University *of Ljubljana*
Faculty *of Computer and Information Science*

# Cross-lingual offensive language identification

Gojko Hajduković, Simon Dimc

**Abstract**

Classification of hate speech in online forums is a important problem for automatic detection and monitoring of hate speech comments. Transferring of learned models from one language to another is also important as not all languages have a proper dataset for developing a classifier. In this work we train classification models on Twitter and online forum hate speech comments. We use Linear Regression, SVM and XGBoost traditional models with tf-idf and topic-modeling features as baseline and compare them to fine-tuned CroSloEngual BERT model on English data. On English data traditional methods perform approximately the same as BERT. When we try to transfer the BERT model trained on English data to Slovenian hate speech classification, we observe that additional training on Slovenian is required for good results.

**Keywords**

Text classification, Hate speech classification

*Advisors: Slavko Žitnik*

## Introduction

Text classification is one of the most important and well researched tasks in Natural Language Processing. It represents a supervised learning approach whose goal is to classify an input text into the correct category, therefore each input example gets assigned a label and as such is used in wide range of applications. Applications that exploit text classification the most are document classification, web search, information retrieval, sentiment analysis and spam detection. Text classification breaks down to different levels of scope:

1. **`Document level`** - In the document classification algorithm, classifies document as a whole into the categories
2. **`Paragraph level`** - Classifies a portion of a document, a single paragraph into the required categories.
3. **`Sentence level`** - Obtains the categories of a portion of a paragraph, a single sentence.

In this paper we focus on online hate speech classification problem. Online hate speech classification has become a well researched part of NLP and an urgent need since human supervision is unable to handle an enlarging volume of online content. We further divide classification problem based on the number of target variables to be classified. At first we classify input text at a paragraph level into binary categories, while in the later phase we classify input text on multi-class labels. In this paper we will explore two types of models for binary classification problems. First, traditional machine learning models such as `Logistic regression`, `SVM`, `Random forest` and an `Ensemble voting model` are to be employed into classifying online hate speech. The later approach will rely on `BERT`, a transformer based machine learning technique. In case of successfully validating our initial hypothesis of correctly classifying hate speech we will introduce a Slovenian labeled hate speech data set consisting of Twitter tweets and news website comments. Additionally, we will transfer the knowledge from pre-trained multi-lingual models using one-shot transfer and evaluate introduced Slovenian data set.

## Related work

With the enlarged amount of online content, the need and the and focus of research has shifted towards online text classification, with a particular focus on content moderation. Within the early stages of text moderation number of methods utilizing shallow approaches have been proposed. Dao et al. [1] introduced methods wich use feature vectors constructed with BoW language model. The main shortcomings of the introduced methods is that it suffers from data sparsity and that it typically leads to many false positives. Other approaches for feature spaces comprising of TF-IDF and character n-grams have been introduced. Waseem et al. [2] explore the traditional, feature-engineered machine learning models such as

Logistic Regression and SVM for automatic content moderation.

In order to tackle the problem of data sparsity, number of approaches such as Word2Vec[3] and GloVe [4] have been introduced for better representation of words and sentences. These methods outshine sparse feature vectors since similar words are located closer together in the latent space and dense word representation produce better encoding of such features. Recent studies show that Neural Network based approaches are superior compared to the traditional machine learning methods. Advantage of neural network approaches lays in it's capability to model larger sequences of text. Sutskever et al. [5] show that gated RNNs are capable of capturing long term dependencies in text leading to a better results in content classification.

The main shortcoming of NN based approaches lays in it's inability to catch global context of the words. A breakthrough in the field of NLP happened with introduction of BERT [6]. Bidirectional Encoder Representations from Transformers - `BERT` represents a SOTA approach in NLP. Compared to the other dense word representations BERT takes into account a larger left and right semantic space thus better modelling words with respect to the context around it. There are a number of pre-trained BERT models which can be fine tuned on various downstream NLP tasks. Furthermore, there are a number of available multilingual pre-trained BERT models, thus allowing a transfer of model knowledge to the less resourced languages.

## Methods

In this section we introduce used methods, data sets and preprocessing steps.

### Data

As aforementioned, we have subdivided speech classification task into two categories, a binary and a multi class label text classification. For both task categories we have used english and Slovenian datasets. For binary task we have used four publicly available english data sets that have been further processed and adopted in order to be combined into one uniform binary class data set. The dataset consists of hate-speech labelled with **1** while non-hate speech is annotated with **0**. While, for mutliclass task we have used two english data sets which are then processed and adopted into one uniform multiclass dataset. Both tasks were tested on one Slovenian dataset with additional hand collected comments from web.

### Binary english datasets

#### Dataset 1

First dataset represents a 1528 Fox news comments [7] which are manually classified into hate and non-hate categories. publicly available at url. The original data set is a json file where each comment has an additional metadata and previous contextual comments. All the metadata have been removed, preserving only the text and the binary label. The dataset is highly imbalanced consisting of 435 hateful and 1093 non-hateful comments.

#### Dataset 2

Second dataset represents a 26581 comments [8] from Reddit posts which are manually classified into hate and non-hate categories, publicly available at url.The original data set is a csv file where each comment has an additional metadata and previous contextual comments. All the metadata have been removed, preserving only the text and the binary label. The dataset is highly imbalanced consisting of 5257 hateful and 17052 non-hateful comments .

#### Dataset 3

Third dataset represents a 38018 comments [8] from Gab posts which are manually classified into hate and non-hate categories, publicly available at url.The original data set is a csv file where each comment has an additional metadata and previous contextual comments. All the metadata have been removed, preserving only the text and the binary label. Dataset consists of 14614 hateful comments and 19162 non-hateful comments.

#### Dataset 4

Fourth dataset represents a 10945 commments [9] from a white supremacists forum which are manually classified into hate and non-hate categories publicly available at url. The original dataset consists of textual documents which hold each comment and a csv file marking whether a document is classified as a hateful or non-hateful one. The dataset has been processed into one csv shaped file consisting of comments and label categories. The dataset is highly imbalanced consisting of 1196 hateful comments and 9748 non-hateful comments.

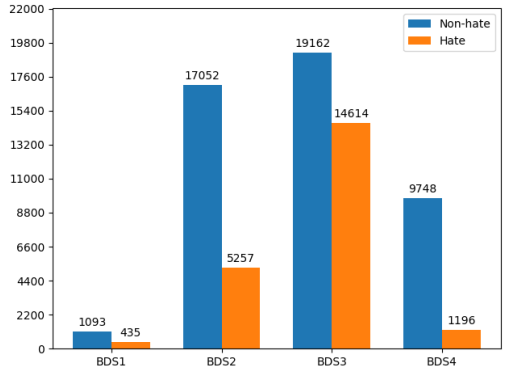


**Figure 1.** Distribution of classes in English binary datasets.

### Multiclass english datasets

#### Dataset 1

First multiclass dataset consists of 10236 comments [10] from Twitter posts which are classified into 7 categories showed in table 1. The dataset tweet ids are publicly available at url. The dataset was collected by searching the Twitter's 1% public stream from January 2016 to July 2017 for tweets containing hate keywords. Those tweets were then additionally

checked with Perspective API developed by Jigsaw and the Google Counter-Abuse technology team. The API was used for checking tweet toxicity and attack on commenter property. The quality of dataset was tested by human annotators, which labeled 97.8% of tweets as hate speech with 92.8% agreement percentage.

### Dataset 2

Second multiclass dataset consists of 25000 comments [11] from Twitter posts which are classified into 5 categories showed in table 1. The dataset was collected by searching the Twitter from December 2016 to January 2017 for tweets containing at least one hate keyword. After that additional 3 human annotators were used to annotate the presence of hate in fetched tweets, with agreement percentage of 80.6%. Of those 25000 tweets, only 2958 contain hate speech.

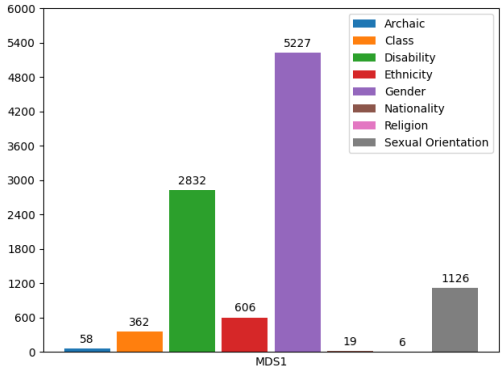


**Figure 2.** Distribution of classes in the 1. English multiclass dataset.

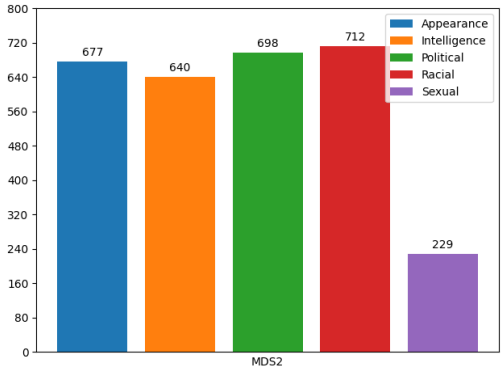


**Figure 3.** Distribution of classes in the 2. English multiclass dataset.

## Slovenian datasets

### Dataset 1

First dataset consists of Slovenian comments from two news websites 24ur.com and Nova24.si. We manually collected and annotated 141 comments into 6 hate classes: non hate, racism, sexual, intelligence, appearance, and other.

### Dataset 2

Second dataset consists of 60000 Slovenian tweets [12]. Tweets are classified into 4 hate speech types: appropriate, inappropriate, offensive, and violent. Tweets are also classified into 11 hate speech targets: racism, migrants, islamophobia, antisemitism, religion, homophobia, sexism, ideology, media, politics, individual, and other. Tweets were fetched from Twitter between December 2017 and August 2020. Tweets were annotated by ten annotators, with agreement percentage of 60%. We sample 673 violent and 3000 appropriate, inappropriate, and offensive (1000 each) tweets, for our needs in this project.

## Datasets combining

We combine all English datasets into one final binary and one final multiclass dataset. Those two datasets are then used for training the classification models. The final binary dataset is obtained by combining binary datasets and multiclass datasets, which were converted to binary by grouping all hate classes into one class. The final multiclass dataset is obtained by mapping different classes from each multiclass dataset into a predefined 6 final classes, and additional non-hate classes from binary datasets. The class mapping rule is shown in Table 1.

Slovenian datasets were also combined into one binary and one multiclass dataset. Binary dataset is obtained by merging all hate classes into one hate class, and combining it with non-hate classes. Multiclass dataset is obtained by merging dataset 1, which was already collected with the correct classes, and mapped dataset 2. Class mapping is shown in Table 2.

The final classes for binary dataset: Non-hate (0), Hate (1). The final classes for multiclass dataset: Non-hate (0), Racism (1), Sexual (2), Intelligence (3), Appearance (4), Other (5).

**Table 1.** English multiclass datasets classes (left columns) and mapping rule for creating the final dataset of 6 classes.

| DS1 | Final DS | DS2 | Final DS |
|---|---|---|---|
| Archaic | Other | Racial | Racism |
| Ethnicity | Racism | Appearance | Appearance |
| Nationality | Racism | Intelligence | Intelligence |
| Class | Intelligence | Political | Other |
| Disability | Intelligence | Sexual | Sexual |
| Gender | Sexual | | |
| Sex | Sexual | | |
| Religion | Other | | |

## Preprocessing

Preprocessing is applied to all the instances of each dataset that the classifier could predict class for. Preprocessing is conducted through eight steps. Since our datasets are collected mainly from the social media or news media comments there are high number of occurences of emoticons. With step 1 all the emoticons are removed. Same reasoning is behind

**Table 2.** Slovenian multiclass 2. dataset classes (left columns) and mapping rule for creating the final dataset of 6 classes.

| DS2 | Final DS |
|---|---|
| Non hate | Non hate |
| Racism | Racism |
| Migrants | Other |
| Islamophobia | Other |
| Antisemitism | Racism |
| Religion | Other |
| Homophobia | Sexual |
| Sexism | Sexual |
| Ideology | Other |
| Media | Other |
| Politics | Other |
| Individual | Other |
| Other | Other |

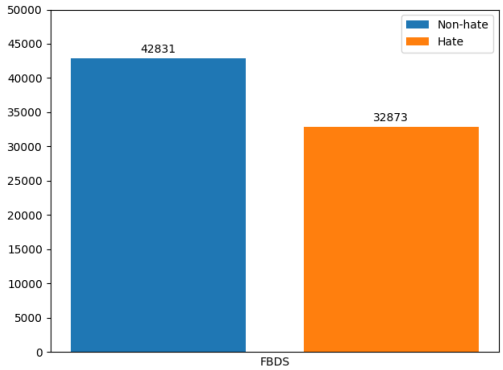


**Figure 4.** Distribution of classes in the combined English binary dataset.

step 2 and 3 where all twitter urls, hashtags and @name mentions are removed. Steps 4 and 5 remove all non-alphabetical characters and lower down all words. Step 6 tokenizes each input comment with nltk function word_tokenize which firsts tokenizes text into sentences than it further tokeniizes it into tokens. Lemmatization is performed in step 7 in order to get word in a canonical form and get higher frequency of similar words. Nltk WordNetLemmatizer is used for converting a word into lemmas. Additionaly, step 8 combines english stop words removal as well as filtering out words of length 1. Lemmatization and stop words removing is preformed only before training on traditional methods. On BERT training, no lemmatization and stop words removing is applied because that may remove context from the text, which may lower the performance of fine-tuned BERT models.

## Features

For the feature extraction in our baseline model we have used sparse-word embeddings. Moreover, we have combined sparse-word embedding language models such as `BoW` and `tf-idf` with topic-modelling features constructed from the given data sets with Latent Dirichlet Allocation - `LDA`.

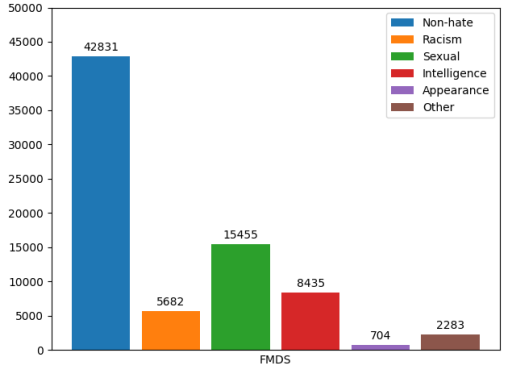


**Figure 5.** Distribution of classes in the combined English multiclass dataset.

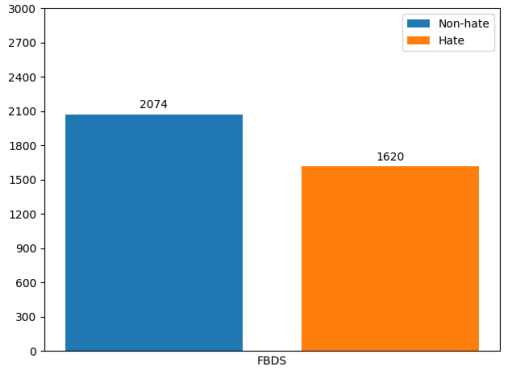


**Figure 6.** Distribution of classes in the combined Slovenian binary dataset.

First, we have extracted ten topic categories from the data and one-hot encoded each comment to the belonging category which are then combined with BoW or tf-idf features as aforementioned. With both approaches we have used unigrams and bigrams while using Sklearn implementations of `CountVectorizer` and `TfidfVectorizer` respectively, with using `gensim` implementation of `LDA`.

## Models

Figures 5 and 7 show that multiclass datasets are highly imbalanced which can lead to classification algorithms having low accuracy towards the minority classes. In order to tackle the problem of imbalanced datasets we have tested undersampling and over-sampling on training data as well as adjusting class weights.

### Traditional ML models

For the baseline models we have selected SKlearn implementation of Logistic regression, Support-Vector Machine and and Ensemble voting classifier - XGBoost.

**Logistic regression** - We have used sklearn implementation of Logistic Regression classifier. For the model parameter tuning the GridSearchCV is used to choose the best model for predicting classes of each dataset as well as predicting classes for a dataset that is a concatenation of all datasets.

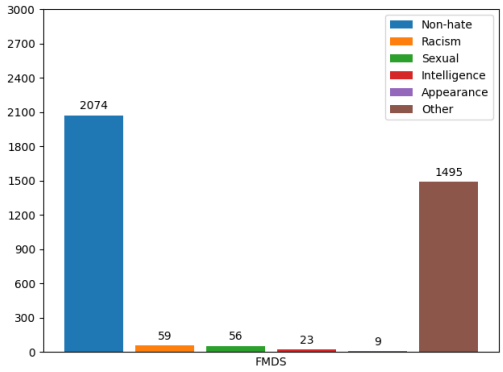


**Figure 7.** Distribution of classes in the combined Slovenian multiclass dataset.

Best performant model is using solver=lbfgs ,max_iter=5000, class_weight=balanced, C=1 with penalty=L2.

**Support-Vector Machine** - We have used sklearn implementation of linear Support-Vector Machine classifier, LinearSVC. For the model parameter tuning the GridSearchCV is used to choose the best model for predicting classes of each dataset as well as predicting classes for a dataset that is a concatenation of all datasets. Best performant model is using kernel=rbf, class_weight=balanced with C=1.

### Ensemble models

**XGBoost** - We have used XGBoost ensemble model for classifying comments data. The boosting algorithm is used with a learning_rate = 0.05, initial n_estimators=300, and max_depth = 5.

### Transformer ML models

**BERT** - In order to utilize BERT for text classification we have used BertForSequenceClassification model with classification head from https://huggingface.co/transformers/ library. In the experiments classification head is based on either mBert or CroSloEngualBert. Both models are pre-trained on a large corpora of multilingual data and are suitable for cross-language knowledge transfer. The model fine-tuning is powered by https://pytorch.org/ framework.

## Experiments and Results

### Traditional ML models

All traditional methods are trained and evaluated on the combined English datasets for binary and multiclass tasks. The process of training and testing is conducted via stratified K-fold validation with $K = 5$ on feature sets composed from tfidf features and topics features, obtained with LDA from 5 passes over the data. In the each fold, the data is split on training and testing data with $80 - 20\%$ ratio.

As stated above, our multiclass data is highly imbalanced. Since in our data we have a dominant majority class, which is non-hate class, a natural effect would be that trained traditional models would be biased towards minority class, thus the model would have a low accuracy on minority class. In order to stimulate the models to focus and reduce the error on minority class, we adjust class weights, giving more weightage to the minority class in the cost function. As before mentioned, due to the data being imbalanced model would mostly predict the majority class, thus having a high overall accuracy while having poor results on minority class. Therefore, for evaluation metric we chose weighted average f1-score which presents the weighted average of precision and recall.

Table 3 shows models evaluation results on binary and multiclass tasks. All methods perform good on binary tasks, achieving weighted average f1-score of 0.90. While for multiclass task, the XGBoost model performs the best with weighted average f1-score of 0.86, but not by a lot. Reasons for good binary performance may be in good defined binary dataset, which is not very unbalanced, in comparison to multiclass dataset, which is very unbalanced and doesn't have a lot of data for hate classes, except Sexual class.

**Table 3.** Performance of traditional models, on binary and multiclass tasks, on combined datasets. Precision, recall and f1-score are weighted.

| Model & data | precision | recall | f1-score |
|---|---|---|---|
| LR eng binary tfidf+topic | 0.91 | 0.90 | 0.90 |
| LR eng multi tfidf+topic | 0.86 | 0.84 | 0.85 |
| SVM eng binary tfidf+topic | 0.90 | 0.90 | 0.90 |
| SVM eng multi tfidf+topic | 0.84 | 0.84 | 0.84 |
| XGBoost eng binary tfidf+topic | 0.91 | 0.90 | 0.90 |
| XGBoost eng multi tfidf+topic | 0.86 | 0.87 | 0.86 |

### Transformer ML models

#### BERT

For BERT fine-tuning we used the CroSloEng pretrained model with Transformers library and BertForSequenceClassification architecture, which adds a single classification linear layer on top of BERT arhitecture. We set the max length of BERT input text to 512. All the text longer that that is truncated. We then prepared our dataset for training with passing it into a BertTokenizer, which uses WordPiece algorithm for tokenization. First the text is tokenized and special tokens are added. For marking the task as classification [CLS], for splitting the sentences [SEP], for padding the text if its shorten that max input text length [PAD], and for marking if the token is not in the pre-trained vocabulary [UNK]. If the token is not in pre-trained vocabulary, the token is split into several

subwords. After tokenization all the tokens are converted into ids according to the token position in vocabulary. Additional attention mask token is computed to tell the model which token should be attended to.

Dataset is then split into 80%-20% training and testing sets. And training set is additionally split into 90%-10% training and validation sets, for tracking the training loss and checking for overfitting.

Adam with weight decay fix optimizer is used for training, with parameters: learning rate 5e-5, epsilon 1e-6, betas (0.9, 0.999), no weight decay and with bias correction. Learning rate is set to linearly decrease from initial learning rate to 0 during the training.

As the BERT model is quite large and we use the max text length, we had to freeze the embeddings and first 8 encoder layers in BERT. This gives us 4 encoding BERT layers and 1 classification layer for training. The batch size during training was 20 and we trained each model for 3 epochs, and take the model where training loss and validation loss come the closest together.

Table 4 shows the performance of the fine-tuned CroSloEng BERT model on different training and testing scenarios. When training on English data and testing on English data, the results are very good, better than the traditional models. Fine-tuned BERT model achieves weighted average f1-score 0.91 for binary and 0.88 for multiclass. But if we consider the training time it took to get to this results, the traditional methods still perform quite good and are acceptable solution to classificating hate speech.

Traditional methods fail when we try to transfer the learned model to another language. As CroSloEngual BERT was trained on English and Slovenian language, we can use this to try to get a Slovenian hate speech classifier, even though we don't have a lot of training data from Slovenian language.

When we test the English fine-tuned BERT model on Slovenian data, we get poor results. f1-scores of 0.42 for binary and 0.40 for multiclass. Only fine-tuning with English data seems to not be enough. So we additionally fine-tune on Slovenian data. We split Slovenian datasets into 80%-20% training and testing and use the same training parameters as with English training. We fine-tune both binary and multiclass models. When we test the Slovenian data on additionally trained models, we get better results than before. The f1-scores are now 0.71 for binary and 0.67 for multiclass.

With the distribution of classes in Slovenian binary dataset, which is quite balanced, we may say that the performance of the binary model is quite representative and good, for the amount of Slovenian data we have. But for multiclass task, we can't really say the model learned anything as the distribution of classes is basically the same as in binary dataset, with additional 4 hate classes, which don't have enough data for useful training. So the model could only learn to classify non-hate class and other hate class. Additional Slovenian multiclass data is required for good classification on the multiclass task.

**Table 4.** Performance of BERT models, on binary and multiclass tasks, on combined datasets. Precision, recall and f1-score are weighted.

| Model & data | precision | recall | f1-score |
|---|---|---|---|
| BERT binary trained on eng tested on eng | 0.91 | 0.91 | 0.91 |
| BERT multi trained on eng tested on eng | 0.88 | 0.88 | 0.88 |
| BERT binary trained on eng tested on slo | 0.58 | 0.56 | 0.42 |
| BERT multi trained on eng tested on slo | 0.31 | 0.55 | 0.40 |
| BERT binary trained on eng-slo tested on slo | 0.71 | 0.70 | 0.71 |
| BERT multi trained on eng-slo tested on slo | 0.68 | 0.68 | 0.67 |

## Conclusion

We train traditional Linear Regression, SVM and XGBoost models on classifying binary and multiclass hate speech from Twitter and online forums. All three methods perform approximately the same with weighted f1-scores of 0.90 for binary and 0.86 for multiclass. Then we fine-tune the CroSloEngual BERT model on the same tasks. We test BERT models on English and Slovenian data, with and without additional Slovenian training. We observe that traditional methods still perform good in comparison to BERT, when we consider the training effort. BERT tested on English data achieves weighted f1-scores of 0.91 for binary and 0.88 for multiclass. When we compare Slovenian data testing on BERT, we observe that additional Slovenian data training is needed for good results (weighted f1-scores of 0.71 for binary and 0.67 for multiclass), and that no additional Slovenian training gives poor results (weighted f1-scores of 0.42 for binary and 0.40 for multiclass).

## References

[1] I Kwok and Y Wang. Locate the hate: Detecting tweets against black. *AAAI*, 2013.

[2] Z Waseem and D Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. *NAACL-HLT, pages 88–93*, 2016.

[3] T Mikolov, I Sutskever, K Chen, G Corrado, and J Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems, pages 3111-3119*, 2013.

[4] J Pennington, R Socher, and C D Manning. Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing, pages 1532-1543*, 2014.

[5] I Sutskever, O Vinyals, and Q V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems, pp. 3104–3112*, 2014.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, 2018.

[7] L Gao and R Huang. Detecting online hate speech using context aware models. *ArXiv*, 2017.

[8] J Qian, A Bethke, Y Liu, E Belding, and W Y Wang. A benchmark dataset for learning to intervene in online hate speech. *ArXiv*, 2019.

[9] O de Gibert, N Perez, A García-Pablos, and M Cuadros. Hate Speech Dataset from a White Supremacy Forum. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 11–20, Brussels, Belgium, October 2018. Association for Computational Linguistics.

[10] M ElSherief, S Nilizadeh, D Nguyen, G Vigna, and E Belding. Peer to peer hate: Hate speech instigators and their targets. *Proceedings of the Twelfth International AAAI Conference on Web and Social Media, pp.52-61*, 2018.

[11] M Rezvan, S Shekarpour, L Balasuriya, K Thirunarayan, V L Shalin, and A Sheth. A quality type-aware annotated corpus and lexicon for harassment research. *ArXiv*, 2018.

[12] P Kralj Novak, I Mozetič, and N Ljubešić. Slovenian twitter hate speech dataset IMSyPP-sl, 2021. Slovenian language resource repository CLARIN.SI.