

Queen of Swords at Touché 2023: Intra-Multilingual Multi-Target Stance Classification using BERT

Notebook for the Touché Lab on Argument and Causal Retrieval at CLEF 2023

Karla Schäfer^{1,†}

¹*Fraunhofer Institute for Secure Information Technology SIT | ATHENE - National Research Center for Applied Cybersecurity, Rheinstraße 75, Darmstadt 64295, Germany*

Abstract

Stance classification can be used in various scenarios, such as fake news detection or public opinion measurement. However, little work has been done on stance detection in multilingual data. For this reason, this work uses a multilingual, multi-target, and multi-topic dataset to develop a classifier for detecting stance in such data. The classifier was trained using pre-trained BERT models, with various experiments showing superior performance of a fine-tuned multilingual BERT model with self-training. Since the dataset was unbalanced, with the main label being "in favor", the macro-averaged F1 score was used for measurement. The best performing model achieved a macro-average F1 score of 0.8862 using the same proposals in stance classification for training and testing. The same approach was used to train two classifiers for the CLEF 2023 Touché Lab Task 4 Subtask 1 and 2, using new, previously unseen proposals for testing. However, by using new unseen proposals, the results deteriorated significantly, and in the challenge only a macro F1 score of 0.324 and 0.417 was achieved.

Keywords

Stance Detection, BERT, Fine-tuning, Self-training

1. Introduction


With the rise of the Internet, people can publish their opinions at any time, e.g. in user forums, blogs or social media platforms, and with stance detection, these comments on various topics can be automatically evaluated. Given a proposal and comments on this proposal, the task of stance detection is to identify the stance of the comment author towards a target (proposal) [1]. This information can then be used, e.g., in fake news detection to classify the stance of headlines to their article bodies to determine if the title is related to the content [2]. Another application scenario is document retrieval tasks, e.g. to measure public opinion towards an event (or entity), such as the Brexit [3] or the US elections [4].


Stance classification tasks can be divided according to language (mono- or multilingual), topic and target (number of target labels). In this paper, an approach to classify multilingual and multi-target stances has been developed as part of the Shared Task 4 of Touché ("Intra-Multilingual Multi-Target Stance Classification") [5, 6]. Using Barriere and Balahur's dataset [7], I fine-tuned a multilingual BERT classifier and applied self-training.

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

✉ karla.schaefer@sit.fraunhofer.de (K. Schäfer)

ORCID [0009-0004-1731-7925](https://orcid.org/0009-0004-1731-7925) (K. Schäfer)

 © 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

In the following, the related work is presented first (Section 2). Then, in Section 3.2, the dataset is briefly introduced and the experimental methods are explained. In Section 4, first results on a test set are presented, together with the results of the Touché Challenge Task 4. The final Section (Section 5) provides a conclusion that identifies limitations and presents future work.

2. Related Works

In supervised learning, learning-based stance classification approaches can be divided into traditional machine learning, deep learning and ensemble learning approaches [8]. Traditional machine learning uses feature-based learning such as support vector machines (SVM) or decision trees. Predefined features such as ngrams, POS tags or sentiments are used to train the classifier [8]. Deep learning based approaches often use classifiers such as LSTMs and CNNs [8].

Tran et al. [9] used CNN with BERT for stance detection in the low-resource language Vietnamese. BERT was used to extract contextual word embeddings, followed by a CNN for classification. The averaged accuracy was compared with Bi-LSTM using different word2vec and BERT embedding approaches. The best performer was BERT-CNN.

In the 2017 Fake News Challenge (FNC-1), the task was to estimate the stance of articles toward a given headline (i.e., claim). The best performing system used an ensemble based on a gradient-boosted decision tree and a convolutional neural network (CNN), along with textual features [10].

As a traditional linear classifier with bag of words representations was compared with a multilingual BERT, which performed better than the traditional approach [11]. While the cross-lingual performance of BERT increased when all questions were in English.

According to Ghosh et al. [12], BERT outperformed feature-based and other neural approaches in the area of stance detection in a monolingual English environment. Based on these good results, this paper takes a closer look at BERT as a stance classifier.

3. Methodology

The dataset used to train and evaluate the different BERT models used is called CoFE, created by Barriere et al. [7, 13]. The following is a brief introduction to this dataset, followed by an explanation of the classification method used.

3.1. Dataset

Only the CoFE dataset [13] provided by the challenge was used. The dataset contains comments on proposals on socially important issues from an online debate platform. The dataset consists of proposals consisting of a title and a text, both in English and in the native language, where the native language can be any of the 24 EU languages (plus Catalan and Esperanto). Additional metadata, such as topic and the name of the native language name, are also provided, but not used here. The comments on the proposals are written in the native language and contain other

Table 1
Overview of the Datasets.

Feature	Dataset Subtask 1 (CF_EDevS dataset)			Dataset Subtask 2		
	CF_S	CF_U	CF_E-Dev	CF_S	CF_U	CF_E-Dev
entries	4145	5785	901	7002	13213	1414
number labels	2	0	3	2	0	3
label (in favor)	3214	-	496	5440	-	753
label (against)	931	-	64	1562	-	118
label (others)	-	-	341	-	-	543

information such as the topic, language, upvotes and downvotes. The comments can be divided into the CF_S, CF_U and CF_E-Dev subsets.

I merged the title and the proposal and used only these data together with the corresponding comments for training the classifier. The other data (topic, downvotes, etc.) were not used for classification. Due to time constraints, the main experiments were performed on the Subtask 1 dataset only. This limited the training data to the part of the CoFE dataset without the comments from the debates of the CF_E-Test test set. No other datasets were used.

The CF_E-Dev dataset contains a small set of comments annotated with three stance labels, the CF_S dataset is a larger set with binary self-annotations (labels: "in favor" or "against"). These two datasets have been used primarily. The CF_U dataset contains unlabeled data, so the labels had to be determined first. This dataset was used in the second part of the training for self-training. In total, there are 4247 different proposals in the dataset. These proposals are linked to the comments by an ID (id_prop). An overview of the used datasets can be found in Table 1.

Only the dataset from Subtask 1 was used to determine the hyperparameters. To do this, I combined the CF_E-Dev and CF_S datasets, resulting in the CF_EDevS dataset with 5,046 labeled entries. Since the dataset is very unbalanced, with 3,710 entries labeled "in favor", the macro-averaged F1 score was used for the following evaluation.

The CF_EDevS dataset was used to create a training (3,633 entries), validation (404 entries), and test (1,009 entries) dataset for evaluating the different approaches. For participation in the Touché Task 4 challenge, only a split between validation (505 entries) and training (4,541 entries) was made.

3.2. Approach

Due to the great success of BERT in related work [11, 12], I decided to use different pre-trained BERT models [14] for stance classification. In stance detection, sentence pairs (proposal and comment) are passed to BERT. In creating this input, I followed the guidelines of Devlin et al. [14] and passed the two specifications, separated by a special token (*[SEP]*), as input to BERT. In addition, the token type ids were stored and passed to the model (segment embedding). The sequence is preceded by the special token *[CLS]*.

Since sentence pairs consisting of longer texts were given as input, different truncation strategies were tested. The BERTokenizer of each pre-trained model was used as the tokenizer. First, the average number of tokens was manually calculated for the proposal and the comment

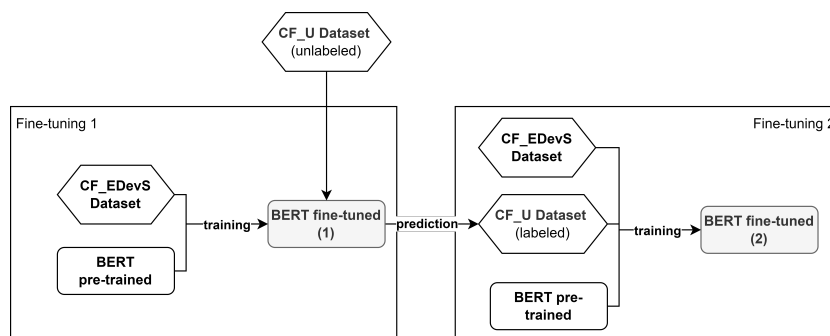


Figure 1: Overview of the approach, divided into Fine-tuning 1 and 2 (self-training).

separately. Since the proposal contained more tokens on average than the comments (proposal: 227 tokens on average, comments: 86 tokens on average in the non-translated dataset; proposal in the native language), the truncation strategy longest or truncation first was chosen (for results of different truncation strategies see Table 2 in Section 4.1). The truncation length was set to 512 as the longest possible input length for BERT.

Since the dataset is multilingual, a multilingual model¹ pre-trained on 104 languages and recommended by the developers² was tried first. Since many pre-trained models are trained on English data, the dataset was translated into English using GoogleTranslator³. On this translated dataset, I applied another BERT pre-trained on English data⁴ only. The pre-trained BERT model was combined with a linear layer for sentence classification. In a second approach, BERT was implemented with a Bi-LSTM layer.

First, the CF_EDevS dataset (from Subtask 1) was used to fine-tune the different BERT models (Fine-tuning 1). Different combinations of the dataset language were tried (see Table 2 in Section 4.1 for the results). The resulting fine-tuned models were used to make predictions for the unlabeled CF_U dataset, including probabilities for classified labeling. Subsequently, the now labeled CF_U dataset was used together with the CF_EDevS dataset to fine-tune the BERT model again (so-called self-training; Fine-tuning 2). Different probability thresholds were tried. Only those comment-proposal pairs from the CF_U dataset whose labels were predicted above a certain probability were used for training. For the whole process, a batch size of 8 was used and different learning rates (5e-5, 3e-5, 2e-5) were tried, number of epochs: 5, 10. For a summary of the process, see Figure 1.

4. Results

First, initial trials were conducted locally with the dataset in different languages. For this, the respective BERT model was fine-tuned only once (Section 4.1). After the first parameters (language, truncation strategy and method) were determined, a longer training including the

¹<https://huggingface.co/bert-base-multilingual-cased>

²<https://github.com/google-research/bert/blob/master/multilingual.md>

³<https://pypi.org/project/deep-translator/#google-translate-1>

⁴<https://huggingface.co/bert-base-uncased>

Table 2
Results for Fine-tuning 1.

No.	model	language (proposal)	language (comment)	truncation strategy	macro-averaged F1 score
1	multilingual	english	native	end	0.8097
2	multilingual	english	native	first	0.7938
3	multilingual	english	native	longest	0.8017
4	multilingual	native	native	longest	0.8067
5	english	english	english	longest	0.8235
6	Bi-LSTM	english	english	longest	0.8145

second fine-tuning on the CF_U dataset was performed on a GPU (Section 4.2). The best performing model was then used to determine the results for the challenge (Section 4.3).

4.1. Results for Fine-tuning 1

First, I translated the comments and computed the results of the classifiers (simply BERT fine-tuned) locally (see Table 2 for results). All results were obtained after 4 epochs of fine-tuning and a learning rate of $2e-5$. This shows that different truncation strategies do not have a big impact on the results (slightly better: truncation end or longest). In the following experiments, truncation longest was used.

Next, I tried different combinations of the dataset in different languages. I first used the proposals in English and the comments in their native language (Table 2, No.1-3) and compared them with proposals and comments in their native language (Table 2, No.4) and everything in English (Table 2, No.5). The best results were obtained for the whole dataset in English (No.5; 0.8235) and everything in the native language (No.4; 0.8067). In another experiment (No.6), I implemented BERT with a Bi-LSTM layer. However, the results here were worse than with the English dataset (No.5). Therefore, this approach was not explored further here, but future work could explore this approach in more detail.

4.2. Results for the entire Process

After the best results were obtained in the first fine-tuning trials (Section 4.1) with the English and native-language datasets, these were used in the subsequent trials. First, different learning rates and epochs were tried. In most cases, overfitting occurred after 2 epochs and training was stopped. The best results for the first fine-tuning step were obtained on the native language dataset with a learning rate of $2e-5$ after 2 epochs (Table 3, No.5). Subsequently, the labels of the CF_U dataset were predicted using this best model (creating weak labels) and the second fine-tuning (self-training) was performed, similar to the approach of Barriere et al. [15].

Different thresholds for the amount of data from the CF_U dataset were tried. More data gave better results. At a threshold of 90% probability for the predicted label, I stopped to avoid adding too many weak labels to the training dataset. The best classifier achieved a macro-averaged F1 score of 0.8862 on the test set (Table 3, No.6). The second fine-tuning on the CF_U dataset increased the F1 score by 0.08.

Table 3

Results for the whole process (Fine-tuning 1+2).

No.	language (dataset)	learning rate	epochs	threshold CF_U	count CF_U used	macro-averaged F1 score
1	english	2e-5	1	-	-	0.7604
2	english	3e-5	1	-	-	0.7604
3	english	5e-5	4	-	-	0.7663
4	native	5e-5	2	-	-	0.7842
5	native	2e-5	2	-	-	0.8079
6	native	2e-5	2	>0.9	3 304 entries	0.8862
7	native	2e-5	2	>0.93	2 955 entries	0.8402
8	native	2e-5	2	>0.99	746 entries	0.7897

Table 4

Results in the CLEF 2023 Touché Lab Task 4 Challenge.

Team	Run timestamp	all-accuracy	all-macro f1-score	all-micro f1-score
touche23-queen-of-swords (Subtask 1)	2023-05-19-07-51-03	0.605	0.417	0.605
touche23-queen-of-swords (Subtask 2)	2023-05-19-07-51-35	0.616	0.324	0.616
touche23-baseline	2023-04-09-12-20-42	0.552	0.237	0.552

4.3. Results of the Challenge

For the challenge, one model each was trained for Subtask 1 and Subtask 2, using two different sized datasets (see Table 1). Both models were trained using the methodology already presented in Section 3.2. The parameters were taken from the model with the best performance in Table 3 (language of the dataset: native; learning rate: 2e-5; epochs:2; threshold for the dataset CF_U > 0.9). To train the classifier for Subtask 1, 3,304 entries of the CF_U dataset, i.e. weak labels, were used. For the Subtask 2 model, 10,726 entries of the CF_U dataset were used. The results of the Subtask 1 and 2 models are shown in Table 4, together with the baseline.

5. Conclusion

In this paper, different approaches for fine-tuning BERT models (English and multilingual) on a multilingual multi-target dataset were evaluated. Although a satisfactory result with a macro-averaged F1 score of 0.8862 was obtained with the self-generated CF_EDevS dataset, the results in the challenge were rather poor. This is probably because I trained, validated, and tested the model using the same proposals. The challenge then used different proposals for testing, which led to the poor results in Section 4.3.

Only the CoFE dataset provided by the challenge was used, training BERT on more data, such as the X-Stance dataset, might improve the results. Other works have achieved very good results with ensemble methods on stance classification tasks. For example, the fine-tuned models could be combined with a decision tree to improve the results.

Acknowledgments

This work was supported by the German Federal Ministry of Education and Research (BMBF) and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of "ATHENE – CRISIS" and "Lernlabor Cybersicherheit" (LLCS).

References

- [1] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, C. Cherry, Semeval-2016 task 6: Detecting stance in tweets, in: Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016), 2016, pp. 31–41.
- [2] P. Bourgonje, J. M. Schneider, G. Rehm, From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles, in: Proceedings of the 2017 EMNLP workshop: natural language processing meets journalism, 2017, pp. 84–89.
- [3] V. Simaki, C. Paradis, A. Kerren, Stance classification in texts from blogs on the 2016 british referendum, in: Speech and Computer: 19th International Conference, SPECOM 2017, Hatfield, UK, September 12-16, 2017, Proceedings 19, Springer, 2017, pp. 700–709.
- [4] M. Lai, D. I. Hernández Fariás, V. Patti, P. Rosso, Friends and enemies of clinton and trump: using context for detecting stance in political tweets, in: Advances in Computational Intelligence: 15th Mexican International Conference on Artificial Intelligence, MICAI 2016, Cancún, Mexico, October 23–28, 2016, Proceedings, Part I 15, Springer, 2017, pp. 155–168.
- [5] Overview of Touché 2023: Argument and Causal Retrieval, Springer, 2023.
- [6] A. Bondarenko, M. Fröbe, J. Kiesel, F. Schlatt, V. Barriere, B. Ravenet, L. Hemamou, S. Luck, J. Reimer, B. Stein, M. Potthast, M. Hagen, Overview of Touché 2023: Argument and Causal Retrieval, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 14th International Conference of the CLEF Association (CLEF 2023), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2023, p. to appear.
- [7] V. Barriere, A. Balahur, Multilingual multi-target stance recognition in online public consultations, *Mathematics* 11 (2023) 2161.
- [8] D. Küçük, F. Can, Stance detection: A survey, *ACM Comput. Surv.* 53 (2020). URL: <https://doi.org/10.1145/3369026>. doi:10.1145/3369026.
- [9] O. Tran, A. C. Phung, B. X. Ngo, Using convolution neural network with bert for stance detection in vietnamese, in: Proceedings of the Thirteenth Language Resources and Evaluation Conference, 2022, pp. 7220–7225.
- [10] D. S. Sean Baird, Y. Pan, Talos targets disinformation with fake news challenge victory, 2017. URL: <https://blog.talosintelligence.com/talos-fake-news-challenge/>.
- [11] J. Vamvas, R. Sennrich, X-Stance: A multilingual multi-target dataset for stance detection, in: Proceedings of the 5th Swiss Text Analytics Conference (SwissText) & 16th Conference on Natural Language Processing (KONVENS), Zurich, Switzerland, 2020. URL: <http://ceur-ws.org/Vol-2624/paper9.pdf>.
- [12] S. Ghosh, P. Singhanian, S. Singh, K. Rudra, S. Ghosh, Stance detection in web and social media: a comparative study, in: Experimental IR Meets Multilinguality, Multimodality, and

Interaction: 10th International Conference of the CLEF Association, CLEF 2019, Lugano, Switzerland, September 9–12, 2019, Proceedings 10, Springer, 2019, pp. 75–87.

- [13] V. Barriere, G. G. Jacquet, L. Hemamou, Cofe: A new dataset of intra-multilingual multi-target stance classification from an online european participatory democracy platform, in: Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, 2022, pp. 418–422.
- [14] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: <https://aclanthology.org/N19-1423>. doi:10.18653/v1/N19-1423.
- [15] V. Barriere, A. Balahur, B. Ravenet, Debating europe: A multilingual multi-target stance classification dataset of online debates, in: Proceedings of the LREC 2022 workshop on Natural Language Processing for Political Sciences, 2022, pp. 16–21.