DLRG@HASOC 2020: A Hybrid Approach for Hate and Offensive Content Identification in Multilingual Tweets

Ratnavel Rajalakshmi, Yashwanth Reddy. B

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai

Abstract

In recent times, most of the people prefer social media platforms as a communication tool and express their views publicly and anonymously. Hate speech and posting offensive contents has become a major issue nowadays. To handle these problems, automated methods are necessary that can help to analyse the social media posts and to identify the hate speech. Existing methods do not focus more on multilingual posts and it poses more challenges, not only due to the linguistic properties but also due to the class imbalance problem. The task of identifying hate and offensive content posted in Hindi or German languages has the same issues. To address the problem of class imbalance, we have combined a over sampling technique with a suitable feature weighting method. In the proposed approach, Multi-class imbalance-based feature selection method is combined with an SVM classifier to classify the tweet as a hate speech or not. This work was submitted to Hate and Offensive Content Identification (HASOC) task@FIRE2020 and scored third rank. We have achieved an accuracy of 80% and 72% on the released German and Hindi language tweets respectively.

Keywords

Hate Speech Detection, SVM, Class Imbalanced data, Multilingual Tweets

1. Introduction

With the advancements in Science and Technology, nowadays many people post their opinion, thoughts and comments on social websites like Facebook, twitter, etc. This has also resulted in the widespread of Hate and offensive content over the web. It becomes difficult to distinguish offensive tweets as they contain different hash tags, emojis, language styles. As most of the harmful incidents of hate speech have created a mental stress among the users of the web, it is very important to take preventive measures for such offensive contents. The large volume of data makes it highly impractical to monitor the posts manually. Many automated techniques applying machine learning algorithms like SVM, Naïve Bayes [1] were used to perform the task. In this paper, we propose a method to determine the hate speech in tweets and perform categorization (hate and offensive speech or not) based on the social media posts in the German Language. The given data is an imbalanced one and it is required to translate the German Tweets to English before applying any machine Learning algorithm. Always it is preferable to balance the data set, as the class imbalance may affect the performance. In many of the existing

FIRE '20, Forum for Information Retrieval Evaluation, December 16–20, 2020, Hyderabad, India

rajalakshmi.r@vit.ac.in (R. Rajalakshmi)

works, TF-IDF based feature weighting method was followed. In this paper, the class imbalance problem is addressed by applying a suitable method and detailed analysis is performed in identifying the tweet expressed in German language as a hate speech or not. This work has been submitted to the HASOC 2020 Data Challenge, organized as a part of FIRE 2020 (Forum for Information Retrieval Evaluation) conference.

The paper is organized as follows: The related works are discussed in section 2 followed by the proposed approach in section 3. The experimental results are discussed in detail in section 4 and the concluding remarks are presented in the last section 5.

2. Related work

There had been many studies made on classifying the offensive content on the web. Hate base is an online repository of hate speech words. T. Davidson, D. Warmsley had built a classifier for Hate base [2]. They have created unigram, bigram, trigram features weighted with its TF-IDF, Part of Speech (POS) tag and suggested Linear classifiers for classifying the offensive language. But the Model was biased towards the offensive language and failed to differentiate between the commonplace offensive language with serious hate speech (e.g., queer in "He's a damn good actor. As a gay man, it's awesome to see an openly queer actor given the lead role for a major film.", from HatebaseTwitter dataset. [2]).Waseem et al. (2017) have proposed a typology that differentiates between whether the (abusive) language is directed towards a specific person or entity, or towards a particular group, and whether the abusive content is explicit or implicit (eg., racist, sexist, neither or both) [3]. GermEval is a shared task focused on offensive language identification in German tweets(8500 tweets). Wiegand et al. (2018) further applied the idea to Waseem et al. to this task. They experimented with detecting offensive vs. non-offensive tweets, and also with a second task on further sub-classifying the offensive tweets as, insult, abuse or profanity[4]. TRAC: The 2018 Workshop on Trolling, Aggression, and Cyber bullying (TRAC) hosted a shared task focused on detecting aggressive text in both English and Hindi [5]. The data set from this task is available to the public and contains 15,869 Facebook comments labeled as overtly aggressive(OAG), covertly aggressive(CAG), or non-aggressive(CAG). The best-performing scores was obtained convolutional neural networks (CNN), recurrent neural networks, and LSTM[6]. OffensEval: Offensive Language Identification Dataset (OLID) dataset, which was built specifically for this task. OLID was annotated using a hierarchical three-level annotation model introduced in Zampieri et al[7]. Three sub-tasks include Offensive Language Identification(Not Offensive, Offensive), Categorization of Offensive Language (Targeted Insult, Untargeted), Offensive Language Target Identification(Individual, Group, Other) [8].Greevy and Smeaton used SVM and bag of words to detect offensive content on web pages [9]. They have used PRINCP corpus of 3 million words with 2 class labels namely offensive and not offensive. BOW, n-gram word sequences and POS tagged documents were used to represent the dataset. But they used only SVM classifier for detection without considering the results of other classifiers. A similar approach was made by Warner and Hirschberg (2012) using unigrams with SVM to detect offensive content of the web [10]. A research group founded by Jigsaw and Google are trying to develop a tool for identifying the toxicity of comments between the range of 0 to 100. C. Nobata, J. Tetreault had proposed annotation of hate speech versus clean speech

[11] They have collected news and finance data set for the binary classification of abusive and clean tweets. They have employed Vowpal Wabbit's regression model for the features obtained through N-grams, Linguistic, Syntactic and Distributional Semantics. They have compared the performance of it using all the above features but focused only on English language and did not consider any other language. D. Gitari had further classified the tweets into strong or weak using lexicon-based approaches [12]. They have used a semantic and subjectivity approach to the created lexicon and use this features for a classifier. But they used rule-based classifier instead of Machine Learning Model which lead to low precision and recall scores.Deep learning methods were applied in various NLP tasks such as web page classification [13] and analysing the tweets. Sentiment analysis in movie reviews has been reported in [14]. The aspect based sentiment analysis has many applications and a detailed survey has been performed [15]. The overview of the tasks submitted to FIRE 2020 is summarized in [16]

3. Proposed Methodology

In any classification task, identifying the relevant features from the given data is an important step. In many applications, class imbalance is observed which is inevitable. In such cases, balancing the data set also plays a crucial role, as it may affect the classification performance. In this HASOC 2020 task, the released data set is highly imbalanced one and we have handled it by applying appropriate method. In this paper, to perform the hate and offensive task classification a supervised learning approach is proposed with an approach to solve the class imbalance problem and the steps involved are summarized below:

- Translation of Tweets;
- Pre-Processing and Feature Extraction
- Handling Imbalance Data
- Building the Model

3.1. Translation of Tweets

In this task, we have been provided with two different language data sets (German and Hindi). As a first step, the tweets are translated to English language. For example, a tweet in German **"Frank Rennicke – Ich binxa0stolz"** was converted by employing MLtranslate and it results in the corresponding English tweet **Frank Rennicke - I am proud**. For this translation process, ML Translator API was used, which is a Google's Neural Machine Translation (NMT) system . This translation method was widely used because of its simplicity and zero-shot translation. Melvin et al. [17] proposed a single Neural Translation multilingual model that shares the same encoder, decoder and attention modules for all the languages without increasing the complexity of model. Also, as the parameters are shared across all the languages, it generalizes well to multiple languages. This NMT model has the advantage of zero-shot translation, as several language pairs are used in a single model and unseen word pairs in different languages were also learnt by the model. We found this translation process as suitable for this task and hence applied the same for converting the tweets in German / Hindi to English.

3.2. Pre-processing and Feature Extraction

Hash tags provide insights about a specific ideology by a group of people. These tags provide vital information for text classification, especially in the case of identification of offensive language in tweets. So we have processed the hash tags and obtained tokenized words out of it, after segmenting the tokens. For example, after applying the hash tag segmentation on the pre-processed tweet **everyhingisgood**, we obtain **everything is good**. Lemmatization is the process of reducing the word to its root form, which is helpful. We have used NLTK (Natural Language Tool Kit) WordNet Lemmatizer for performing lemmatization. Consider the following example, **Koeln Mohamed recognizes no German right but only the Scharia. That he wanted to break Cologne Cathedral was just a joke but when he comes out of jail, he has no more pity**. After lemmatising, it becomes **koeln mohamed recognizes german right scharia wanted break cologne cathedral joke come jail pity**.

In any text classification task, the feature extraction plays an important role. To extract suitable features from the pre-processed data, we have used TF-IDF (Term Frequency – Inverse Document Frequency) as it is the well-known weighting scheme in many NLP tasks and this score is calculated based on the count of terms that are present in every tweet with the terms present in the entire corpus. As it extracts most descriptive terms from the tweet collection and simple to implement, we have chosen this feature weighting scheme. In our experiments, the minimum frequency of the word is set to 5 and maximum number of words is set to 5000.

3.3. Handling Imbalance Data

German and Hindi data sets were highly imbalanced data set, so SOUP(Similarity-based Oversampling and Undersampling Preprocessing)was performed. German data is imbalanced one with unequal number of tweets for positive and negative classes. It contains 1700 non-hate and offensive tweets, but only 673 hate speech samples. After applying SOUP the samples of both labels are balanced with 1186 on both the classes as shown in Figure 3.3.

We have applied SOUP (Similarity-based Oversampling and Undersampling Preprocessing) from multi-balance package. It is an oversampling technique in which the number of the minority class samples are increased and the number of majority class samples are decreased to obtain a balanced data set. This is performed by removing the most unsafe examples until a desired class cordinality is obtained. The calculation of the safe level is done by using the Heterogeneous Value difference metric (HVDM) [18]. By this method, the class imbalance problem is solved and then we used this balanced data for performing classification task. To perform classification, we have applied different machine learning algorithms viz., Logistic Regression, Naive Bayes Classifier, SVM and Random Forest method and the effect of applying SOUP method was studied.

4. Experiments and Results

To study the performance of the proposed method, various experiments were conducted using German and Hindi data sets. For implementation, we used Python 3 and scikit-learn library. All the experiments were carried on a workstation with Intel Xeon Quad Core Processor, 32

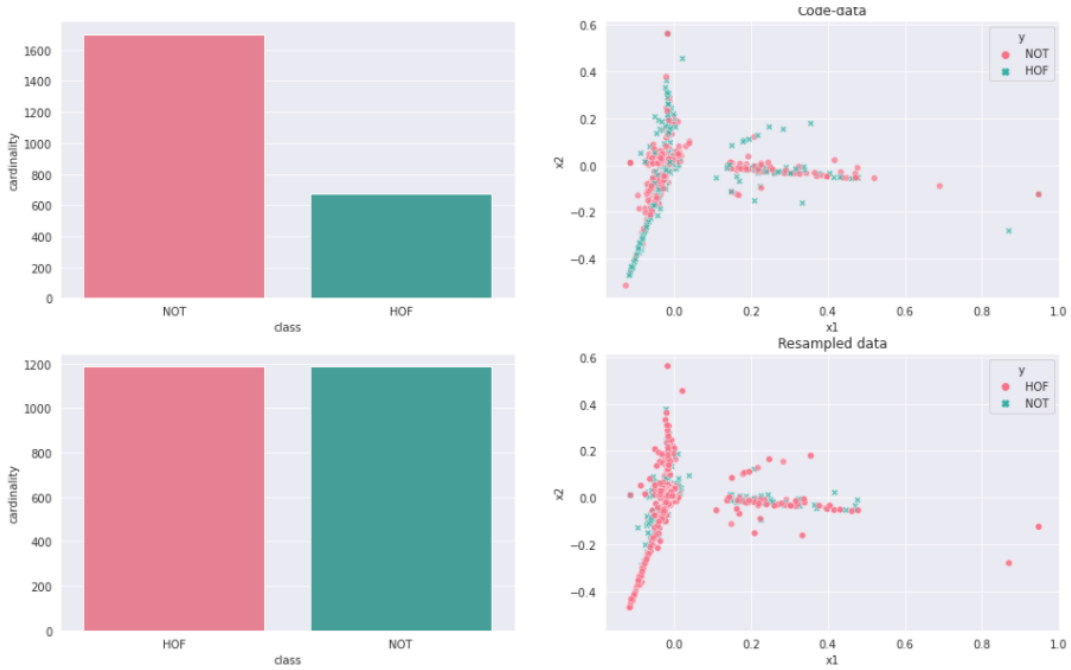

Figure 1: Data Visualization before and after applying SOUP

GB RAM, NVIDIA Quadro P4000 GPU 8GB. For the initial experiments, we have divided the released training data into training set and validation set and conducted the experiments using accuracy as the performance metric. Finally the performance of the proposed system was tested on the test set released by the organizers. For these experiments, we combined all the training and validation data into a single training set and applied the algorithm. We have reported the validation accuracy and test accuracy obtained on both German and Hindi data sets. After translation and pre-processing of tweets, tokenization was performed. Then to extract the suitable features, we have applied TF-IDF. First, TF-IDF vectorizer (using sklearn) was used to get maximum of 4,378 features with the minimum occurrence frequency of 2 for German data set and 6,789 features for Hindi data set. We have used SOUP method to handle the imbalanced data and then we have used the above features to build the Logistic Regression (LR) and Random Forest (RF) classifier and SVM models. The performance of different classifiers was studied by applying SOUP technique and the results are in Table1. It is observed from Table 1 that, on German and Hindi data set, among four classifiers, SVM performs better than the other three methods viz. Logistic regression, Naive Bayes and Random Forest. So, it can be concluded that, the class imbalance problem can be addressed and it can improve the performance of the classifier. The summary of the results are presented in Table 2.

Table 1
Performance of proposed approach - Test Accuracy

Data set	Logistic Regression	Naive Bayes	SVM	Random Forest
German	79	57	80	79
Hindi	69	55	72	68

Table 2
Performance of proposed approach - Test Accuracy

Data set	SOUP with SVM
German	80
Hindi	72

5. Conclusion

This work was submitted to the FIRE2020 task, Identification of Hate and Offensive Speech in Indo-European Languages (HASOC 2020). In this research, the problem of identifying the hate and offensive content in tweets have been experimentally studied on two di erent language data sets German and Hindi that has class imbalance. The importance of feature weighting method was analysed by using TF-IDF based feature selection by applying on different classifiers. Also, the effect of class imbalance problem was studied. As the released German and Hindi data sets were highly imbalanced, we applied SOUP analysis and then performed classification. From the experimental results, it is shown that the performance of the SVM classifier is better than the other methods and a test accuracy of 80% and 72% were achieved on German and Hindi data set respectively. In this work, we have restricted to machine learning approaches with suitable feature selection method and deep learning techniques will be explored in future.

6. Acknowledgement

The authors would like to thank the management of Vellore Institute of Technology, Chennai for providing the support to carry out this work. Also, the authors thank the Science and Engineering Research Board, Govt. of India for their financial support (ECR/2016/000484).

References

[1] I. Kwok, Y. Wang, Locate the hate: Detecting tweets against blacks, in: Proceedings of the twenty-seventh AAAI conference on artificial intelligence, 2013, pp. 1621–1622.
[2] T. Davidson, D. Warmsley, M. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, arXiv preprint arXiv:1703.04009 (2017).
[3] Z. Waseem, Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter, in: Proceedings of the first workshop on NLP and computational social science, 2016, pp. 138–142.
[4] M. Wiegand, M. Siegel, J. Ruppenhofer, Overview of the germeval 2018 shared task on the

identification of offensive language, Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria – September 21, 2018, Austrian Academy of Sciences, Vienna, Austria, 2018, pp. 1 – 10. URL: http://nbn-resolving.de/urn:nbn:de:bsz:mh39-84935.

[5] M. Janicka, M. Lango, J. Stefanowski, Using information on class interrelations to improve classification of multiclass imbalanced data: A new resampling algorithm, International Journal of Applied Mathematics and Computer Science 29 (2019) 769–781.

[6] R. Kumar, A. K. Ojha, S. Malmasi, M. Zampieri, Benchmarking aggression identification in social media, in: Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), 2018, pp. 1–11.

[7] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, Predicting the type and target of offensive posts in social media, arXiv preprint arXiv:1902.09666 (2019).

[8] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval), arXiv preprint arXiv:1903.08983 (2019).

[9] E. Greevy, A. F. Smeaton, Classifying racist texts using a support vector machine, in: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 2004, pp. 468–469.

[10] W. Warner, J. Hirschberg, Detecting hate speech on the world wide web, in: Proceedings of the second workshop on language in social media, 2012, pp. 19–26.

[11] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, Y. Chang, Abusive language detection in online user content, in: Proceedings of the 25th international conference on world wide web, 2016, pp. 145–153.

[12] N. D. Gitari, Z. Zuping, H. Damien, J. Long, A lexicon-based approach for hate speech detection, International Journal of Multimedia and Ubiquitous Engineering 10 (2015) 215–230.

[13] A. P. C. A. Rajalakshmi, Joel Raymann, Deep URL: design of adult URL classifier using deep neural network, ACM Conference Proceedings 20 (2019) 1–5.

[14] R. R. Sivakumar Soubraylu, Hybrid convolutional bidirectional recurrent neural network based sentiment analysis on movie reviews, Computational Intelligence – (2020) –.

[15] R. R. Vaishali Ganganwar, Implicit aspect extraction for sentiment analysis: A survey of recent approaches, Procedia Computer Science 165 (2019) 485–491.

[16] T. Mandl, S. Modha, G. K. Shahi, A. K. Jaiswal, D. Nandini, D. Patel, P. Majumder, J. Schäfer, Overview of the HASOC track at FIRE 2020: Hate Speech and Offensive Content Identification in Indo-European Languages), in: Working Notes of FIRE 2020 - Forum for Information Retrieval Evaluation, CEUR, 2020. URL: http://ceur-ws.org/.

[17] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, M. Hughes, J. Dean, Google's multilingual neural machine translation system: Enabling zero-shot translation, Transactions of the Association for Computational Linguistics 5 (2017) 339–351. URL: https://www.aclweb.org/anthology/Q17-1024. doi:10.1162/tacl_a_00065.

[18] M. Lango, K. Napierala, J. Stefanowski, Evaluating difficulty of multi-class imbalanced data, in: International Symposium on Methodologies for Intelligent Systems, Springer, 2017, pp. 312–322.