Enhancing Word-Level Language Identification in Code-Mixed Dravidian Languages

Sonith D[1,*], Kavya G[2], Asha Hegde[3] and H L Shashirekha[4]

Department of Computer Science, Mangalore University, Mangalore, Karnataka, India

Abstract

Code-mixing is the practice of combining two or more languages in a single utterance and users on social networking platforms often employ code-mixed text for the ease of use. This phenomena reflects the dynamic linguistic landscape of multilingual societies, where speakers fluidly switch between languages. Language Identification (LI) which aims to recognize the language of text automatically is a crucial and preliminary step for many Natural Language Processing (NLP) applications. Word-Level Language Identification (WLLI) is LI of each word in a given code-mixed text. The difficulties presented by informal and non-standard language, such as slang, abbreviations, and partial words, in user-generated code-mixed text prompt the need for WLLI. To explore the strategies for WLLI, in this paper, we - team MUCS describe the models submitted to "Word Level Language Identification in Code-Mixed Dravidian Languages" - a shared task organized at Forum for Information Retrieval Evaluation (FIRE) 2024. The shared task is offered in four code-mixed Dravidian languages - Malayalam, Kannada, Tamil, and Tulu. We have explored WLLI as: i) Sequence Labeling (CoLi_CNN - using Multilingual Representations for Indian Languages (MuRIL) and Convolutional Neural Network (CNN) and CoLi_TNN - customized Transformer Neural Network (TNN) model) problem and ii) Sequence-to-Sequence (Seq2Seq) learning approach (using Bidirectional Long Short Term Memory (BiLSTM)-to-Long Short Term Memory (LSTM) model), for WLLI in code-mixed Dravidian languages. Among the proposed models, CoLi_CNN model outperformed other models with macro F1 scores of 0.8028, 0.8400, 0.6994, and 0.7854 for Malayalam, Kannada, Tamil, and Tulu datasets respectively, securing 6th rank in all the languages.

Keywords

Word-Level Language Identification, Sequence Labeling, Sequence-to-Sequence Labeling Approach, Code-mixed Text

1. Introduction

LI refers to the process of determining the natural language in which a given piece of text is written. The increase in multilingual text comprising multiple languages or dialects on digital platforms, particularly in regions with diverse linguistic landscapes, makes LI as essential task. For tasks like sentiment analysis, information retrieval, content moderation, and machine translation, accurate LI is crucial as it enables systems to process and comprehend text correctly. Without effective LI, processing multilingual data can lead to errors, misinterpretations, and inefficiencies, making LI a crucial task in modern NLP applications.

India is a multilingual country with a rich heritage of languages and Indians who are often hooked to social media platforms can often read, write and speak two-three languages comfortably in addition to English. They usually use a combination of two or more languages in their informal communication on social media platforms such as Twitter, Instagram, and Facebook, to express themselves more comfortably [1, 2, 3]. This phenomenon of mixing languages at different linguistic units such as sentence, word, or sub-word, is known as code-mixing and it poses significant challenges for identifying the language of these linguistic units. LI involves analyzing various linguistic features and patterns within the text to accurately determine the language it belongs to [4, 5, 6]. To process the code-mixed content, it is necessary to go beyond traditional LI and focus on identifying the language of each word in a sentence. WLLI which addresses the challenge of automatically discerning the language of each word

Forum for Information Retrieval Evaluation (FIRE), December 12-15, 2024, India

*Corresponding author.
sonithksd@gmail.com (S. D); kavyamujk@gmail.com (K. G); hegdekasha@gmail.com (A. Hegde); hlsrekha@mangaloreuniversity.ac.in (H. L. Shashirekha)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR-WS.org/Vol-4054/T3-9.pdf
CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

within a sentence or phrase is crucial for effectively processing and understanding multilingual content on social media and other digital platforms. By accurately identifying languages at the word level, WLLI not only enhances the usability of digital tools and social media analytics but also contributes to preserve linguistic diversity enabling more inclusive communication platforms [7, 8]. As digital interactions continue to evolve in multilingual societies like India, the significance of WLLI in code-mixed text remains paramount for fostering effective communication across diverse linguistic contexts. Further, exploring the complexities of code-mixed text and developing innovative solutions for WLLI provides new opportunities for language technology and promote greater linguistic diversity and inclusion in the digital sphere. However, challenges in WLLI include the fluidity of language switching within sentences, variations in spelling and grammar across languages, and the scarcity of annotated data particularly for under-resourced languages.

Malayalam, Tamil, Kannada, and Tulu languages, primarily spoken in southern part of the country belong to Dravidian language family and are known for their unique linguistic features and scripts. In spite of their popularity, these languages are under-resourced. Further, code-mixing of these languages with English is quite common on social media platforms. To address the challenges of WLLI in code-mixed Dravidian languages - Malayalam, Kannada, Tamil, and Tulu, in this paper, we - Team MUCS describe the models submitted to "Word-Level Language Identification in Dravidian Languages" shared task[1] organized at FIRE 2024. With the aim of developing robust models for WLLI despite the challenges posed by code-mixing text, we propose: sequence labeling (CoLi_CNN: using MuRIL with CNN and CoLi_TNN: customized TNN model) and Seq2Seq learning approach with BiLSTM2LSTM model, to identify the language at word level in Malayalam, Kannada, Tamil, and Tulu code-mixed texts. The given dataset is in romanized form and the sample Malayalam, Kannada, Tamil, and Tulu words, from the given datasets are shown in Table 1.

The rest of paper is organized as follows: Section 2 describes the recent literature on WLLI and Section 3 focuses on the description of the proposed models followed by the experiments and results in Section 4. The paper concludes with future works in Section 5.

2. Related Work

WLLI in code-mixed language environments is crucial for accurately processing multilingual texts found on social media platforms. This not only improves user engagement, but also enhances content personalization, and fosters better communication across diverse linguistic communities online. In this direction, several studies have been conducted on WLLI in Dravidian languages and some of the notable studies are mentioned below:

Sushma et al. [9] proposed two distinct models: i) CoLIEnsemble - an ensemble of Machine Learning (ML) classifiers (Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR)) with hard voting trained with Term Frequency-Inverse Document Frequency (TF-IDF) of character n-grams in the range (1, 3) and fastText pre-trained word vectors individually, and ii) CoLI-CRF - a Conditional Random Field (CRF) algorithm trained with text-based features, for WLLI in Tulu. Their proposed CoLI-CRF model outperformed the other model with a macro F1 score of 0.77. Yigezu et al. [10] proposed LSTM, BiLSTM, and RF models, to identify the language of words in code-mixed Kannada texts in CoLI-Kanglish shared task at ICON2022. Their proposed BiLSTM model outperformed other models with a weighted F1-score of 0.82. Tash et al. [11] proposed ML models (k-Nearest Neighbors (k-NN), SVM) trained with TF-IDF of word n-grams in the range (1, 2) for WLLI in Kannada-English Texts and their proposed kNN and SVM models achieved macro F1 scores of 0.58 and 0.47 respectively. Thara and Poornachandran [12] employed a transformer model with various Bidirectional Encoder Representations from Transformers (BERT) variants (Cross-lingual Language Model - Robustly Optimized BERT approach (XLM-RoBERTa), CamemBERT, Distilled Version of BERT (DistilBERT), and Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA)) for the WLLI in Malayalam-English code-mixed dataset and their proposed ELECTRA model outperformed other models with a weighted

[1]https://codalab.lisn.upsaclay.fr/competitions/19357

Table 1
Description and samples of tokens in code-mixed Dravidian languages

Category	Tag	Description	Samples
Malayalam	MALAYALAM	Malayalam words	Nammude (ours), kanan (to watch), vere (other), abhinayikkan (acting)
Kannada	kn	Kannada words	Yavudu (which), mamuli (simple), channagide (good), tumba (more)
Tamil	tm	Tamil words	ivar (them), pesuvaru (speak up), pandre (do), solunga (tell me), enga (where)
Tulu	Tulu	Tulu words	Anda (is it), nerle (scold), bodu (want), edde (nice)
English	ENGLISH	Pure English words	like, Happy, enegy, feel, good, range, Never, phone, Nice, Super
Mixed language	MIXED	Combination of Malayalam and English words	seenillum, stonga, Kolamass, Trailerinu
	mixed	Combination of Kannada and English words	Camerada, sweetagiddale
	tmen	Combination of Tamil and English words	Doubleilla, gardenneye
	Mixed	Combination of Tulu and English words	Lastda, comedyla
Name	NAME	Words that indicates name of person (Including Indian names)	Mamookkha, Laletta, mohan, guru, Vasukiii, Nishanth, vaishnavi, jai, rai
Place	PLACE / Location	Words that indicates locations	Tamilnadu, KASARAGOD, Trivandrum, Singapore, Andhra, karnataka, Mangalore, Chennai, Kapikad, thulunadu
Other	OTHER	Words not belonging to any of the above categories and words of other languages	trlr, Mmk, sath, btata, aap, mast, ಏರ, unte, Badhoos, niranthar
Number	NUMBER	Words that indicates numerical values	12, 2O, 7k, 730k
Symbol	Sym / SYM	End of each sequence of words in-terms of sentences	. , *

F1 score of 0.99. Bansal et al. [13] proposed ML models (LR, Decision Tree (DT), and Gaussian Naive Bayes (GNB)) for LI in English-Punjabi code-mixed sentiment analysis social media dataset. Among the proposed models LR classifier outperformed other ML classifiers with an accuracy and F1 score of 86.63% and 0.88 respectively.

Shashirekha et al. [6] developed code-mixed Kannada-English dataset, code-mixed Kannada-English embeddings (for words, sub-words, and characters) and implemented four learning models: i) CoLI-ngrams : an ensemble of ML classifiers (Linear Support Vector Classifier (LSVC), Multi-Layer Perceptron (MLP) and LR) with 'soft' voting trained with Byte Pair Embeddings, ii) CoLI-vectors: an ensemble model trained with CountVectorizer of sub-words in the range (1, 5) and characters in the range (2, 5), iii) CoLI-BiLSTM: a sequence processing model based on BiLSTM architecture, and iv) CoLI-ULMFiT: a Universal Language Model Fine-Tuning (ULMFiT) utilizing Transfer Learning (TL) based approach, for Kannada-English code-mixed LI task at word level. Among the proposed models, CoLI-ngrams model outperformed all other models with an average macro F1 score of 0.64.

The related work emphasizes research on WLLI using various ML, DL, and transformer models. However, the performance of all models are not promising due to the challenges of processing variations

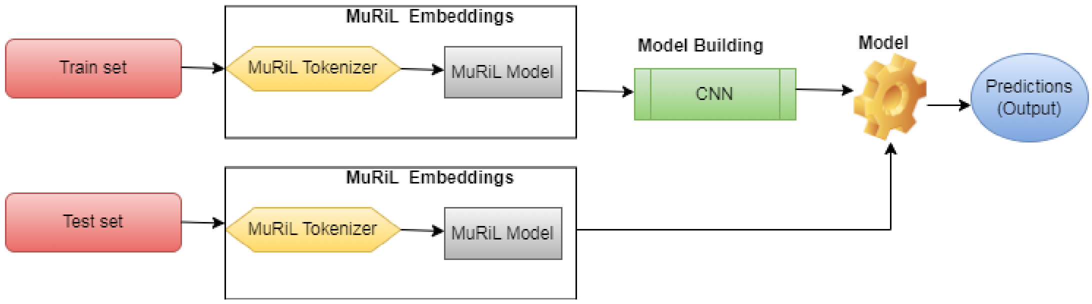

Figure 1: The framework of CoLi_CNN Model

in code-mixed text generated by creative users. Further, scarcity of annotated data for WLLI in low-resource Dravidian languages such as Malayalam, Kannada, Tamil and Tulu, adds its share of challenges to develop models for WLLI. This creates a significant opportunity for further research in this field.

3. Methodology

Pre-processing involves cleaning the data to remove noise in order to enhance the performance of the learning models. But as the given dataset is clean, to enhance the models ability to process and identify the language of the words accurately, all numerical values in the text are converted into their corresponding word forms. For instance, the number "100" is transformed into "one hundred". This conversion will help in avoiding potential confusion caused by numerical digits and ensures that all elements of the text are treated uniformly.

While sequence labeling problem assigns a label to each and every element in a sequence like tagging each word in a sentence with its part-of-speech tag, Seq2Seq learning on the other hand focuses on mapping the entire input sequence to an output sequence. The methodology for the proposed models are explained below:

3.1. Sequence Labeling

Two models: i) CoLi_CNN and ii) CoLi_TNN, are proposed using sequence labeling. CoLi_CNN model employs MuRIL embeddings to train a CNN, while CoLi_TNN model utilizes self-attention mechanisms to effectively capture contextual relationships in a sequence labeling approach. The description of the models is given below:

3.1.1. CoLi_CNN Model

In this approach, MuRIL[2] - a transformer model pre-trained on 17 Indian languages (including English, Malayalam, Tamil, Kannada, etc.) is used to represent the given text. MuRIL excels at capturing the semantic meaning of text through its deep layers and provides contextualized representations of text [14]. These embeddings are then passed to CNN, which applies convolutional filters to detect local patterns and features in the data. CNN architecture includes multiple convolutional layers followed by pooling layers to reduce dimensionality, and a dense layer with a softmax activation function to generate the final classification probabilities. The CNN classifier, a type of feed forward artificial neural network, effectively learns complex patterns and sequential dependencies within the data. Dropout component is also used to regularize the model to prevent overfitting. This approach combines the contextual understanding provided by MuRIL with CNNs capability to optimize the models performance for WLLI in code-mixed content. The framework and hyperparameters used in proposed CoLi_CNN model is shown in Figure 1 and Table 2 respectively.

[2]https://huggingface.co/google/muril-base-cased

Table 2
Hyperparameter and their values used in CoLi_CNN model

Hyperparameter	Values
Embedding Dimension	768
Number of Convolutional Layers	3 (implied by the number of filter sizes)
Batch Size	8
Learning Rate	2e-5
Optimizer	Adam
Max Sequence Length	128
Activation Function	ReLU
Number of convolution kernel	100
Dropout Rate	0.2

Table 3
Hyperparameter and their values used in CoLi_TNN model

Hyperparameter	Value
Vocabulary Size (Vx)	20,000
Number of Unique Labels (Vy)	Dynamic (based on training data)
Maximum Sequence Length	128
Embedding Dimension	100
Number of Attention Heads	4
Feed-Forward Dimension	64
Batch Size	32
Epochs	15
Dropout Rate	0.1
Loss Function	Sparse Categorical Cross Entropy

3.1.2. CoLi_TNN Model

CoLi_TNN is a customized TNN architecture proposed for WLLI. Unlike traditional sequence transduction models that rely on RNNs or CNNs [15], the transformer architecture in CoLi_TNN uses self-attention mechanisms to compute representations of input sequence. In this study, a standard transformer architecture is customized to suit token-level classification. This customized model has embedding layers that convert input tokens and positional information into dense vectors followed by a series of transformer blocks consisting of multi-head attention and feed-forward networks that allow the model to capture complex relationships between tokens in the sequence. Custom residual connections are used to retain the original token-level information in case if any token-specific information is missed, and layer normalization is applied to stabilize training. This ensures consistent activations within each layer, leading to smoother learning and improved training efficiency. The model outputs tag predictions for each token via a dense layer with sparse categorical cross entropy loss, dropout layers for regularization followed by a softmax layer. The hyperparameter and their values used in this model is shown in Table 3.

3.2. Sequence to Sequence Learning (Seq2Seq)

Deep Neural Networks (DNNs) are powerful models that have achieved excellent performance on difficult learning tasks. A Seq2Seq model is a type of DNN architecture designed to transform one sequence into another [16, 17] and framework of the proposed Seq2Seq model is shown in Figure 2. This model consists of an encoder-decoder architecture designed for Seq2Seq learning. While tokenization converts text sequences into numerical tokens, padding ensures uniform sequence lengths for batch processing. The encoder, implemented with a BiLSTM layer, processes the input sequence by capturing underlying patterns in both directions (forward and backward), creating a rich sequence representation. The decoder, utilizing a LSTM layer, generates the output sequence based on the context vector produced

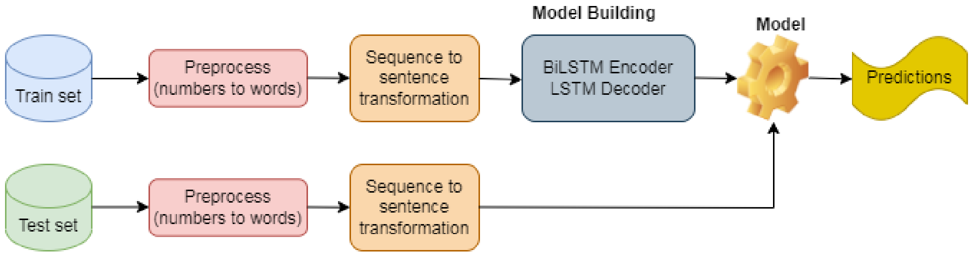

Figure 2: The framework of proposed Seq2Seq model

Table 4
Hyperparameter and their values used in Seq2Seq model

Hyperparameter	Value
Embedding Dimension	100
LSTM Units	128
Learning Rate	0.001
Batch Size	8
Epochs	13
activation	softmax
Optimizer	Adam
Loss Function	Sparse Categorical Crossentropy

Table 5
Label distribution of Train and Validation datasets

LANGUAGES							
Malayalam		Kannada		Tamil		Tulu	
Labels	Total samples	Labels	Total samples	Labels	Total samples	Labels	Total samples
MALAYALAM	12,408	kn	4,260	tm	8,064	Tulu	12,900
ENGLISH	6,030	en	18,777	en	3,259	English	8,222
MIXED	839	mixed	1,257	tmen	1,399	Mixed	600
OTHER	2,287	other	2,626	Other	77	Other	723
NAME	2,120	name	1,381	name	1,309	Name	1,636
PLACE	123	location	134	Location	11	Location	560
SYM	3,071	sym	4,064	sym	1,394	sym	4,665
NUMBER	645	-	-	-	-	Kannada	3,223
Total	27,523	Total	32,499	Total	15,513	Total	32,529

by the encoder. Both the input text and labels are embedded into high-dimensional vector spaces using embedding layers, while the final output is predicted using a fully connected softmax layer, providing a probability distribution over possible labels for each time step. The hyperparameter and their values used in Seq2Seq model is given in Table 4.

4. Experiments and Results

Various experiments were carried out using different learning models to identify the language of the words in the given code-mixed Kannada, Malayalam, Tamil and Tulu text. The label distribution of Malayalam, Kannada, Tamil, and Tulu datasets, is shown in Table 5. The performances of the models are evaluated based on macro F1 score and performances of the proposed models on the Validation and Test sets using sequence problem (CoLi_CNN and CoLi_TNN) and Seq2Seq approach are shown in tables 6 and 7 respectively.

Figure 3 gives a comparison of macro F1 scores of all the participating teams in the shared task for all

Table 6
Performance of the proposed models on Validation sets

Language	Model	Precision	Recall	Macro F1 score	Weighted F1 score	Accuracy
Malayalam	**CoLi_CNN**	0.72	0.72	0.71	0.89	0.89
	CoLi_TNN	0.86	0.61	0.67	0.82	0.84
	Seq2Seq	0.32	0.29	0.30	0.67	0.67
Kannada	**CoLi_CNN**	0.71	0.76	0.73	0.94	0.94
	CoLi_TNN	0.93	0.67	0.72	0.84	0.82
	Seq2Seq	0.65	0.49	0.52	0.83	0.82
Tamil	**CoLi_CNN**	0.64	0.64	0.64	0.92	0.92
	CoLi_TNN	0.72	0.48	0.49	0.73	0.78
	Seq2Seq	0.25	0.26	0.25	0.52	0.55
Tulu	**CoLi_CNN**	0.63	0.61	0.61	0.84	0.86
	CoLi_TNN	0.87	0.62	0.67	0.81	0.83
	Seq2Seq	0.55	0.51	0.53	0.78	0.77

Table 7
Performance of the proposed models on Test sets

Language	Model	Macro P	Macro R	Macro F1 score	W P	W R	WF1 score	Acc
Malayalam	**CoLi_CNN**	0.8861	0.7751	**0.8028**	0.9105	0.9132	0.9086	0.9132
	CoLi_TNN	0.8225	0.7295	0.7144	0.9281	0.7325	0.7969	0.7325
	Seq2Seq	0.1013	0.1177	0.1081	0.2555	0.2951	0.2730	0.2951
Kannada	**CoLi_CNN**	0.8304	0.8582	**0.8400**	0.9382	0.9333	0.9346	0.9333
	CoLi_TNN	0.9279	0.7601	0.8083	0.9127	0.8865	0.8838	0.8865
	Seq2Seq	0.1259	0.1450	0.1300	0.3302	0.4353	0.3676	0.4353
Tamil	**CoLi_CNN**	0.7343	0.6997	**0.6994**	0.9279	0.9279	0.9257	0.9279
	CoLi_TNN	0.5036	0.5057	0.4718	0.8128	0.6665	0.7089	0.6665
	Seq2Seq	0.1193	0.1466	0.1309	0.3114	0.4012	0.3496	0.4012
Tulu	**CoLi_CNN**	0.8224	0.7659	**0.7854**	0.8799	0.8779	0.8769	0.8779
	CoLi_TNN	0.8343	0.6494	0.6824	0.8625	0.7459	0.7737	0.7459
	Seq2Seq	0.1122	0.1303	0.1196	0.2463	0.2985	0.2695	0.2985
P: Precision; R: Recall; W: Weighted; WF1 score: Weighted F1 score; Acc: Accuracy								

the four languages. Among the submitted models, proposed CoLi_CNN model obtained better macro F1 scores securing 6th rank for all the four languages in the shared task. These macro F1 scores indicate that proposed CoLi_CNN model have performed competitively.

5. Conclusion and Future Work

In this paper, we - team MUCS, describe the models submitted to 'Word-Level Language Identification in Dravidian Languages' a shared task at 'FIRE 2024', to identify the languages in code-mixed Malayalam, Kannada, Tamil, and Tulu texts. Experiments are carried out with sequence labeling (CoLi_CNN and CoLi_TNN), and Seq2Seq approaches. CoLi_CNN model employs MuRIL word embeddings to train the CNN model, whereas CoLi_TNN and Seq2Seq models incorporate Keras embeddings for feature extraction. Among the proposed models, CoLi_CNN model outperformed other models with macro F1 scores of 0.8028, 0.8400, 0.6994, and 0.7854 for Malayalam, Kannada, Tamil, and Tulu languages respectively, securing 6th rank for all the languages in the shared task. Optimized feature combinations and diverse learning approaches will be explored, in addition to examining methods for addressing data imbalance.

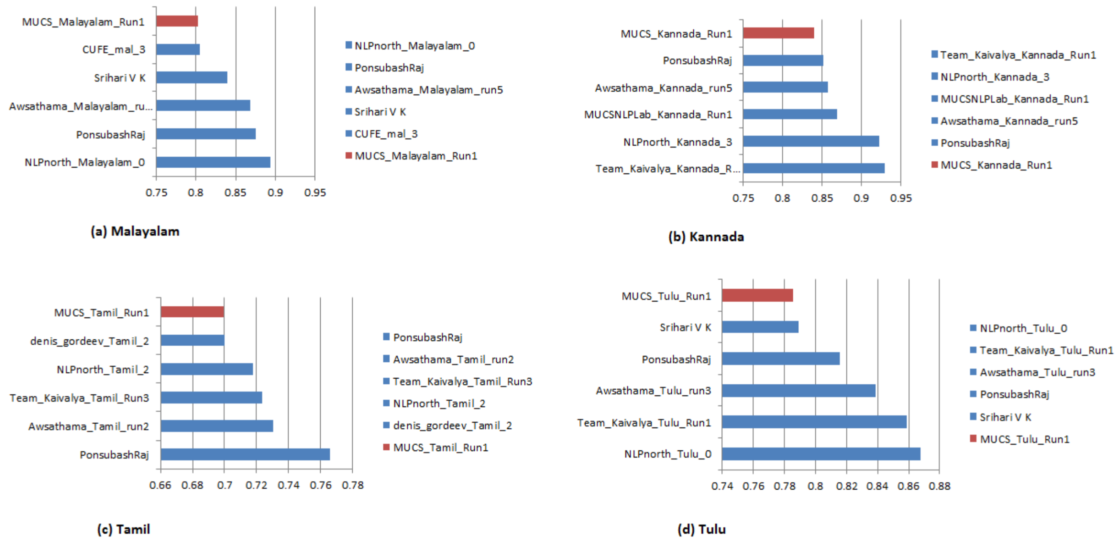

Figure 3: Comparison of macro F1 scores of the participating teams in the shared task

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT in order to: Grammar and spelling check. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] A. Hegde, F. Balouchzahi, S. Butt, S. Coelho, K. G, H. S Kumar, S. D, S. Hosahalli Lakshmaiah, A. Agrawal, Overview of CoLI-Dravidian: Word-level Code-mixed Language Identification in Dravidian Languages, in: Forum for Information Retrieval Evaluation FIRE - 2024, 2024.

[2] F. Balouchzahi, S. Butt, A. Hegde, N. Ashraf, H. Shashirekha, G. Sidorov, A. Gelbukh, Overview of coli-kanglish: Word Level Language Identification in Code-mixed Kannada-English Texts at Icon 2022, in: Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts, 2022, pp. 38–45.

[3] A. Hegde, F. Balouchzahi, S. Coelho, S. H L, H. A. Nayel, S. Butt, CoLI@FIRE2023: Findings of Word-level Language Identification in Code-mixed Tulu Text, FIRE '23, Association for Computing Machinery, New York, NY, USA, 2024, p. 25–26. URL: https://doi.org/10.1145/3632754.3633075. doi:10.1145/3632754.3633075.

[4] P. Shetty, Word-Level Language Identification of Code-Mixed Tulu-English Data., in: FIRE (Working Notes), 2023, pp. 198–204.

[5] F. Balouchzahi, S. Butt, A. Hegde, N. Ashraf, S. Hosahalli Lakshmaiah, G. Sidorov, A. Gelbukh, Overview of CoLI-Kanglish: Word Level Language Identification in Code-mixed Kannada-English Texts at ICON 2022, in: 19th International Conference on Natural Language Processing Proceedings, 2022.

[6] H. L. Shashirekha, F. Balouchzahi, M. D. Anusha, G. Sidorov, CoLI-Machine Learning Approaches for Code-mixed Language Identification at the Word Level in Kannada-English Texts, in: arXiv preprint arXiv:2211.09847, 2022.

[7] A. Hegde, M. D. Anusha, S. Coelho, H. L. Shashirekha, B. R. Chakravarthi, 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, 2022, pp. 33–40.

[8] S. H. Lakshmaiah, F. Balouchzahi, M. D. Anusha, G. Sidorov, CoLI-Machine Learning Approaches

for Code-mixed Language Identification at the Word Level in Kannada-English Texts, in: Acta Polytechnica Hungarica, volume 19, 2022.
[9] N. Sushma, A. Hegde, H. L. Shashirekha, Word-level Language Identification in Code-mixed Tulu Texts., in: FIRE (Working Notes), 2023, pp. 213–222.
[10] M. G. Yigezu, A. L. Tonja, O. Kolesnikova, M. S. Tash, G. Sidorov, A. Gelbukh, Word Level Language Identification in Code-mixed Kannada-English Texts using Deep Learning Approach, in: Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts, 2022, pp. 29–33.
[11] M. S. Tash, Z. Ahani, A. Tonja, M. Gemeda, N. Hussain, O. Kolesnikova, Word Level Language Identification in Code-mixed Kannada-English Texts using Traditional Machine Learning Algorithms, in: Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts, 2022, pp. 25–28.
[12] S. Thara, P. Poornachandran, Transformer based Language Identification for Malayalam-English Code-mixed Text, in: IEEE Access, volume 9, IEEE, 2021, pp. 118837–118850.
[13] N. Bansal, V. Goyal, S. Rani, Experimenting Language Identification for Sentiment Analysis of English Punjabi Code mixed Social Media Text, in: International Journal of E-Adoption (IJEA), volume 12, IGI Global, 2020, pp. 52–62.
[14] T. Bao, N. Ren, R. Luo, B. Wang, G. Shen, T. Guo, A Bert-based Hybrid Short Text Classification Model Incorporating CNN and Attention-based bigru, in: Journal of Organizational and End User Computing (JOEUC), volume 33, IGI Global, 2021, pp. 1–21.
[15] M. Bilkhu, S. Wang, T. Dobhal, Attention is All You Need for Videos: Self-attention Based Video Summarization using Universal Transformers, 2019.
[16] I. Sutskever, O. Vinyals, Q. V. Le, Sequence to Sequence Learning with Neural Networks, 2014. URL: https://arxiv.org/abs/1409.3215. arXiv:1409.3215.
[17] S. Palaskar, F. Metze, Acoustic-to-word Recognition with Sequence-to-Sequence Models, in: 2018 IEEE Spoken Language Technology Workshop (SLT), IEEE, 2018, pp. 397–404.