Sentiment Analysis on Code-Switched Dravidian Languages with Kernel Based Extreme Learning Machines

Mithun Kumar S R
Uber R&D India, Bangalore
mithunkumar.sr@uber.com,

Lov Kumar
BITS Pilani, Hyderabad
(lovkumar,

Aruna Malapati
BITS Pilani, Hyderabad
arunam)@hyderabad.bits-pilani.ac.in

Abstract

Code-switching refers to the textual or spoken data containing multiple languages. Application of natural language processing (NLP) tasks like sentiment analysis is a harder problem on code-switched languages due to the irregularities in the sentence structuring and ordering. This paper shows the experiment results of building a Kernel based Extreme Learning Machines(ELM) for sentiment analysis for code-switched Dravidian languages with English. Our results show that ELM performs better than traditional machine learning classifiers on various metrics as well as trains faster than deep learning models. We also show that Polynomial kernels perform better than others in the ELM architecture. We were able to achieve a median AUC of 0.79 with a polynomial kernel.

1 Introduction

Because of the expansion of user-generated material, it is now possible to automatically detect linked attitudes. A "sentiment" is a good or negative opinion, emotion, feeling, or thinking conveyed by a sentiment bearer (user). In general, sentiment analysis attempts to extract certain sentiments from text automatically (Sakuntharaj and Mahesan, 2021, 2017, 2016; Thavareesan and Mahesan, 2019, 2020a,b, 2021). Sentiment analysis seeks to analyse textual patterns in order to find a sentiment at the word, phrase, or document level. Sentiment analysis is widely used in a variety of sectors today, including public-health monitoring, electoral patterns, predicting terrorist actions, and social network analysis (Sampath et al., 2022; Ravikiran et al., 2022).

Dravidian languages, Tamil, Kannada and Malayalam are widely spoken by over 250 million people, but still is a a sparse language for NLP tasks (Chakravarthi et al., 2021, 2022; Bharathi et al., 2022; Priyadharshini et al., 2022). Dravidian languages are a spoken mostly in southern India, north-east Sri Lanka, and south-west Pakistan (Anita and Subalalitha, 2019b,a; Subalalitha and Poovammal, 2018). There have been tiny but important immigrant groups in Mauritius, Myanmar, Singapore, Malaysia, Indonesia, the Philippines, the United Kingdom, Australia, France, Canada, Germany, South Africa, and the United States since the colonial era (Subalalitha, 2019; Srinivasan and Subalalitha, 2019; Narasimhan et al., 2018). Tamil is a member of the southern branch of the Dravidian languages, a group of about 26 languages indigenous to the Indian subcontinent. It is also classed as a member of the Tamil language family, which contains the languages of around 35 ethnolinguistic groups, including the Irula and Yerukula languages.

The influence of English in the regions where these languages are spoken is higher due to the colonial history and the medium of schooling (Priyadharshini et al., 2021; Kumaresan et al., 2021). However the ease of expression of sentiments switches between the words in the Dravidian language and English with most of the bilinguists versatile in both, especially on online social platforms (Chakravarthi, 2020; Chakravarthi and Muralidaran, 2021). The sentiment analysis of text written in code-switched language between the Dravidian languages and English is analysed in this paper through a novel kernel based ELM.

,

2 Related work

Multi-class classification of text sentiment has been approached in both, traditional machine learning models as well as in deep learning models in the past. Chakravarthi et al. has previously shown the performance of traditional classifiers for Dravidian

Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages, pages 184 - 190

Language	Positive	Negative	Mixed Feelings	Unknown State
Tamil	20,070	4,271	4,020	5,628
Malayalam	6,421	2,105	926	5,279
Kannada	2,823	1,188	574	711

Table 1: Data split between various classes.

languages. Kumar et al. (2021) showed that the performance metrics was the best with ensemble models in Dravidian language code-mixed dataset. Deep learning models like LSTM have been used by Yadav and Chakraborty (2020) for sentiment classification. However most of the pre-trained models like BERT takes as longer as 84 hours to train and there are optimisation efforts on reducing the time as experimented by You et al. (2020). One of the parallel optimisation technique on neural network is to use a single layer hidden layer which is explored in Extreme learning Machines (ELM) by Huang et al. (2004). There has been no work so far in exploring ELM on code-switched languages and hence this paper explores the possibility of using ELM for sentiment analysis. The following research questions (RQ) are explored through our experiments.

- **RQ1:** Will ELM be faster to train than deep-learning models and yield better results for sentiment analysis on code-switched languages?
- **RQ2:** Will sentiment analysis models perform better with dimensionality reduction, word embedding and data balancing techniques, which we hypothesise to be true.

3 Dataset

We conducted our experiments on the labelled data from the YouTube comments using three code-mixed benchmark datasets published for Dravidian languages. Kannada code-switched corpus, published by Hande et al. (2020) was our primary source. Similarly Tamil code-switched corpus, published by Chakravarthi et al. (2020b) was used. For Malayalam code-switched corpus, we used the data published by Chakravarthi et al. (2020a).

The multi-class dataset contains manually labelled sentiments for code-switched data. This dataset is an imbalanced one with a skew towards the labels containing 'Positive' sentiments. The split between various classes is shown in Table 1.

4 Experiment Setup

A multi-staged pipeline was setup for our experiments as depicted in Figure 1.

4.1 Data preprocessing

The raw corpus in code-switched languages were preprocessed with steps such as case conversion, removing stopwords and emoticons, lemmatizing to retain only the root form of the morpheme. Most of the preprocessing was done using NLTK[1]. Labels in the original dataset were 'Positive', 'Negative', 'Mixed Feelings', 'Unknown State' and 'Not in the target language'. Since we were using an explicit language identifier, langdetect[2], and primarily focusing on sentiment classification, we removed the data with the label 'Not in the target language' and retained the rest for our training.

4.2 Word embedding

Our focus during the experiment was to use a language specific word embedding technique. One such pre-trained word embedding model is provided by FastText[3] in multiple languages including Tamil, Kannada and Malayalam. Sentence vectorisation after the language identification was done using the pre-trained FastText word vectors in 300 dimensions on the preprocessed dataset.

4.3 Feature selection

The vectorised sentences along with the labels after the word embedding was either retained as-is, with all the features (All) or was subjected to dimensionality reduction using Principal Component Analysis (PCA). Two different datasets were created for each of the languages, one with All and the other constrained through PCA.

4.4 Data balancing techniques

Since the data is skewed, the vectorised dataset was then subjected to data balancing techniques. We wanted to study the effect of both, imbalanced as

[1]https://www.nltk.org/
[2]https://pypi.org/project/langdetect/
[3]https://fasttext.cc/docs/en/crawl-vectors.html

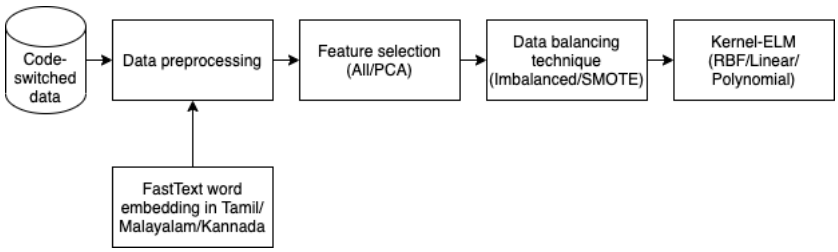

Figure 1: Pipeline of the experimental setup

well as the balanced data. Hence we created two other copies of the data. The first was to retain the data imbalance. The second was to overcome the class imbalance using an oversampling technique, Synthetic Minority Over-sampling TEchnique (SMOTE). This used synthetic minority class samples to build a dataset of equal number of samples in all classes. The dataset was then subjected to a split of training and test data. Data normalisation was done using a 5-fold cross validation on the dataset.

4.5 Kernel-ELMs

We setup a Extreme Learning Machine (ELM) through a single layer feed forward neural network with the same number of hidden layer nodes as the dimension of the sentence vectors in the dataset. The activation layer was through various kernels like Radial Basis Functions (RBF), Linear and Polynomial. Each set of data was trained and evaluated through the Kernel-based ELM. We also ensured that 98% of the variance in the data is present. The training time was around 60 minutes for most of the languages which was faster than deep learning model training time.

5 Observations and Analysis

The combination of features and data-balancing techniques from the pipeline was evaluated separately with each of the ELM kernels. Performance metrics like accuracy as well as the receiver operating characteristic (ROC) curve was determined for each of the dataset. We also measured the Area under the ROC curve (AUC) for each combination of the dataset as observed in Table 2.

5.1 Accuracy analysis

One of the major observations was that all the code-switched languages in combination with the features and data balancing techniques was yielding the best accuracy when all the features were selected instead of dimensionality constraining with techniques like PCA. Balancing techniques like SMOTE was worsening the accuracy instead of bettering it. This pattern is observed with all the language datasets irrespective of the Kernel chosen. Our hypothesis is that this might be due to the over-generalisation with the minority synthetic dataset which might be from the overlapping areas. Since there is larger and less specific decision boundary in SMOTE, there is also a possibility of augmenting noisy regions as also studied by Santos et al. (2018).

5.2 Kernel analysis

One of our research objectives was to analyse the various activation kernels. Linear kernels (LIN) generally perform good for text data. But in our experiments, we subjected the code-switched text data to higher dimension word embedding, where linear kernels did not perform better. This was validated through our experiments where a non-linear kernel like RBF or Polynomial (POLY) of degree 2 was always performing better than linear across the languages. However, between the RBF and Polynomial Kernels, it was a close contest between them, where the values were very similar. For instance, we achieved an accuracy of 0.67 for Malayalam imbalanced data with all features considered, in both RBF and Polynomial Kernels.

5.3 Boxplot analysis

We evaluated the median through the boxplot as in Figure 2 of both accuracy and AUC across the language-feature-data combination. We notice that Polynomial kernel compares better than both, linear as well as RBF kernels in AUC as well as Accuracy evaluation. The median accuracy is 0.63 with a Polynomial kernel compared to 0.55 with Linear and 0.62 with RBF kernels. AUC is also better with Polynomial kernels where it yields 0.79 at the median compared to 0.77 of RBF and 0.74 of linear kernels. Polynomial kernels are known to favor discrete data that has no natural notion of

Code-mixed with English	Features	Data	Acc RBF	Acc LIN	Acc POLY	AUC RBF	AUC LIN	AUC POLY
Tamil	All	Imbalanced	**0.67**	**0.67**	**0.68**	0.72	0.72	0.75
Tamil	PCA	Imbalanced	0.67	0.67	0.67	0.70	0.68	0.71
Tamil	All	SMOTE	0.56	0.51	0.57	**0.79**	**0.76**	**0.80**
Tamil	PCA	SMOTE	0.49	0.46	0.49	0.74	0.72	0.74
Mal	All	Imbalanced	**0.67**	**0.64**	**0.67**	0.76	0.74	0.81
Mal	PCA	Imbalanced	0.61	0.61	0.61	0.70	0.66	0.70
Mal	All	SMOTE	0.63	0.54	0.64	**0.84**	**0.78**	**0.85**
Mal	PCA	SMOTE	0.48	0.43	0.48	0.75	0.70	0.74
Kannada	All	Imbalanced	0.71	**0.59**	**0.70**	0.84	0.77	0.86
Kannada	PCA	Imbalanced	0.60	0.55	0.59	0.78	0.73	0.78
Kannada	All	SMOTE	**0.74**	0.53	0.69	**0.89**	**0.78**	**0.89**
Kannada	PCA	SMOTE	0.57	0.51	0.56	0.81	0.75	0.80

Table 2: Accuracy and AUC values through various kernels and data selection techniques (Best values in bold).

smoothness as studied by Smola et al. (1998).

5.4 Dimensionality reduction analysis

We hypothesised that dimensionality reduction techniques like PCA will better the performance of the model relative to selecting all the features. But across the kernels as well as languages, PCA performed worse by dropping the accuracy margin, than when selecting all the features. Our analysis is that in text embeddings like FastText, the higher the dimensions it better captures the context generally for each word in a 300x1 column vector. The embedding size can be reduced by constraining with techniques like PCA while training in the word vectors but higher dimensions are preferred. Hence, vital spatial information which is important for classification is lost and hence the accuracy degrades.

5.5 Data balancing analysis

While we also hypothesised that data balancing techniques like SMOTE might improve the model's performance, during the experiments we found that the AUC is the best when SMOTE is used along with all the features. This is evident across all the three code-switched languages. For instance, for the Kannada code-switched dataset, selecting all the features yield better results as seen in Figure 3 relative to using SMOTE as shown in Figure 4. We believe that the sentiment classifier achieves good performance on the positive class (high AUC) at the cost of a high false negatives rate (or a low number of true negative).

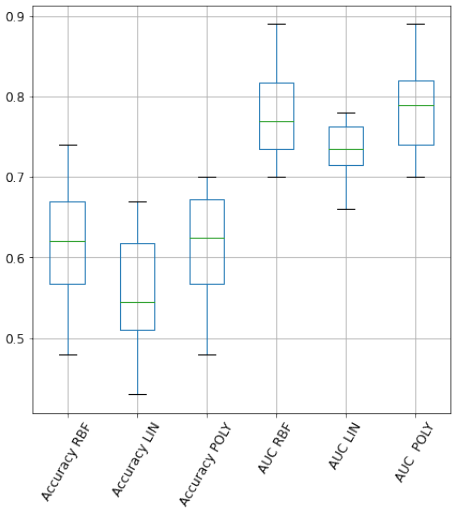

Figure 2: Boxplot of accuracy and AUC with various ELM Kernels

6 Conclusion

In this paper, various Kernel based ELMs like RBF, Linear and Polynomial have been experimented, along with combination of data constraining techniques like PCA and data balancing techniques like SMOTE for accuracy and AUC determination for code-switched languages. Our experimental results show that:

- ELM based techniques are faster to train relative to deep-learning models.
- Polynomial Kernels outperform Linear and RBF Kernels in ELMs across languages.
- SMOTE techniques with all the features favour better AUC in ELM models.

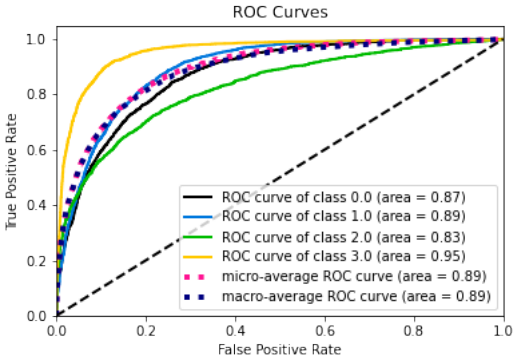

Figure 3: ROC curves of various classes for Kannada dataset with all the features in a Polynomial Kernel

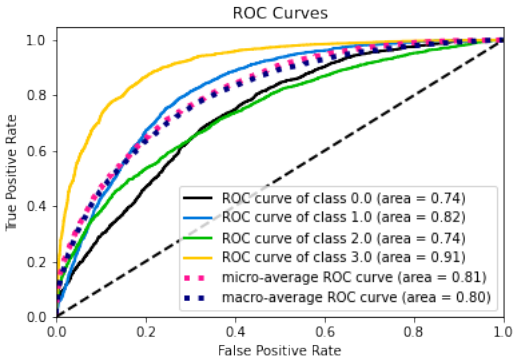

Figure 4: ROC curves of various classes for Kannada dataset constraining with PCA and SMOTE in a Polynomial Kernel

- ELM perform better in the chosen metrics relative to the traditional ensemble classifiers.

The next steps would be to improve on the word embedding and language identification on code-switched data for kernel based ELMs.

References

R Anita and CN Subalalitha. 2019a. An approach to cluster Tamil literatures using discourse connectives. In *2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP)*, pages 1–4. IEEE.

R Anita and CN Subalalitha. 2019b. Building discourse parser for Thirukkural. In *Proceedings of the 16th International Conference on Natural Language Processing*, pages 18–25.

B Bharathi, Bharathi Raja Chakravarthi, Subalalitha Chinnaudayar Navaneethakrishnan, N Sripriya, Arunaggiri Pandian, and Swetha Valli. 2022. Findings of the shared task on Speech Recognition for Vulnerable Individuals in Tamil. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi. 2020. HopeEDI: A multilingual hope speech detection dataset for equality, diversity, and inclusion. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*, pages 41–53, Barcelona, Spain (Online). Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John Philip McCrae. 2020a. 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)*, pages 177–184, Marseille, France. European Language Resources association.

Bharathi Raja Chakravarthi and Vigneshwaran Muralidaran. 2021. Findings of the shared task on hope speech detection for equality, diversity, and inclusion. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 61–72, Kyiv. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020b. 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)*, pages 202–210, Marseille, France. European Language Resources association.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Thenmozhi Durairaj, John Phillip McCrae, Paul Buitaleer, Prasanna Kumar Kumaresan, and Rahul Ponnusamy. 2022. Findings of the shared task on Homophobia Transphobia Detection in Social Media Comments. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Vigneshwaran Muralidaran, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John P McCrae. Dravidiancodemix: Sentiment analysis and offensive language identification dataset for dravidian languages in code-mixed text. *Language Resources and Evaluation*.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Rahul Ponnusamy, Prasanna Kumar Kumaresan, Kayalvizhi Sampath, Durairaj Thenmozhi, Sathiyaraj Thangasamy, Rajendran Nallathambi, and John Phillip McCrae. 2021. Dataset for identification of homophobia and transophobia in multilingual YouTube comments. *arXiv preprint arXiv:2109.00227*.

Adeep Hande, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2020. KanCMD: Kannada CodeMixed dataset for sentiment analysis and offensive language detection. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*, pages 54–63, Barcelona, Spain (Online). Association for Computational Linguistics.

Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. 2004. Extreme learning machine: a new learning scheme of feedforward neural networks. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, volume 2, pages 985–990 vol.2.

S R Mithun Kumar, Nihal Reddy, Aruna Malapati, and Lov Kumar. 2021. An ensemble model for sentiment classification on code-mixed data in dravidian languages. *Forum for Information Retrieval Evaluation, FIRE 2021*.

Prasanna Kumar Kumaresan, Ratnasingam Sakuntharaj, Sajeetha Thavareesan, Subalalitha Navaneethakrishnan, Anand Kumar Madasamy, Bharathi Raja Chakravarthi, and John P McCrae. 2021. Findings of shared task on offensive language identification in Tamil and Malayalam. In *Forum for Information Retrieval Evaluation*, pages 16–18.

Anitha Narasimhan, Aarthy Anandan, Madhan Karky, and CN Subalalitha. 2018. Porul: Option generation and selection and scoring algorithms for a tamil flash card game. *International Journal of Cognitive and Language Sciences*, 12(2):225–228.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Subalalitha Chinnaudayar Navaneethakrishnan, Thenmozhi Durairaj, Malliga Subramanian, Kogilavani Shanmugavadivel, Siddhanth U Hegde, and Prasanna Kumar Kumaresan. 2022. Findings of the shared task on Abusive Comment Detection in Tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Sajeetha Thavareesan, Dhivya Chinnappa, Durairaj Thenmozhi, and Rahul Ponnusamy. 2021. Overview of the DravidianCodeMix 2021 shared task on sentiment detection in Tamil, Malayalam, and Kannada. In *Forum for Information Retrieval Evaluation*, pages 4–6.

Manikandan Ravikiran, Bharathi Raja Chakravarthi, Anand Kumar Madasamy, Sangeetha Sivanesan, Ratnavel Rajalakshmi, Sajeetha Thavareesan, Rahul Ponnusamy, and Shankar Mahadevan. 2022. Findings of the shared task on Offensive Span Identification in code-mixed Tamil-English comments. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Ratnasingam Sakuntharaj and Sinnathamby Mahesan. 2016. A novel hybrid approach to detect and correct spelling in Tamil text. In *2016 IEEE International Conference on Information and Automation for Sustainability (ICIAfS)*, pages 1–6.

Ratnasingam Sakuntharaj and Sinnathamby Mahesan. 2017. Use of a novel hash-table for speeding-up suggestions for misspelt Tamil words. In *2017 IEEE International Conference on Industrial and Information Systems (ICIIS)*, pages 1–5.

Ratnasingam Sakuntharaj and Sinnathamby Mahesan. 2021. Missing word detection and correction based on context of Tamil sentences using n-grams. In *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, pages 42–47.

Anbukkarasi Sampath, Thenmozhi Durairaj, Bharathi Raja Chakravarthi, Ruba Priyadharshini, Subalalitha Chinnaudayar Navaneethakrishnan, Kogilavani Shanmugavadivel, Sajeetha Thavareesan, Sathiyaraj Thangasamy, Parameswari Krishnamurthy, Adeep Hande, Sean Benhur, Kishor Kumar Ponnusamy, and Santhiya Pandiyan. 2022. Findings of the shared task on Emotion Analysis in Tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Miriam Santos, Jastin Soares, Pedro Henriques Abreu, Helder Araujo, and Joao Santos. 2018. Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches. *IEEE Computational Intelligence Magazine*, 13:59–76.

Alex J. Smola, Bernhard Schölkopf, and Klaus-Robert Müller. 1998. The connection between regularization operators and support vector kernels. *Neural Netw.*, 11(4):637–649.

R Srinivasan and CN Subalalitha. 2019. Automated named entity recognition from tamil documents. In *2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP)*, pages 1–5. IEEE.

C. N. Subalalitha. 2019. Information extraction framework for Kurunthogai. *Sādhanā*, 44(7):156.

CN Subalalitha and E Poovammal. 2018. Automatic bilingual dictionary construction for Tirukural. *Applied Artificial Intelligence*, 32(6):558–567.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2019. Sentiment analysis in Tamil texts: A study on machine learning techniques and feature representation. In *2019 14th Conference on Industrial and Information Systems (ICIIS)*, pages 320–325.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2020a. Sentiment lexicon expansion using Word2vec and fastText for sentiment prediction in Tamil texts. In *2020 Moratuwa Engineering Research Conference (MERCon)*, pages 272–276.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2020b. Word embedding-based part of speech tagging in Tamil texts. In *2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS)*, pages 478–482.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2021. Sentiment analysis in Tamil texts using k-means and k-nearest neighbour. In *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, pages 48–53.

Siddharth Yadav and Tanmoy Chakraborty. 2020. Unsupervised sentiment analysis for code-mixed data.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2020. Large batch optimization for deep learning: Training bert in 76 minutes.