Tohoku at SemEval-2016 Task 6: Feature-based Model versus Convolutional Neural Network for Stance Detection

Yuki Igarashi
Tohoku University
yuki.i@dc.tohoku.ac.jp

Hiroya Komatsu
Tohoku University
h-komatsu@ecei.tohoku.ac.jp

Sosuke Kobayashi
Tohoku University
sosuke.k@ecei.tohoku.ac.jp

Naoaki Okazaki
Tohoku University
okazaki@ecei.tohoku.ac.jp

Kentaro Inui
Tohoku University
inui@ecei.tohoku.ac.jp

Abstract

In this paper, we compare feature-based and Neural Network-based approaches on the supervised stance classification task for tweets in SemEval-2016 Task 6 Subtask A (Mohammad et al., 2016). In the feature-based approach, we use external resources such as lexicons and crawled texts. The Neural Network based approach employs Convolutional Neural Network (CNN). Our results show that the feature-based model outperformed the CNN model on the test data although the CNN model was better than the feature-based model in the cross validation on the training data.

1 Introduction

To solve supervised short text classification tasks, there are two major approaches; feature-based and Neural Network based approaches. In traditional feature-based approaches, we extract various features from a text. The features are usually constructed from n-grams (e.g., bigrams) of the texts and external resources such as lexicons and unlabeled corpora.

In Neural Network based approaches, a number of models for text classifications exist; for example, Feed-Forward Neural Network model using an average of embeddings of target word sequences as the input layer (Iyyer et al., 2015), Recursive Neural Network (Socher et al., 2011; Socher et al., 2013), and Convolutional Neural Network (CNN) (Johnson and Zhang, 2015; dos Santos and Gatti, 2014; Kim, 2014).

In this paper, we compare feature-based and Neural Network based approaches on the supervised stance classification task for tweets, SemEval-2016 Task 6 Subtask A (Mohammad et al., 2016). The feature-based approach classifies tweets using logistic regression model. The features are extracted using external knowledge such as SentiWordNet (Esuli and Sebastiani, 2006) and a collection of crawled tweets, in addition to unigrams or bigrams in the target tweet. For the Neural Network approach, we implement CNN based on Kim (2014). As the input embeddings, we use word embeddings trained by Continuous Bag-Of-Words (CBOW) model (Mikolov et al., 2013) on Wikipedia articles.

The experimental results show that the CNN based approach performed the best in the cross validation on the training data. However the tendency was opposite on the test data probably because the CNN model overfitted to the training data. In contrast, the feature-based approach was more robust, leveraging the external knowledge.

2 Datasets

We use the dataset of the SemEval-2016 Task 6 Subtask A, which is a supervised tweet classification task for five topics. There are three stances to classify; `NONE`, `FAVOR`, `AGAINST`. Table 1 shows the topics and distributions of the training data.

Proceedings of SemEval-2016, pages 401–407,
San Diego, California, June 16-17, 2016.

Topic	FAVOR	AGAINST	NONE
Atheism	92	304	117
Climate Change is a Real Concern	212	15	168
Feminist Movement	210	328	126
Hillary Clinton	112	361	166
Legalization of Abortion	105	334	164

Table 1: Distributions of the labels for each topics in the training data.

Topic	Query	# tweets
Atheism	"atheism"	24124
Climate Change is a Real Concern	"climate", "climate change"	22703
Feminist Movement	"feminist", "feaminism", "feminist movement", "gender equality"	131677
Hillary Clinton	"hillary", "clinton", "hillary clinton"	980080
Legalization of Abortion	"abortion"	54846

Table 2: Crawled tweets used for HighPMI Features. This table shows search queries and the number of tweets we collected for each topic.

Topic	Keywords
Atheism	"atheism"
Climate Change is a Real Concern	"climate", "change"
Feminist Movement	"feminist", "feminism"
Hillary Clinton	"hillary", "clinton"
Legalization of Abortion	"abortion"

Table 3: Seed keywords for each topic for TargetSentiment and HighPMI features.

To classify tweets into their stances, we consider two configurations: *Three-way Polarity Classifier* which detects three stance labels at once, and a combination of *Topic Classifier* and *Two-way Polarity Classifier*. Topic Classifier judges the relevance of a tweet to the topic, in other words, whether a stance label is NONE or not (FAVOR/AGAINST). Two-way Polarity Classifier then labels FAVOR or AGAINST for tweets that were not judged as NONE by the Topic Classifier.

3 Feature-Based Approach

3.1 Preprocessing Tweets

We remove reply and mention expressions (@UserName) in tweets to prevent overfitting, and keep flags indicating whether tweets contain them or not. We also remove hashtags based on the following rules to prevent overfitting.

Rule 1. Hashtags embedded in the sentence with capitals or digits at non-inital letters
e.g., #WeLoveJapan, #Pray4all

Rule 2. Hashtags at the end of a tweet
e.g., #SemST, #2014, #LylicTweet

Rule 1 removes hashtags that are too long or unpopular. **Rule 2** removes hashtags that do not contain a stance. Remaining hashtags such as #hillary and #god may provide important features to detect the stances.

We also expand shortened forms such as "I'm" and "can't" based on simple rules. Finally, we obtain part-of-speech (POS) tags and dependency trees of tweets by using Stanford CoreNLP[1].

3.2 Features

Reply (R): If a tweet has a flag that indicates a reply or mention expression, we generate R=is_reply or R=is_mention as a feature. This feature may be effective because a reply or mention may provides a clue for detecting a stance.

BagOfWords (BoW): For detecting stances, words in a tweet are very informative. We include all unigrams of lemmas in a tweet as features. (e.g. BoW=think, BoW=not)

BagOfDependencies (BoD): Dependency relations such as adjectival modifier and negation are important for detecting stances. We include all dependency relations in a tweet as features. (e.g. BoD=hate=>i, BoD=like=>not)

BagOfPOSTag (BoP): We also extract features from POS tags. For example, if a tweet contains several interjections, the user probably has a negative opinion to the topic. We include all unigrams of POS tags in a tweet as features. (e.g. BoP=NOUN, BoP=UH)

SentiWordNet (SWN): Content words in a tweet may express some sentiment, which indicates stances and emotions of the user. We use SentiWordNet (Esuli and Sebastiani, 2006) for introducing sentiment of a word. It assigns positive/negative/objective scores to each word. In sentiment classification task, Pang et al. (2002) introduce SentiWordNet features. Following their work, we include sentiment polarity features for nouns, verbs,

[1] http://stanfordnlp.github.io/CoreNLP/

Classifier	Atheism	ClimateChange is a Real concern	Feminist Movement	Hillary Clinton	Legalization Of Abortion	ALL topics
3-way Polarity	0.5314	0.5144	0.5735	0.5273	0.5277	0.6083
Topic + 2-way Polarity	**0.5327**	**0.5248**	**0.5860**	**0.5502**	**0.5290**	**0.6188**

Table 4: Comparison of 3-way Polarity Classifier with Topic + 2-way Polarity Classifier on 10-fold cross validation using the feature-based approach. The scores were measured in a macro average of micro-F1 scores of `FAVOR` and `AGAINST` for each topic and all topics.

Feature sets	Atheism	ClimateChange is a Real concern	Feminist Movement	Hillary Clinton	Legalization Of Abortion	ALL topics
ALL	0.5327	0.5248	0.5860	0.5502	0.5290	0.6188
- R	0.5373	0.5248	0.5776	0.5539	0.5349	0.6180
- BoW	0.5440	0.5248	**0.5936**	0.4927	0.5341	0.6185
- BoD	**0.5672**	0.5097	0.5895	**0.5561**	**0.5746**	**0.6276**
- BoP	0.5525	0.5248	0.5785	0.5527	0.5034	0.6169
- SWN	0.5357	0.5168	0.5760	0.5475	0.5308	0.6162
- SWS	0.5316	0.5248	0.5783	0.5520	0.5342	0.6174
- TS	0.5327	0.5248	0.5882	0.5502	0.5349	0.6200
- P	0.5360	0.5248	0.5834	0.5520	0.5406	0.6204
Best	0.5672	0.5642	0.5208	0.5883	0.5208	0.6297

Table 5: Ablation Test of Topic + 2-way Polarity Classifier on 10-fold cross validation using the feature-based approach. The scores were measured in a macro average of micro-F1 scores of `FAVOR` and `AGAINST` for each topic and all topics.

adjectives and adverbs in the tweet based on the following rules.

1. For a given word, look up the top item in SentiWordNet and obtain a negative and positive score of the word.
2. If the negative score is equal to the positive score, no features are generated.
3. If the negative score is larger than the positive score, generate a negative polarity feature, otherwise generate a positive polarity feature. (e.g. `SWN=love=>p, SWN=hate=>n`)

SentiWordSubject (SWS): This feature focuses on sentiment expressed by subjective pronouns such as "I" or "we", which may indicate emotions or stances of the user of a tweet. We obtain a sentiment polarity from the word modifying a subjective pronoun in a tweet, and include it as a feature. A sentiment polarity is obtained by SentiWordNet using the same rules for SWN features. (e.g. `SWS=I=love=>p, SWS=We=hate=>n`)

TargetSentiment (TS): We also consider sentiment or emotion for the topics. Jiang et al. (2011) add words modifying target words as features. Similarly, we extract words modifying target words in a tweet, and include sentiment polarity features using the same rules in SWN features.

We calculate similarities between words and seed keywords using word embeddings. If the similarity is higher than 0.7, we use it as the target word. Table 3 shows the seed keywords for each topic.

For example, given a tweet "We hate feminist", we extract "hate" that modifies the target word "feminist". Then we get a feature `TS=n` using the same rules in SWN features. (e.g. `TS=p, TS=n`)

HighPMI (P): We crawled tweets containing target words, and collected words cooccuring with seed keywords (Table 3) in all crawled tweets for each topic. Table 2 shows query words and the number of crawled tweets for each topic. Then we calculate Point-wise Mutual Information (PMI) for all words. If the word in a tweet is in top 300 of the PMI, we generate a feature. This feature detects a tweet containing words related to the topic. This feature may be effective to classify whether `NONE` or not. (e.g. `P=humanist, P=meninist`)

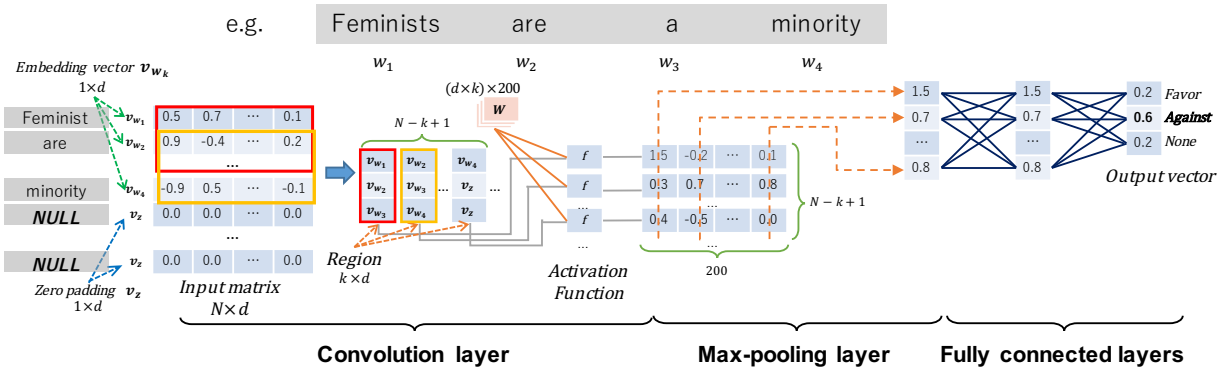

Figure 1: Overview of our CNN model.

3.3 Evaluation

3.3.1 Experimental Setups

We used L2 logistic regression as the classification algorithm, and measured the classification performance on 10-fold cross validation using the Classias package (Okazaki, 2009). We evaluated each model by a macro average of micro-F1 scores of FAVOR and AGAINST for each topic and all topics.

3.3.2 Comparison of Classifier Combinations

We compared Three-way Polarity Classifier with the combination of Topic Classifier and Two-way Polarity Classifier. Table 3.1 shows the performances of these two classifier configurations. We confirmed that the combination of Topic Classifier and Two-way Polarity Classifier outperformed Three-way Polarity Classifier. Therefore, we used the combination of Topic Classifier and Two-way Polarity Classifier hereafter.

3.3.3 Ablation Test

Through this experiment, we explore the contribution of individual features explained in Section 3.2. Table 5 shows the results of ablation tests. These results show that SWN features were the most effective to classify the stances. Sentiment of the tweet is one of the keys for stance classification. In contrast, BoD features degraded the classifier. We experimented further ablation tests with the feature set except for degraded features in the ablation test. These experiments revealed the best feature sets {**BoW, BoP, R, SWN, P**} (denoted 'Best' in Table 5).

4 CNN Based Approach

4.1 Method Overview

In recent years, Convolutional Neural Network (CNN) models have achieved remarkable results in various fields of research, such as computer vision and speech recognition. In the field of natural language processing, CNN models are also used for text classification tasks (Johnson and Zhang, 2015; dos Santos and Gatti, 2014), sentiment analysis (Kim, 2014), etc.

Following Kim (2014), we constructed CNN models to detect stances, as shown in Figure 1. They consist of one convolution layer with one max-pooling layer, and a three-layered feedforward network with softmax at the end to predict a distribution over classes. The convolution layer has 200 kernel windows whose sizes are $k \times d$, where k is the number of words in a window and d is the dimension size of the word embeddings. We denote an input tweet s as a sequence of words $\mathbf{w_1}, \mathbf{w_2}, ..., \mathbf{w_n}$, and their embeddings $\mathbf{v_{w_1}}, \mathbf{v_{w_2}}, ..., \mathbf{v_{w_n}}$. We use Chainer[2] for creating neural networks. To create a fixed-size input matrix for the implementation on Chainer, we added zero-padding vectors into the end of a sentence so that each input matrix will be $N \times d$ matrix, where N is the upper bound of the length of a sentence.

As we mentioned in Section 2, we consider both **Three-way Polarity Classifier** and a combination of **Topic Classifier** and **Two-way Polarity Classifier**. We also try to find out the best hyper parameter k and activation functions.

[2]http://chainer.org/

Classifier	Activation Functions		Atheism	Climate Change is a Real Concern	Feminist Movement	Hillary Clinton	Legalization of Abortion	Total
3-way	sig		**0.6770**	0.4118	0.5958	0.5681	0.4814	0.6664
	relu		0.6039	0.5186	0.6015	0.6098	0.5421	0.6647
Topic + 2-way	sig	sig	0.6751	0.4080	0.5969	0.6061	0.5528	0.6381
	relu	relu	0.5736	0.5710	0.5990	0.6391	0.5455	0.6365
	sig	relu	0.5826	**0.5774**	0.5974	**0.6402**	0.5425	**0.6713**
	relu	sig	0.6688	0.4017	**0.6005**	0.6090	**0.5612**	0.6398

Table 6: Comparison of 3-way Polarity Classifier with Topic + 2-way Polarity Classifier on 10-fold cross validation using CNN based approach. The scores were measured in a macro average of micro-F1 scores of FAVOR and AGAINST for each topic and all topics.

Kernel Size (k)	Atheism	Climate Change is a Real Concern	Feminist Movement	Hillary Clinton	Legalization of Abortion	ALL topics
2	**0.6390**	0.5743	0.6189	**0.6490**	**0.5611**	**0.6831**
3	0.5826	**0.5774**	0.5974	0.6402	0.5425	0.6713
4	0.5746	0.5746	**0.6276**	0.6354	0.5398	0.6755

Table 7: Tuning of window size per word k on 10-fold cross validation using CNN based approach. The scores were measured in a macro average of micro-F1 scores of FAVOR and AGAINST for each topic and all topics.

4.2 Experimental Setups

We trained 300 dimensional word embeddings using Word2Vec[3] with Wikipedia articles[4] (3950598 articles in total)[5]. We set N to 100, which exceeds the maximum length of all tweets. We use $(300 \times k) \times 200$ matrix as W and three fully connected layers that consist of 200-50-3 units (Three-way Polarity Classifier) or 200-50-2 units (Topic Classifier or Two-way Polarity Classifier).

We measured the performance on 10-fold cross validation. We evaluated each model by a macro average of micro-F1 scores of FAVOR and AGAINST for each topic and all topics.

4.3 Evaluation

4.3.1 Comparison of Classifier Combinations

We compared Three-way Polarity Classifier with the combination of Topic Classifier and Two-way Polarity Classifier with $k = 3$. We tried using all possible combinations of sigmoid and relu functions in the CNN models.

Table 6 shows the performances of the classifiers. The table indicates that the combination of Topic Classifier and Two-way Polarity Classifier outperformed Three-way Polarity Classifier. We confirmed that the combination of these classifier has been found effective for not only the feature-based approach, but also for the CNN-based approach.

We also achieved the best score when we use sigmoid function for Topic Classifier and relu for Two-way Polarity Classifier.

4.3.2 Tuning of Window Size k

We searched for the best value of the hyperparameter k with the highest model in Section 4.3.1. Table 7 shows that the model obtained the highest score with window size $k = 2$. The results show that bigram is appropriate for stance detection on the training data.

4.3.3 Visualization of the CNN Model

In this section, we visualize the CNN model that achieved the highest score in Section 4.3.2. To visualize the CNN model, we define a *region score* as the number of dimensions that are selected by the max-pooling layer per region. Figure 2 shows a heat map reflecting the region score.

The figure provides several observations.

- Topic related words such as *movement* received a high score in both Topic classifier and Two-way Polarity Classifier. This shows that each

[3]https://code.google.com/archive/p/word2vec/

[4]https://dumps.wikimedia.org/enwiki/20151201/enwiki-20151201-pages-articles.xml.bz2

[5]We used the following options: `-size 300 -window 5 -sample 1e-4 -negative 5 -hs 0 -cbow 1 -iter 3`

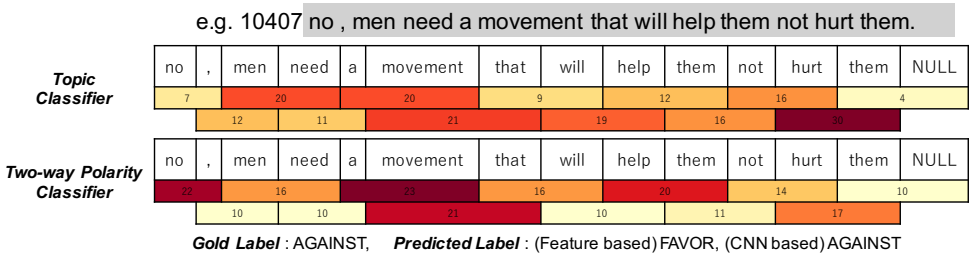

Figure 2: Visualization of the CNN model. A bigram region in deep color receives high *region score*, which indicates that the convolution layer highly focuses on the region.

Method	Train	Test					
	Total	Atheism	Climate Change is a Real Concern	Feminist Movement	Hillary Clinton	Legalization of Abortion	ALL topic
Feature-Best	0.6297	**0.5973**	0.3891	**0.5487**	**0.5360**	**0.5796**	**0.6426**
CNN-Best (Submission)	**0.6831**	0.5890	**0.3951**	0.5241	0.3981	0.3775	0.6221
Majority Baseline	0.5411	0.4210	0.4212	0.3910	0.3683	0.4030	0.6522

Table 8: Comparison of Feature-based Model and CNN Model on test data. The scores were measured in a macro average of micro-F1 scores of FAVOR and AGAINST for each topic and all topics.

CNN model automatically detects the topic words.

- Nouns, verbs and adjectives that appear in SentiWordNet received a higher score in both classifiers. In addition, their scores have some associations with cooccurrence with the topic word.

- Negation words such as *not* and *can't* received high scores in Polarity Classifier, but they received less scores in Topic Classifier.

5 Overall Results

We compared feature-based models with CNN models and the majority baseline in the test data. The feature-based models used Topic + Two-way Polarity Classifiers and the best feature sets mentioned in Section 3.3.3. The CNN models used Topic + Two-way Polarity Classifiers and the best hyperparameters mentioned in Section 4.3. The majority baseline labeled the test data as the stance that was most prevalent in the training data.

Table 8 shows a macro average of micro-F1 scores of FAVOR and AGAINST for two models in the test data and the cross validation results. As a comparison, we also show the majority baseline in Table 8.

We found that the feature-based model outperformed the CNN model in the test data, although the CNN model was better in the cross validation on the training data. We think that the feature-based model was more robust, including broad external knowledge such as SentiWordNet and crawled tweets. In contrast, the CNN model obtained a lower score on the test data than on the cross validation.

6 Conclusion

We compared the feature-based and the CNN based approaches on SemEval-2016 Task 6 Subtask A. The CNN based approach performed the best in the cross validation on the training data although the feature-based approach outperformed the CNN model on the test data. We also visualized the CNN model to reveal what was focused on. We found that the CNN model automatically detected the topic words and effective words to detect the stances.

Acknowledgments

We acknowledge the support of the Step-QI School program, Department of Information and Intelligent Systems, Tohoku University.

References

Cicero dos Santos and Maira Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th In-*

ternational Conference on Computational Linguistics: Technical Papers, pages 69–78.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation*, pages 417–422.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1681–1691.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 151–160.

Rie Johnson and Tong Zhang. 2015. Effective use of word order for text categorization with convolutional neural networks. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 103–112.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 Task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16.

Naoaki Okazaki. 2009. Classias: a collection of machine-learning algorithms for classification.

Bo Pang, 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*, EMNLP '02, pages 79–86.

Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.