COMPARATIVE ANALYSIS FOR OFFENSIVE LANGUAGE IDENTIFICATION OF TAMIL TEXT USING SVM AND LOGISTIC CLASSIFIER

Prabhu Ram. N, Meeradevi.T, Vibin Mammen Vinod, Gothainayaki.A, Anusha S and Agalya T

Electronics and Communication Engineering, Kongu Engineering College, Erode, TamilNadu, India

Abstract

Social media like Twitter, Facebook, YouTube provide an opportunity of the fastest communication between people. The social media texts are largely filled with code-mixed comments/post and reactions and its content may be filled with offensive language or non-offensive language. It is necessary to classify the YouTube comments/post and reactions as offensive label and non-offensive label. As the offensive comments/post is very sensational to something or someone to react in the society, Government has responsibility to identify it in the social media, before it reaches a larger audience. In India, multi-lingual practices use code mixed comments/post in social media, which leads to difficulty in offensive text classification automatically. The Dravidian code mixed data set is used to train the machine learning model to classify the label as offensive language or non-offensive language. The text data set is transformed into numerical data based on relative occurrence in the available datasets of training and testing using TFIDF method. However, the imbalanced dataset may be biased to a particular class of label, and hence it is turned into balanced dataset using SMOTE method. It is trained on SVM classifier and Logistic Classifier. The F1 score is analsyed and it is observed that balanced dataset predictions are better than unbalanced dataset predictions.

Keywords

Multilingual, SMOTE, TFIDF, SVM, Logistic classifier, NLP, Machine Learning

1. Introduction

In the modern era there are 3.78 billion social media users worldwide in 2021. The social media makes communication easier and faster over the world and connecting everyone together. The social media like Facebook, YouTube, Twitter gave us freedom to express opinion in public. It may allow some bad actors in spreading fake news and offensive content. The offensive language in the social platform is one of the most dangerous activities. So people have to be protected themselves from these hateful activities in social media. The main challenges in the social media is to identify offensive text content and deleting the problematic posts. Research based on safety and security in social media has grown substantially in the last decade. In many countries like United Kingdom, Canada, France, these activities are punishable[1].

FIRE 2021, Forum for Information Retrieval Evaluation, December 13-17, 2021, India

prabhuramnphd@gmail.com (P. Ram. N); meeradevi.ece@kongu.edu (Meeradevi.T); vibin.ece@kongu.edu (V. M. Vinod); gothainayakia.18ece@kongu.edu (Gothainayaki.A); anushas.18ece@kongu.edu (A. S)
0000-0003-2769-9790 (P. Ram. N)

Social networks have introduced policies to restrict the offensive speech on people based on racism, gender etc. A fine-drawn hate speech in sentences can be considered as hate or not hate depending upon the person who interprets. The social media texts are represented with multilingual text and code-mix text. The phenomenon of mixing the second language into the first language or mixing the foreign languages into the native language structure is said to be code mix. Such that, Tamil words are written in English script. Multilingual text is the combination of multiple native language in single sentence. Such that Tamil and English words were written in their native script in single sentence. The technique to identify the solution to this problem by NLP(Natural Language Processing). NLP is a field of artificial intelligence, which has an ability to understand, analyse the context of the human language.

2. Related Works

Hate speech identification through sentiment analysis is one of the current research fields in Natural Language Processing. The solution is given by either machine learning approach or lexicon based approach. The machine learning approach involves collecting an annotated data, pre-processing the collected text data, transformation into machine learning input vector by vectorisation technique and trained to classify using machine learning model. Lexicon based approach is widely used in sentiment analysis, where the sentiment are collected from WordNet, SentiwordNet and are used for classification. In lexicon based approach, there is no necessity for labelling which is a time consuming process.

Hate speech identification on monolingual english dataset[2, 3] and code-mix dataset for Tamil and Malayalam scripts, the features extraction is executed by various methods like Hash Vectorizer[2], Count Vectoriser, TFIDF(Term Frequency Inverse Document Frequency)[3, 4, 5, 6] and Word Embedding, customized word embedding, CBOG, Skip-gram, word2vec, doc2vec, fastText[7]. TFIDF vectorizer, Count vectorizer are most commonly used vectorisation algorithms which are not neural network based transformation. However, TFIDF performs well on smaller vocabulary size, but more features are recorded on larger dataset, by modifying IDF(Inverse Document Frequency) feature size with minimum computation time[8]. Neural network based vectorization methods such as word2vec, doc2vec, fastText are used on code-mix dataset. In which fastText vectorization performs better than other neural network based vectorization methods[9]. The neural network based classification architectures like sub-word level LSTM model, Hierarchical LSTM model, BERT, XLM-RoBERT, LSTM, GRU, XLNet[10, 11, 12] were used. Some machine learning based classification models such as Support Vector Machine(SVM), Logistic Regression (LR), Random Forest Classifier (RFC)[3, 4, 13, 14, 15, 16] and K-Nearest Neighbour (KNN)[17] are used. SVM model performs better for code-mix tamil dataset than other machine learning models. Deep learning models such as RNN[11, 18],MLP are also used for classification[19, 20] for enhancement in prediction of classification. The evaluation of predictive model by accuracy, f1-score, precision, recall[14, 15, 17]. Hate speech identification of code mix data, trained model has reduced prediction accuracy due to imbalanced dataset. Section 3 describes the methodology, Section 4 describes about experimental setup for training model of SVM and logistic classifier in different configurations of hyper-parameter. The conversion of imbalanced dataset into balanced dataset using SMOTE method is also described in Section 4. Section 5 describes about the results and discussion. Section 6 describes about the conclusion.

3. Methodology

The flow of methodology have been described in detail in the following sub sections.

3.1. Text Pre-processing

Preprocessing involves the removal of special characters such as reaction smiles, punctuation using standard package. The number of vocabularies gets reduced after removal of special characters. In English language, conversion of token of words into its equivalent base form of word by stemming and lemmitization is done. However, in Dravidian language, such processes are not possible. The stream of text data is converted into token of word as unigram word, bigram word, n-gram words as a token by the process called as tokenisation.

3.2. Vectorisation

The text after pre-processing is vectorised. The vectorisation method include TFIDF(Term Frequency Inverse Document Frequency) used to represent the text data into its equivalent numerical data. TFIDF adds weightage to unique words in the document.

3.3. Training Model

The logistic regression and SVM model are trained by tri-gram based TFIDF vectorization of training dataset. The aim of the task is to classify the text as offensive or not-offensive class. Logistic regression(LR) and Support Vector Machine (SVM) are supervised machine learning algorithms used for classification and regression and they are best suited for binary classification.

3.4. Making Balanced dataset

The dataset may be balanced or imbalanced dataset. The balanced dataset contains equal number of labels as offensive labels and not-offensive labels. The imbalanced dataset is the one which has either one of the labels high. The dataset with 1153 offensive and 4724 not offensive is an example of imbalanced dataset. This imbalancing in the dataset may lead to fit the model on majority class which may give lower prediction results. There are some methods to make imbalanced dataset to balanced dataset. They are:

- Oversampling
- Undersampling
- SMOTE(Synthetic Minority Oversampling Technique)

Algorithm 1 : SMOTE's algorithm

```
procedure SMOTE($X, y$)    ▷ SMOTE of X data array and y target array
    $k_{neighbors} \leftarrow [5]$    ▷ Number of nearest neighbors
    $n_{jobs} \leftarrow 4$    ▷ Number of cores on execution
    $n_{sample} \leftarrow Count(X)$    ▷ Number of input samples
    $min \leftarrow Count(y_{majorityclass}) - Count(y_{minorityclass})$    ▷ Number of majority and minority classes
    $step \leftarrow random(0, 1)$    ▷ Scalar multiplicative value
    while $i \leq n_{sample}$ and $X_i \in \mathrm{y}_{miniorityclass}$ and min $\neq 0$ do
        $\mathbf{X^i_{nn}} \leftarrow [[X^i_{nn_1}], [X^i_{nn_2}], ..., [X^i_{nn_{k_{neighbours}}}]]$    ▷ $X^i$ Nearest neighbour sample
        $\mathbf{X^i_{new}} \leftarrow \mathbf{X_i} + step \times (\mathbf{X_i} - \mathbf{X^i_{nn}})$    ▷ $X_i, X_{nn} \in \mathrm{y}_{miniorityclass}$
        $min \leftarrow min - 1$
        $i \leftarrow i + 1$
    end while
    return $X_{new}$    ▷ Augmented Data
end procedure
```

Oversampling methods is duplicating actual minority data from the dataset. The undersampling method is removal of actual majority data from the data set.These approaches does not add any new information to the dataset.SMOTE is the process of synthetically generating features of minority class[21, 22, 23]. Based on Algorithm 1 the balanced dataset is generated.

3.5. Evaluating Model

The trained model is to be evaluated with the test data set. The metrics used to evaluate the model are accuracy, f1-score, precision and recall. The accuracy of the model alone is insufficient to evaluate as best fitted model. This is due to model may be biased to certain classes which can be identified using f1-score metrics.

4. Experimental Setup

The dataset given in HASOC-Dravidian CodeMix FIRE 2021 [24] for the task of detection of offensive language is split into training samples and testing samples and is described in Table 1. In Table 2 is the description of number of known vocabulary from training set and unknown vocabulary in cross validation dataset and test dataset with respect to the known vocabulary from training samples. The datasets are labelled as offensive label, not-offensive label and not-tamil label. The occurrence of "not-tamil" label in the given dataset is minimum in count, so the samples of "not-tamil" labels are dropped in text pre-processing stage. 30% of training samples is treated as the cross validation data sets. Since samples are imbalanced,it is necessary to make them as a balanced training samples using SMOTE method. The imblearn package from python is used to perform SMOTE[21].

The training samples has been trained by logistic classifier and SVM classifier with certain parameters.The Logistic classifier and SVM models can be trained using open source python

Table 1
HASOC-Dravidian-CodeMix-FIRE 2021 Dataset

	Sample count in Training set		Sample count in Testing set
	Actual Dataset	Regenerated Dataset	
Not Offensive Class	4724	4724	536
Offensive Class	1153	4724	118
Not Tamil	3	-	-
Average Length of sentence	16		17
Maximum Length of sentence	113		164
Minimum Length of sentence	1		2

Table 2
Description of actual dataset based on number of vocabulary

	Vectorization					
	Uni-Gram		Bi-Gram		Tri-Gram	
Number of Vocabulary in the Training Set	19208		44604		48345	
	CV *	Test	CV *	Test	CV *	Test
Number of Unknown Vocabulary with respect to Training Set	14883	16722	41491	43033	46842	47638
Number of Non offensive sample with atleast one unknown vocabulary	537	338	120	62	72	31
Number of Offensive sample with atleast one unknown vocabulary	13	4	6	0	6	0

* Cross Validataion dataset

Table 3
Parameters used in Logistic classifier and SVM

	C	max$_{i}teration$	Kernel
Logistic Classifier	1	500	Sigmoid
SVM	1	No limit	linear

package such as sklearn. The parameter value setting in the logistic classifier and SVM classifier are tabulated in Table 3. The parameter C is termed as inverse of regularization strength. If the value of C is larger, SVM classifier minimises the number of misclassified samples and their by making smaller margin of decision boundary.[1]

5. Results and Discussions

The logistic trained model and SVM classifier model is evaluated using labelled test samples by accuracy, f1-score of average weighted by support, precision and recall metrics in the Table 4 and Table 5. The macro average metrics are calculated for each class and is used to determine the average of it without considering imbalanced classes into account. The weighted average

[1] https://github.com/GothainayakiA/Hatesppech.git

Table 4
Classification Report of Logistic Classifier model using TFIDF vector

	Imbalanced Dateset			Balanced Dataset		
	Precision	Recall	f1-Score	Precision	Recall	f1-Score
Not Offensive class	0.820	1.000	0.901	0.833	0.910	**0.873**
Offensive class	0.000	0.000	0.000	0.333	0.203	**0.253**
Accuracy			**0.820**			0.783
Macro Average	0.410	0.500	0.450	0.586	0.557	**0.563**
Weighted Average	0.672	0.820	0.738	**0.747**	**0.783**	**0.761**

Table 5
Classification Report of SVM Classifier model using TFIDF vector

	Imbalanced Dateset			Balanced Dataset		
	Precision	Recall	f1-Score	Precision	Recall	f1-Score
Not Offensive class	0.823	1.000	0.903	0.837	0.950	**0.890**
Offensive class	1.000	0.025	0.050	0.413	0.161	**0.232**
Accuracy			**0.824**			**0.807**
Macro Average	0.912	0.513	0.476	0.625	0.555	0.561
Weighted Average	**0.855**	**0.824**	0.749	0.761	0.807	**0.771**

metrics which calculate the average weight of number of true instance for each class.
In the Table 4, f1-score of offensive class has been improved to 0.253 and overall weighted average f1-score of balanced dataset by SMOTE is increased from 73.8% to 76.1% in logistic classifier. Similarly, in SVM classifier as shown in Table 5, f1-score of offensive class has been improved to 0.23 and overall weighted average f1-score of balanced dataset by SMOTE is increased from 74.9% to 77.1%. The number of unknown vocabulary described in Table 2 is maximum in cross validation set and testing set as compared to training dataset. This leads to misclassification and reaches an average accuracy.

6. Conclusion

The task of identifying offensive language for the dataset given in HASOC-Dravidian CodeMix FIRE 2021[24] is performed by using TFIDF Vectorisation methods and trained on logistic classifier model and SVM classifier model. It is observed that the models are trained with imbalanced samples provides biased predictions to one specific class. Hence, to improve the level of biased prediction to certain class, the oversampling technique is used to generate new labelled dataset from the existing dataset. The generated balanced dataset is trained on logistic classifier and SVM classifier. It is concluded that there is an improvement in average weighted f1-score prediction by 2.3% and 2.2% with logistic classifier model and SVM classifier model respectively. However, the occurrence of unknown vocabularies in the cross validation and test set is possible, contextual based word representation to the unknown vocabulary may be applied. In future SMOTE can be performed for pre-trained models like word2vec,fastText and also custom trained model of word vectorisation and the model to be trained using sequential neural network like RNN,LSTM,GRU.

References

[1] T. Davidson, D. Warmsley, M. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, in: Proceedings of the International AAAI Conference on Web and Social Media, volume 11, 2017.

[2] S. Kaur, P. Kumar, P. Kumaraguru, Automating fake news detection system using multi-level voting model, Soft Computing 24 (2020) 9049–9069. URL: https://doi.org/10.1007/s00500-019-04436-y. doi:10.1007/s00500-019-04436-y.

[3] A. Muneer, S. M. Fati, A comparative analysis of machine learning techniques for cyberbullying detection on twitter, Future Internet 12 (2020). URL: https://www.mdpi.com/1999-5903/12/11/187. doi:10.3390/fi12110187.

[4] V. Pathak, M. Joshi, P. Joshi, M. Mundada, T. Joshi, KBCNMUJAL@HASOC-Dravidian-CodeMixFIRE2020: Using machine learning for detection of hate speech and offensive code-mixed social media text, CEUR Workshop Proceedings 2826 (2020) 351–361.

[5] B. R. Chakravarthi, V. Muralidaran, R. Priyadharshini, J. P. McCrae, Corpus creation for sentiment analysis in code-mixed Tamil-English text (2020) 202–210. URL: https://www.aclweb.org/anthology/2020.sltu-1.28.

[6] S. Swaminathan, H. K. Ganesan, R. Pandiyarajan, HRS-TECHIE@Dravidian-CodeMix and HASOC-FIRE2020: Sentiment analysis and hate speech identification using machine learning, deep learning and ensemble models, CEUR Workshop Proceedings 2826 (2020) 241–252.

[7] A. V. Mandalam, Y. Sharma, Sentiment Analysis of Dravidian Code Mixed Data, Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages (2021) 46–54. URL: https://www.aclweb.org/anthology/2021.dravidianlangtech-1.6.

[8] S. Manochandar, M. Punniyamoorthy, Scaling feature selection method for enhancing the classification performance of Support Vector Machines in text mining, Computers and Industrial Engineering 124 (2018) 139–156. URL: https://doi.org/10.1016/j.cie.2018.07.008. doi:10.1016/j.cie.2018.07.008.

[9] K. Sreelakshmi, B. Premjith, K. P. Soman, Detection of Hate Speech Text in Hindi-English Code-mixed Data, Procedia Computer Science 171 (2020) 737–744. URL: https://doi.org/10.1016/j.procs.2020.04.080. doi:10.1016/j.procs.2020.04.080.

[10] T. Y. Santosh, K. V. Aravind, Hate speech detection in Hindi-English code-mixed social media text, ACM International Conference Proceeding Series (2019) 310–313. doi:10.1145/3297001.3297048.

[11] B. R. Chakravarthi, R. Priyadharshini, N. Jose, A. K. M, T. Mandl, P. K. Kumaresan, R. Ponnusamy, R. L. Hariharan, J. P. Mccrae, E. Sherly, Findings of the Shared Task on Offensive Language Identification in Tamil , Malayalam , and Kannada (2021) 133–145.

[12] S. Banerjee, A. Jayapal, S. Thavareesan, Nuig-shubhanker@dravidian-codemix-fire2020: Sentiment analysis of code-mixed dravidian text using xlnet, 2020.

[13] N. P. Ram, V. M. Vinod, V. Mekala, M. Manimegalai, A fast and energy efficient path

planning algorithm for offline navigation using SVM classifier, International Journal of Scientific and Technology Research 9 (2020) 2082–2086.

[14] A. Hande, R. Priyadharshini, B. R. Chakravarthi, KanCMD: Kannada CodeMixed Dataset for Sentiment Analysis and Offensive Language Detection, Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media (2020) 54–63. URL: https://www.aclweb.org/anthology/2020.peoples-1.6.

[15] B. R. Chakravarthi, N. Jose, S. Suryawanshi, E. Sherly, J. P. McCrae, A sentiment analysis dataset for code-mixed Malayalam-English (2020) 177–184. URL: https://www.aclweb.org/anthology/2020.sltu-1.25.

[16] N. P. Ram, K. Sandhiya, V. M. Vinod, V. Mekala, Offline navigation: Gps based assisting system in sathuragiri forests using machine learning, in: 2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW), 2018, pp. 326–331. doi:10.1109/I2C2SW45816.2018.8997523.

[17] B. R. Chakravarthi, V. Muralidaran, R. Priyadharshini, J. P. McCrae, Corpus Creation for Sentiment Analysis in Code-Mixed Tamil-English Text (2020) 202–210. URL: http://arxiv.org/abs/2006.00206. doi:10.5281/zenodo.4015253. arXiv:2006.00206.

[18] B. R. Chakravarthi, R. Priyadharshini, V. Muralidaran, S. Suryawanshi, N. Jose, E. Sherly, J. P. McCrae, Overview of the track on Sentiment Analysis for Dravidian Languages in Code-Mixed Text, ACM International Conference Proceeding Series (2020) 21–24. doi:10.1145/3441501.3441515.

[19] Z. Al-Makhadmeh, A. Tolba, Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach, Computing 102 (2020) 501–522. URL: https://doi.org/10.1007/s00607-019-00745-0. doi:10.1007/s00607-019-00745-0.

[20] A. Al-Hassan, H. Al-Dossari, Detection of Hate Speech in Social Networks: a Survey on Multilingual Corpus (2019) 83–100. doi:10.5121/csit.2019.90208.

[21] G. Lemaître, F. Nogueira, C. K. Aridas, Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning, Journal of Machine Learning Research 18 (2017) 1–5. URL: http://jmlr.org/papers/v18/16-365.html.

[22] G. Douzas, F. Bacao, Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE, Information Sciences 501 (2019) 118–135. URL: https://doi.org/10.1016/j.ins.2019.06.007. doi:10.1016/j.ins.2019.06.007.

[23] S. Kiyohara, T. Miyata, T. Mizoguchi, Prediction of grain boundary structure and energy by machine learning 18 (2015) 1–5. URL: http://arxiv.org/abs/1512.03502. doi:10.1126/sciadv.1600746. arXiv:1512.03502.

[24] B. R. Chakravarthi, P. K. Kumaresan, R. Sakuntharaj, A. K. Madasamy, S. Thavareesan, P. B, S. Chinnaudayar Navaneethakrishnan, J. P. McCrae, T. Mandl, Overview of the HASOC-DravidianCodeMix Shared Task on Offensive Language Detection in Tamil and Malayalam, in: Working Notes of FIRE 2021 - Forum for Information Retrieval Evaluation, CEUR, 2021.