MUCS@DravidianLangTech2023: Sentiment Analysis in Code-mixed Tamil and Tulu Texts using fastText

Rachana K[a], Prajnashree M[b], Asha Hegde[c], H L Shashirekha[d]
Department of Computer Science, Mangalore University, Mangalore, Karnataka, India
{[a]rachanak749,[b]prajnapushparaj27,[c]hegdekasha}@gmail.com
[d]hlsrekha@mangaloreuniversity.ac.in

Abstract

Sentiment Analysis (SA) is a field of computational study that focus on analyzing and understanding people's opinions, attitudes, and emotions, towards any entity. An entity could be an individual, an event, a topic, a product etc., which is most likely to be covered by reviews and such reviews can be found in abundance on social media platforms. The increase in the number of social media users and the growing amount of user-generated code-mixed content such as reviews, comments, posts etc., on social media, have resulted in a rising demand for efficient tools capable of effectively analyzing such content to detect the sentiments. However, SA of social media text is challenging due to the complex nature of the code-mixed text. To tackle this issue, in this paper, we - team MUCS, describe the learning models submitted to the shared task "Sentiment Analysis in Tamil and Tulu" - DravidianLangTech@Recent Advances in Natural Language Processing (RANLP) 2023. Using fastText embeddings to train the Machine Learning (ML) models to perform SA in code-mixed Tamil and Tulu texts, the proposed methodology exhibited F1 scores of 0.14 and 0.204 respectively.

1 INTRODUCTION

In this digital era, social media platforms have become an integral part of the life of many people, especially the younger generation and have impacted people's perception of networking and socialising to a greater extent (Bharathi and Agnusimmaculate Silvia, 2021; Bharathi and Varsha, 2022; Swaminathan et al., 2022). This concept has influenced people to communicate efficiently and quickly using various social media platforms and has resulted in the increase in large amount of user-generated text data in the form of posts, comments, opinions, emotions, attitudes and reviews, making them a best source for user sentiments (Chakravarthi et al., 2022a,b; Chakravarthi, 2023). Identifying the sentiments of these text as positive, negative, neutral, etc., is the objective of SA as it is useful for various applications (Anita and Subalalitha, 2019; Thavareesan and Mahesan, 2019, 2020a,b). For example, SA can be used to determine which videos are liked by people on YouTube, based on the words/phrases in the comments for the video. SA can also help to determine whether a user is happy, sad, or angry, with the video.

As there is no barrier of language and content on social media, users feel convenient to post comments very informally by mixing words and sentences of more than one language (usually with one language being English) in more than one script, usually the native script and roman script. Further, due to the limitations of keyboard/keypad in computers/smart phones, users find it easy to key in the posts/comments in roman script (Chakravarthi, 2022b; Kumaresan et al., 2022; Chakravarthi, 2022a). This phenomena of mixing the linguistic units of more than language in one utterance or text is called as Code-mixing and it has almost become the official language of social media due to the increased number of users using this language (Chakravarthi et al., 2023a,b). Analyzing the user sentiments in code-mixed language is challenging due to inadequate resources and tools to address the text in code-mixed language. The complexity of the task increases if the code-mixed text is in low-resource languages such as Kannada, Tulu, Tamil, Malayalam, etc. As the code-mixed language is free from the grammar of any languages, users create words/sentences according to their whims and fancies which makes it interesting and challenging to analyze such texts.

To address the challenges of processing code-mixed Dravidian Languages for SA, in this pa-

Third Workshop on Speech and Language Technologies for Dravidian Languages,
pages 258–265, Varna, Bulgaria, Sep 7, 2023.
https://doi.org/10.26615/978-954-452-085-4_038

Language	Comments	English Translation
Tamil	Today trailer paaka yaarellam wait panreenga	who is waiting to watch today's trailer
	indha gilli padam yaara yematha pakre	This is gilli movie whom you are thinking to cheat
	Nalla concept ana nalla eruintha nalla erukkum	Good concept but if it was good it will be good
Tulu	Supar bro irna comedy masth ista apudu	Super bro I like your comedy so much
	Title d spoiler alert pad doli	Could have include a spoiler alert in the title
	Ithe Encha ullariye?!	How are you now?

Table 1: Examples of Tamil and Tulu comments in romanized script

per, we - team MUCS, describe the models submitted to "Sentiment Analysis in Tamil and Tulu" shared task at DravidianLangTech@RANLP 2023[1] (Hegde et al., 2023). The shared task consists of a message-level polarity classification task for SA in code-mixed Tamil (Chakravarthi et al., 2020) and Tulu (Hegde et al., 2022). Given a Youtube comment in Tamil/Tulu, the aim of the shared task is to develop models to classify the given comment into positive, negative, neutral, or mixed emotions. Few Tulu and Tamil comments in romanized script and their English translations are shown in a Table 1. This shared task is modeled as a multi-class text classification problem with two distinct models: i) Support Vector Machine (SVM) classifier and ii) ensemble of ML classifiers, both trained with fastText embeddings. As the given datasets are imbalanced, Text Augmentation approaches are explored to increase the size of the minority classes in the training set.

Tulu language, a member of the Dravidian language family, is spoken by a community of more than three million people known as Tuluvas. The Tulu-speaking region is primarily located in the coastal districts of Dakshina Kannada and Udupi in the state of Karnataka, India. Tuluvas can also be found in Mumbai, Maharashtra, and various Gulf countries. Tamil is a Dravidian language spoken in the Tamil Nadu and Puducherry states of India and also some parts of Sri Lanka. Tamil has a long literary history, and is spoken by almost 225 million people. Tamil is a Multilingualism language which means that there is a large variation between the written form of the language and the spoken form. Both Tulu and Tamil languages, belong to the category of low-resourced languages. While some Natural Language Processing (NLP) activity is being explored in Tamil language for various applications, NLP in Tulu is yet to takeoff as there is no availability of digital data in Tulu. The only resources available for Tulu are: a small Wikipedia[2], Byte Pair embeddings (BPEmb)[3] and fastText[4] embeddings.

The rest of the paper is organized as follows: Section 2 contains related work and Section 3 describes the methodology. Section 4 describes the experiments and results followed by conclusion and future work in Section 5.

2 RELATED WORK

Many of the techniques explored by researchers for SA focus on high-resource languages like English, Spanish etc. Off late, SA is also being explored in code-mixed low-resource languages. Description of some of the relevant SA works in code-mixed low-resource languages are given below:

CoSaD - a code-mixed SA model for Dravidian Languages proposed by Balouchzahi et al. (2021) makes use of char n-grams, char sequences, and syllables, to train an ensemble (Linear Support Vector Machine (LSVM), Logistic Regression (LR) and Multi-Layer Perceptron (MLP) classifers) model with majority voting to identify sentiments in code-mixed Kannada, Malayalam, and Tamil languages. Their models obtained average weighted F1-scores of 0.628, 0.726,and 0.619 for code-mixed Kannada, Malayalam, and Tamil languages respectively. Ensemble of Random Forest (RF), Multi-Layer Perceptron (MLP) and gradient boosting is proposed by Hegde et al. (2021) to identify hate speech and offensive content in monolingual English, Hindi, and Marathi languages and code-mixed English-Hindi language pairs. These ensemble models trained using a combination of the Term Frequency - Inverse Document Frequency (TF-IDF) of word uni-grams, character n-grams in the range (2, 3),

[1]https://codalab.lisn.upsaclay.fr/competitions/11095

[2]https://en.wikipedia.org/wiki/Tulu_language

[3]https://bpemb.h-its.org/tcy/

[4]https://fasttext.cc/docs/en/pretrained-vectors.html

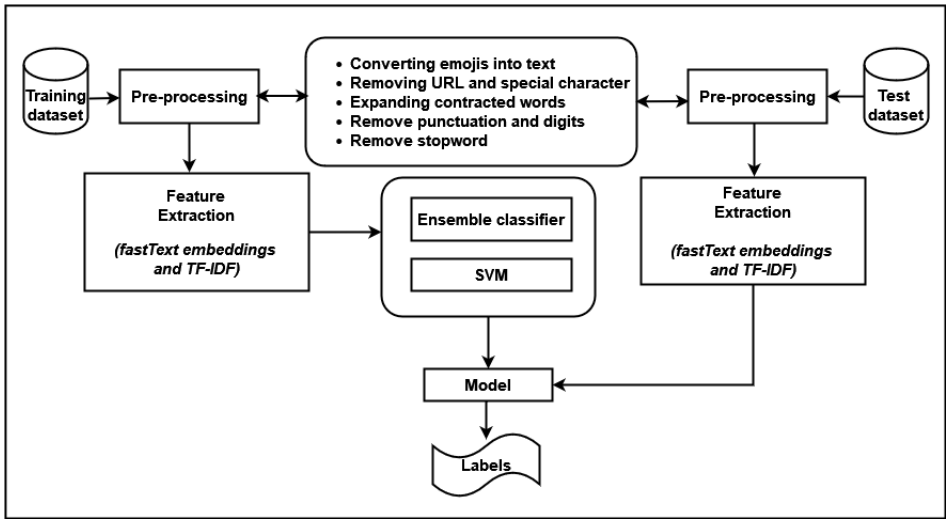

Figure 1: Framework of the proposed methodology

pre-trained word embeddings (Word2Vec), Hashtag embeddings (HastagVec), and Emo2Vec embeddings, obtained F1 scores 0.8251, 0.6323, 0.7830 and 0.6721 for English, Hindi, Marati and code-mixed English-Hindi language pair respectively.

SA of YouTube comments in code-mixed Tamil, Malayalam and Kannada language explored by Babu and Eswari (2021) using Paraphrase Cross-lingual Language Model-Robustly Optimized Bidirectional Encoder Representations from Transformers (XLM-RoBERTa) trained with hyperparameters (epochs = 12, learning rate = 3e-5, batchsize = 16, and dropout = 0.5.) obtained F1-scores of 0.71, 0.75 and 0.62 on Tamil, Malayalam and Kannada languages respectively. Chanda and Pal (2020) experimented feature extraction using Bidirectional Encoder Representations from Transformers (BERT), DistilBERT and fastText to train the LR classifiers to perform SA in code-mixed Tamil and Malayalam languages. Among these models, LR model trained with fastText embeddings outperformed other models with F1 scores of 0.58 and 0.63 for code-mixed Tamil and Malayalam languages respectively.

XLM-Roberta fine-tuned on code-mixed Malayalam and Tamil texts by Bai et al. (2021), to automatically detect sentiments, achieved F1 scores of 0.804 and 0.676 for Malayalam and Tamil languages respectively. Convolutional Neural Networks and Bi-directional Long Short-Term Memory (CNN+Bi-LSTM) model trained with fastText and GloVe pre-trained models by Mengistie and Kumar (2021) for SA of COVID-19 Public Reviews achieved 99.33 and 97.55 accuracy. Zhu and Dong (2020) proposed SA of Dravidian code-mixed text using multilingual Bidirectional Encoder Representations from Transformer (mBERT) model and used self-attention to assign a weight to the output of the BiLSTM. Their models achieved F1-scores of 0.73 and 0.64 for Malayalam and Tamil languages respectively.

From the literature review, it is evident that even though there are many models for SA of code-mixed low-resource languages, very few works have been reported for SA of code-mixed Tamil language and no work has been reported for the SA of code-mixed Tulu language. Hence, there is lot of scope to develop SA models for code-mixed low-resource Tulu and Tamil languages.

3 METHODOLOGY

The proposed methodology for SA in code-mixed Tulu and Tamil includes: Text Augmentation, Preprocessing, Feature extraction, and Model Construction. The framework of the proposed methodology is shown in Figure 1 and the steps are explained below:

3.1 Text Augmentation

Text augmentation is an important aspect of NLP to generate an artificial corpus. This helps in improving the NLP models to generalize better over a lot of different sub-tasks like intent classification, machine translation, chatbot training, image summarization, etc. The training sets for the task shared by the organizers are highly imbalanced and this may affect the performance of the learning models. Hence, several text augmentation methods are

Before Text Augmentation		
Labels	Training set	Development set
Positive	20,070	2,257
Unknown_state	5,628	611
Negative	4,271	480
Mixed_feelings	4,020	438
Total	33,989	3786
After Text Augmentation		
Labels	Training set	Development set
Positive	20,070	2,257
Unknown_state	6,239	611
Negative	4,751	480
Mixed_feelings	4,458	438
Total	35,518	3786

Table 2: Distribution of classes in Tamil dataset

Before Text Augmentation		
Labels	Training set	Development set
Positive	3,118	369
Neutral	1,719	202
Mixed feeling	974	120
Negative	646	90
Total	6,457	781
After Text Augmentation		
Labels	Training set	Development set
Positive	3,118	369
Neutral	1,719	202
Mixed feeling	1,297	120
Negative	1,016	90
Total	7,150	781

Table 3: Distribution of classes in Tulu dataset

explored to overcome the data imbalance to some extent in the training set.

In Tamil training set, the samples belonging to 'Unknown-state', 'Negative' and 'Mixed feelings' classes are less compared to that of 'Positive' class. Hence, the samples belonging to these classes from the Development set are added to the Training set, to balance the dataset to some extent.

In Tulu training set, 'Mixed feeling' and 'Negative' are the two minority classes which are highly imbalanced as compared to 'Positive' class. Hence, to balance the data to some extent, 'Mixed feeling' and 'Negative' classes are upsampled as follows: i) the samples belonging to the above two classes from the Development set are added to the Training set and ii) samples similar to those belonging to the above mentioned classes are collected from various sources (YouTube and Facebook post/comments and WhatsApp chat) and added to the corresponding classes in the training set. The distribution of classes in Tamil and Tulu datasets before and after augmentation are shown in Tables 2 and 3 respectively.

3.2 Pre-processing

Text pre-processing involves removing noise, normalizing and converting the normalized text to a format suitable for feature extraction.

- As emojis mainly depict user's intention, it would be imperative to replace them with their meanings to pick up their cues. Hence, emojis are converted to text using demoji[5] library.

- A contraction is a shortened form of a group of words. For example: hasn't, I'm, I'll etc. Contractions are often used in both written and oral communication. Expanding contractions into their natural form (hasn't – has not, I'm – I am, I'll – I will) will be more useful for processing particularly to extract embeddings from pre-trained models. The contracted words are expanded using the Contractions[6] library.

- URLs (Uniform Resource Locators) in a text are references to a location on the web. URLs, user mentions, hash tags, special characters, punctuation, and numeric information, present in the text data do not contribute to the classification task and hence are removed.

- Stop words are a set of commonly used words in any language. As they are not the distinguishing words, they do not contribute significantly to the classification task and hence are removed. English stopwords available at the Natural Language Tool Kit (NLTK)[7] and Tamil[8] stopwords available at GitHub repository are used as references to remove the English and Tamil stopwords respectively.

[5]https://pypi.org/project/demoji/
[6]https://pypi.org/project/pycontractions/
[7]https://pythonspot.com/nltk-stop-words/
[8]Tamil stopwords

The remaining words are the content bearing words which goes as input to feature extraction.

3.3 Feature Extraction

The process of extracting distinguishing features from the given data is called as Feature Extraction. fastText[9] is an open-source library of pre-trained models providing word embeddings for a total of 157 languages including Tamil and Tulu, developed by Facebook AI Research laboratory. These models trained on character n-grams represent word as the average of character embeddings of the characters a word is made up of. The advantage of using fastText is that it provides word representation even for Out of Vocabulary (OOV) words using their character n-grams. The feature extraction process using fastText pre-trained models for the datasets in both the languages are given below:

- **Tulu** - As the dataset is code-mixed, it consists of English words and Tulu words in native and romanized script. fastText Tulu pre-trained embeddings are used to represent Tulu words in native script and fastText English pre-trained embeddings are used to represent English words. However, Tulu words in romanized script cannot be represented by Tulu or English pre-trained models and hence they result in OOV words. These OOV words are represented as TF-IDF vectors. The vocabulary size of Tulu/English pre-trained models are 7,000 and 20,00,000 respectively and the vector dimension is 300. Concatenation of Tulu and English embeddings and TF-IDF of OOV words is used to train and evaluate the learning models.

- **Tamil** - As the datset is code-mixed, it consist of English words and Tamil words in native and romanized script. fastText Tamil pre-trained embeddings are used to represent Tamil words in native script and fastText English pre-trained embeddings are used to represent English words. The vocabulary size of both Tamil/English pre-trained models is 20,00,000 and the vector dimension is 300. Concatenation of Tamil and English embeddings are used to train and evaluate the learning model.

[9]https://fasttext.cc/docs/en/crawl-vectors.html

3.4 Model Construction

SVM and Ensemble Voting Classifier are used to detect the sentiments in the given unlabeled Tamil and Tulu comments.

- **Support Vector Machine** - maps data to a high-dimensional feature space so that data points can be categorised, even when the data are not otherwise linearly separable. The objective of the SVM Ahmad et al. (2017) algorithm is to find a hyperplane in an n-dimensional space that distinctly classifies the data points. The dimension of the hyperplane depends upon the number of features.

- **Ensemble model** - is a method of generating a new classifier from multiple base classifiers taking advantage of the strength of one classifier to overcome the weakness of another classifier with the intention of getting better performance for the classification task. This arrangement of more than one classifier will outperform when compared to the performance of any constituent classifier in the ensemble. It may be noted that any number of classifiers can be ensembled with compatible parameters Hegde and Shashirekha (2021). As more than one classifier is used in the ensemble model, majority voting of the classifiers is used to predict the class labels for the given unlabeled sample and hence, ensemble of classifiers is also called as Voting classifier.

  An ensemble of three ML classifiers, namely: Logistic Regression (LR), Bernoulli Naive Bayes (BNB) and Support Vector Classifer (SVC) classifiers with hard voting is used to identify the sentiment of the given unlabeled comment.

  LR is a ML classifier utilized for predicting categorical variables, employing dependent variables and regularization to mitigate overfitting. In LR, features from the input data are linearly combined and then transformed using the logistic function, allowing the algorithm to make predictions and classify instances into one of the two classes (Hassan et al., 2022).

  BNB classifier is a probabilistic ML algorithm based on the Naive Bayes principle, specifically designed for binary classification tasks. This algorithm computes the probability of a specific class label based on a set of binary

Model Name		Hyper-parameter and values
SVM		kernal='linear'
Ensemble	LR	random_state=1
	BNB	-
	SVC	penalty='l2', c=1.0

Table 4: Hyperparameters and their values used in ensemble model

Before Text Augmentation			
Language	Model	Development set	Test set
Tamil	SVM	0.20	0.05
	Ensemble	0.29	**0.13**
Tulu	SVM	0.33	0.35
	Ensemble	0.39	**0.35**
After Text Augmentation			
Language	Model	Development set	**Test set**
Tamil	SVM	-	0.05
	Ensemble	-	**0.14**
Tulu	SVM	0.16	**0.20**
	Ensemble	0.15	0.15

Table 5: Results of the proposed models

features using Bayes' theorem by incorporating the assumption of feature independence, making it efficient for text classification tasks (Singh et al., 2019).

SVC is an ML algorithm commonly used for text classification tasks. It aims to find the optimal hyperplane that best separates different classes of text data in a high-dimensional feature space. SVC seeks to find the most discriminative features that can separate different classes of documents/text effectively (Kalcheva et al., 2020). The hyperparameters and their values used in the classifiers of the ensemble model is shown in Table 4.

4 EXPERIMENTS AND RESULTS

Several experiments were conducted by combining various features and classifiers. The combination of features and classifiers which gave good performance on the Development sets are used to train the proposed models. The proposed models are evaluated on the Test set and the predictions are assessed by the organizers based on macro F1-score for the final evaluation and ranking. The performance of the proposed models for both Tamil and Tulu datasets are shown in Table 5.

The results illustrate that, ensemble model exhibited better performance over the other model with macro F1 score of 0.13 for Tamil text. Even though text augmentation is used to increase the samples of the minority classes (3 classes in Tamil and 2 classes in Tulu) to some extent, the datasets still remains imbalanced. Tamil dataset has a very large difference between the number of samples in 'Positive' class and other classes where as the difference between the number of samples in 'Positive' class and other classes in Tulu dataset is comparatively less. This clearly indicates the effect of data imbalance on the performance of the classifiers.

5 CONCLUSION

This paper describes the models submitted to "Sentiment Analysis in Tamil and Tulu" - DravidianLangTech@RANLP 2023 shared task. The proposed methodology consists of balancing the imbalance data using text augmentation, using fastText embeddings and TF-IDF as features to train SVM and ensemble model (LR, BNB and SVC models) with hard voting to perform SA. The proposed models exhibited F1 scores of 0.14 and 0.20 securing 13th and 15th rank for Tamil and Tulu datasets respectively.

References

Munir Ahmad, Shabib Aftab, and Iftikhar Ali. 2017. Sentiment Analysis of Tweets Using SVM. In *Int. J. Comput. Appl*, pages 25–29.

R Anita and CN Subalalitha. 2019. 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.

Yandrapati Prakash Babu and Rajagopal Eswari. 2021. Sentiment Analysis on Dravidian Code-Mixed YouTube Comments using Paraphrase XLM-RoBERTa Model. In *Working Notes of FIRE*.

Yang Bai, Bangyuan Zhang, Y Gu, T Guan, and Q Shi. 2021. Automatic Detecting the Sentiment of Code-Mixed Text by Pre-training Model. In *Working Notes of FIRE*.

Fazlourrahman Balouchzahi, Hosahalli Lakshmaiah Shashirekha, and Grigori Sidorov. 2021. CoSaD-Code-Mixed Sentiments Analysis for Dravidian Languages. In *CEUR Workshop Proceedings*, pages 887–898. CEUR-WS.

B Bharathi and A Agnusimmaculate Silvia. 2021. SSNCSE_NLP@DravidianLangTech-EACL2021: Offensive language identification on multilingual code

mixing text. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 313–318, Kyiv. Association for Computational Linguistics.

B Bharathi and Josephine Varsha. 2022. SSNCSE NLP@TamilNLP-ACL2022: Transformer based approach for detection of abusive comment for Tamil language. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 158–164, Dublin, Ireland. Association for Computational Linguistics.

Bharathi Raja Chakravarthi. 2022a. Hope speech detection in Youtube comments. *Social Network Analysis and Mining*, 12(1):75.

Bharathi Raja Chakravarthi. 2022b. Multilingual hope speech detection in English and Dravidian languages. *International Journal of Data Science and Analytics*, 14(4):389–406.

Bharathi Raja Chakravarthi. 2023. Detection of homophobia and transphobia in Youtube comments. *International Journal of Data Science and Analytics*, pages 1–20.

Bharathi Raja Chakravarthi, Adeep Hande, Rahul Ponnusamy, Prasanna Kumar Kumaresan, and Ruba Priyadharshini. 2022a. How can we detect homophobia and transphobia? experiments in a multilingual code-mixed setting for social media governance. *International Journal of Information Management Data Insights*, 2(2):100119.

Bharathi Raja Chakravarthi, Manoj Balaji Jagadeeshan, Vasanth Palanikumar, and Ruba Priyadharshini. 2023a. Offensive language identification in Dravidian languages using MPNet and CNN. *International Journal of Information Management Data Insights*, 3(1):100151.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020. 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, Shubanker Banerjee, Manoj Balaji Jagadeeshan, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Sean Benhur, and John Philip McCrae. 2023b. Detecting abusive comments at a fine-grained level in a low-resource language. *Natural Language Processing Journal*, 3:100006.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Thenmozhi Durairaj, John McCrae, Paul Buitelaar, Prasanna Kumaresan, and Rahul Ponnusamy. 2022b. Overview of the shared task on homophobia and transphobia detection in social media comments. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 369–377, Dublin, Ireland. Association for Computational Linguistics.

Supriya Chanda and Sukomal Pal. 2020. IRLab@ IITBHU@ Dravidian-CodeMix-FIRE2020: Sentiment Analysis for Dravidian Languages in Code-Mixed Text. In *FIRE (Working Notes)*, pages 535–540.

Sayar Ul Hassan, Jameel Ahamed, and Khaleel Ahmad. 2022. Analytics of Machine Learning-Based Algorithms for Text Classification. In *Sustainable Operations and Computers*, pages 238–248. Elsevier.

Asha Hegde, Mudoor Devadas Anusha, Sharal Coelho, Hosahalli Lakshmaiah Shashirekha, and Bharathi Raja Chakravarthi. 2022. Corpus Creation for Sentiment Analysis in Code-Mixed Tulu Text. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 33–40, Marseille, France. European Language Resources Association.

Asha Hegde, Mudoor Devadas Anusha, and Hosahalli Lakshmaiah Shashirekha. 2021. Ensemble Based Machine Learning Models for Hate Speech and Offensive Content Identification. In *Forum for Information Retrieval Evaluation (Working Notes)(FIRE), CEUR-WS. org*.

Asha Hegde and Hosahalli Lakshmaiah Shashirekha. 2021. Urdu Fake News Detection Using Ensemble of Machine Learning Models.

Chakravarthi Bharathi Raja Hegde, Asha, Hosahalli Lakshmaiah Shashirekha, Rahul Ponnusamy, Lavanya CN, SUBALALITHA andS K, Thenmozhi D, Martha Karunakar, Shreya Shreeram, and Sarah Aymen. 2023. "Findings of the Shared Task on Sentiment Analysis in Tamil and Tulu Code-Mixed Text". In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, Varna, Bulgaria. Recent Advances in Natural Language Processing.

Neli Kalcheva, Milena Karova, and Ivaylo Penev. 2020. Comparison of the Accuracy of SVM Kemel Functions in Text Classification. In *2020 International Conference on Biomedical Innovations and Applications (BIA)*, pages 141–145. IEEE.

Prasanna Kumar Kumaresan, Rahul Ponnusamy, Elizabeth Sherly, Sangeetha Sivanesan, and Bharathi Raja Chakravarthi. 2022. Transformer based hope speech comment classification in code-mixed text. In *International Conference on Speech and Language Technologies for Low-resource Languages*, pages 120–137. Springer.

Tajebe Tsega Mengistie and Deepak Kumar. 2021. Deep Learning Based Sentiment Analysis On COVID-19 Public Reviews. In *2021 International*

Conference on Artificial Intelligence in Information and Communication (ICAIIC), pages 444–449.

Gurinder Singh, Bhawna Kumar, Loveleen Gaur, and Akriti Tyagi. 2019. Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification. In *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*, pages 593–596. IEEE.

Krithika Swaminathan, Bharathi B, Gayathri G L, and Hrishik Sampath. 2022. SSNCSE_NLP@LT-EDI-ACL2022: Homophobia/transphobia detection in multiple languages using SVM classifiers and BERT-based transformers. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 239–244, Dublin, Ireland. Association for Computational Linguistics.

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. IEEE.

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. IEEE.

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. IEEE.

Yueying Zhu and Kunjie Dong. 2020. YUN111@ Dravidian-CodeMix-FIRE2020: Sentiment Analysis of Dravidian Code Mixed Text. In *FIRE (Working Notes)*, pages 628–634.