Training multilabel classifiers for hate speech and target analysis on German offensive language in Tweets

Hate speech and target classification

C. K. Dietze, M. Führer, S. Grambau, F. Härtwig, Y. A. Hujjon, A. A. Pick, M. Pinno, M. Strauß
General and Digital Forensics, Submission Software Project Forensic Tools

04.01.2023

Abstract

In recent years, in particular, hate speech detection and the prosecution of offences committed on social networks have become more essential. Networks such as Twitter, Facebook and Instagram developed into platforms where radical ideologies and criticism are disseminated. Various approaches have been tested so far for the automated evaluation of such postings. In this scientific submission, we compare the performance of a selected number of classifiers trained on a training set consisting of $32,374$ sets. With our approach to feature engineering, we are combining the latest models in fields of sentiment analysis, dictionary approaches and simple linguistic features to enrich our training dataset for better classification results of the task given labels `'hate speech'` and `'target'`. Ultimately, we compare the results of each classifier for unbalanced and balanced performance and examine which classifiers are best suited for the subtasks of hate speech and target entities detection.

Keywords: machine learning, model comparison, text mining, sentiment analysis.

1 Introduction

Nowadays, the expression of hate on the internet is a common occurrence, which is steadily increasing due to advancing digitalisation. Social Networks are often platforms that are misused to humiliate, verbally assault or insult people. Using Twitter as an example, we analysed this phenomenon by looking at the occurrence of hate speech in Tweets. The collection of rules and guidelines known as 'Twitter Rules' prohibits such behaviour. It explains the category of 'Hateful conduct' with the following description: 'You may not promote violence against, threaten, or harass other people on the basis of race, ethnicity, national origin, caste, sexual orientation, gender, gender identity, religious affiliation, age, disability, or serious disease' [1]. Regarding this important topic, the paper addresses hate speech and its algorithmic detection as well as the classification in relation to individual Tweets as part of a large dataset. The following sections will highlight methodologies that have been used to help with this concern regarding the topic of hate speech detection.

2 Task statement

The objective of the software project is to solve the following two subtasks.

Subtask a: Hate speech detection. The aim is to use the development dataset consisting of Tweets to train a classifier that can decide whether a message contains hate speech or not. The classifier will be evaluated using k-fold cross validation.

Subtask b: Target detection. The aim is to use the development dataset consisting of Tweets to train a classifier that can decide whether a message is undirected or directed at a group or individual. The classifier will be evaluated using k-fold cross validation.

3 Related work

Hate speech detection. Hate speech detection is an already well researched field. The selection of classifiers and useful features with n-grams [2], as well as the application of stemmers for bigram, unigram and trigram features using TFIDF [3] and dictionary approaches, as in Tulkens, Hilte, Lodewyckx, *et al.* [4], have already been tested. Proper preprocessing is a recurring topic here. According to [5], the assignment of sentiments in preprocessing is effective and makes it possible to weight the texts in advance via a sentiment scoring. For the German language the use of such methods have not yet been sufficiently scientifically investigated. According to [6], it is popular to carry out such sentiment scoring in sentiment analysis in English; with the German adaptation GerVADER, this has also been tested with German texts.

Target entities detection. Compared to hate speech detection, the detection of targeted entities required by the task section 2 has not yet been sufficiently researched. So far, these labels have only been used in a publication in the field of German language by Demus, Pitz, Schütz, *et al.* [7] and determined by means of classification processes.

Classifiers & baseline models. Scientifically well researched classifiers in the field of hate speech detection are numerous. One classifier that is often used as a reference is the methodology of Support Vector Machines. These algorithms can be used as a pre-training algorithm [8]. In addition, there are simple baseline models, such as Naive Bayes and Decision Trees [9], as well as Random Forest and Neural Networks in which the features of the texts are encoded as vectors [10]. Salles, Gonçalves, Rodrigues, *et al.* [11] propose a lazy version of Random Forest to solve the problem of overfitting that usually occurs in complex trees generated from noisy data. They use nearest neighbours to classify and obtain training data. Their approach proved to be the best performing in a series of experiments and seems to be a great alternative for automatically solving text classification compared to state-of-the-art classifiers.

4 Dataset

This paper uses the given development dataset of Tweets with textual annotated hate speech and target entities as a baseline.

Further, the dataset is enriched with other German annotated data such as GermEval-2018 Corpus (DE) [12], GermEval-2019 Corpus (DE) [13], HASOC 19 [14] and translated Twitter Toxic Comments corpus [15]. In total, the performance of the baseline dataset and the enriched dataset is compared.

4.1 Dataset structure

The given development dataset consists of 'c_id', 'c_text', 'date', 'author_id', 'like_count', 'quote_count', 'retweet_count', 'reply_count', 'hate speech' and 'toxicity'.

4.2 Annotations

The annotation guideline of the dataset we used was designed very comprehensively, resulting in an abundance of important as well as neutral aspects which form the foundation for our paper. Since only part of the annotation guideline deals with our shared task, there are of course aspects such as toxicity, which were of no use to our work. The annotation of the present dataset was carried out manually by six students of the University of Applied Sciences Mittweida in cooperation with the University of Applied Sciences Darmstadt. The annotation tool was developed for another project by a member of the University of Applied Sciences Mittweida and was adapted for the work on this dataset. For optimal classification it was necessary to add the additional labels 'Hate speech', 'Toxicity' and 'Target' to the dataset. In order to evaluate these categories properly, there were priorities that were emphasised in the process of the annotation. The primary question was to determine how incomprehensible a comment was, which can occure due to a lack of context. Even if there was a problem of comprehension, the aim was to annotate everything that could be possibly annotated. The focus on the sentiment provides information about the author's state of mind and was divided into three categories where a regular sentiment analysis score could be determined with the labels 'Positive', 'Neutral' and 'Negative'. The Tweets were also investigated for criminal relevance. It can be assumed that as soon as the Tweet has violated a paragraph of the German Criminal Code, criminal relevance is fulfilled. In the item expression, it is determined whether the content of the comment was expressed explicitly or implicitly, which allows a conclusion to be made on linguistic features such as irony. In addition, there were extremism and danger, which were simultaneously responsible for evaluating whether extreme, radical statements in relation to political or religious views were portrayed. This was followed by an examination regarding if a person was seriously threatened or whether there were active calls for violent activity. All these factors were included in the annotation process in order to be finally able to correctly classify the Tweets with the labels 'Hate speech', 'Toxicity' and 'Target'.

4.3 Data exploration

In the following section, the data exploration analysed the characteristics of the data including distribution of subtask labels, timeline analysis and frequent words.

4.3.1 Distribution of labels

The development dataset contains 8,132 data records.

Thereby, distribution of the labels is unbalanced for both hate speech labels and target entities are shown in Figure 1.

LABEL: 'HATE SPEECH'	
hate speech	1,173
non-hate speech	6,959
TOTAL	**8,132**

LABEL: 'TARGET'	
person	2,865
group	3,213
public	1,283
NAs	771
TOTAL	**8,132**

Figure 1: Exploratory data analyses for the labled distributions 'Hate speech' and 'Target'

4.3.2 Timeline analysis

For data exploration, a timeline analysis was performed for the collected data to achieve a better identification of certain patterns. Looking at the temporal distribution of the Tweets, peaks in Tweets can be identified. Particularly noticeable are the periods from 04/19 to 05/10, as well as from 06/13 to 06/20.

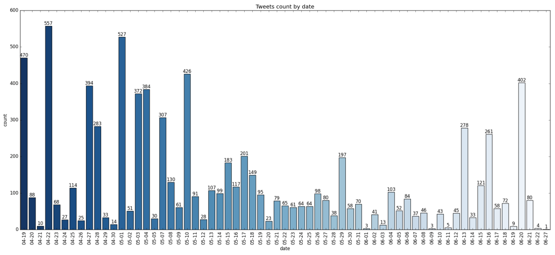

Figure 2: Timeline analysis, Tweet counts by date

4.3.3 Frequent words

For further processing of the data, in particular the text data, a frequency analysis of the occurring terms was carried out. The frequently occurring stop words of the German language are conspicuous here, as are certain topic words of the text collection, such as 'Querdenker' or 'AfD' (*'Sie': 1104, 'Ich': 847, 'Querdenker': 537, 'Menschen': 428, 'Es': 420, 'AfD': 403*). These findings were used in the process of data cleaning.

Sample	Counts	Frequency	Approx. procentage
`Sie`	1104	0.135759960649287	13.58%
`Ich`	847	0.104156419085096	10.42%
`Querdenker`	537	0.066035415641909	6.60%
`Menschen`	428	0.052631578947368	5.26%
`Es`	420	0.051647811116576	5.16%
`AfD`	403	0.049557304476144	4.96%

Table 1: Most frequent terms after text cleaning

4.4 Additional datasets

Additional datasets were used to balance the datasets and to improve the accuracy of the classification. The datasets were adjusted to the given development dataset. The following datasets were used:

–**GERMEVAL 2018** [12] To balance the label distribution for hate speech in this corpus, 5,008 datasets labelled as ‘OFFENSE’ were added to the development dataset under formatting as binary ‘Hate speech’ label.

–**GERMEVAL 2019** [13] For this corpus, records labelled as ‘OFFENSE’ for hate speech. 1,957 records were added to the development dataset under formatting as a binary ‘Hate speech’ label to balance the label distribution.

–**HASOC 2019** [14] For this corpus, to balance the label distribution for hate speech, 407 records labelled as ‘HOF’ and 3,412 labelled as ‘NOT’ were added to the development dataset under formatting as a binary ‘Hate speech’ label.

–**Toxic Comments Corpus** [15] First, the Toxic Comments Corpus was restricted to only those records labelled as ‘Toxic’. These 13,458 records were then translated from the original (English) into German. In order to balance the label distribution for hate speech, labelled records were added to the development dataset and formatted as a binary ‘Hate speech’ label.

By taking the balancing of the dataset into account, 32,374 data records are passed to the classification process in the training. That’s an increase of four times the original development dataset containing only 8,132 sets. The additional and newly created dataset with the labels ‘Hate speech’ is saved and available as a new collection as a corpus at GitHub[1].

5 Classification approaches

5.1 Baseline model

The underlying model process is the same for both classification problems. After the data import, which consists of preprocessing the textual data and then transferring the tokenized texts to a TFIDF transformer, the transformer transfers them as the main feature to the classification models for training via a term matrix. Subsequently, all models are trained on the TFIDF matrices and then basically evaluated for accuracy on train and test sets. To cross-evaluate the models, a 5-fold cross-validation is used to validate the models and compare the cross-validation performance. The best model is then used to label the evaluation dataset.

5.2 Hate speech method

The baseline model is adapted for the hate speech classification subtask in the way that additional datasets are prepared and embedded for the enlargement of the development dataset. In addition, further features were selected for the hate speech classification in order to improve the model performance. These features, as described in section 6 Feature engineering, are included and tested in the training process. Afterwards, the process of hate speech classification follows the baseline model process.

5.3 Target method

The target entities classification subtask is divided in two different classification approaches. The model processes are explained below.

5.3.1 Target binary classification approach

The simplistic binary classification approach, in its procedure and training, is similar to the baseline model process. After preparing the data, the categorical labels of the target entities are converted into binary labels (here: numerical labels 0, 1, 2). This adaptation makes it possible to proceed in the hate speech Classification subtask. In the end, the best model for the labelling is selected.

5.3.2 Target concatenated classification approach

As for the more complex approach of the Linked Classification Process, the labels of the target entities are considered linked. In order to obtain the relations between the individual labels, so-called ‘classification chains’ are considered in the classification process. This should make it possible to neglect the binary relevance of non-visible relations between ‘person’, ‘public’ and ‘group’. Here, the preprocessed data is transferred to the classifier chains. The models are then trained and evaluated. The best model is then used for labelling.

6 Feature engineering

For model training and to improve model performance, other features are aggregated from the data in addition to the TFIDF matrices as the main feature. A certain number of features from the areas of surface, linguistic and sentiment were selected in order to avoid overfitting and other negative influences. These are described below in their aggregation and application.

6.1 Surface features

The surface features available in the development dataset are on the one hand meta information about the individual Tweets already contained in the dataset, on the other hand aggregated features from the basic feature `‘c_text’`, such as the length of the text and the number of emojis, punctuation marks or CapsLock terms. This meta-information contained in the dataset cannot be used for surface features because the dimensionality of the information is not given due to the enlargement of the dataset. Therefore, the pre-selection of surface features is limited to those extracted from `‘c_text’`.

6.2 Linguistic features

For the feature creation of the linguistic features, a dictionary methodology was used. In so-called profanity scoring, certain terms annotated as profane were taken from a dictionary and compared with the tokens from Tweets of the dataset. If a term is included in the dictionary, it will be assigned a meaning via a score. Terms that are not included are not scored. First, a collection from Davidson et al. [3] was used. For better results, the usage of a bigger dictionary was preferred. Therefore, an annotated collection from Surge AI [16] was utilised for the underlying dictionary. This was supplemented with additional terms from the corpus used here.

6.3 Sentiment features

The underlying model for the sentiment analysis was VADER (Valence Aware Dictionary and sEntiment Reasoner) [17]. As a pre-trained model using rule-based values tuned to social media sentiments, the Tweets previously translated into English were labelled. The results of the Vader sentiment analysis were incorporated into the training as four different features consisting

[1] German Hate Speech Corpus, `https://github.com/HSMW-TargetGruppe4/Softwareprojekt/tree/main/data`

of ‘neg’, ‘pos’, ‘neu’ and ‘comp’. A German approach using GerVADER [6] was discarded after evaluating the resulting scores and pretraining size, comparing German and English.

7 Results & performance analysis

The following results were achieved using the given resources. The results are subdivided into subtasks and then compared. In the end, the best-performing model is selected.

7.1 Subtask hate speech classification

The classification process described in 5 Classification approaches, subsection 5.2 hate speech, was applied in the hate speech classification. Thereby, 32,374 datasets were considered in the training and testing of the classifiers to be checked and compared. The root feature was `‘c_text’` which was preprocessed in text preprocessing. Additionally, the features `‘c(c_text)’`, `‘c(emojis)’`, `‘sentiment_neg’`, `‘sentiment_pos’`, `‘sentiment_neu’`, `‘sentiment_comp’` and `‘profanity_score’` were included in the classifications.

First, the effect of dataset enlargement and balancing was investigated, and the following results were observed. As shown in Figure 3, the cross-validation score increases at the first balancing, then returns to normal after the dataset size is increased. For best results on the corpus, the dataset with 32,734 sets is used and then balanced.

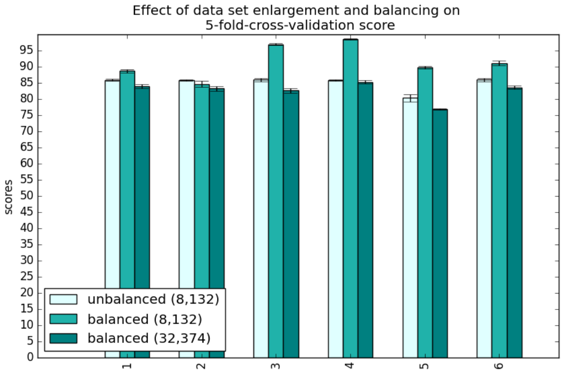

Figure 3: Effect of dataset enlargement and balancing the hate speech data corpus in hate speech detection, 5-fold-cv-scores of LogisticRegression(1), NaiveBayes(2), RandomForestClassifier(3), SVM (4), DecisionTreeClassifier (5), RigdeClassifier(6).

The classification results of all six methods used for the classification are shown in Table 2. It is remarkable that the performance indicators of the used methods do not significantly deviate and include a score range of approx. 0.790 – 0.850 (f1-score) for the hate speech classification and a score range of approx. 0.730 – 0.820 (f1-score) for the non-hate speech classification.

After the evaluation using a 5-fold cross-validation, the results of this were consolidated in Table 3. Both tables display the classification performance for all six models. The DecisionTreeClassifier performed the weakest out of all models for hate speech classification, achieving a cv-score (mean) of 76.83%. The most effective model for hate speech classification of all models with a cv-score (mean) of 85.29% is SVC.

	precision	**recall**	**f1-score**
LogisticRegression			
hate speech	0.911	0.786	0.844
non-hate speech	0.755	0.896	0.820
NaiveBayes			
hate speech	0.911	0.784	0.843
non-hate speech	0.753	0.895	0.818
RandomForestClassifier			
hate speech	0.888	0.802	0.843
non-hate speech	0.763	0.863	0.810
SVM			
hate speech	0.933	0.795	0.859
non-hate speech	0.768	0.922	0.838
DecisionTreeClassifier			
hate speech	0.811	0.777	0.794
non-hate speech	0.713	0.755	0.733
RidgeClassifier			
hate speech	0.895	0.800	0.845
non-hate speech	0.763	0.873	0.814

Table 2: Results of the hate speech models with the extended hate speech dataset

When considering specifically the label classification scores, the DecisionTreeClassifier performed the weakest of all models for hate speech classification, achieving a f1-score of 0.794. The best model for hate speech classification of all models with a f1-score of 0.859 is the SVC. The classifiers NaiveBayes(0.843), RandomForestClassifier (0.843), LogisticRegression (0.844) and RidgeClassifier (0.845) rank in between, spanning a not significantly different interval of 0.843 (f1-score) - 0.845 (f1-score). The results are very similar for the non-hate speech classification. Similarly, the least efficient model is the DecisionTreeClassifier with a f1-score of 0.733 and the best model, the SVC, with a f1-score of 0.838. The classifiers RandomForestClassifier (0.810), RidgeClassifier (0.814), NaiveBayes (0.818) and LogisticRegression (0.820) score in between.

	mean cross-validation scores	**standard deviation cross-validation scores**
LogisticRegression	0.840421	0.006270
NaiveBayes	0.833045	0.007396
RandomForestClassifier	0.826443	0.007681
SVC	0.852941	0.005337
DecisionTreeClassifier	0.768302	0.001775
RidgeClassifier	0.836414	0.004234

Table 3: Comparison between models based on evaluation metrics (5-fold-cross-validation)

It is evident from Table 2 and Table 3 that the best performing model, both in the hate speech and non-hate speech classification, is the SVM with a f1-score of 0.859 (hate speech) and 0.838 (non-hate speech), as well as an overall cross-validation score of 85.29 %.

7.2 Subtask target entities classification

The classification process described in 5 Classification approaches, subsubsection 5.3.1 Target binary classification approach and subsubsection 5.3.2 Target concatenated classification approach which was applied in the classification process. Thereby, after additional annotation of the given corpus consisting of 32,374 datasets, 10,090 annotated datasets were considered in the training and testing the classifiers to be checked and compared. The root feature was the feature `‘c_text’` which was preprocessed in text preprocessing. In the target-entities methodology, no other features were included in the classifications.

The classification results of all six methods which were used for the classification are shown in Table 4. It is important to state that the key figures of the classifications in the particular case (regarding the f1-scores of the labels ‘person’, ‘group’ and ‘public’) vary greatly. This can be seen in the key figures of the classifier between person and public, which exhibit a significant divergence from 0.636 (f1-score, ‘person’) to 0.469 (f1-score, ‘public’) (Table 4). In general, the performance indicators of the Target binary classification approach used do deviate between a range of approx. $0.600 - 0.650$ (f1-score) for person-entity classification, a range of approx. $0.650 - 0.730$ (f1-score) for group-entity and a range of approx. $0.460 - 0.530$ (f1-score) for public-entity.

After the evaluation using a 5-fold cross-validation, the results of this were consolidated in Table 3. Consequently, the NaiveBayes performed the weakest of all models for target entity classification, achieving a cv-score (mean) of 69.5% (and 33.0% in concatenated approach (subsubsection 5.3.2)). The most effective model for target entity classification of all models with a cv-score (mean) of 76.0% is RandomForestClassifier. Examining the cross-validation performance of the individual approaches, there is significant evidence that the binary classification approach (subsubsection 5.3.1) has higher scores. This approach is used in the following to identify the best models.

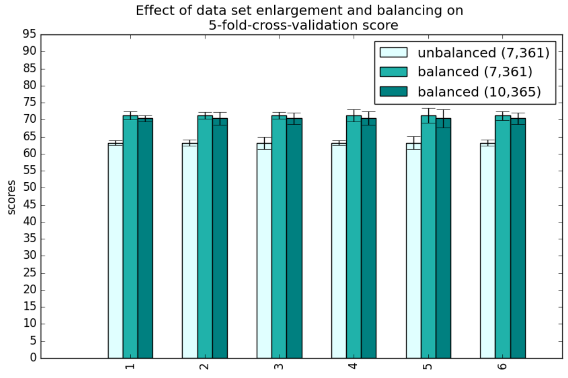

Figure 4: Effect of dataset enlargement and balancing the hate speech data corpus in target detection, 5-fold-cv-scores of LogisticRegression(1), NaiveBayes(2), RandomForestClassifier(3), SVM (4), DecisionTreeClassifier (5), RigdeClassifier(6)

	precision	recall	f1-score
LogisticRegression			
person	0.736	0.588	0.653
group	0.767	0.708	0.737
public	0.437	0.673	0.530
NaiveBayes			
person	0.738	0.559	0.636
group	0.726	0.745	0.736
public	0.406	0.556	0.469
RandomForestClassifier			
person	0.709	0.611	0.656
group	0.739	0.698	0.718
public	0.445	0.612	0.515
SVM			
person	0.700	0.608	0.650
group	0.763	0.704	0.732
public	0.427	0.604	0.501
DecisionTreeClassifier			
person	0.609	0.592	0.600
group	0.681	0.620	0.649
public	0.418	0.525	0.466
RidgeClassifier			
person	0.714	0.599	0.652
group	0.738	0.708	0.723
public	0.430	0.594	0.499

Table 4: Results of the target models with the extended target dataset, subsubsection 5.3.1 Target binary classification approach

When considering specifically the label classification scores using the methodology described in subsubsection 5.3.1, the DecisionTreeClassifier performed the weakest of all models for person-entity classification, achieving a f1-score of 0.600. The best model for person-entity classification of all models with a f1-score of 0.656 is the RandomForestClassifier. The classifiers NaiveBayes(0.636), SVM (0.650), RidgeClassifier (0.652) and LogisticRegression (0.653) rank in between.
Hereafter, the least efficient model for group-entity classification is the DecisionTreeClassifier with a f1-score of 0.649 and the best model the LogisticRegression with a f1-score of 0.737. The classifiers RandomForestClassifier (0.718), RidgeClassifier (0.723), SVM (0.732) and NaiveBayes (0.736) score in between.
Finally, the least efficient model for public-entity classification is the DecisionTreeClassifier with a f1-score of 0.466 and the best model the LogisticRegression with a f1-score of 0.530. The classifiers NaiveBayes (0.469), RidgeClassifier (0.499), SVM (0.501) and RandomForestClassifier (0.515) score in between.

It is evident from Table 4 and Table 5 that the best performing model overall (regarding cross-validation score) is RandomForestClassifier with a cross-validation score of 76.0%. However, the ranking of the classifiers for the individual labels

	m. cross-val. scores	std cross-val. scores
LogisticRegression		
Binary approach (5.3.1)	0.732	0.011
Concatenated approach (5.3.2)	0.678	0.014
NaiveBayes		
Binary approach (5.3.1)	0.695	0.018
Concatenated approach (5.3.2)	0.330	0.010
RandomForestClassifier		
Binary approach (5.3.1)	0.760	0.020
Concatenated approach (5.3.2)	0.569	0.013
SVM		
Binary approach (5.3.1)	0.731	0.015
Concatenated approach (5.3.2)	0.687	0.009
DecisionTreeClassifier		
Binary approach (5.3.1)	0.737	0.026
Concatenated approach (5.3.2)	0.593	0.014
RidgeClassifier		
Binary approach (5.3.1)	0.740	0.015
Concatenated approach (5.3.2)	0.678	0.010

Table 5: Comparison of classification approaches based on evaluation metrics (5-fold-cross-validation)

differs here. It can be concluded that for the person-entity classification, the classifier RandomForestClassifier with a f1-score of 0.656, for the group-entity classification the classifier LogisticRegression with a f1-score of 0.737 and for the public-entity classification the classifier LogisticRegression with a f1-score of 0.530 are the most suitable models for the specific labels.

8 Discussion

In the context of applying a best-practise model, it appears that while high individual accuracies for hate speech classification are attractive, the predictive accuracy for non-hate speech is significantly lower. This represents a problem in that high individual accuracies can be achieved with a small dataset, but these are not representative for the performance of the models at a larger scale. In examining the effect of dataset enlargement and balancing, this can be seen with hate speech (in Figure 3) and target entities (in Figure 4) cross-validation-scores. It can be observed that the cross-validation scores increase with larger datasets, but the single label accuracies decrease due to more training and testing data (see e.g. Table 4). Word ambiguities and linguistic deviations, such as character restrictions or restrictions in the area of swear words, represent another problem for classifiers. These deviations create neologisms that are challenging even for humans to classify. It can be stated that "[...] these factors are significantly hindering the performance [...] [18], especially in subjective entity classification [18]. Another problem according to Davidson, Warmsley, Macy, *et al.* [19] is the mixing of hate speech and offensive statements. Thereby, a consideration between common insults and serious hate speech is mistakenly not executed [19]. The integration of the profanity score as a feature is a crucial point that should be explored in future work.

9 Conclusion

Machine learning methods require a systematic amount of data for a proper learning process. A training dataset with Tweets forms the baseline, which was labelled by various classification algorithms. To improve the accuracy of the output, NA values were removed. In the further development of our project, an attempt was made to improve the scores by annotating comments specifically for the label target. The used dataset was obtained from the corpora GERMEVAL, 2018 [12], GERMEVAL, 2019 [13], HASOC, 2019 [14] and the Toxic Comments Corpus, 2018 [15] (in subsection 4.4). However, this process did not lead to significant improvements in the scores (Figure 3). The training data we relied on was then utilised to perform training and classification on an unknown dataset. We used different classifiers to evaluate which one fits best for our shared task. As a result the best model for hate speech classification in both hate speech and non-hate speech detection was SVC. This model has shown that it is able to successfully perform the shared task of hate speech classification. For the shared task of target classification, it should be noted that the data used in our model was extended manually due to a lack of elements to train a classifier properly. The possibility to improve the trained model in future work by others is given. Here, the RandomForestClassifier performed best in terms of cross-validation, and thus represents the best model for the approach.

In the process of working on the shared task for this paper, some observations were made that could be the content of future work and projects. The identification of hate speech using the dataset provided to us was relatively easy to realise, as there was always the given possibility to use other labelled datasets to train the classifier. To sum up, there was never the problem of being limited by the data. On the other hand, there was a lack of alternative datasets for target detection, as this topic has very rarely been the subject of investigations into Tweets. These datasets neither exist in German nor in English. For this reason, it would be a future task to have further datasets labelled by annotators under the aspect of target entity labels. Another consideration is the manual creation of swearword dictionaries. Although some of these dictionaries exist, they are mostly context-dependant and require manual completion. In this case, it could be helpful to use sentiment scores to correctly classify all possible spellings and word types depending on the context. Alternative solution strategies were also examined in the scope of this work. The use of computationally intensive models such as RoBERTa already grants a very high classification quality. The different RoBERTa models can be enriching for future studies [20]. The study of the learning curves of different species in relation to the amounts of the set of training data is a helpful resource. However, as was found, it is not the decision for or against a classifier that is decisive. Rather, it is the feature selection that determines how well or badly a word is classified.

10 Acknowledgements

To gain further insight into classification performance, we provided confusion matrices for each classifier used in the classification reports in the appendix. The submission, including code and reports, is available on Github at `https://github.com/HSMW-TargetGruppe4/Softwareprojekt`.

References

[1] *The Twitter Rules*, https://help.twitter.com/en/rules-and-policies/twitter-rules, Accessed: 2022-11-26, 2022.

[2] M. S. A. Sanoussi, C. Xiaohua, G. K. Agordzo, M. L. Guindo, A. M. Al Omari, and B. M. Issa, "Detection of hate speech texts using machine learning algorithm," in *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, 2022, pp. 0266–0273.

[3] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proceedings of the international AAAI conference on web and social media*, vol. 11, 2017, pp. 512–515.

[4] S. Tulkens, L. Hilte, E. Lodewyckx, B. Verhoeven, and W. Daelemans, "A dictionary-based approach to racism detection in dutch social media," *arXiv preprint arXiv:1608.08738*, 2016.

[5] K. Dinakar, B. Jones, C. Havasi, H. Lieberman, and R. Picard, "Common sense reasoning for detection, prevention, and mitigation of cyberbullying," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 2, no. 3, pp. 1–30, 2012.

[6] K. Tymann, M. Lutz, P. Palsbröker, and C. Gips, "Gervader-a german adaptation of the vader sentiment analysis tool for social media texts.," in *LWDA*, 2019, pp. 178–189.

[7] C. Demus, J. Pitz, M. Schütz, N. Probol, M. Siegel, and D. Labudde, "Detox: A comprehensive dataset for german offensive language and conversation analysis," in *Proceedings of the 6th Workshop on Online Abuse and Harms (WOAH 2022), Association for Computational Linguistics, Online*, 2022, pp. 54–61.

[8] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," *Advances in neural information processing systems*, vol. 28, 2015.

[9] J. Huang, J. Lu, and C. X. Ling, "Comparing naive bayes, decision trees, and svm with auc and accuracy," in *Third IEEE International Conference on Data Mining*, IEEE, 2003, pp. 553–556.

[10] S. Almatarneh, P. Gamallo, F. J. R. Pena, and A. Alexeev, "Supervised classifiers to identify hate speech on english and spanish tweets," in *International Conference on Asian Digital Libraries*, Springer, 2019, pp. 23–30.

[11] T. Salles, M. Gonçalves, V. Rodrigues, and L. Rocha, "Improving random forests by neighborhood projection for effective text classification," *Information Systems*, vol. 77, pp. 1–21, 2018.

[12] M. Wiegand, *GermEval-2018 Corpus (DE)*, version V1, 2019. DOI: 10.11588/data/0B5VML. [Online]. Available: https://doi.org/10.11588/data/0B5VML%5C%7D.

[13] J. M. Struß, M. Siegel, J. Ruppenhofer, M. Wiegand, M. Klenner, *et al.*, "Overview of germeval task 2, 2019 shared task on the identification of offensive language," 2019.

[14] T. Ranasinghe, M. Zampieri, and H. Hettiarachchi, "Brums at hasoc 2019: Deep learning models for multilingual hate speech and offensive language identification.," in *FIRE (working notes)*, 2019, pp. 199–207.

[15] S. V. Georgakopoulos, S. K. Tasoulis, A. G. Vrahatis, and V. P. Plagianakos, "Convolutional neural networks for toxic comment classification," in *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*, ser. SETN '18, Patras, Greece: Association for Computing Machinery, 2018, ISBN: 9781450364331. DOI: 10.1145/3200947.3208069. [Online]. Available: https://doi.org/10.1145/3200947.3208069.

[16] E. Chen, *The Obscenity List*, 2021. [Online]. Available: https://github.com/surge-ai/profanity.

[17] S. Elbagir and J. Yang, "Twitter sentiment analysis using natural language toolkit and vader sentiment," in *Proceedings of the international multiconference of engineers and computer scientists*, vol. 122, 2019, p. 16.

[18] K. Kumaresan and K. Vidanage, "Hatesense: Tackling ambiguity in hate speech detection," in *2019 National Information Technology Conference (NITC)*, 2019, pp. 20–26. DOI: 10.1109/NITC48475.2019.9114528.

[19] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 11, no. 1, pp. 512–515, May 2017. DOI: 10.1609/icwsm.v11i1.14955. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/14955.

[20] A. Warstadt, Y. Zhang, H.-S. Li, H. Liu, and S. R. Bowman, "Learning which features matter: Roberta acquires a preference for linguistic generalizations (eventually)," *arXiv preprint arXiv:2010.05358*, 2020.

Appendix

The appendix contains the classification results of the individual classifiers.

[1] Classification_Report(LogisticRegression).pdf

[2] Classification_Report(NaiveBayes).pdf

[3] Classification_Report(SVM).pdf

[4] Classification_Report(RandomForestClassifier).pdf

[5] Classification_Report(DecisionTreeClassifier).pdf

[6] Classification_Report(RidgeClassifier).pdf

Classification Report

Logistic Regression

Set Exploration

The exploration of the dataset led to the following dates

Dataset	Hatespeech-Extend-Dataset
Size	32,374 sets
Features	1: c_text 2: c(c_text) 3: c(emojis) 4: sentiment_neg 5: sentiment_pos 6: sentiment_neu 6: sentiment_comp 6: profanity_score
Label	1: hatespeech 2: target
Label distribution (Label 1)	0: 13,728 1: 18,646
Label distribution (Label 2)	person: 4,834 group: 3,513 public: 2,018 NaN: 22,009

The data set was then examined for the distribution and frequency of occurring terms for the processing of the text data. Background noise was removed.

Classification Metrics

After training the classifier in a predefined test-train split, the following metric was determined to evaluate the classifier performance.

Dataset	Hatespeech-Extend-Dataset
training set size	0.8
testing set size	0.2
Hatespeech Metrics	
5fold-CV-Score	0.840421± 0.006270
execution time	0.192s
Target Metrics, Binary approach	
5fold-CV-Score	0.732± 0.011
execution time	1.648s
Target Metrics, Concatenated approach	
5fold-CV-Score	0.678± 0.014
execution time	44.967s

Examined best performing model score

–Hatespeech –

The Logistic Regression classifier achieved a **placement of 2** in hatespeech classification, which ranks it **2/6 in the overall ranking**.

–Target –

The Logistic Regression classifier achieved a **placement of 4** in target classification, which ranks it **4/6 in the overall ranking**.

Hatespeech classification Ranking		
1 st.	SVM	0.852
2 nd.	Logistic Regression	0.840
3 rd.	Ridge Classifier	0.836
4 th.	NaiveBayes	0.833
5 th.	RandomForest Classifier	0.826
6 th.	DecisionTree Classifier	0.768

Target classification Ranking		
1 st.	RandomForest Classifier	0.760
2 nd.	Ridge Classifier	0.740
3 rd.	DecisionTree Classifier	0.737
4 th.	Logistic Regression	0.732
5 th.	SVM	0.731
6 th.	NaiveBayes	0.695

Classification Report

Naive Bayes

Set Exploration

The exploration of the dataset led to the following dates

Dataset	Hatespeech-Extend-Dataset
Size	32,374 sets
Features	1: c_text 2: c(c_text) 3: c(emojis) 4: sentiment_neg 5: sentiment_pos 6: sentiment_neu 6: sentiment_comp 6: profanity_score
Label	1: hatespeech 2: target
Label distribution (Label 1)	0: 13,728 1: 18,646
Label distribution (Label 2)	person: 4,834 group: 3,513 public: 2,018 NaN: 22,009

The data set was then examined for the distribution and frequency of occurring terms for the processing of the text data. Background noise was removed.

Classification Metrics

After training the classifier in a predefined test-train split, the following metric was determined to evaluate the classifier performance.

Dataset	Hatespeech-Extend-Dataset
training set size	0.8
testing set size	0.2
Hatespeech Metrics	
5fold-CV-Score	0.833± 0.007396
execution time	0.011s
Target Metrics, Binary approach	
5fold-CV-Score	0.695± 0.018
execution time	0.041s
Target Metrics, Concatenated approach	
5fold-CV-Score	0.330± 0.010
execution time	8.239s

Examined best performing model score

–Hatespeech –

The Naive Bayes classifier achieved a **placement of 4** in hatespeech classification, which ranks it **4/6 in the overall ranking**.

–Target –

The Naive Bayes classifier achieved a **placement of 6** in target classification, which ranks it **6/6 in the overall ranking**.

Hatespeech classification Ranking		
1 st.	SVM	0.852
2 nd.	Logistic Regression	0.840
3 rd.	Ridge Classifier	0.836
4 th.	NaiveBayes	0.833
5 th.	RandomForest Classifier	0.826
6 th.	DecisionTree Classifier	0.768

Target classification Ranking		
1 st.	RandomForest Classifier	0.760
2 nd.	Ridge Classifier	0.740
3 rd.	DecisionTree Classifier	0.737
4 th.	Logistic Regression	0.732
5 th.	SVM	0.731
6 th.	NaiveBayes	0.695

Classification Report

SVM

Set Exploration

The exploration of the dataset led to the following dates

Dataset	Hatespeech-Extend-Dataset
Size	32,374 sets
Features	1: `c_text` 2: `c(c_text)` 3: `c(emojis)` 4: `sentiment_neg` 5: `sentiment_pos` 6: `sentiment_neu` 6: `sentiment_comp` 6: `profanity_score`
Label	1: `hatespeech` 2: `target`

Label distribution (Label 1)	`0: 13,728`
	`1: 18,646`

Label distribution (Label 2)	`person: 4,834`
	`group: 3,513`
	`public: 2,018`
	`NaN: 22,009`

The data set was then examined for the distribution and frequency of occurring terms for the processing of the text data. Background noise was removed.

Classification Metrics

After training the classifier in a predefined test-train split, the following metric was determined to evaluate the classifier performance.

Dataset	Hatespeech-Extend-Dataset
training set size	0.8
testing set size	0.2
Hatespeech Metrics	
5fold-CV-Score	0.852± 0.005337
execution time	76.769s
Target Metrics, Binary approach	
5fold-CV-Score	0.731± 0.015
execution time	21.414s
Target Metrics, Concatenated approach	
5fold-CV-Score	0.687± 0.009
execution time	1,328.920s

Examined best performing model score

–Hatespeech –

The SVM classifier achieved a **placement of 1** in hatespeech classification, which ranks it **1/6 in the overall ranking**.

–Target –

The SVM classifier achieved a **placement of 5** in target classification, which ranks it **5/6 in the overall ranking**.

Hatespeech classification Ranking		
1 st.	SVM	0.852
2 nd.	Logistic Regression	0.840
3 rd.	Ridge Classifier	0.836
4 th.	NaiveBayes	0.833
5 th.	RandomForest Classifier	0.826
6 th.	DecisionTree Classifier	0.768

Target classification Ranking		
1 st.	RandomForest Classifier	0.760
2 nd.	Ridge Classifier	0.740
3 rd.	DecisionTree Classifier	0.737
4 th.	Logistic Regression	0.732
5 th.	SVM	0.731
6 th.	NaiveBayes	0.695

Classification Report

RandomForestClassifier

Set Exploration

The exploration of the dataset led to the following dates

Dataset	Hatespeech-Extend-Dataset
Size	32,374 sets
Features	1: `c_text` 2: `c(c_text)` 3: `c(emojis)` 4: `sentiment_neg` 5: `sentiment_pos` 6: `sentiment_neu` 6: `sentiment_comp` 6: `profanity_score`
Label	1: `hatespeech` 2: `target`

Label distribution (Label 1)	`0: 13,728`
	`1: 18,646`

Label distribution (Label 2)	`person: 4,834`
	`group: 3,513`
	`public: 2,018`
	`NaN: 22,009`

The data set was then examined for the distribution and frequency of occurring terms for the processing of the text data. Background noise was removed.

Classification Metrics

After training the classifier in a predefined test-train split, the following metric was determined to evaluate the classifier performance.

Dataset	Hatespeech-Extend-Dataset
training set size	0.8
testing set size	0.2
Hatespeech Metrics	
5fold-CV-Score	0.826± 0.007681
execution time	33.651s
Target Metrics, Binary approach	
5fold-CV-Score	0.760± 0.020
execution time	13.615s
Target Metrics, Concatenated approach	
5fold-CV-Score	0.569± 0.013
execution time	145.415s

Examined best performing model score

–Hatespeech –
The RandomForestClassifier classifier achieved a **placement of 5** in hatespeech classification, which ranks it **5/6 in the overall ranking**.

–Target –
The RandomForestClassifier classifier achieved a **placement of 1** in target classification, which ranks it **1/6 in the overall ranking**.

Hatespeech classification Ranking		
1 st.	SVM	0.852
2 nd.	Logistic Regression	0.840
3 rd.	Ridge Classifier	0.836
4 th.	NaiveBayes	0.833
5 th.	RandomForest Classifier	0.826
6 th.	DecisionTree Classifier	0.768

Target classification Ranking		
1 st.	RandomForest Classifier	0.760
2 nd.	Ridge Classifier	0.740
3 rd.	DecisionTree Classifier	0.737
4 th.	Logistic Regression	0.732
5 th.	SVM	0.731
6 th.	NaiveBayes	0.695

Classification Report

DecisionTreeClassifier

Set Exploration

The exploration of the dataset led to the following dates

Dataset	Hatespeech-Extend-Dataset
Size	32,374 sets
Features	1: c_text 2: c(c_text) 3: c(emojis) 4: sentiment_neg 5: sentiment_pos 6: sentiment_neu 6: sentiment_comp 6: profanity_score
Label	1: hatespeech 2: target
Label distribution (Label 1)	0: 13,728 1: 18,646
Label distribution (Label 2)	person: 4,834 group: 3,513 public: 2,018 NaN: 22,009

The data set was then examined for the distribution and frequency of occurring terms for the processing of the text data. Background noise was removed.

Classification Metrics

After training the classifier in a predefined test-train split, the following metric was determined to evaluate the classifier performance.

Dataset	Hatespeech-Extend-Dataset
training set size	0.8
testing set size	0.2
Hatespeech Metrics	
5fold-CV-Score	0.768± 0.001775
execution time	10.947s
Target Metrics, Binary approach	
5fold-CV-Score	0.737± 0.026
execution time	3.211s
Target Metrics, Concatenated approach	
5fold-CV-Score	0.593± 0.014
execution time	40.647s

Examined best performing model score

–Hatespeech –

The DecisionTreeClassifier classifier achieved a **placement of 6** in hatespeech classification, which ranks it **6/6 in the overall ranking**.

–Target –

The DecisionTreeClassifier classifier achieved a **placement of 3** in target classification, which ranks it **3/6 in the overall ranking**.

Hatespeech classification Ranking		
1 st.	SVM	0.852
2 nd.	Logistic Regression	0.840
3 rd.	Ridge Classifier	0.836
4 th.	NaiveBayes	0.833
5 th.	RandomForest Classifier	0.826
6 th.	DecisionTree Classifier	0.768

Target classification Ranking		
1 st.	RandomForest Classifier	0.760
2 nd.	Ridge Classifier	0.740
3 rd.	DecisionTree Classifier	0.737
4 th.	Logistic Regression	0.732
5 th.	SVM	0.731
6 th.	NaiveBayes	0.695

Classification Report

RidgeClassifier

Set Exploration

The exploration of the dataset led to the following dates

Dataset	Hatespeech-Extend-Dataset
Size	32,374 sets
Features	1: `c_text` 2: `c(c_text)` 3: `c(emojis)` 4: `sentiment_neg` 5: `sentiment_pos` 6: `sentiment_neu` 6: `sentiment_comp` 6: `profanity_score`
Label	1: `hatespeech` 2: `target`

Label distribution (Label 1)	`0: 13,728`
	`1: 18,646`

Label distribution (Label 2)	`person: 4,834`
	`group: 3,513`
	`public: 2,018`
	`NaN: 22,009`

The data set was then examined for the distribution and frequency of occurring terms for the processing of the text data. Background noise was removed.

Classification Metrics

After training the classifier in a predefined test-train split, the following metric was determined to evaluate the classifier performance.

Dataset	Hatespeech-Extend-Dataset
training set size	0.8
testing set size	0.2
Hatespeech Metrics	
5fold-CV-Score	0.836± 0.004234
execution time	0.060s
Target Metrics, Binary approach	
5fold-CV-Score	0.740± 0.015
execution time	0.113s
Target Metrics, Concatenated approach	
5fold-CV-Score	0.678± 0.010
execution time	139.191s

Examined best performing model score

–Hatespeech –

The RidgeClassifier classifier achieved a **placement of 3** in hatespeech classification, which ranks it **3/6 in the overall ranking**.

–Target –

The RidgeClassifier classifier achieved a **placement of 2** in target classification, which ranks it **2/6 in the overall ranking**.

Hatespeech classification Ranking		
1 st.	SVM	0.852
2 nd.	Logistic Regression	0.840
3 rd.	Ridge Classifier	0.836
4 th.	NaiveBayes	0.833
5 th.	RandomForest Classifier	0.826
6 th.	DecisionTree Classifier	0.768

Target classification Ranking		
1 st.	RandomForest Classifier	0.760
2 nd.	Ridge Classifier	0.740
3 rd.	DecisionTree Classifier	0.737
4 th.	Logistic Regression	0.732
5 th.	SVM	0.731
6 th.	NaiveBayes	0.695