Svandiela @ HaSpeeDe: Detecting Hate Speech in Italian Twitter Data with BERT

Svea Klaus **Anna-Sophie Bartle** **Daniela Rossmann**
Eberhard Karls Universität Tübingen
{svea-kristin.klaus, anna-sophie.bartle, daniela.rossmann}
@student.uni-tuebingen.de

Abstract

English. This paper explains the system developed for the Hate Speech Detection (HaSpeeDe) shared task within the 7th evaluation campaign EVALITA 2020 (Basile et al., 2020). The task solution proposed in this work is based on a fine-tuned BERT model. In cross-corpus evaluation, our model reached an F1 score of 77,56% on the tweets test set, and 60,31% on the news headlines test set.

Italiano. *Questo articolo spiega il sistema sviluppato per il tesk finalizzato all'individuazione dei discorsi d'odio all'interno della campagna di valutazione EVALITA 2020 (Basile et al., 2020). La soluzione proposta per il task è basata su un raffinemento di un modello BERT. Nella valutazione finale il nostro modello raggiunge un valore F1 di 77,56% sul dataset di tweets e di 60,31% sul dataset di titoli di giornale.*

1 Introduction

The detection of Hate Speech has been a popular task in Natural Language Processing. Because there is no universal definition of the term 'hate speech', we follow the EVALITA 2018 organizers in defining it as any expression *"that is abusive, insulting, intimidating, harassing, and/or incites to violence, hatred, or discrimination. It is directed against people on the basis of their race, ethnic origin, religion, gender, age, physical condition, disability, sexual orientation, political conviction, and so forth"* (Erjavec and Kovačič, 2012).

Apart from being hurtful to the person or group that the hateful message is aimed at, its systematic usage can be the cause of hate crime and other criminal acts towards these groups. Mass and social media help to spread hate speech a lot faster than traditional communication channels (Sponholz, 2018). However, social media platforms like Twitter, YouTube and Facebook lack systematic control in monitoring and removing hateful comments. Although these platforms discourage hateful content, its removal depends on individual users and trusted reports (Erjavec and Kovačič, 2012), thus indicating that automated detection of such utterances is a crucial problem to solve. Our goal within the HaSpeeDe task was to develop a system for automated detection of hateful messages against muslims, roma, and immigrants. The first section introduces related works on the topic. In the second section, we explain the task setup, followed by the description of our approach. Finally, we show our results and discuss them with regards to possible future work on hate speech detection.

2 Related Work

In previous work, automated detection of hateful messages has been approached in various ways, starting from simpler lexicon-based approaches and Naive Bayes classifiers to more state of the art Convolutional Neural Networks (Zhang and Luo, 2018). The EVALITA 2020 shared task follows SemEval 2019 (May et al., 2019) and EVALITA 2018 (Bosco et al., 2018), where the automated

detection of hateful speech has also been among the core topics.

Early work in this area includes Spertus' automatic recognition of hostile messages with the Smokey system. She found that only 12% of such messages contained explicit keywords. Therefore, she compiled a set of rules resulting in a 47-element feature vector per sentence to capture semantic and syntactic information. For instance, imperative statements have higher chances of containing insulting content than indicative utterances. The same applies to sentences starting with *you*. For evaluation, decision trees were trained on the vectors and the results were compared to human assessments. Overall, in 36% of the cases the instances labeled as insulting matched with the human classification. (Spertus, 1997).

Another approach introduced by Greevy and Smeaton in 2004 involved support vector machines for classifying racist texts. In their work, they compared part-of-speech distributions across racist and non-racist documents as well as different feature representations like bag-of words and bigrams. The bag-of-words model was found to be more useful than the bigram model (accuracy of 87.77% vs. 84.77%) (Greevy and Smeaton, 2004).

Since around 2015 and with the gaining popularity of deep learning, various methods involving neural networks have been proposed. For instance, Kamble and Joshi compared a CNN, LSTM, and BiLSTM to one another for detecting code-mixed Hindi-English hate speech within the context of ICON 2018. The CNN was fed with domain-specific embeddings and showed the best performance (F1 score of 80.85%) (Kamble and Joshi, 2018). The growing interest in hate speech detection is further reflected in other shared tasks, workshops, and data mining competitions on Abusive Language, Trolling, Aggression, Cyberbyullying, Misogyny detection and so forth (Zhang and Luo, 2018). For the most part, these models are trained on English text data, paying little attention to other languages. Therefore, Italian hate speech detection has been introduced within the context of EVALITA (Sanguinetti et al., 2020a).

In 2018, the EVALITA organizers presented three subtasks: In the first task, Facebook data was used to classify a message as not hateful (0) or hateful (1) and in Task 2, the same challenge was conducted on Twitter data. Task 3 asked the participants to train on the Facebook data and test on the Twitter data, and vice versa. With an F1 score of 0.82, the best performance on the Facebook task was achieved by a team that used polarity and subjectivity lexicons as well as two word-embedding lexicons as external resources together with a 2-layer BiLSTM. The same team reached the best performance for the Twitter data (F1 score of 0.79). However, systems that were cross-corpus tested performed significantly worse with an F1 score of 0.65% with the Facebook training set and 0.69% with the Twitter train data. The former score was achieved with a neural network with three hidden layers involving word embeddings that were trained on previously extracted Facebook comments; the latter was once again achieved by the team with the 2-layer BiLSTM (Cimino et al., 2018).

3 Task Description and Dataset

We participated in subtask A of HaSpeeDe – a binary classification task to predict the presence or absence of hate speech in Italian Twitter messages (Sanguinetti et al., 2020b). The training dataset provided by the task organizers consists of 6837 text samples collected from Twitter and corresponding binary labels: 1 if the text sample contains hate speech and 0 otherwise. Among the tweets, 4071 are labeled as not containing hate speech, 2766 are labeled as hate speech. Table 1 shows two examples with their labels.

id	text	hs
1940	Ma quindi solo io sono preoccupato che il terrorista stava in Italia?	0
6777	Cacciamo tutti gli immigrati visto che sono un pericolo	1

Table 1: Example Tweets from the training data

4 Experiments

To solve the task, we fine-tuned the language model ***B**idirectional **E**ncoder **R**epresentations from **T**ransformers* (*BERT*). BERT was developed

by Google and offers great possibilities not only for hate speech detection, but for all kinds of tasks that involve processing natural language (Devlin et al., 2019). Since BERT is available for multiple languages, we were interested in which version of BERT – the multilingual BERT (bert-base-multilingual-cased) or the Italian version of BERT (dbmdz/bert-base-italian-cased) (Wolf et al., 2019) – would perform best for the task at hand to determine Italian hate speech in tweets and news headlines. The multilingual BERT cased is a language model that has been trained on 104 languages whereas the latter version has been pretrained solely on Italian language.

For faster and more efficient processing while fine-tuning the model, we used Google Colab (`https://colab.research.google.com`) in all experiments as it provides free GPU. We further experimented with the training data by comparing model performance on the data as it was provided by the event organizers and after cleaning it. Leaving data as is could have several advantages: On the one hand, it can be helpful to leave in junk characters that appear in tweets as well as trailing white spaces. For instance, a tweet written in all capital letters might indicate an insult and therefore contain useful information for the classifier. On the other hand, the task at hand did not solely require hate speech detection on social media but was evaluated on newspaper articles. Therefore, the model might adapt too much to the specific style of the Twitter genre and lower classifier performance when trying to generalize to another domain (like newspaper articles where these kinds of characters do not occur). For both our runs of the final model we cleaned the data as previous test runs showed better performance.

4.1 System Description

To solve the task, we fine-tuned a BERT model. After experimenting with the different language models as described in the previous section, we found the *bert-base-italian-cased* model to be the best fit. The data was split into training and validation set during the first phase of the training. Cross-validation was used on the training set to prevent overfitting, and the validation set was used to assess how the model will generalize to unseen data. In the second training phase, the whole training data was used for training purposes.

Before experimenting with different estimators, the data was cleaned from @user-marks, trailing whitespaces, and we corrected errors like "&amp" to "&". Since BERT is an already trained language model, extensive preprocessing of the data is not unnecessary. However, we assume that some preprocessing will be useful for cross-domain evaluation. After preprocessing, the text data was tokenized by the Italian BERT tokenizer (AutoTokenizer) that splits texts into tokens. It adds special [CLS] and [SEP] tokens to mark that the sentences can now be used for classification purposes and to separate sentences so that each token within a sentence can be assigned a segment token. Afterwards, the tokens are converted into token ids using the pre-defined indices of BERT's tokenizer vocabulary. Additionally, those embeddings are also assigned attention masks that specify how much attention the system should pay to each of the words.

Since we implemented BERT with PyTorch, we used the optimization module AdamW for finetuning. Finding a good learning rate can be difficult. AdamW takes care of this issue by adapting the learning rates for different parameters which makes the training process more efficient (Kingma and Ba, 2015). Following the recommendations of the developers of AdamW, we set the learning rate to *5e-5* as default which also achieved the best results overall. Moreover, we tried various epochs, again using the recommended number of epochs, to see whether the performance of the model would improve. The best F1-score and overall accuracy was achieved with only two epochs. During each epoch the model is trained and evaluated on the validation set. The batch size was set to 16 and we set the random seed to 42 to ensure reproducibility.

Even though we are dealing with binary classification, the model makes predictions by calculating probabilities using the softmax function. Moreover, we used a threshold of 0.9% to reduce prediction errors; 90% certainty is very high when we compare the default threshold of 50% that is typically used for this purpose. However, after manually going through some of the test data, it is sometimes fairly difficult even for a human to uncover hate speech, especially for the *news* dataset.

Therefore, our goal was to produce realiable predictions. For both our runs we used the same system playing around with some of its parameters according to the results received from the first run. Therefore, our second run performs slightly better.

5 Results

When evaluating our model with the two test sets provided by the EVALITA organizers, we received the scores shown in Table 2. Our model performed 17% better on test data containing Tweets (Basile et al., 2020) compared to the news data with overall F1 macro-scores of **77.56%** (on tweets) and **60%** (on news).

The organizers provided two baseline models (see Table 3 – *most frequent class* (MFC) and *Linear SVM* with unigrams, char-grams and TF-IDF representation. Our model achieved higher scores for the news headlines and the twitter test set compared to the MFC baseline that achieved Macro-F1 scores of 38.94% and 33.66% respectively. However, our model failed to beat the baseline of the Linear SVM for the news test set which scored 62.1%. Nevertheless, it performed better on the tweets test set compared to the Linear SVM (72.12%).

Test Data	non-hate			hate		
	F1	P	R	F1	P	R
News	0.82	0.70	0.98	0.39	0.25	0.9
Tweets	0.79	0.75	0.83	0.76	0.81	0.72

Table 2: System Evaluation

Test Data	non-hate			hate		
	F1	P	R	F1	P	R
News MFC	0.78	0.64	1	0	0	0
News SVC	0.78	0.71	0.87	0.46	0.61	0.38
Tweets MFC	0.67	0.51	1	0	0	0
Tweets SVC	0.72	0.73	0.71	0.72	0.71	0.73

Table 3: Baseline Results (Basile et al., 2020)

As expected, model performance decreases in cross-corpus evaluation, especially in the news headlines test data. We assume that our model learned characteristics of the Twitter data alongside the characteristics of hate speech. Therefore, the model performs worse when applied to domains that entail different linguistic surface structures. The F1 macro-scores in Table 2 show that the scores for the two labels are evenly distributed (79% for non-hate and 76% for hate). Contrary to this, the model tested on the news data is a lot more likely to detect non-hate items (with 82%) whereas its performance on finding hate items only lies at 39%. The confusion matrices for both test sets for the second run can be seen in Table 4 and Table 5.

		Predicted	
		Positive	Negative
Actual	Positive	314	5
	Negative	136	45

Table 4: Confusion Matrix of news headlines test set

		Predicted	
		Positive	Negative
Actual	Positive	534	107
	Negative	175	447

Table 5: Confusion Matrix of tweets test set

6 Error Analysis

Identifying hate speech in Twitter data was obviously easier for our model because it had been trained on similar data. However, the model had more difficulties in making predictions on the news headlines as hints towards hate speech were much more subtle and harder to grasp. This became especially clear when we tried to identify hate speech in the test data ourselves. For the tweets test data, the use of hate speech was more obvious and direct. Another and bigger problem might have been missing context information as we were limited to the headlines, thereby missing the content of the article. Since we had difficulties identifying especially hate speech for the news headlines test data it is only reasonable that our model had similar difficulties and performed worse compared to the tweets test set. Table 6 and 7 show some examples where our system failed to detect hate speech correctly. Table 6 contains examples with upper-cased words which are used to highlight strong ideas and opinions. In this context, the upper-cased language is used to highlight the rage of the user. Therefore, our model should have been made more sensible towards the intentional use of capital letters to classify content containing hate speech more accurately. Nevertheless,

none of these examples, including Table 7 were correctly classified as hate speech.

id	text
11834	@user A me pare una scelta politica suicida puntare tutto su una battaglia sicuramente perdente in favore dell'immigrazione incontrollata...Meglio così, spariranno più velocemente!
11846	Rosarno, le case popolari? Solo agli immigrati Hanno avuto bisogno di governi non eletti, di gente imposta ad un popolo disarmato. Una volta messi li, i VIGLIACCHI hanno dato inizio alla ns fine! Se e quando si scatenerà la rabbia vera, ne farò parte!!URL
11220	I CRISTIANI ATTACCATI DAL MONDO ISLAMICO: IRAQ SIRIA SRI LANKA E ED EUROPA.E LA CHIESA DIVISA TRA DUE PAPI, BENEDETTO AUTOREVOLE RINTUZZA LA RIVOLUZIONE TRASGRESSIVA DEI COSTUMI, FRANCESCO LASCIA FARE. CRISTIANI PERSEGUITATI MA IL PROBLEMA SONO I MIGRANTI URL

Table 6: Example Tweets wrongly classified

id	text
10547	L'Europa caccia i clandestini
10130	Italia? Immigrati e sfottò: Mr Europa ci rende onore ma non fermerà l'invasione
10247	Immigrazione, la rotta dei sospetti jihadisti: in Italia su moderni gommoni

Table 7: Example News Headlines wrongly classified

7 Discussion

Our goal was to develop a system for Hate Speech Detection in Italian Twitter data. After cleaning the data, we fine-tuned a BERT model with a batch size of 16 and a learning rate of *5e-5*. Overall, our model reached an F1 score of 77.56% on the Twitter test data, and 60% on the news data. Ideas for future work include adding training data that has been collected from other sources apart from Twitter, incorporating a lexicon of hate words, such as Hurtlex (Bassignana et al., 2018), or using topic modelling techniques to extract information about topics that are likely to be involved in hate speech on social media.

References

Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro. 2020. EVALITA 2020: Overview of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Online. CEUR.org.

Elisa Bassignana, Valerio Basile, and Viviana Patti. 2018. Hurtlex: A Multilingual Lexicon of Words to Hurt. In Elena Cabrio, Alessandro Mazzei, and Fabio Tamburini, editors, *Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Torino, Italy, December 10-12, 2018*, volume 2253 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Christina Bosco, Felice Dell'Orletta, Fabio Poletto, Manuela Sanguinetti, and Maurizio Tesconi. 2018. Overview of the EVALITA 2018 Hate Speech Detection Task. *EVALITA@CLiC-it*, pages 1–9.

Andrea Cimino, Lorenzo De Mattei, and Felice Dell'Orletta. 2018. Multi-task learning in deep neural networks at EVALITA 2018. In Tommaso Caselli, Nicole Novielli, Viviana Patti, and Paolo Rosso, editors, *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Turin, Italy, December 12-13, 2018*, volume 2263 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT*, pages 4171–4186.

Karmen Erjavec and Melita Poler Kovačič. 2012. "You Don't Understand, This Is a New War!" Analysis of Hate Speech in News Web Sites' Comments. *Mass Communication and Society*.

Edel Greevy and Alan F. Smeaton. 2004. Classifying racist texts using a support vector machine. In *SIGIR 2004 - the 27th Annual International ACM SIGIR Conference, 25-29 July 2004, Sheffield, UK*.

Satyajit Kamble and Aditya Joshi. 2018. Hate speech detection from code-mixed hindi-english tweets using deep learning models.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980.

Jonathan May, Ekaterina Shutova, Aurelie Herbelot, Xiaodan Zhu, Marianna Apidianaki, and Saif M. Mohammad, editors. 2019. *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Manuela Sanguinetti, Gloria Comandini, Elisa Di Nuovo, Simona Frenda, Marco Stranisci, Cristina Bosco, Tommaso Caselli, Viviana Patti, and Irene Russo. 2020a. HaSpeeDe 2@EVALITA2020: Overview of the EVALITA 2020 Hate Speech Detection Task. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Online. CEUR.org.

Manuela Sanguinetti, Gloria Comandini, Elisa Di Nuovo, Simona Frenda, Marco Stranisci, Cristina Bosco, Tommaso Caselli, Viviana Patti, and Irene Russo. 2020b. Overview of the EVALITA 2020 Hate Speech Detection (HaSpeeDe 2) Task. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*, Online. CEUR.org.

Ellen Spertus. 1997. Smokey: Automatic Recognition of Hostile Messages. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence*, pages 1058–1065. AAAI Press.

Liriam Sponholz. 2018. *Hate Speech in den Massenmedien: Theoretische Grundlagen und empirische Umsetzung*. VS Verlag für Sozialwissenschaften.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv*, abs/1910.03771.

Ziqi Zhang and Lei Luo. 2018. Hate speech detection: A solved problem? the challenging case of long tail on twitter. *Semantic Web*, 10.