bits2020@Dravidian-CodeMix-FIRE2020: Sub-Word Level Sentiment Analysis of Dravidian Code Mixed Data

Yashvardhan Sharma, Asrita Venkata Mandalam

Department of Computer Science and Information Systems, Birla Institute of Technology and Science Pilani, Pilani Campus

Abstract

This paper presents the methodologies implemented while classifying Dravidian code-mixed comments according to their polarity in the evaluation of the track 'Sentiment Analysis for Davidian Languages in Code-Mixed Text' proposed by Forum of Information Retrieval Evaluation in 2020. The implemented method used a sub-word level representation to capture the sentiment of the text. Using a Long Short Term Memory (LSTM) network along with language-specific preprocessing, the model classified the text according to its polarity. With F1-scores of 0.61 and 0.60, the model achieved an overall rank of 5 and 12 in the Tamil and Malayalam tasks respectively.

Keywords

Sentiment Analysis, Recurrent Neural Networks, Sub-word Analysis

1. Introduction

Code-mixing usually involves two languages to create another language that consists of elements of both in a structurally understandable manner. It has been noted that bilingual and multilingual societies use a code-mix of languages in informal speech and text. Dravidian code-mixed languages, including but not limited to Malayalam and Tamil, are increasingly used by younger generations in advertising, entertainment and social media. The language is commonly written in Roman script.

With the rise in the number of non-English and multilingual speakers using social media, there is an interest in analysing the sentiment of the content posted by them. As code-mixed data does not belong to one language and is often written using Roman script, identifying its polarity cannot be done using traditional sentiment analysis models. In social media, low-resourced languages such as Tamil and Malayalam have been increasingly used along with English. Identifying the sentiment of this data can prove to be useful in social media monitoring and feedback of users towards other online content.

The 'Sentiment Analysis for Davidian Languages in Code-Mixed Text' proposed by FIRE 2020 [1] [2] task contains a code-mixed dataset consisting of comments from social media

FIRE 2020: Forum for Information Retrieval Evaluation, December 16-20, 2020, Hyderabad, India
EMAIL: yash@pilani.bits-pilani.ac.in (Y. Sharma); f20171179@pilani.bits-pilani.ac.in (A. V. Mandalam)
URL: www.bits-pilani.ac.in/pilani/yash/profile (Y. Sharma)

Table 1
Distribution of Data in the Tamil and Malayalam Dataset

Dataset	Positive	Negative	Mixed Feelings	Unknown State	Other Languages
Tamil	10,559	2,037	1,801	850	497
Malayalam	2,811	738	403	1,903	884

websites for both Tamil and Malayalam. Each team had to submit a set of predicted sentiments for the Tamil-English and Malayalam-English mixed test sets [3].

Along with language specific preprocessing techniques, the implemented model makes use of sub-word level representations to incorporate features at the morpheme level, the smallest meaningful unit of any language. Evaluated by F1-score, the presented approach achieved the 5th highest score in the Tamil task and the 12th rank in the Malayalam task. The implemented model is available on Github[1].

2. Related Work

Analysing the sentiment of code-mixed data is important as traditional methods fail when given such data. Barman et al. [4] concluded that n-grams proved to be useful in their experiments that involved multiple languages with Roman script.

Qurat Tul Ain et al. [5] wrote a review paper that highlighted studies regarding the implementation of deep learning models in sentiment analysis models. Due to more hidden layers, deep learning models extracted data that had heavier weights and used those features to proceed.

Bojanowski et al. [6] used character n-grams in their skip-gram model. The lack of preprocessing resulted in a shorter training time and outperformed baselines that did not consider sub-word information. Joshi et al. [7] outperformed existing systems as well by using a sub-word based LSTM architecture. Their dataset consisted of 15% negative, 50% neutral and 35% positive comments. As their dataset was imbalanced like the one used in this paper, the submitted approach involved morpheme extraction as it would help in identifying the polarity of the dataset. In more recent work, Jose et al. [8] surveyed publicly available code-mixed datasets. They noted statistics about each dataset such as vocabulary size and sentence length. Priyadharshini et al. [9] used embeddings of closely related languages of the code-mixed corpus to predict Named Entities of the same corpus.

3. Dataset

The model has been trained, validated and tested using the Tamil [10] and Malayalam [11] datasets provided by the organizers of the Dravidian Code-Mix FIRE 2020 task. The Tamil code-mix dataset consists of 11,335 comments for the train set, 1,260 for the validation set and 3,149 comments for testing the model. In the Malayalam code-mix dataset, there are 4,851

[1]https://github.com/avmand/SA_Dravidian

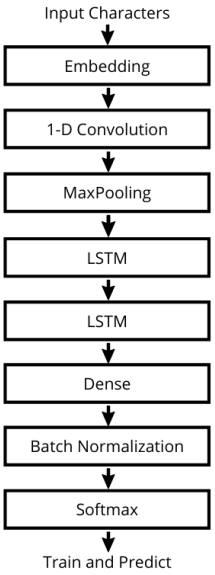

Figure 1: The classification model implemented using sub-word analysis.

comments for training, 541 for validating and 1,348 for testing the model. Table 1 gives the distribution of each sentiment in each dataset.

4. Proposed Technique

Word-level models such as Word2Vec [12] and GloVe [13] are popularly used in a variety of NLP tasks. However, they did not seem to be suited for a code-mixed dataset. It is not possible to use a word-level model due to the sparsity of words in the dataset. The implemented approach uses a sub-word level model as it accounts for words that have a similar morpheme. For example, in the Tamil dataset, *Ivan*, *Ivanga* and *Ivana* have similar meanings due to their root word *Ivan*.

First, the dataset is preprocessed to replace all emojis with their corresponding description in English. As the dataset contains both Roman and Tamil (or Malayalam) characters, the latter is replaced with its corresponding Roman script representation.

From the preprocessed data, a set of characters was obtained . The input to the model is a set of character embeddings. The sub-word level representation is generated through a 1-D convolution layer with activation as ReLU, size of convolutional window as 5 and number of output filters as 128. After getting a morpheme-level feature map, a 1-D maximum pooling layer is used to obtain its most prominent features. To obtain the connections between each of these features, LSTMs are used due to their ability to process sequences and retain information. The first and second LSTM layers have a dropout of 0.4 and 0.2 respectively. Finally, it is passed to a fully connected layer. Batch normalization has been used in the model to prevent overfitting. While training the model, early stopping has been utilized to stop training when the validation loss shows no improvement after 4 epochs. The training data was shuffled before each epoch. Figure 1 gives a representation of the discussed methodology.

Table 2
Classification report for each dataset and class.

Language	Report	Precision	Recall	F1-Score	Support
Tamil	Mixed Feelings	0.26	0.17	0.21	377
	Negative	0.42	0.22	0.29	424
	Positive	0.72	0.91	**0.80**	2075
	Not Tamil	0.71	0.34	0.46	100
	Unknown State	0.67	0.01	0.02	173
	Macro Avg	0.55	0.33	0.35	3149
	Weighted Avg	**0.62**	**0.66**	**0.61**	3149
Malayalam	Mixed Feelings	0.21	0.51	0.30	70
	Negative	0.35	0.55	0.43	138
	Positive	0.73	0.73	**0.73**	565
	Not Malayalam	0.58	0.68	0.63	177
	Unknown State	0.81	0.38	0.52	398
	Macro Avg	0.54	0.57	0.52	3149
	Weighted Avg	**0.67**	**0.59**	**0.60**	1348

5. Result

The submitted run achieved a rank of 5 and 12 for the Tamil and Malayalam tasks respectively. The final rank was evaluated based on the weighted average F1-score. The classification report is shown in Table 2. The Tamil task received Precision, Recall and an F1-score of 0.62, 0.66 and 0.61 respectively. For the Malayalam task, the submission received scores of 0.67, 0.59 and 0.60 respectively.

6. Error Analysis

6.1. Tamil Task

From Table 2, one can see that the F1-score of the positive comments is the highest with a value of 0.80. The next highest score is only at 0.46, attained by the class of comments that are not in Tamil. The order of classes from the highest to the lowest F1-scores are Positive, Not Tamil, Negative, Mixed Feelings and Unknown State. The weighted F1-score is lower than the Precision and Recall as the weighted score takes into account the proportion of each class in the dataset.

Due to the higher number of positive comments in the overall dataset, it is not surprising that the model trains well and produces the best results for that class. Non-Tamil comments get the next highest score due to the different morphemes used in them. These comments are usually in a different Indian language like Hindi or Telugu and are written using the English alphabet. Some comments are written in the script of their respective language. This class does not achieve a higher score due to words that they have in common with the Tamil-English code-mixed tweets such as *Rajinikanth* and *Thalaiva*. The same can be concluded for the negative label as well as it had many words that were common with those of the positive comments. Comments from the mixed feelings class were misclassified as either positive or negative. They

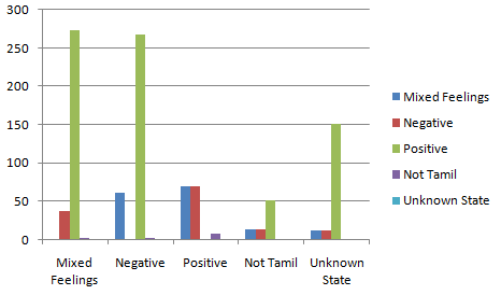

Figure 2: Tamil misclassified comments.

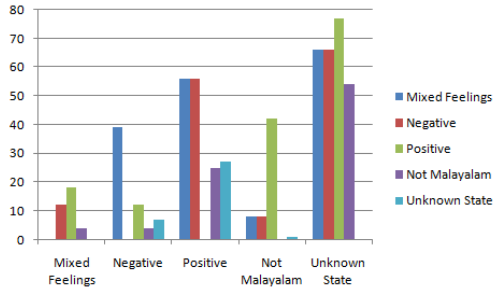

Figure 3: Malayalam misclassified comments.

were not misclassified as comments from the unknown state class possibly due to the relatively lower ratio of unknown comments as compared to the positive and negative classes. As these comments contained both positive and negative sentiments, there was a much higher chance of them being classified into one of those classes. The unknown state class receives the lowest F1-score. Its precision is 0.67 but its recall is very low at 0.01. This implies that there is a high false negative rate and is because all of the comments use words from the Tamil vocabulary. Most of those words are common with those of the positive class. Figure 2 gives a representation of the misclassified Tamil comments.

6.2. Malayalam Task

The classification report of the Malayalam task can be seen in Table 2. The F1-score of the positive comments is the highest at 0.73. The next highest is at 0.63, for the class of comments that are not in Malayalam. The order of classes from the highest to the lowest F1-scores are Positive, Not Malayalam, Unknown State, Negative and Mixed Feelings.

The Malayalam dataset was more balanced as compared to the Tamil dataset with the second largest class less than 1000 comments behind the largest one. Similar to the Tamil dataset, the positive class has the highest number of comments. This led to the relatively higher F1-score and an equally low false positive and false negative rate. For the class of comments that were not in Malayalam, the classifier identified all of the comments that were not written in the Roman or Malayalam script. However, words that were commonly found in positive comments, such as names of Malayalam actors, and were used with English words were classified incorrectly. For the unknown state class, it is noted that the misclassified comments were majorly assigned a positive, negative or mixed feelings tag. Although the overall sentiment of the sentence was unknown, a portion of that sentence had similarities with one of the other classes. For the mixed feelings class, the same was deduced and most of the wrongly classified comments were assigned either a positive or negative tag. Most of the misclassified comments from the negative class were labelled as comments with mixed feelings. Sarcastic comments that used positive words but implied negative sentiments were not accounted for by the model. The distribution of misclassified comments can be seen in Figure 3.

7. Conclusion

This paper presents the submitted approach for the Sentiment Analysis for Dravidian Languages in Code-Mixed Text track of Forum for Information Retrieval Evaluation (FIRE) 2020. The model implemented sub-word level representations along with LSTM networks. Morpheme level representations have proven to be useful in code-mixed data as they club words with similar root words and meanings together. The results show that the positive class in each dataset receives the highest F1-scores. This is possibly due to the higher ratio of the same as compared to the rest of the classes. Comments that were not in the language of their dataset received the next highest score as their vocabulary included sub-words that were not a part of the rest of the datasets. For future work, a sarcasm detection feature could be included to avoid misclassification.

Acknowledgments

The authors would like to convey their sincere thanks to the Department of Science and Technology (ICPS Division), New Delhi, India, for providing financial assistance under the Data Science (DS) Research of Interdisciplinary Cyber Physical Systems (ICPS) Programme [DST/ICPS/CLUSTER/Data Science/2018/Proposal-16:(T-856)] at the department of computer science, Birla Institute of Technology and Science, Pilani, India.

References

[1] 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.

[2] B. R. Chakravarthi, Leveraging orthographic information to improve machine translation of under-resourced languages, Ph.D. thesis, NUI Galway, 2020.

[3] 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.

[4] U. Barman, A. Das, J. Wagner, J. Foster, Code mixing: A challenge for language identification in the language of social media, in: Proceedings of the first workshop on computational approaches to code switching, 2014, pp. 13–23.

[5] Q. T. Ain, M. Ali, A. Riaz, A. Noureen, M. Kamran, B. Hayat, A. Rehman, Sentiment analysis using deep learning techniques: a review, Int J Adv Comput Sci Appl 8 (2017) 424.

[6] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.

[7] A. Joshi, A. Prabhu, M. Shrivastava, V. Varma, Towards sub-word level compositions for sentiment analysis of hindi-english code mixed text, in: Proceedings of COLING 2016, the

26th International Conference on Computational Linguistics: Technical Papers, 2016, pp. 2482–2491.

[8] 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), IEEE, 2020, pp. 136–141.

[9] 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), IEEE, 2020, pp. 68–72.

[10] 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.

[11] 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.

[12] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781 (2013).

[13] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.