SSNCSE_NLP@Dravidian-CodeMix-FIRE2020: Sentiment Analysis for Dravidian Languages in Code-Mixed Text

Nitin Nikamanth Appiah Balaji, B. Bharathi and J. Bhuvana

Department of CSE, Sri Siva Subramaniya Nadar College of Engineering,Tamil Nadu, India

Abstract

Social media has become a place for expressing people's emotions, love, and hatred. With the ubiquitous availability of the internet entertains individuals to spend more time and express their feelings good or bad openly on social media platforms. In this study, we compare and analyze the methods for comment-level text polarity classification task using the Dravidian-CodeMix-FIRE2020 data-set. We contrast machine learning models with features extracted from techniques such as TF, TFIDF, BERT, fastText, and LSTM. The TF and the BERT embedding gave the best results compared. Our models scored F1 scores of 0.61 and 0.71 for the Tamil-English and the Malayalam-English tasks respectively.

Keywords

Machine Learning, NLP, Code-mixed text, Sentiment analysis, BERT embeddings

1. Introduction

The recent boom in technology has exhorted people to spend an extensive amount of time engrossed in social media. Sometimes even a passive expression of emotions could be incendiary to stir up extreme conflict among different groups or individuals. It becomes important to build a reliable model that could limit such messages surfacing the internet. With the usage of native languages increasing precipitously, it becomes important to consider the mixture of native and English languages for comments in use [1, 2, 3, 4, 5]. Considering the massive amount of people using the languages of Malayalam and Tamil in India and across the world, the code-mix of Tamil-English and Malayalam-English becomes necessary to study [6, 7, 8, 9]. This analysis then can be easily interpolated to other Dravidian or other languages as well.

In this analysis different feature extraction models such as count vectorization, LSTM, BERT embedding are implemented, and the performance of each feature extraction technique with various machine learning models are compared. As the data-set consists of social media (YouTube) comments with a mixture of Tamil and Malayalam with English, it is harder to fine-tune a model rather than building a feature extractor from scratch, such as TFIDF and LSTM with one-hot word embedding. But multilingual BERT embedding is considered as it contains weight

FIRE 2020: Forum for Information Retrieval Evaluation, December 16-20, 2020, Hyderabad, India
nitinnikamanth17099@cse.ssn.edu.in (N. N. A. Balaji); bharathib@ssn.edu.in (B. Bharathi); bhuvanaj@ssn.edu.in (J. Bhuvana)
0000-0002-6105-0998 (N. N. A. Balaji); 0000-0001-7279-5357 (B. Bharathi)

Table 1
Data Distribution

Task Description	Data Set	Number of comments
Tamil-English	Train-set	11,335
	Dev-set	1,260
	Test-set	3,149
	Total	**15,744**
Malayalam-English	Train-set	4,851
	Dev-set	541
	Test-set	1,348
	Total	**6,739**

by unsupervised training on a large corpus containing different languages. The BERT model produced results on par with the count vectorization model.

This work is an account of the submissions made to the Dravidian-Codemix challenge [10, 11, 12]. The subsequent sections are arranged as follows: The Section 2 explaining the data-set distribution and preprocessing steps. The Section 3 details the experimental setup and the various feature that was trialed for the task. The Section 4 provides a subjective analytical comparison of the performance of various models in the test-data. Finally the Section 5 briefs the objectives.

2. Data Description

The data-set consists of YouTube comments with message-level sentiment polarity labels. The classes for the tasks include positive, negative, neutral, mixed emotions, or if the comment is not in the intended language for label. This becomes a multi-class classification task. The individual comments contain an average of sentence length 1, making it more balanced and easier to analyze. But there is some imbalance in the classes as it simulates the real-life scenario. The data-set distribution among the training, development, and test sets are described in table 1. More detailed description on the data-set is provided in [13] and [14] for the Tamil-English and Malayalam-English tasks respectively.

3. Experiments

The experimental structure for the task can be expressed in two stages - the feature extraction stage and the classifier stage. Techniques such as Count vectorization, TFIDF vectorization, BERT, fastText are analyzed for the feature extraction stage and different classifiers such as Logistic regression, Multi Layer Perceptron, Naive Bayes, and Random Forest classifiers are compared. The machine learning algorithms, count, and TFIDF vectorization are studied using the sci-kit learn [1] implementations. The sentence transformers [15] implementation for the

[1] https://scikit-learn.org/stable/

multilingual BERT and the pymagnitude [16] implementation for fastText is considered. The Keras version of the LSTM was used for this study.

The features extracted from the first stage are used to train the machine learning models in the second stage and their performance is compared using the F1 score and the accuracy score. The metrics methods of the sci-kit learn package is used to measure the performance. The implementation with experimented and selected hyper-parameters are available in the link [2].

3.1. Count And TFIDF Vectorization

The content of the text comments is a mix of various languages, their syntax, and the interchanging between different symbols. It becomes hard to capture the coherent intensity of the comments with the extant pre-trained models. So bag of words based word and char count models are deployed and analyzed by varying the n-gram range. The n-gram range of 2-3 and 1-5 gave the optimal result on the dev-set for Tamil-English and Malayalam-English sub-tasks respectively. The Term Frequency Inverse Document frequency model helps to give a lesser weight-age to the banal words in the corpus. This technique emphasizes more on the unique terms in the corpus than the repeated words, rendering a better model.

3.2. Bidirectional LSTM

Bidirectional Long short-term memory network is trained using word one-hot embedding. This structure helps to learn the semantic syntax of the mixing of different languages and consolidates structures from the comments. An LSTM could get relations from long distance in the sequence. A multi-input, single-output RNN network is constructed with a single layer 1024 dimension hidden biLSTM layer and a dense layer. The words are converted into one-hot vectors of 150-time frames and 100 dimensions for each word. This vector is fed to the LSTM network. The network is trained for 7 epochs with a batch size of 128 and a learning rate of 0.001.

3.3. Multilingual Embedding Models

The YouTube comments selected for the study contains text from English and Dravidian languages coalesced together. This becomes a major problem to consider when applying monolingual pre-trained models and fine-tuning for this particular task. But with the option of pre-trained models in an unsupervised manner on a large collection of languages, it is possible to fine-tune such multilingual models for fitting well for the Code-mix application.

As the fastText and the BERT multilingual models [17] had shown fruitful results, it is considered for this experiment. For FastText, a fixed length of 300 dimension vector is generated by averaging the word-wise vectors of the entire sentence with the pymagnitude implementation [16]. The Tamil specific and Malayalam specific pre-trained models from the FastText multilanguage resources is used [3]. Similarly a 512 dimension vector is generated by the BERT (distiluse-

[2] https://github.com/nikamanthab/SSN_NLP-FIRE2020/tree/master/Codemix
[3] https://fasttext.cc/docs/en/crawl-vectors.html

Table 2
Results of dev-set for Tamil-English and Malayalam-English sub-tasks.

Task	Features	n-gram	Classifier	Precision	Recall	F1 score
Tamil-English	char-count vec	2-3	MLP	0.62	0.64	**0.63**
	char-count vec	2-3	NB	0.61	0.63	0.62
	char-TFIDF	2-3	MLP	0.61	0.63	0.62
	char-TFIDF	2-3	NB	0.61	0.63	0.62
	biLSTM	-	softmax	0.59	0.65	0.61
	multi BERT	-	MLP	0.61	0.69	**0.63**
	multi fastText	-	MLP	0.64	0.69	0.62
Malayalam-English	char-count vec	1-5	LR	0.72	0.71	**0.71**
	TFIDF	1-5	LR	0.72	0.71	**0.71**
	TFIDF	1-5	NB	0.72	0.71	0.70
	biLSTM	-	softmax	0.64	0.65	0.63
	multi BERT	-	MLP	0.67	0.67	0.66
	multi fastText	-	MLP	0.63	0.63	0.60

base-multilingual-cased) pre-trained model from the SentenceTransformers implementation [15]. The extracted features are used to train a classification model. The Multi Layer Perceptron with 0.001 learning rate trained for 25 iterations shined better when compared to the Random Forest classifier or the Naive Bayes classifier.

4. Observations

The code-mix data-set presents a new challenge of applying alternating symbols and syntax from majorly two different languages - English and a Dravidian language. Due to this very reason, the primitive pre-trained fastText model showed relatively poorer results than the models trained from scratch. But in contrast, the multilingual BERT which is an attention-based transformer model, much convoluted from trained on a larger corpus produced comparable results to that of the TFIDF and count vectorization models. The biLSTM didn't perform on par with the vectorization models but showed better performance than the fastText model.

Out of all the analyzed models the count vectorization and the BERT model produced the best performance for the Tamil-English corpus. For the Malayalam-English corpus the TFIDF and count vectorization techniques generated equal performance models. The performance of each model on the dev-set is presented in Table 2 and the results on the test-set is presented in Table 3.

5. Conclusion

Social media platforms are growing rapidly and entrench more and more people in taking part in these platforms. Even though some opinions may be acceptable by one, it may be hurting for others, so it becomes necessary to devise an effective model for the sentiment polarity detection for the text comments. In this study, we have analyzed a variety of feature extraction techniques

Table 3
Results of test-set for Tamil-English and Malayalam-English sub-tasks.

Task	Features	n-gram	Classifier	Precision	Recall	F1 score
Tamil-English	char-count vec	2-3	MLP	0.59	0.64	**0.61**
	biLSTM	-	softmax	0.58	0.63	0.60
	multi BERT	-	MLP	0.59	0.66	**0.61**
Malayalam-English	char-TFIDF vec	1-5	LR	0.70	0.71	**0.71**
	biLSTM	-	softmax	0.64	0.66	0.64
	multi BERT	-	MLP	0.62	0.62	0.62

and conclude that the Count, TFIDF based vectorization, and multilingual BERT technique performs well on code-mix polarity labeling task. With these features, we reach a weighted F1 score of 0.61 for the Tamil-English task and 0.71 for the Malayalam-English tasks respectively.

References

[1] 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.

[2] 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.

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

[4] P. Rani, S. Suryawanshi, K. Goswami, B. R. Chakravarthi, T. Fransen, J. P. McCrae, A comparative study of different state-of-the-art hate speech detection methods for Hindi-English code-mixed data, in: Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying, European Language Resources Association (ELRA), Marseille, France, 2020.

[5] S. Suryawanshi, B. R. Chakravarthi, P. Verma, M. Arcan, J. P. McCrae, P. Buitelaar, A dataset for troll classification of Tamil memes, in: Proceedings of the 5th Workshop on Indian Language Data Resource and Evaluation (WILDRE-5), European Language Resources Association (ELRA), Marseille, France, 2020.

[6] 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, The Global WordNet Conference 2018 Committee, 2018. URL: http://compling.hss.ntu.edu.sg/events/2018-gwc/pdfs/GWC2018_paper_16.

[7] 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), volume 70 of *OpenAccess Series in Informatics (OASIcs)*, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2019, pp.

6:1–6:14. URL: http://drops.dagstuhl.de/opus/volltexte/2019/10370. doi:10.4230/OASIcs.LDK.2019.6.

[8] 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, European Association for Machine Translation, Dublin, Ireland, 2019, pp. 1–7. URL: https://www.aclweb.org/anthology/W19-7101.

[9] B. R. Chakravarthi, R. Priyadharshini, B. Stearns, A. Jayapal, S. S, M. Arcan, M. Zarrouk, J. P. McCrae, Multilingual multimodal machine translation for Dravidian languages utilizing phonetic transcription, in: Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages, European Association for Machine Translation, Dublin, Ireland, 2019, pp. 56–63. URL: https://www.aclweb.org/anthology/W19-6809.

[10] 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.

[11] 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.

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

[13] 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.

[14] 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.

[15] N. Reimers, I. Gurevych, Making monolingual sentence embeddings multilingual using knowledge distillation, arXiv preprint arXiv:2004.09813 (2020). URL: http://arxiv.org/abs/2004.09813.

[16] A. Patel, A. Sands, C. Callison-Burch, M. Apidianaki, Magnitude: A fast, efficient universal vector embedding utility package, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2018, pp. 120–126.

[17] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).