word indices to the real-valued embedding vectors of length $L$. The convolutional layer consists of $N_C$ kernels containing $k \times 1$ units and uses rectified linear unit (ReLU) activation function. The convolutional layer is followed by a max-pooling layer and dropout for regularization. The max-pooling layer takes maxima from patches of size $(M-k+1) \times 1$. The output of the max-pooling layer is fed into a fully-connected layer. Follows the output layer with 3 neurons which corresponds to the number of classes. It has softmax activation function.

In our experimental setup we use the embedding dimensionality $L = 300$ and $N_C = 40$ convolutional kernels with $5 \times 1$ units. The penultimate fully-connected layer contains 256 neurons. We train the network using adaptive moment estimation optimization algorithm [5] and cross-entropy is used as the loss function.

## 6 Results

We used 20-fold cross-validation for models evaluation to compensate the small size of dataset and to prevent overfitting.

For all experiments we report the macro-averaged F1-score of two classes $F1\_2$ (*In favor* and *Against*) – the official metric for the SemEval-2016 stance detection task[10], accuracy, and the macro-averaged F1-score of all three classes ($F1\_3$).

Table 3 shows results for each dataset. CNN-1 is described in Section 5 and CNN-2 is the architecture proposed in [4]. We achieved the best results on average with the maximum entropy classifier with the ***feature set*** consisting of lemma unigrams, word shape, bag of adjectives, bag of adverbs, and character *n*-grams ($n \in \{3,5,7\}$). We further performed ablation study of this combination of features. In Table 3 the bold numbers denote five best results for given column and in the ablation study they denote features with no gain in the given column (i.e. feature sets with no loss).

Both CNNs achieved good results, CNN-2 was slightly better, this is not surprising as it was designed for sentiment analysis while CNN-1 was previously used for document classification. Surprisingly stem worked better than lemma as the word input for both neural networks. The ablation study shows that word shape, bag of adjectives, and bag of adverbs features present little to no information gain for the classifier, thus these features should be discarded or

Table 3: Results on Czech stance detection datasets in %. We report accuracy (*Acc*), the macro-averaged F1-score of two classes ($F1\_2$) and the macro-averaged F1-score of all three classes ($F1\_3$). *Feature set* consists of lemma unigrams, word shape, bag of adjectives, bag of adverbs, and character *n*-grams ($n \in \{3,5,7\}$). The bold numbers denote five best results for given column and in the ablation study they denote features with no gain in the given column (i.e. feature sets with no loss).

| Classifier | Features | Zeman | | | Smoking All | | | Smoking Gold | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $F1\_3$ | $F1\_2$ | *Acc* | $F1\_3$ | $F1\_2$ | *Acc* | $F1\_3$ | $F1\_2$ | *Acc* |
| SVM | Random Class | 32.7 | 34.6 | 33.4 | 32.4 | 34.4 | 33.0 | 31.2 | 27.2 | 32.2 |
| SVM | Majority Class | 21.6 | 32.4 | 47.9 | 21.0 | 31.5 | 46.0 | 20.8 | 0.0 | 45.5 |
| CNN-1 | lemma | 48.6 | 52.1 | 51.9 | 51.4 | 54.2 | 54.2 | 61.2 | 55.6 | **65.1** |
| CNN-1 | stemm | **50.7** | 55.3 | **54.5** | 51.7 | 54.6 | 54.5 | 60.6 | 54.8 | 64.8 |
| CNN-2 | lemma | 48.3 | 51.7 | 51.3 | 51.8 | 54.9 | 54.5 | 61.2 | 55.9 | 64.8 |
| CNN-2 | stemm | **51.3** | **55.7** | **54.9** | **52.1** | 54.9 | 54.6 | **61.7** | 56.4 | **65.5** |
| MaxEnt | lemma | 47.7 | 51.8 | 50.2 | 48.8 | 52.3 | 50.9 | 58.1 | 52.2 | 61.6 |
| SVM | lemma | 46.7 | 52.0 | 50.7 | 50.4 | 55.3 | 53.8 | 60.1 | 54.5 | 63.5 |
| MaxEnt | stem | 47.2 | 50.9 | 49.5 | 49.5 | 52.5 | 51.8 | 58.3 | 52.2 | 62.2 |
| SVM | stem | 48.3 | 52.8 | 51.8 | 51.5 | 55.3 | 54.2 | 57.3 | 52.4 | 60.6 |
| MaxEnt | char. *n*-gram 3,5,7 | 50.4 | **55.7** | 53.7 | 50.3 | 54.9 | 53.1 | **61.6** | **56.8** | 65.0 |
| SVM | char. *n*-gram 3,5,7 | 47.4 | 53.4 | 52.2 | 51.3 | **57.2** | **54.9** | 57.6 | 53.2 | 60.8 |
| MaxEnt | shape | 45.0 | 50.2 | 48.4 | 45.7 | 50.2 | 47.9 | 53.9 | 48.6 | 57.0 |
| SVM | shape | 45.5 | 50.3 | 49.7 | 48.1 | 52.0 | 50.6 | 56.5 | 50.8 | 60.7 |
| MaxEnt | feature set | **50.6** | **56.0** | **53.9** | **51.9** | **55.8** | **54.7** | **62.6** | **57.5** | **66.5** |
| SVM | feature set | 47.9 | 54.3 | 52.7 | **52.6** | **58.2** | **56.0** | 59.8 | 55.3 | 62.9 |
| MaxEnt | feature set + emoticons | **50.5** | **56.0** | **53.9** | 51.6 | 55.7 | 54.2 | **62.7** | **57.7** | **66.4** |
| SVM | feature set + emoticons | 47.3 | 53.3 | 51.9 | **52.3** | **58.1** | **55.6** | 61.0 | **56.8** | 63.5 |
| MaxEnt | feature set + emoticons + stem | **50.7** | **56.0** | **53.9** | **51.9** | 55.5 | 54.5 | **62.6** | **57.6** | **66.3** |
| SVM | feature set + emoticons + stem | 47.7 | 53.5 | 52.2 | 51.6 | **57.4** | **55.0** | 60.6 | 55.6 | 64.1 |
| MaxEnt | feature set - shape | **50.8** | **56.0** | **54.0** | 51.6 | **56.1** | 54.4 | **63.0** | **58.3** | **66.5** |
| MaxEnt | feature set - bag of adj. | **50.7** | **56.1** | **54.0** | 51.8 | 55.4 | 54.4 | **62.7** | **57.7** | 66.4 |
| MaxEnt | feature set - bag of adv. | **50.9** | **56.4** | **54.3** | 51.8 | 55.4 | 54.6 | 62.6 | 57.4 | **66.5** |
| MaxEnt | feature set - lemma | 50.2 | 55.6 | 53.6 | 50.8 | 55.1 | 53.6 | 62.4 | 57.3 | 66.2 |
| MaxEnt | feature set - char. *n*-gram 3,5,7 | 46.9 | 51.7 | 49.9 | 48.6 | 52.3 | 50.9 | 58.1 | 52.3 | 61.8 |

readjusted to better capture the stance in comments. However, the selected feature combination still performed reasonably well.

The best results for the target "Miloš Zeman" were achieved by CNN-2 in terms of accuracy and $F1\_3$, $F1\_2$ was the highest for maximum entropy classifier with lemma unigrams, word shape, bag of adjectives, and character *n*-grams. The entity "Smoking Ban in Restaurants" was best assessed by SVM with the selected feature set for all data and by maximum entropy classifier with the same feature set for the gold dataset.

Character *n*-grams alone present a strong baseline for this task.

## 7 Conclusion

The paper describes our system created to detect stance in online discussions. We evaluated top-performing models used for sentiment analysis and stance detection. We conducted feature ablation and concluded that more features still need to be readjusted for this task.

The used features are very common in natural language processing, however even in the SemEval-2016 stance detection task, the best results were achieved by commonly used features. This suggests that stance detection is still in its infancy and more gain can be expected in the future as researchers better understand this new task.

In future work, we plan to extend the dataset to other domains, include more target entities and comments, which will let us draw stronger conclusions and move the task closer to the industrial expectations. Given that there are vast amounts of news comments related to highly discussed topics, we will study stance summarization which should aim at identifying the most important arguments. Another interesting experiment would be supplementing the dataset with sentiment annotation.

## Acknowledgments

This publication was supported by the project LO1506 of the Czech Ministry of Education, Youth and Sports under the program NPU I., by Grant No. SGS-2016-018 Data and Software Engineering for Advanced Applications and by project MediaGist, EU's FP7 People Programme (Marie Curie Actions), no. 630786.

## References

[1] Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing*, pages 194–201. Association for Computational Linguistics, 1997.

[2] Tomáš Brychcín and Miloslav Konopík. Hps: High precision stemmer. *Information Processing & Management*, 51(1):68–91, 2015.

[3] Barbora Hourová. Automatic detection of argumentation. Master's thesis, University of West Bohemia, Faculty of Applied Sciences, 2017.

[4] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/D14-1181`.

[5] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[6] Michal Konkol. Brainy: A machine learning library. In Leszek Rutkowski, Marcin Korytkowski, Rafal Scherer, Ryszard Tadeusiewicz, Lotfi Zadeh, and Jacek Zurada, editors, *Artificial Intelligence and Soft Computing*, volume 8468 of *Lecture Notes in Computer Science*, pages 490–499. Springer International Publishing, 2014. ISBN 978-3-319-07175-6.

[7] Peter Krejzl, Barbora Hourová, and Josef Steinberger. Stance detection in online discussions. In Mária Bieliková and Ivan Srba, editors, *WIKT & DaZ 2016 11th Workshop on Intelligent and Knowledge Oriented Technologies 35th Conference on Data and Knowledge*, pages 211–214. Vydatel'stvo STU, Vazovova 5, Bratislava, Slovakia, November 2016. ISBN 978-80-227-4619-9.

[8] Ladislav Lenc and Pavel Král. Deep neural networks for czech multi-label document classification. *CoRR*, abs/1701.03849, 2017. URL `http://arxiv.org/abs/1701.03849`.

[9] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014. URL `http://www.aclweb.org/anthology/P/P14/P14-5010`.

[10] Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California, June 2016. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/S16-1003`.

[11] Milan Straka, Jan Hajič, and Jana Straková. UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Paris, France, May 2016. European Language Resources Association (ELRA). ISBN 978-2-9517408-9-1.

[12] Wan Wei, Xiao Zhang, Xuqin Liu, Wei Chen, and Tengjiao Wang. pkudblab at semeval-2016 task 6 : A specific convolutional neural network system for effective stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 384–388, San Diego, California, June 2016. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/S16-1062`.