Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)

UPB at GermEval-2019 Task 2: BERT-Based Offensive Language Classification of German Tweets

Andrei Paraschiv
Computer Science Department
University Politehnica of
Bucharest, Romania
andrei.paraschiv74@stud.acs.upb.ro

Dumitru-Clementin Cercel
Computer Science Department
University Politehnica of
Bucharest, Romania
clementin.cercel@gmail.com

Abstract

In this paper, we describe our participation to GermEval-2019 Task 2, which requires identifying and classifying offensive content in German tweets. For all three challenging subtasks, i.e. i) Subtask 1 – a binary classification between Offensive and Non-Offensive tweets, ii) Subtask 2 – a fine-grained classification into three different categories: Profanity, Insult, Abuse and iii) Subtask 3 – detecting whether the tweets contain Explicit or Implicit Offensive language, we used the **B**idirectional **E**ncoder **R**epresentations from **T**ransformers (BERT) model with a pre-training phase based on German Wikipedia and German Twitter corpora and then performed fine-tuning on the competition dataset. Thus, our approach focuses on how to pre-train, fine-tune and deploy a BERT model to classify German tweets. Our best submission achieves on test data 76.95% average F1-score on Subtask 1, 53.59% on Subtask 2 and 70.84% on Subtask 3.

1 Introduction

Online social networks today are more popular than ever. However, the freedom of communication leads sometimes to abusive and undesired behavior. For example, hate speech, racism, abusive language, doxing or offensive speech has become a real problem for all major online social networks. Due to its short messages and very interactive nature, this behavior is mostly present in Twitter. The huge amount of user-generated content renders a manual review impossible. Bound by the law[1] to remove hate speech from their websites, online media companies have invested a lot of effort and resources to detect and classify hate speech and abusive language automatically.

The task of abusive and offensive language identification has been recently addressed in several papers and competitions (Kumar et al., 2018) (Zampieri et al., 2019b), with a large focus on English language. The GermEval campaign (Wiegand et al., 2018) tries to overcome this shortage and proposed at the 2019 edition a second shared task on the identification of offensive language in German tweets.

Waseem et al. (2017) identified the dimensionality for the typology of abusive language - the uttering can target a particular person, or it can be directed at a generalized group, for instance, an ethnic minority, immigrants, sexual minority. In the GermEval-2019 Task 2 training set, we can find a higher number of directed abusive statements like "Das Weib hat wirklich einen Vogel", and some generalized abuse (e.g., "Der Islam hat den Bundesgerichtshof gekauft"). This skewed distribution is consistent with the distribution in the dataset created by (Zampieri et al., 2019a) for English tweets. The other dimension proposed by Waseem et al. is the extent to which the hate speech is explicit or implicit and it is directly addressed by our Subtask 3.

In our research, we deployed a deep learning system based on Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018), a general language model that is by default pre-trained on two corpora, i.e., English Wikipedia and BooksCorpus (Zhu et al., 2015), using a "masked language model" and "next sentence prediction". In contrast to classical word embedding models like GloVe (Pennington et al., 2014) or Word2Vec (Mikolov et al., 2013), BERT uses a limited vocabulary

[1] https://www.bmjv.de/SharedDocs/Gesetzgebungsverfahren/Dokumente/NetzDG_engl.pdf

(around 30,000 words, compared to 400,000 words used by GloVe), since it relies not only on word embeddings but also on segment and positional embeddings. Our paper assesses the performance of a slightly modified BERT model, pre-trained from scratch on the German Wikipedia followed by German Twitter data.

The remainder of this work is organized as follows. The next section briefly shows an overview of the current methods employed for detecting offensive language. Section 3 describes the GermEval-2019 Task 2 dataset and our runs for this task. Section 4 illustrates the performances of each run. Section 5 outlines the conclusions that can be drawn from our work and possible future improvements.

2 Related Work

In recent years, the issues of toxic comments and abusive language have come to the forefront of text classification research. Generally, most classification studies have focused on three main topics:

- Identifying offensive texts (binary classification);
- Distinguishing between explicit and implicit offensive language;
- A fine-grained classification of offensive texts.

One major problem for researchers is the difficulty to clearly define hate speech, and also the lack of large labeled datasets, and thus the lack of consensus among the annotators (Waseem, 2016). Moreover, in most large corpora, the percent of offensive speech is very low and labeling enough positive samples takes a lot of tedious work. Gilbert et al. (2018) narrows the search down by choosing a white supremacist forum to extract the samples and then to manually annotate them. Their work not only provides an insightful annotation procedure but also shows that, for an accurate labeling, a sentence needs extra context, for example, the whole conversation or the forum thread title.

For German language, Ross et al. (2017) showed that reliability in the annotation work is relatively low and more guidelines and clear definitions can bring improvements. Notably, Köffer et al. (2018) has shown that methods developed for hate speech detection in English language can be successfully applied to similar tasks in German language. However, due to German language characteristics, the process can be more complex, and results might achieve lower scores than their English counterparts.

Burnap and Williams (2015) used a feature-based classification employing various machine learning algorithms - Bayesian Logistic Regression, Random Forest Decision Trees, Support Vector Machines and an ensemble of all three models. They used different feature sets, such as n-grams, hateful terms and typed dependencies.

Recently, deep learning models have been widely applied to handle natural language processing tasks. For example, Gao and Huang (2017) proposed the utilization of context information by employing BiLSTM (Bidirectional Long-Short Term Memory Networks) with attention layer (Bahdanau et al., 2014) for hate speech detection. Founta et al. (2018) developed a deep neural network architecture that also takes various tweet metadata into account (e.g., number of followers, number of retweets, etc.) besides the content of the tweets. Schäfer (2018) is building upon this architecture and proposed a classification model aimed at German texts.

Wu et al. (2019) used the BERT model to detect and classify offensive language in English tweets. They used the base, uncased version with 768-dimensional embeddings and obtained good results in the binary classification task.

3 Methodology

3.1 Data analysis

For both Subtasks 1 and 2 (i.e., binary and fine-grained classification respectively), the training dataset supplied for the GermEval-2019 competition Task 2 consists of 3,995 annotated German tweets. Additionally, for Subtask 3 (i.e., explicit or implicit offensive language classification) the training dataset contains 1,958 annotated German tweets.

Annotations for the Subtask 1 classification were OFFENSE for the positive class and OTHER for the negative class. For Subtask 2, the positive class was split into three categories, i.e., PROFANITY, INSULT, ABUSE. Subtask 3 annotations were EXPLICIT and IMPLICIT, since all tweets in this case were marked as OFFENSE.

3.2 Additional training set

To increase the training data size, we also used the annotated data from the previous GermEval-2018 edition (Wiegand et al., 2018), including

8,541 German tweets. This additional dataset was only suitable for Subtasks 1 and 2, since in this dataset there were no implicit/explicit labels. Table 1 shows the final distribution of classes in our training set for Subtasks 1 and 2 and Table 2 presents the distribution for Subtask 3.

Class	Tweets	%
Other	8,359	66.70%
Profanity	271	2.10%
Insult	1,601	12.80%
Abuse	2,305	18.40%
Offensive	4,177	33.30%
Total	12,536	100.00%

Table 1: The final distribution of the tweets in the training dataset for both Subtasks 1 and 2

Class	Tweets	%
Explicit	1,699	86.80%
Implicit	259	13.20%
Total	1,958	100.00%

Table 2: The distribution of the tweets in the training dataset for Subtask 3

3.3 Data preprocessing

All tweets were pre-processed before the classification step. Some basic replacements were performed:

- Emojis encoded in strings like <U+0001F44D> were converted to their unicode representation;
- Emoji characters were spelled out into words like <thumbs_up> or <rolling_on_the_floor_laughing>;
- Usernames, weblinks and newline markers were converted to the standard tokens <user>, <url> and <nl>;
- Numbers, dates and timestamps were converted to standard tokens <number> and <time> using the Ekphrasis text processing tool (Baziotis et al., 2017).

Further, we tried to split each hashtag into atomic words, for instance, hashtags like #EheFürAlle into "Ehe für Alle". Since not all hashtags are camel-cased, we tried to split all non-camel-cased hashtags by using unigrams and bigrams (Baroni et al., 2009).

We checked for spelling errors and unresolved hashtags using the German GloVe vocabulary[2] as reference. Abbreviations, misspellings and spaced out words were manually replaced, for instance, "E N D L I C H" to Endlich, #noAfD to "no AfD", "innenminist" to "Innenminister", "sooooooo" to "so", and "schonlängerhierlebende" to "schon länger hier lebende".

Due to the fact that German words can change their meaning for different capitalization, we tried to preserve or correct the upper/lower case of words.

3.4 Model description

We used the BERT-Base, cased, model pre-trained from scratch using German Wikipedia, OpenLegalData corpus and news articles by deepset.ai[3].

Model BERT (Devlin et al., 2018) is a bidirectional model and consists of 12 transformer blocks, 12 attention heads and 110M parameters. There are two pre-training phases for BERT: "masked language modeling" and "next sentence prediction". For masked language modeling, the model predicts the probabilities for a percentage of random "masked" words from a sentence. The next sentence prediction phase trains the language model to predict if one sentence might follow another sentence.

Pre-training Since tweets have a specificity not captured by Wikipedia or news articles, we pre-trained the BERT model on the 6.2M tweets, consisting of three corpora of German tweets as follows:

- A corpus of 1,212,220 tweets collected by Kratzke (2017) in the context of German Federal Elections of 2017;
- A corpus of 5,964,889 tweets collected by Kratzke (2019) in the months of April and May 2019 around the European Elections 2019;
- A collection of 70,745 tweets of well-known trolling or aggressive Twitter accounts, namely SiffTwitter[4], that were collected by us using Tweepy[5].

For the purposes of this step, we pre-processed the above-mentioned Twitter corpora similar to the training data and experimented with the pre-training hyperparameters. The optimal results were achieved for 150,000 training steps with 10,000 warmup steps and a learning rate of 0.0001.

[2] https://deepset.ai/german-word-embeddings

[3] https://deepset.ai/german-bert

[4] https://medium.com/@trolltwitter/sifftwitter-infos-über-die-schlimmste-hasscommunity-im-netz-dc1f943c0227

[5] https://www.tweepy.org/

Unlike models based on GloVe or Word2Vec, BERT uses by default a WordPiece tokenization (Schuster and Nakajima, 2012), that helps to reduce the vocabulary file to ~31,000 words. Additionally, we extended the BERT vocabulary file with 181 frequent out-of-vocabulary words (e.g., "Rechtspopulismus", "Migrationspakt", "Ibizagate", etc.) from our training data and pre-training corpora. Using WordPiece, the model handles out-of-vocabulary words by breaking them in subwords. For example, words like "Versprechungen" that are not in the vocabulary are tokenized into "Versp", "##rech", "##ungen", or "einlösen" into "ein", "##lösen".

Classification For the classification step, we modified the original BERT model. For Subtasks 1 and 3, we removed the last nine layers from the model. Then, we added a LSTM layer and its output is fed into a fully connected layer with a two-dimensional output vector. In contrast, for Task 2, we removed the last six layers from the model and added a fully connected layer with a three-dimensional output vector, since we predicted only the labels for the entries that were detected as offensive in Task 1.

Fine-tuning The model-training stage was performed with all 12,536 tweets for Subtasks 1 and 2 respectively and with 1,958 tweets for Subtask 3.

Submissions We submitted three runs for evaluation on the test data. The first one was based on above mentioned steps. The second run

Model	Accuracy	Precision	Recall	F1
BiLSTM	76.94%	73.49	73.06	73.27
TUWienKBS	75.32%	72.43	74.49	73.44
BERT Run 1	79.38%	76.35	77.55	**76.95**
BERT Run 2	79.64%	76.6	77.12	76.86
BERT Run 3	79.38%	76.35	77.55	76.95
BERT no pre-train	78.62%	75.51	76.64	76.07

Table 3: Performance comparison of various models on test data for Subtask 1. Precision, Recall and F1-measure are average values over the two classes (OTHER, OFFENSE). The best result is shown in boldface.

Model	Accuracy	Precision	Recall	F1
BiLSTM	67.44%	50.53	39.97	44.64
TUWienKBS	70.47%	53.21	49.42	51.24
BERT Run 1	73.61%	58.53	49.42	**53.59**
BERT Run 2	71.66%	54.4	50.7	52.48
BERT Run 3	73.57%	55.63	49.02	52.11
BERT no pre-train	70.01%	53.04	47.3	50.01

Table 4: Performance comparison of various models on test data for Subtask 2. Precision, Recall and F1 are average values over the four classes (PROFANITY, INSULT, ABUSE and OTHER). The best result is shown in boldface.

Model	Accuracy	Precision	Recall	F1
BiLSTM	85.59%	42.80	50.00	46.12
BERT Run 1	87.85%	77.44	65.28	**70.84**
BERT Run 2	86.88%	73.88	63.48	68.29
BERT Run 3	87.20%	74.39	66.77	70.37
BERT no pre-train	86.13%	71.35	66.76	68.98

Table 5: Performance comparison of various models on test data for Subtask 3. Precision, Recall and F1 are average values over the two classes (EXPLICIT and IMPLICIT OFFENSE). The best result is shown in boldface.

had an additional pre-processing step compared to Run 1, i.e. the words that were not in the GloVe vocabulary were split if possible by using unigrams and bigrams (Baroni et al., 2009), in order to tackle the problem of German compound words. Run 3 was similar to that of Run 1, only instead of 2 epochs, we trained the model for 4 epochs.

For baseline comparison, we used the best performing model at GermEval-2018, namely TUWienKBS, proposed by Montani and Schüller (2018) and also a BiLSTM-based model with German GloVe embeddings[6] of 300 dimensions and a vocabulary of 20,000 words.

4 Experiments

The results for Subtask 1 of the three runs and the baseline models are given in Table 3. As can be seen, the additional preprocessing step did not increase the score for Run 2, since it increased the precision but lowered the recall.

Table 4 shows the results for the fine-grained classification. Thus, the additional training epochs did not improve the results, on the contrary, it seemed to overfit the model.

Finally, as seen in Table 5, the model performed well on the explicit/implicit task and we can see that the additional preprocessing step decreased the score for Run 2.

For Subtask 1, we can see in Figure 1 the learning curve of the average F1-score for one training epoch. After 90% of the training set, the score improvement is less significant and even with 50% of the training data, we can reach good scores.

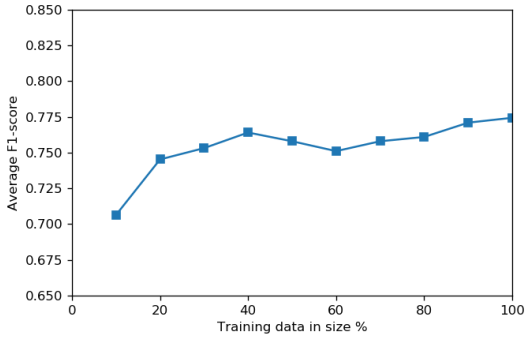

Fig1: Run 1 learning curve of average F1-score

Confusion matrices for the run with the best score, namely BERT Run 1, can be viewed in Tables 6-8 for each subtask. As we can see in Table 6, the model has the tendency to underpredict offensive content. Tweets as "*@morgenpost Das ist eine Sie? h…*" or "*Jetzt daheim. Vielen Dank an den Hurentisch 30, der noch eineinhalb Stunden nach Ladenschluss fröhlich Caipirinas bestellt hat.*" prove difficult to predict as OFFENSE.

The model has also difficulty in telling INSULT and ABUSE apart (see Table 7). Tweets like "*@SPIEGELONLINE Merkel eine Schande für Deutschland überall machen wir Deutsche uns zum Narren*" or "*Deutschland, die anderen 149 Länder wollen ihre Kriminelle Unterschicht loswerden. @WELT_Politik*" were classified as ABUSE rather than INSULT.

Finally, due to the imbalance in the classes for Subtask 3, the model has the tendency to over predict explicit offenses. Tweets like "*Infotweet: Es gibt nur 2 Geschlechter #GenderDay*" or "*Immer wenn ich Deutsche Kinder sehe krieg ich wieder Hoffnung für mein Land*" were wrongly classified as EXPLICIT.

		Predicted Label	
		OTHER	OFFENSE
True Label	OTHER	**1825**	236
	OFFENSE	330	**640**

Table 6: The confusion matrix of BERT Run 1 model for Subtask 1

		Predicted Label			
		OTH	PROF	INS	ABU
True Label	OTH	**1825**	6	89	141
	PROF	37	**19**	25	30
	INS	171	7	**147**	134
	ABU	122	2	36	**240**

Table 7: The confusion matrix of BERT Run 1 model for Subtask 2.

		Predicted Label	
		IMPLICIT	EXPLICIT
True Label	IMPLICIT	**45**	89
	EXPLICIT	24	**772**

Table 8: The confusion matrix of BERT Run 1 model for Subtask 3.

[6] https://deepset.ai/german-word-embeddings

5 Discussion and Conclusions

In this study, we used the BERT-Base version in order to classify German tweets into different categories related to offensive language. Our results show that BERT is a powerful model, capable of detecting offensive language accurately. BERT outperforms a BiLSTM model with GloVe word embeddings for all three subtasks, and the TUWienKBS model in both Subtasks 1 and 2. However, a more subtle classification into nuances of offensive language can lead to lower scores. This can also be a result of the highly unbalanced three subcategories, or due to an unclear delimitation between them.

As for the detection of implicit versus explicit offensive language, it achieves a higher score if one takes into account the fact that the error from Subtask 1 will propagate into Subtask 3. As seen in the results, the pre-trained model with political targeted tweets leads to a slight increase in performance for the binary tasks, but a significant increase for the fine-grained classification.

For future work, we noticed that the treatment of emoticons could be significantly improved, since the spelling out of certain emojis does not always improve the detection. Additionally, a larger set of tweets for pre-training would improve the language understanding of the model. Also, we will work to better distinguish between Insult and Abuse languages, which would allow us to improve the results for Subtask 2.

References

Dzmitry Bahdanau, Kyunghyun Cho and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*

Marco Baroni, Silvia Bernardini, Adriano Ferraresi and Eros Zanchetta. 2009. The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. In *Language Resources and Evaluation*, 43(3): 209-226.

Pete Burnap and Matthew L Williams. 2015. Cyber Hate Speech on Twitter: An application of machine classification and statistical modeling for policy and decision making. In *Policy & Internet* 7(2):223–242.

Christos Baziotis, Nikos Pelekis and Christos Doulkeridis. "DataStories at SemEval-2017 Task 4: Deep LSTM with Attention for Message-level and Topic-based Sentiment Analysis." SemEval@ACL (2017).

Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *ArXiv abs/1810.04805*

Antigoni-Maria Founta, Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Athena Vakali, and Ilias Leontiadis. 2018. A unified deep learning architecture for abuse detection. *CoRR, abs/1802.00385*

Lei Gao and Ruihong Huang. 2017. Detecting online hate speech using context aware models. *arXiv preprint arXiv:1710.07395*

Ona de Gibert, Naiara Pérez, Aitor García Pablos and Montse Cuadros. 2018. Hate Speech Dataset from a White Supremacy Forum. In *ALW2: 2nd Workshop on Abusive Language Online*, pages 11-20.

Sebastian Köffer, Dennis M. Riehle, Steffen Höhenberger and Jörg Becker. 2018. Discussing the Value of Automatic Hate Speech Detection in Online Debates. In *Multikonferenz Wirtschaftsinformatik (MKWI 2018): Data Driven X - Turning Data in Value, Leuphana, Germany*.

Nane Kratzke. 2017. The #BTW17 Twitter Dataset - Recorded Tweets of the Federal Election Campaigns of 2017 for the 19th German Bundestag [Data set]. Data. Zenodo. http://doi.org/10.5281/zenodo.835735

Nane Kratzke. 2019. Monthly Samples of German Tweets (Version 2019-05) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.3236750

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking Aggression Identification in Social Media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbulling (TRAC)*, Santa Fe, USA.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural Information Processing Systems*, 26, pages 3111–3119.

Joaquın Padilla Montani and Peter Schüller. 2018. TUWienKBS at GermEval 2018: German Abusive Tweet Detection. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processings, KONVENS 2018*.

Jeffrey Pennington, Richard Socher and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing*. pages 1532-1543.

Björn Ross, Michael Rist, Guillermo Carbonell, Benjamin Cabrera, Nils Kurowsky and Michael

Wojatzki. 2016. Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis. *ArXiv abs/1701.08118*

Johannes Schäfer. 2018. HIIwiStJS at GermEval-2018: Integrating Linguistic Features in a Neural Network for the Identification of Offensive Language in *Microposts Proceedings of the Workshop Germeval 2018 - Shared Task on the Identification of Offensive Language. Vienna, Austria.*

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conferenceon, pages 5149–5152. IEEE.

Zeerak Waseem. 2016. Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142.

Zeerak Waseem, Thomas Davidson, Dana Warmsley and Ingmar Weber. 2017. Understanding Abuse: A Typology of Abusive Language Detection Subtasks. *ALW@ACL. arXiv:1705.09899*.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language. In *Proceedings of the GermEval Workshop. Vienna, Austria*, pages 1–10.

Zhenghao Wu, Hao Zheng, Jianming Wang, Weifeng Su and Jefferson Fong. 2019. BNU-HKBU UIC NLP Team 2 at SemEval-2019 Task 6: Detecting Offensive Language Using BERT model. In *SemEval 2019 at North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 551-555.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of NAACL*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval)*.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.