Determining Rumour Veracity and Support for Rumours

Ivor Leon Matijašić, Mihaela Bakšić, Karina Ewa Szubert

University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
{Ivor-Leon.Matijasic,Mihaela.Baksic,Karina.Szubert}@fer.hr

Abstract

The tasks of rumour veracity and stance classification have captured the interest of researchers with the rise of social media and its consumption for relevant information spreading. Traditional approaches of feature engineering have been challenged by neural language models. This paper provides a broad overview of the performance of different feature setups across several classifier models and showcases that this approach is still relevant for the stance classification task. Furthermore, we also investigate how including the simplest form of discussion environment, the discussion source tweet, influences the performance of a simple neural model classifier.

1. Introduction

In light of the rise of social media and online discussion platforms as one of the main sources of information, the issue of the veracity of claims made and detection of rumours becomes substantial. A rumour can be defined as a "circulating story of questionable veracity, which is apparently credible but hard to verify, and produces sufficient skepticism and/or anxiety to motivate finding out the actual truth" (Derczynski et al., 2017). Considering the vast amount of claims generated on online social networks, the task of determining rumour veracity ought to be automated.

This paper is concerned with the task of stance classification of Twitter claims and responses. Stance classification is the task of determining the attitude of a short text towards a claim. The author's attitude towards the claim is typically classified into four categories:

- **Support** - Author of the short text supports the veracity of the original claim
- **Deny** - Author of the short text denies the veracity of the original claim
- **Query** - Author of the short text poses a question regarding the contents of the original claim or requesting additional evidence regarding veracity of the claim
- **Comment** - Author of the short text provides additional information that is neutral regarding the veracity of the original claim.

The stance of responses can be highly indicative of the original claim veracity and should be utilized to verify claims as a form of crowd response analysis (Aker et al., 2017). The task of stance classification is generally regarded as difficult and the nature of the task allows for numerous approaches. Feature engineering has traditionally been used for similar natural language processing tasks but has recently been challenged by neural and deep models that learn implicit features of the data. This paper will focus on two main points. The first issue revolves around how including the contents of source-tweet into the classification impacts classification performance. The second point is concerned with the lack of proper comparative analysis between non-neural classifier models for the task of stance classification and will aim at providing such.

2. Related Work

Up until the 2017 RoumourEval shared task the interest for roumour veracity has been relatively low (Derczynski et al., 2017). The shared tasks have spurred more interest in the topic, but state-of-the-art has still not fully matured, considering the most successful model from 2019 SemEval reported an F1 measure of 0.6187 for the task of rumour stance classification (Gorrell et al., 2019).

The task has been approached from several perspectives and with distinct goals. The two most common approaches have been developing and extracting the set of most informative features from the contents of tweets and similar short texts or developing the neural model that can capture intrinsic features of the content the best (Gorrell et al., 2019). Aker et al. (2017) highlight a significant improvement in classifier performance due to the inclusion of informed hand-crafted content features.

Training neural models for the task of stance classification is predominantly done by either training the model from scratch or using pre-trained language models and transformers such as BERT. LSTM-based language models have proven most adequate for many natural language processing tasks and developing different neural architectures around LSTM and attention models shows satisfactory results for the task of stance classification (Du et al., 2017). However, pre-trained language models have shown to be top of the class at solving the task of stance classification (Bharathi et al., 2020).

Contrary to previous content-based works, UTCNN, a deep learning model of stance classification, utilizes the usually discarded information from the social media platforms (Chen and Ku, 2016). Such information consists of extra-linguistic features such as post metadata, user metadata, and statistics regarding the user's preference, topic taste, and the number of "likes". Authors have reported that UTCNN yields a 0.842 accuracy on the English online debate forum test dataset.

On the other hand, the importance of rapid response of state-of-the-art systems has been recognised and explored (Vosoughi et al., 2017). Authors approach the task of rumour veracity as a real-time verification task, aiming at minimizing impact of false information distributed through

Twitter.

Apart from the research and state-of-the-art development within the field of rumour stance classification, improvements in related fields such as sentiment analysis and stance classification in ideological debates have been of great use for this purpose (Hasan and Ng, 2014).

The distinction between the task of rumour stance classification and sentiment analysis is the environment in which the short text is observed. For sentiment analysis the claim is observed in isolation, while the claim in stance classification task is observed in comparison to the source claim (Gorrell et al., 2019). However, most models used for stance classification are only utilizing the reply claim and not the original one.

Considering the distinction between the tasks, this paper aims to further investigate the importance of including the source claim for success of stance classification task.

3. Dataset

We use the RumourEval2019 dataset, which consists of a set of English tweets and their discussion threads. Twitter is a particularly valuable source of data for the rumour veracity detection task considering the number of active users, dynamics of responses, and content in the format of short texts.

Tweets have been collected about several controversial topics on Twitter that have created a noticeable amount of attention on the platform. Within every topic, the tweets were organized into multiple discussion threads.

The source tweet is the rumour and response tweets from the thread have a tree-like structure. Every tweet in the thread is assigned one of the four stance labels. The source tweet is labeled as true. Labels were assigned by annotators from crowdsourcing platforms. The annotator agreement for all labels is at least 70%. The dataset also contains metadata on the tweets and metadata on Twitter users who have authored the tweets (Gorrell et al., 2019).

Table 1 shows the distribution of labels across the training and test datasets. The distribution of the total of 5568 labels is skewed towards the comment label, which tends to be least indicative of the veracity of the claim (Gorrell et al., 2019).

Table 1: Distribution of labels in training and test dataset

	Train	Test
Support (S)	1004	141
Deny (D)	415	92
Query (Q)	464	62
Comment (C)	3685	771
Total	5568	1066

4. Experimental Setup

The experimental setup consists of data extraction and preprocessing, feature extraction, and data adjustment for neural models. All experiments are conducted on the contents of tweets.

To investigate the importance of original claim presence for the performance of stance classifiers, a simple setup using bidirectional LSTM networks aims at comparing results of using the combination of source and reply tweets and using only source tweets. The results for both setups with LSTM are compared with the results of classifiers with hand-crafted features and n-gram models.

4.1. Data Preparation and Preprocessing

Our data preparation consists of loading the contents of tweets into data frames and generating the source-tweet and reply-tweet pairs. The source tweet in the pair is the root of the discussion thread and the reply in the pair is a tweet in the discussion thread, regardless of the depth in the thread. String labels have been replaced by numerical representations. Source and reply tweets have also been lemmatized and cleansed of external links and punctuation. Numbers have been replaced by the '#' sign. This constitutes the preprocessed dataset.

4.2. Feature Extraction

Feature extraction has been performed both on the preprocessed and original tweets. Features have been created solely based on the reply tweet in the tweet pair. Features have been crafted with regards to observed linguistic and structural properties of tweets indicative of stance towards the original claim, closely related to the notion of sentiment (Aggarwal and Aker, 2019). We consider the following features:

- 25 Word Glove embeddings - 25-Dimensional Word Vectors Trained on Tweets that represent words in the tweet as vectors
- Number of standard negative words[1] - number of occurrences of words with a strong negative sentiment of the tweet author
- Number of standard positive words - number of occurrences of words with a strong positive sentiment of the tweet author
- Indicator of the presence of question mark - indicator variable that strongly indicates the affiliation to the comment label
- Indicator of the presence of exclamation mark - indicator variable that often indicates the affiliation to the deny label
- Indicator of the presence of numbers - indicator variable that often indicates the affiliation to either comment or deny label, considering that numbers are often used when denying the claim or providing additional information on the topic.
- Capitalized characters ratio - the proportion of the capitalized characters in the whole sequence, where a higher ratio indicates a high likeliness of denying the claim

[1]The list of positive and negative words have been obtained from (Hu and Liu, 2004).

- Tweet content length - lengthy tweets are often indicative of denying the original claim.

4.3. N-Gram Models

The motivation for including n-gram models is the fact that they allow for better capturing of the local context information and can yield quality results for many natural language processing and stance classification tasks (Liu and Forss, 2014). This paper compares the performance of monogram, bigram and trigram features combined with several classification models.

4.4. Majority Class Baseline

As a baseline system, we leverage the majority class baseline. The majority class in both training and test datasets is the comment class. The training dataset is comprised of 3685 comment tweets and the test dataset is comprised of 771 comment tweets, as presented in Table 1. The accuracy for the majority class baseline on the test dataset is 0.7233 and macro F1 score is 0.2098.

4.5. Neural Models

Two bidirectional LSTM networks have been modeled, one that takes vectorized reply-tweet as input and the other that takes concatenation vectorized source-tweet and reply-tweet. The model architecture uses the dropout layer to combat overfitting. The optimal number of epochs is 10 for both LSTM networks.

The aim of this experiment is not to showcase the state-of-the-art model for the stance classification task, but rather to examine the impact of including the original claim in the classification on the performance of a simple model.

4.6. Hyperparameter Optimization

Hyperparameters for SVC model with RBF kernel were obtained from grid search over C = $\{0.1, 1, 10, 100, 1000\}$ and $\gamma = \{1, 0.1, 0.01, 0.001, 0.0001\}$.

The optimal number of epochs was determined by training neural models with number of epochs ranging from 1 to 20.

5. Results

5.1. Performance of Stance Classifiers

The first part of the experimental setup provides an overview of different classifier models and how they respond to different feature arrangements. Table 2 shows the accuracy and macro F1 scores for models and feature arrangements. Due to the test dataset having a high share of replies labeled as a comment, the referent measure taken for comparison is macro F1.

Looking at the n-gram features used, we can conclude that the monogram setup performed the best on all models, regardless of our assumption that it is unlikely that it can fully capture the complexity of the task and that the bigram feature setup would have utilized the possibility of capturing negations. The only exception is the SVC model with RBF kernel that yielded similar F1 scores for both monogram and bigram feature setups. The poor performance of all models on the trigram setup can be attributed to the overfitting of the model to the training dataset.

Table 2: Accuracy and macro F1 scores of models on different features ranges.

Features	Model	Accuracy	F1
Hand-crafted	Naive Bayes classifier	0.7233	0.2098
	Logistic regression	**0.7467**	**0.3977**
	Linear SVM (SGD[2])	0.7345	0.3452
	Linear SVC	0.7401	0.378
	SVC (RBF kernel)[3]	0.7336	0.3483
	Majority class baseline	0.7233	0.2098
Monograms	Naive Bayes classifier	0.7223	0.2098
	Logistic regression	**0.7242**	0.3142
	Linear SVM (SGD)	0.712	0.3513
	Linear SVC	0.7016	**0.3679**
	SVC (RBF kernel)[4]	0.7214	0.302
Bigrams	Naive Bayes classifier	0.7232	0.2098
	Logistic regression	**0.7261**	0.2679
	Linear SVM (SGD)	0.7232	0.299
	Linear SVC	0.7214	0.2911
	SVC (RBF kernel)[5]	0.7148	**0.3001**
Trigrams	Naive Bayes classifier	0.7232	0.2098
	Logistic regression	0.7223	**0.226**
	Linear SVM (SGD)	0.7214	0.2258
	Linear SVC	0.7214	0.2258
	SVC (RBF kernel)[6]	**0.7233**	0.2132

Best results per feature category in Table2 are bolded. The best performing model overall is the logistic regression model trained on hand-crafted features, with the macro F1 score of 0.3977. Models based on hand-crafted features consistently outperformed n-gram feature setups. Logistic regression classifier with hand-crafted features would have taken the ninth place in the SemEval-2019 RumourEval shared task, where most submissions are neural classifiers (Gorrell et al., 2019). This shows that it is possible for the hand-crafted features to face some simpler neural models and that feature engineering should not yet be discarded as a valid approach to the task of rumour stance classification.

5.2. LSTM Classifier Performance

We observed differences in performance of a simple neural classifier in the case of training and classification only of reply tweets versus in the case of training and classification of source-reply tweet pairs. Per Table 3, the LSTM model performed noticeably better on the task of reply tweet stance classification. The reply tweet classification has yielded both better accuracy and F1 scores. This is in contrast with the claims of (Derczynski et al., 2017) that including the source tweet might be beneficial for the performance of stance classification models. Presumably, the

[2]Stohastic gradient descent
[3]C=1000 γ=0.001
[4]C=1 γ=1
[5]C=10 γ=1
[6]C=1000 γ=1

Table 3: Accuracy and macro F1 scores of LSTM models on reply tweet and source-reply tweet datasets

Model	Data	Accuracy	F1
LSTM	reply tweets	**0.7392**	**0.3492**
LSTM	source and reply tweets	0.7223	0.3102

source tweet introduced too much noise for the classifier, and some indicative implicit features of the reply tweet were not learned.

6. Conclusion

The problem of rumour veracity can be approached from several angles. Rumour veracity task is intertwined with the task of rumour stance classification. The stance of the replies' authors can be highly suggestive of the rumour veracity. That makes the task of stance classification a legitimate proxy for determining the rumour veracity.

Relevant features for the task can be manually extracted through the feature engineering mechanisms or learned implicitly by neural models. This paper showcased how different types of features interact with different classification models and concluded that hand-crafting features remains a relevant approach to solving the task of rumour stance classification.

Furthermore, seeing that the reply tweet stance is made concerning the original claim of the source tweet, the question arises whether including the source tweet in classification would benefit the classifier performance. However, the experiment on a simple neural classifier denied such an idea.

References

Piush Aggarwal and Ahmet Aker. 2019. Identification of good and bad news on Twitter. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 9–17, Varna, Bulgaria, September. INCOMA Ltd.

Ahmet Aker, Leon Derczynski, and Kalina Bontcheva. 2017. Simple open stance classification for rumour analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 31–39, Varna, Bulgaria, September. INCOMA Ltd.

B. Bharathi, J. Bhuvana, and Nitin Nikamanth Appiah Balaji. 2020. Ssncse-nlp @ evalita2020: Textual and contextual stance detection from tweets using machine learning approach (short paper). In *EVALITA*.

Wei-Fan Chen and Lun-Wei Ku. 2016. UTCNN: a deep learning model of stance classification on social media text. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1635–1645, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. 2017. SemEval-2017 task 8: RumourEval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 69–76, Vancouver, Canada, August. Association for Computational Linguistics.

Jiachen Du, Ruifeng Xu, Yulan He, and Lin Gui. 2017. Stance classification with target-specific neural attention. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3988–3994.

Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski. 2019. SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845–854, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Kazi Saidul Hasan and Vincent Ng. 2014. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 751–762, Doha, Qatar, October. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. pages 168–177, 08.

Shuhua Liu and Thomas Forss. 2014. Combining n-gram based similarity analysis with sentiment analysis in web content classification. In *Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - Volume 1*, IC3K 2014, page 530–537, Setubal, PRT. SCITEPRESS - Science and Technology Publications, Lda.

Soroush Vosoughi, Mostafa 'Neo' Mohsenvand, and Deb Roy. 2017. Rumor gauge: Predicting the veracity of rumors on twitter. *ACM Trans. Knowl. Discov. Data*, 11(4), jul.