CUSATNLP@HASOC-Dravidian-CodeMix-FIRE2020: Identifying Offensive Language from Manglish Tweets

Sara Renjit[a], Sumam Mary Idicula[b]

[a]Department of Computer Science, Cochin University of Science and Technology, Kerala, India
[b]Department of Computer Science, Cochin University of Science and Technology, Kerala, India

Abstract

With the popularity of social media, communications through blogs, Facebook, Twitter, and other platforms have increased. Initially, English was the only medium of communication. Fortunately, now we can communicate in any language. It has led to people using English and their own native or mother tongue language in a mixed form. Sometimes, comments in other languages have English transliterated format or other cases; people use the intended language scripts. Identifying sentiments and offensive content from such code mixed tweets is a necessary task in these times. We present a working model submitted for Task2 of the sub-track HASOC Offensive Language Identification- DravidianCodeMix in Forum for Information Retrieval Evaluation, 2020. It is a message level classification task. An embedding model-based classifier identifies offensive and not offensive comments in our approach. We applied this method in the Manglish dataset provided along with the sub-track.

Keywords

Offensive Language, Social Media Texts, Code-Mixed, Embeddings, Manglish

1. Introduction

As code-mixing has become very common in present communication media, detecting offensive content from code mixed tweets and comments is a crucial task these days [1, 2]. Systems developed to identify sentiments from the monolingual text are not always suitable in a multilingual context. Hence we require efficient methods to classify offensive and non-offensive content from multilingual texts. In this context, two tasks are part of the HASOC FIRE 2020 sub-track [3]. The first task deals with the message-level classification of code mixed YouTube comments in Malayalam, and the second task deals with the message-level classification of tweets or Youtube comments in Tanglish and Manglish (Tamil and Malayalam using written using Roman characters). Tamil and Malayalam languages are Dravidian languages spoken in South India [4, 5, 6, 7].

The following sections explain the rest of the contents: Section 2 presents related works in offensive content identification. Task description and dataset details are included in Section

FIRE 2020: Forum for Information Retrieval Evaluation, December 16-20, 2020, Hyderabad, India
sararenjit.g@gmail.com (S. Renjit); sumam@cusat.ac.in (S.M. Idicula)
0000-0002-9932-2039 (S. Renjit)

3. Section 4 explains the methodology used. Section 5 relates to experimental details and evaluation results. Finally, Section 6 concludes the work.

2. Related Works

We discuss works done related to offensive content identification in the past few years. Offensive content detection from tweets is part of some conferences as challenging tasks. In 2019, SemEval (Semantic Evaluation) [8] conducted three tasks, out of which the first task was the identification of offensive and non-offensive comments in English tweets. The dataset used was OLID. It has 14000 tweets annotated using a hierarchical annotation model. The training set has 13240 tweets, and a test set has 860 tweets. They used different methods like Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), LSTM with attention, Embeddings from Language Models (ELMo), and Bidirectional Encoder Representations from Transformers (BERT) based systems. Also, few teams attempt traditional machine learning approaches like logistic regression and support vector machine (SVM) [8]. A new corpus developed for sentiment analysis of code-mixed text in Malayalam-English is detailed in [9]. SemEval in 2020 presented offensive language identification in multilingual languages named as OffensEval 2020, as a task in five languages, namely English, Arabic, Danish, Greek, and Turkish [10]. The OLID dataset mentioned above extends with more data in English and other languages. Pretrained embeddings from BERT-transformers, ELMo, Glove, and Word2vec are used with models like BERT and its variants, CNN, RNN, and SVM.

The same task was conducted for Indo-European languages in FIRE 2019 for English, Hindi, and German. The dataset was created by collecting samples from Twitter and Facebook for all the three languages. Different models such as LSTM with attention, CNN with Word2vec embedding, BERT were used for this task. In some cases, top performance resulted from traditional machine learning models, rather than deep learning methods for languages other than English [3]. Automatic approaches for hate speech also includes keyword-based approaches and TF-IDF based multiview SVM, which is described in [11].

3. Task Description & Dataset

HASOC sub-track, Offensive Language Identification of Dravidian CodeMix [12] [13] [14] consists of two tasks: message level classification of YouTube comments in code mixed Malayalam (Task1) and message level classification of Twitter comments in Tamil and Malayalam written using Roman characters(Task2). This task is in continuation with the HASOC, 2019 task as in [15]. The details about corpus creation and its validation are detailed in [16]. Orthographic information is also utilized while creating the corpus, as mentioned in [17]. This paper discusses the methods used for Task 2 classification of twitter comments in Manglish text. The training dataset consists of comments in two different classes for the task, as shown in Table 1.

Table 1
Dataset statistics for Task2: Malayalam Code-Mix

Class	Count
Offensive	1953
Not Offensive	2047
Total	4000

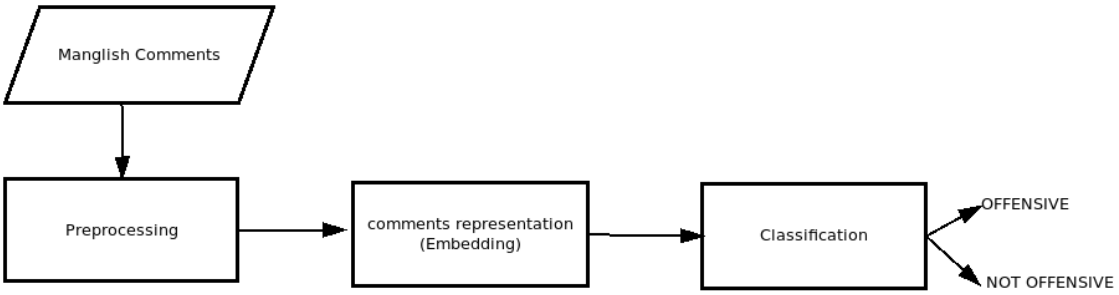

Figure 1: Design of the proposed methods

4. Proposed Methods

We present three submissions experimenting with three different methods based on the general system design, as in Figure 1. The offensive language identification system consists of the following stages:

1. Preprocessing: This stage includes preprocessing texts based on removing English stopwords present in the Manglish comment texts, URLs defined with prefix "www" or "https", usernames with the prefix "@", hash in hashtags, repeated characters, unwanted numbers. All text is converted to lowercase and tokenized into words. Tweet preprocessor [1] for cleaning tweets is used to remove hashtags, URLs, and emojis.
2. Comments representation: Here, we embed the comments based on two embedding mechanisms:
    - Using Keras embedding[2], we represent the sentences using one-hot representation, adequately padded to a uniform length, and passed to Keras embedding layer to produce 50-dimensional sentence representations.
    - Using paragraph vector, an unsupervised framework that learns distributed representations from texts of variable length [18]. The paragraph vector is an algorithm that uses Word2vec based word vector representations [19].
3. Classification: In this step, the comments represented as n-dimensional vectors are trained with the following network parameters:
    - System A: Classifier with an LSTM layer and recurrent dropout(0.2) followed by a

[1]https://pypi.org/project/tweet-preprocessor
[2]https://keras.io/api/layers/core_layers/embedding

Table 2
Evaluation results for System A: based on Keras Embedding

System A	Precision	Recall	F1-score	Support
NOT	0.52	0.60	0.56	488
OFF	0.55	0.48	0.51	512
micro avg	0.54	0.54	0.54	1000
macro avg	0.54	0.54	0.53	1000
weighted avg	0.54	0.54	0.53	1000

dense layer with sigmoid activation and binary cross-entropy classifies comments as offensive or not offensive in 5 epochs.

- System B: Classifier with three dense layers with Relu activation and the final layer is dense with sigmoid activation and binary cross-entropy and trained for 50 epochs for classification.
- System C: Combination of two classifiers: We use a mathematical combination of predictions from both classifiers to produce the third submission results, based on a decision function. Prob(X) denotes the probability values output by system X, and Pred(X) denotes the predicted class of System X.

  Decision function:

  if Pred(A) == Pred(B) **then**
  Pred(C)= Pred(A or B)
  else if Prob(A)+Prob(B) greater than 1 **then**
  Pred(C) = "Offensive"
  else
  Pred(C) = "Not Offensive"
  end if

5. Experimental Results

The proposed system[3] is trained on 4000 comments from the training set and tested on 1000 comments. The weighted average F1-score is used for evaluation as it is an imbalanced binary classification task. Table 2 shows the performance of System A based on Keras embedding and LSTM layer, Table 3 shows System B's results based on document embedding using Doc2Vec, and Table 4 is the result of the combined classifier based on mathematical logic. We use precision, recall, and F1-score as evaluation metrics. The weighted average F1-score is more significant as it handles class imbalance. Its presence in the data sample weights each class score, showing a balance between precision and recall. We calculate these metrics using the Scikit-Learn[4] package. All the results show an average performance, which shows the scope for improvement.

[3]https://github.com/SaraRenG/Code1
[4]https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

Table 3
Evaluation results for System B: based on Doc2Vec

System B	Precision	Recall	F1-score	Support
NOT	0.49	0.67	0.56	488
OFF	0.51	0.32	0.40	512
micro avg	0.49	0.49	0.49	1000
macro avg	0.50	0.50	0.48	1000
weighted avg	0.50	0.49	0.48	1000

Table 4
Evaluation results for System C: based on Decision function

System C	Precision	Recall	F1-score	Support
NOT	0.53	0.49	0.51	488
OFF	0.55	0.58	0.57	512
micro avg	0.54	0.54	0.54	1000
macro avg	0.54	0.54	0.54	1000
weighted avg	0.54	0.54	0.54	1000

6. Conclusion

This paper presents offensive content identification systems for Manglish tweets or comments. It is as part of the HASOC sub-track in FIRE, 2020. We implemented simple methods using sentence representations and binary classification using neural networks. Significant challenges in this task are the representation of text in Manglish(Malayalam written using Roman characters), it's embedding without losing much information, which we can further improve in future attempts.

References

[1] R. Priyadharshini, B. R. Chakravarthi, M. Vegupatti, J. P. McCrae, Named entity recognition for code-mixed indian corpus using meta embedding, in: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 68–72.

[2] N. Jose, B. R. Chakravarthi, S. Suryawanshi, E. Sherly, J. P. McCrae, A survey of current datasets for code-switching research, in: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 136–141.

[3] T. Mandl, S. Modha, P. Majumder, D. Patel, M. Dave, C. Mandlia, A. Patel, Overview of the HASOC Track at FIRE 2019: Hate Speech and Offensive Content Identification in Indo-European Languages, in: Proceedings of the 11th Forum for Information Retrieval Evaluation, FIRE '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 14–17. URL: https://doi.org/10.1145/3368567.3368584. doi:10.1145/3368567.3368584.

[4] B. R. Chakravarthi, M. Arcan, J. P. McCrae, Improving wordnets for under-resourced languages using machine translation, in: Proceedings of the 9th Global WordNet Conference (GWC 2018), 2018, p. 78.

[5] B. R. Chakravarthi, M. Arcan, J. P. McCrae, Wordnet gloss translation for under-resourced languages using multilingual neural machine translation, in: Proceedings of the Second Workshop on Multilingualism at the Intersection of Knowledge Bases and Machine Translation, 2019, pp. 1–7.

[6] B. R. Chakravarthi, M. Arcan, J. P. McCrae, Comparison of Different Orthographies for Machine Translation of Under-resourced Dravidian Languages, in: 2nd Conference on Language, Data and Knowledge (LDK 2019), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

[7] B. R. Chakravarthi, P. Rani, M. Arcan, J. P. McCrae, A survey of orthographic information in machine translation, arXiv e-prints (2020) arXiv–2008.

[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), in: Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, pp. 75–86.

[9] B. R. Chakravarthi, N. Jose, S. Suryawanshi, E. Sherly, J. P. McCrae, A sentiment analysis dataset for code-mixed Malayalam-English, in: Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL), European Language Resources association, Marseille, France, 2020, pp. 177–184. URL: https://www.aclweb.org/anthology/2020.sltu-1.25.

[10] M. Zampieri, P. Nakov, S. Rosenthal, P. Atanasova, G. Karadzhov, H. Mubarak, L. Derczynski, Z. Pitenis, Ç. Çöltekin, SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020), arXiv preprint arXiv:2006.07235 (2020).

[11] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, O. Frieder, Hate speech detection: Challenges and solutions, PloS one 14 (2019) e0221152.

[12] B. R. Chakravarthi, M. A. Kumar, J. P. McCrae, P. B, S. KP, T. Mandl, Overview of the track on "HASOC-Offensive Language Identification- DravidianCodeMix", in: Proceedings of the 12th Forum for Information Retrieval Evaluation, FIRE '20, 2020.

[13] 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, in: Proceedings of the 12th Forum for Information Retrieval Evaluation, FIRE '20, 2020.

[14] 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, in: Working Notes of the Forum for Information Retrieval Evaluation (FIRE 2020). CEUR Workshop Proceedings. In: CEUR-WS. org, Hyderabad, India, 2020.

[15] T. Mandl, S. Modha, P. Majumder, D. Patel, M. Dave, C. Mandlia, A. Patel, Overview of the HASOC track at FIRE 2019: Hate Speech and Offensive Content Identification in Indo-European Languages, in: Proceedings of the 11th Forum for Information Retrieval Evaluation, ACM, 2019, pp. 14–17.

[16] B. R. Chakravarthi, V. Muralidaran, R. Priyadharshini, J. P. McCrae, Corpus creation for sentiment analysis in code-mixed Tamil-English text, in: Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL), European Language Resources association, Marseille, France, 2020, pp. 202–210. URL: https://www.

aclweb.org/anthology/2020.sltu-1.28.
[17] B. R. Chakravarthi, Leveraging orthographic information to improve machine translation of under-resourced languages, Ph.D. thesis, NUI Galway, 2020. URL: http://hdl.handle.net/10379/16100.
[18] Q. Le, T. Mikolov, Distributed representations of sentences and documents, in: International conference on machine learning, 2014, pp. 1188–1196.
[19] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781 (2013).