International Journal of Electrical and Computer Engineering (IJECE)
Vol. 13, No. 5, October 2023, pp. 5632~5640
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i5.pp5632-5640

Identification of monolingual and code-switch information from English-Kannada code-switch data

Ramesh Chundi[1], Vishwanath R. Hulipalled[2], Jay Bharthish Simha[3]
[1]School of Computer Science and Applications, REVA University, Bangalore, India
[2]School of Computing and Information Technology, REVA University, Bangalore, India
[3]CTO and RACE Labs, Abiba Systems, REVA University, Bangalore, India

Article Info

Article history:

Received Jul 16, 2022
Revised Oct 10, 2022
Accepted Mar 9, 2023

Keywords:

Character level n-gram
Code-switch text
English-Kannada
Machine learning techniques
Monolingual text

ABSTRACT

Code-switching is a very common occurrence in social media communication, predominantly found in multilingual countries like India. Using more than one language in communication is known as code-switching or code-mixing. Some of the important applications of code-switch are machine translation (MT), shallow parsing, dialog systems, and semantic parsing. Identifying code-switch and monolingual information is useful for better communication in online networking websites. In this paper, we performed a character level n-gram approach to identify monolingual and code-switch information from English-Kannada social media data. We paralleled various machine learning techniques such as naïve Bayes (NB), support vector classifier (SVC), logistic regression (LR) and neural network (NN) on English-Kannada code-switch (EKCS) data. From the proposed approach, it is observed that the character level n-gram approach provides 1.8% to 4.1% of improvement in terms of Accuracy and 1.6% to 3.8% of improvement in F1-score. Also observed that SVC and NN techniques are outperformed in terms of accuracy (97.9%) and F1-score (98%) with character level n-gram.

Corresponding Author:

Ramesh Chundi
School of Computer Science and Applications, Research Scholar, REVA University
Rukmini Knowledge Park, Kattigenahalli, Srinivasa Nagar, Yelahanka, Bangalore-560064, India
Email: chundiramesh@gmail.com

1. INTRODUCTION

Social networking websites like YouTube, Facebook and Twitter are providing online communication without boundaries for internet users. Communication in social media can be either in code-switch or monolingual. Referring two or more languages in communication is known as code-mixing or code-switching [1]. Code-switch text appears frequently in social media text since the users are not mandatory to follow any linguistic or grammatical rules [2]–[4]. Also code-switch can be used to decrease the social distance between people or to draw attention in social media [5].

There are three types of code-switch text in social media, namely word level code-switch, intra-sentential code-switch and inter-sentential code-switch: i) word level code-switch: user's switch from one language to another language within a word; ii) intra-sentential code-switch: user's switch from one language to another language within the sentence; and iii) inter-sentential code-switch: user's switch from one language to another language in between the sentences.

According to the leading newspaper the times of India report (Nov 7th, 2018), 52% of Indians are bilingual (at least they can read and write two languages) and 18% are multilingual (can read and write more than two languages) [6]. Kannada is one of the popular and the oldest Dravidian language in south India.

56.9 million speakers use Kannada and it is the 8th most spoken language in India. Due to the impact of English language, internet users are merging English and Kannada (sentence level) or Kannada writing in English (word level) in social media communication through Tweets, WhatsApp and Facebook.

Some of the examples to illustrate monolingual and code-switch text are discussed below. [E1] and [E2] are the examples for monolingual text, [E1] is a pure Kannada monolingual text and [E2] is a pure English monolingual text. In monolingual text, both script and source languages belong to the same language.

- [E1] ಗಿರೀಶ ಕಾರ್ನಾಡ್ ಅವರು ಆತ್ಮಕ್ಕೆ ಶಾಂತಿ ಸಿಗಲಿ ಎಂದು ಹಾರೈಸುತ್ತೇನೆ
  Translation: Let Girish Karnad soul rest in peace.
- [E2] I am proud of you I'm fan of you from this minute.
- [E3] Sir ಆದಷ್ಟು ಈ ಸಾಂಗ್ ನ್ ನಾನ್ share ಮಾಡ್ತೀನಿ. Nice song
  Translation: sir I will try my best to share the song. Nice song
- [E4] Dekshakkagi tyaga madiddare jeevana
  Translation: Sacrifice the life for country.

There are various ways to write the code-switch text in social networking websites. For example, [E3] refers to one way of writing the code-switch text (English and Kannada both the languages are used in the same sentence). [E4] is another way of writing the code-switch text (source and script both are different), source language is Kannada and scripting language is English. The relevant literature on this topic has been carried out by several researchers who experimented on language identification (identification of monolingual and code-switch information) in the recent past. Research on code-switch started from the year 1970, and few hypotheses are proposed. This leads to the motivations behind the study of code-switching. Some of the examples includes the markedness model [3], diglossia [7], communication accommodation theory [8] and conversational analysis model [9].

Ahmed *et al.* [10] proposed an efficient way for language identification using a cumulative frequency addition (CFA) of N-grams. However, in this approach more testing is required on large datasets to evaluate the performance of CFA. Rosner and Farrugia [11] discussed language identification in English-Maltese code-mixed data and achieved nearly 95% of the accuracy. However, numeric and punctuation entities were completely ignored in this approach. If it includs these, which will help to build more accurate model. Solorio and Liu [12] predicted the possible code-switch points in Spanish-English code-switch data. They trained various learning algorithms using transcription of code-switch speech. The average values for the code-switch sentences produced by machine learning approach were near to the values of those produced by the humans. Further, the accuracy can be improved by including a multi-word expression recognition system. Piergallini *et al.* [13] performed word-level language identification and prediction of code-switch point from Swahili-English code-switch data. The proposed approach achieved high accuracy in language identification and moderate improvement in code-switch point prediction. This approach needs to focus on social analysis of code-switch behaviour like the association between linguistic accommodation and power or code-switch and social solidarity.

Yirmibeşoğlu and Eryiğit [14] proposed a system to identify the code-switching between Turkish-English by using character level n-gram and conditional random fields (CRFs) have achieved 95.6% of micro-average F1-score. However, still there is a scope for improvement in F1-score by increasing the corpus size. Barman *et al.* [15] presented a preliminary study on instinctive language identification with Indian language code-mixing from social media messages. They performed word-level classification on Bengali-Hindi-English code-mixing data and concluded that character n-gram features are useful for language identification in code mixed data. Word-level code-mixing completely ignored in this work. Veena *et al.* [16] developed a system for word-level language identification from Tamil-English and Malayalam-English code-mixed data. This approach is executed based on character-based embedding with context information and achieved 93% of accuracy on Malayalam-English and 95% of accuracy on Tamil-English code-mixed data. Further, more features can be used to improve the system performance.

Mave *et al.* [17] examined different code-switching metrics and found that CRF model performed better with the boundary of 2-5 percentage for Spanish-English and 3-5 percentage for Hindi-English in comparison with deep learning model. This work can be extended to match the code-switching manners from various domains like chat conversations, song lyrics, and movie scripts across various language pairs. Gundapu and Mamidi [18] presented a study on different models for language identification in English-Telugu code-mixed data. It is found that CRF model outperformed for word-level language identification with 0.91 F1-score. Mandal and Singh [19] tested a multichannel neural networks on two different code-mixed languages such as Bengali-English and Hindi-English. They attained 93.28% of accuracy on Bengali-English code-mixed data and 93.32% of accuracy on Hindi-English code-mixed data.

Singh *et al.* [20] build an automatic named entity recognition (NER) system for Hindi-English code-mixed data and the proposed system outperformed with 33.18% of F1-score in comparison with existing baseline systems. However, this work can be extended to build natural language processing (NLP)

models like entity-specific sentiment analysers or semantic role labelling which use of NER for code-mixed data. Das *et al.* [21] presented a supervised learning model for word-level language identification in Bengali-English code-mixed data. Two types of word encoding methods such as character and phonetic are used along with stacking and threshold techniques. The stacking method achieved 91.78% of accuracy and Threshold method achieved 92.35%. Chaitanya *et al.* [22] tried to identify different languages from Facebook code-mixed comments and compared two different word embedding methods such as continuous bag of word (CBOW) and skip-gram. Shekhar *et al.* [23] applied two different assessment models such as statistical and neural-based learning models on Hindi-English code-mixed data. The outcome of proposed model illustrates that the word embedding is capable to spot the language parting by identifying source of the word and similarly mapping to its language label.

Very few researchers are focused for language identification in English-Kannada code-switch (EKCS) data. Lakshmi and Shambhavi [24] addressed the problem of word-level language identification for English-Kannada code-mixed data. Performed various supervised classifiers and found that the dictionary based model is better to handle word-level code-mixed in English-Kannada data. However, identifying monolingual and code-switch information from EKCS data problem needs to be addressed. James *et al.* [25] provided an investigational evidence to demonstrat that the accuracy of cloud-based multilingual systems (Google and Microsoft Azure) is low when identifying Maori language. The proposed study shows that, the BiLSTM with bilingual embeddings to identify Maori-English code-switching points with an accuracy of 87%. Hybrid models using hand-crafted rules based on the phonotactic variances between the deep learning techinques and languages can improve the performance of the proposed approach.

2. METHOD

2.1. Pre-processing and annotation

To implement monolingual and code-switch information identification task, 10,396 EKCS comments are collected from YouTube.com and these comments are written using English script. There are two types of comments in the collected dataset, comments that are having the combination of monolingual words (i.e., English words written in English script) and code-switch words (i.e., Kannada words written in English script) and the comments that are completely with code-switch words. Pre-processing is a fundamental and important task in NLP, since the performance of any model depends on quality data. To generate quality data, it is required to remove the noisy data from the dataset such as unwanted symbols, special characters, and digits since this noisy data will not play any significance role in some of the tasks like language identification and part of speech (POS) tagging. Hence, we performed pre-processing task on EKCS dataset to remove noisy data. Frist, removed digits, special characters, and emojiees from the dataset and then converted into lower case to bring the uniformity in the text. After the pre-processing, tokenization task is carried out to split each comment into tokens. Finally, 123,249 tokens are presented in the EKCS corpus out of 10,396 comments.

Once the tokens are generated, next step is annotation i.e., assign each token to its relevant class. Annotation is required to train and test the performance of supervised techniques in machine learning. The entire corpus is annotated with six classes such as monolingual (MN), code-switch (CS), names (NE), mix of monolingual and code-switch as MIX, location (LC), and remaining all other tokens as unknown (UN). Table 1 shows the sample data of EKCS corpus. Table 2 shows the statistics (number of tokens in each class) of EKCS corpus. Motive of this work is to identify monolingual and code-switch information, so the corpus has a smaller number of tokens in NE, LC and UN classes in comparison with code-switch and monolingual classes and particularly number of tokens in MIX class is the least (448) since the mix of monolingual and code-switch is a rare occurrence in social media text.

2.2. Proposed methodology

The proposed approach called character level n-gram is discussed in this section to identify monolingual and code-switch information from EKCS corpus. Four types of supervised learning approaches such as naïve Bayes (NB), support vector classifier (SVC), logistic regression (LR), and neural networks (NN) are implemented with two types of feature extraction such as word-level term frequency-inverse document frequency (TF-IDF) and character level n-gram. To implement proposed method, the EKCS corpus needs to be divided into two parts such as train dataset and test dataset. Performed 80:20 split by using sklearn.model_selection.tain_test_split on the EKCS corpus. Here, 80% of data is used to train the various models and 20% of the data is used to test the performance of the models. Figure 1 shows the proposed approach for language identification. Totally 24,650 tokens are used as input to test the performance of proposed model and the input dataset is pre-processed. Table 3 shows the statistics of input dataset.

Table 1. Sample of EKCS corpus

Token	Label
Thumballa	CS
Super	MN
Rama	NE
Correctagi	MIX
Karnataka	LC
Congi	UN

Table 2. Statistics of EKCS corpus

Class	Value
CS	85,178
MN	26,326
NE	7,048
MIX	448
LC	1,360
UN	2,889
Total No of Tokens	123, 249

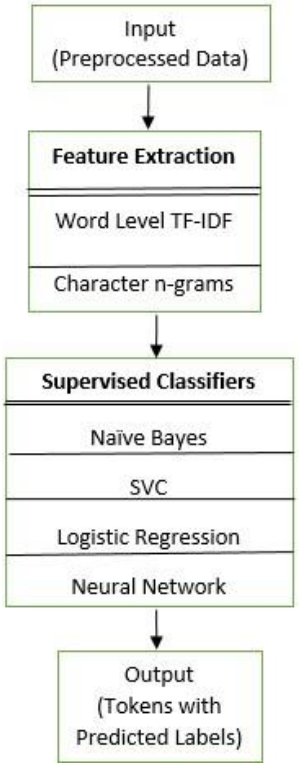

Figure 1. Language identification

Table 3. Statistics of test data

Class	Value
CS	16,983
MN	5,313
NE	1,350
MIX	96
LC	278
UN	630

In next step, feature extraction from the input dataset is carried out. The process of transforming the text data into features is called feature extraction. The TF-IDF is a technique which provides numerical values for text data and also gives the significance of a specific words in the corpus. Calculating term frequency is the first step in TF_IDF process and this can be done by using the (1).

$$TF = \frac{W}{N} \quad (1)$$

where, *W* is the frequency of a word in the document, and *N* is the total no of words in the document.

Once the term frequency is calculated, next step is to calculate the inverse document frequency and this can be done by using the (2).

$$IDF = \log(D/WD) \quad (2)$$

where, *D* is the total no of documents, and *WD* is the no of documents containing the word. Finally, we can calculate TF-IDF value by using the (3).

$$TF - IDF = TF * IDF \quad (3)$$

A character level n-gram is an order of *n* characters in a word and *n* is the number of characters in the sequence. Algorithm 1 shows the process of generating character level n-grams for each token. Algorithm 2 shows the process of features extraction by using *char_ngrams( )* and byte-pair encoding (*bpemb_en.encode()*) functions.

Algorithm 1. Character level n-gram generation

```
Input: word, n/*word–input text and n–Number of characters in the sequence */

Output: Character Level N-gram

/*Generating Character N-grams */
char_ngrams(word, n)
{
        cng=[]/* is empty list */

      /* Left and Right padding */
      cng.append(list(ngrams(word, n, pad_left=True, pad_right=True,
                        left_pad_symbol=' ',right_pad_symbol=' ')))

          /* Remove duplicate combinations */
          if(n>2)
          {
                    rc=n-2
                    cng=[ng_list [rc:-rc] for cng in cng]
       }

          ng_tupl=[ngram for cng in cng for ngram in cng]
          form_string=' '

          for i in range (0,n)
               form_string+=('%s')

      ng_tupl=[form_string % ngram_tuple for ngram_tuple in ng_tupl]

          return ng_tupl
}
```

Algorithm 2. Extraction of Word2features

```
Input: w/* input word */
Output: Features of the word
/* Features Extraction */
word2features(w)
{
          features={'word': word}
          tmp=list(features.values())+char_ngrams(w)
       for sw in bpemb_en.encode(w)
                    if sw.startswith('▁')
                              sw=sw[1:]
          tmp.extend(char_ngrams(sw))
          return tmp
}
```

3. RESULTS AND DISCUSSION

Various supervised classification methods such as NB, SVC, LR and NN are compared with two feature extractions such as word level TF-IDF and character level n-gram. The comparison is made based on two parameters such as accuracy and F1-score. It is observied that, the proposed method character level n-gram is more efficient interms of accuracy and F1-Score in comparison with word level TF-IDF.

3.1. Word level TF-IDF

Table 4 shows the comparison of word level TF-IDF F1-score for various classifiers. It is observed that, all four classifiers are producing almost similar F1-score value with variation of about 1% for code-switch class. LR produces slightly less F1-score i.e., 93% in comparison with other classifiers for monolingual class. Surprisingly NB classifier produces 0% as shown in Table 4 for mixed class, since the number of tokens is less about 96 as shown in Table 3 in comparison with other classes. Further, rest of the classes, NB produces less F1-score in comparison with other classifiers as per Table 4 since NB works well for large datasets.

Table 4. F1-score of word level TF-IDF

Class	F1-Score			
	NB	SVC	LR	NN
CS	96	97	96	97
MN	96	96	93	96
NE	76	87	79	87
MIX	00	36	08	36
LC	66	92	79	92
UN	62	89	77	89

3.2. Character level n-gram

Table 5 shows the comparison of character level n-gram F1-score for various classifiers. It is observed that, all four classifiers are producing almost similar F1-score for code-switch class. NB produces slightly less F1-score (96%) value in comparison with other classifiers for monolingual class as shown in Table 5. NB and LR performed better i.e., 47 and 42% respectively for mixed class in comparison with word level TF-IDF. If we considered overall, there is an improvement in F1-score with character level n-gram for all the classes in comparison with word level TF-IDF.

Table 5. F1-Score of character level n-gram

Class	F1-Score			
	NB	SVC	LR	NN
CS	98	99	99	99
MN	96	98	98	98
NE	84	92	91	92
MIX	47	62	50	61
LC	91	96	95	94
UN	81	89	89	90

Table 6 shows the F1-score comparison between word level TF-IDF and character level n-gram for each classifier. SVC and NN performed better with character level n-gram feature extraction (98%) in comparison with word level TF-IDF followed by LR classifier (97.9%) and NB (96.1%). Figure 2 shows the improvement of F1-score for various supervised classifiers. From the Figure 2, it is evident that a good amount of improvement in LR classifier with character level n-gram (an improvement of 3.8%) followed by SVC and NN with 1.7% and NB with 1.6%.

Table 6. F1-Score of word level TF-IDF and character level n-gram

Classifier	Word level TF-IDF	Character level N-Gram	Improvement
Naïve Bayes	94.5	96.1	1.6
SVC	96.3	98.0	1.7
Logistic regression	94.1	97.9	3.8
Neural networks	96.3	98.0	1.7

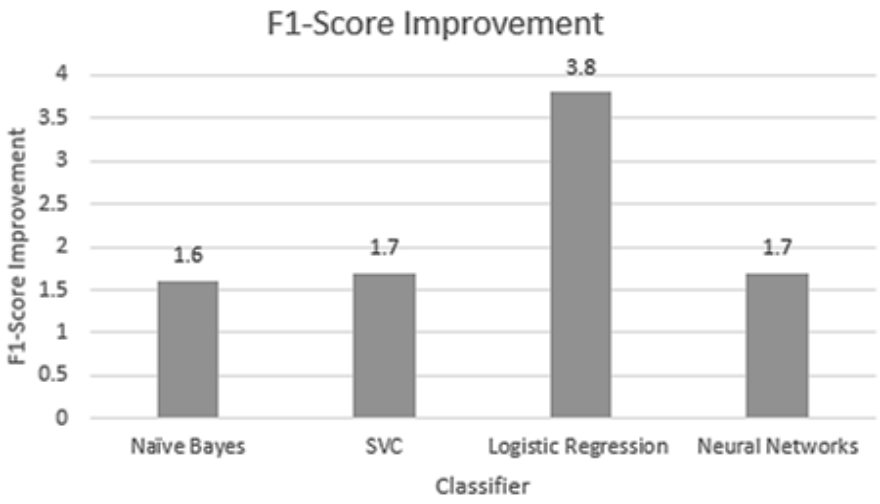

Figure 2. F1-Score improvement

Table 7 shows the accuracy comparison of word level TF-IDF and character level n-gram for each classifier. SVC and NN are performed better with character level n-gram feature extraction (97.9%) in comparison with word level TF-IDF followed by LR classifier (97.8%) and NB (96%). Figure 3 shows the improvement of accuracy for various supervised classifiers and it is evident that a good amount of improvement in LR classifier with character level n-gram (an improvement of 4.1%) followed by NB (2.2%), SVC and NN are with 1.8% respectively.

Table 7. Accuracy of word level TF-IDF and character level n-gram

Classifier	Word Level TF-IDF	Character Level N-Gram	Improvement
Naïve Bayes	93.8	96.0	2.2
SVC	96.1	97.9	1.8
Logistic Regression	93.7	97.8	4.1
Neural Networks	96.1	97.9	1.8

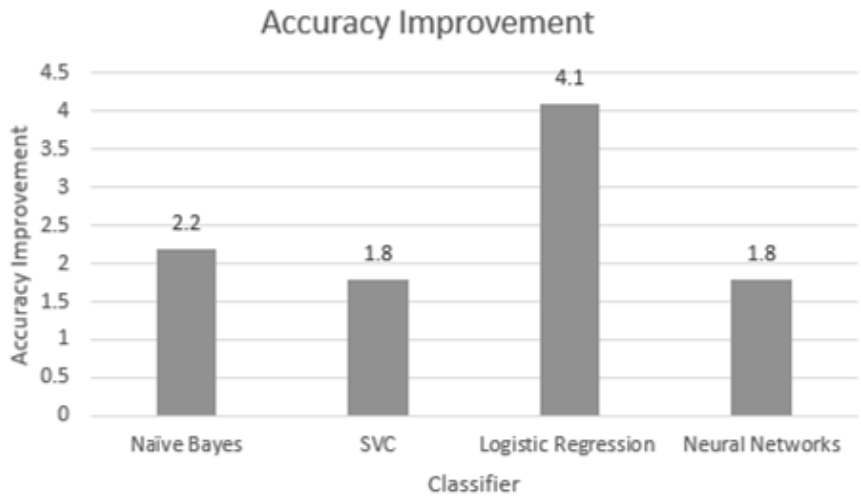

Figure 3. Accuracy improvement

4. CONCLUSION

In this work, we performed "Identification of monolingual and code-switch information from English-Kannada code-switch data" and conducted an experiment with various supervised classifiers by using two feature extraction techniques such as word level TF-IDF and character level n-gram. From the proposed

approach, it is observed that there is a considerable amount of improvement in F1-score and accuracy with character level n-gram for all classifiers in comparison with word level TF-IDF. In terms of F1-score, LR with 3.8% followed by SVC and NN are with 1.7% and NB with 1.6% improvement. In terms of accuracy, LR with 4.1% followed by NB with 2.2%, SVC and NN are with 1.8% improvement. Considering overall performance, SVC and NN are performed better (with 98% of F1-score and 97.9% of accuracy) in comparison with LR (with 97.9% of F1-score and 97.8% of accuracy) and NB (with 96.1% of F1-score and 96% of accuracy). In future, we are planning to develop an automatic language identification system for English-Kannada code-switch data.

ACKNOWLEDGEMENTS

The authors want to thank the REVA University management for their support of this research activity.

REFERENCES

[1] J. Lipski, "Code-switching and the problem of bilingual competence," *Aspects of bilingualism*, vol. 250, 1978.
[2] M. Gysels, "French in urban Lubumbashi Swahili: Codeswitching, borrowing, or both?," *Journal of Multilingual and Multicultural Development*, vol. 13, pp. 41–55, Jan. 1992, doi: 10.1080/01434632.1992.9994482.
[3] E. McClure, "Duelling languages: grammatical structure in codeswitching," *Studies in Second Language Acquisition*, vol. 17, no. 1, pp. 117–118, Mar. 1995, doi: 10.1017/S027226310001408X.
[4] M. G. Moyer, "Bilingual speech: A typology of code-mixing," *Language in Society*, vol. 31, no. 4, pp. 621–624, Oct. 2002, doi: 10.1017/S004740450224405X.
[5] C. Myers-Scotton, *Social motivations for codeswitching: Evidence from Africa*. Oxford University Press, 1995.
[6] R. Nagarajan, "52% of India's urban youth are now bilingual, 18% speak three languages," *The Times of India*, 2018.
[7] J. J. Gumperz and J.-P. Blom, "Social meaning in linguistic structures: Code-switching in Norway," *Directions in sociolinguistics*, pp. 407–434, 1971.
[8] H. Giles and R. N. St. Clair, *Language and social psychology*. Edition of Language and Social Psychology, 1979.
[9] P. Auer, Ed., *Code-switching in conversation*. Routledge, 2013.
[10] B. Ahmed, S.-H. Cha, and C. Tappert, "Language identification from text using n-gram based cumulative frequency addition," *Proceedings of Student/Faculty Research Day, CSIS, Pace University*, vol. 12, 2004.
[11] M. Rosner and P.-J. Farrugia, "A tagging algorithm for mixed language identification in a noisy domain." *Eighth Annual Conference of the International Speech Communication Association*, pp. 190–193, 2007.
[12] T. Solorio and Y. Liu, "Learning to predict code-switching points," *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008, doi: 10.3115/1613715.1613841.
[13] M. Piergallini, R. Shirvani, G. S. Gautam, and M. Chouikha, "Word-level language identification and predicting codeswitching points in Swahili-English language data," in *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, 2016, pp. 21–29, doi: 10.18653/v1/W16-5803.
[14] Z. Yirmibeşoğlu and G. Eryiğit, "Detecting code-switching between Turkish-English language pair," in *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, 2018, pp. 110–115, doi: 10.18653/v1/W18-6115.
[15] U. Barman, A. Das, J. Wagner, and J. Foster, "Code mixing: a challenge for language identification in the language of social media," in *Proceedings of the First Workshop on Computational Approaches to Code Switching*, 2014, pp. 13–23, doi: 10.3115/v1/W14-3902.
[16] P. V. Veena, M. A. Kumar, and K. P. Soman, "An effective way of word-level language identification for code-mixed facebook comments using word-embedding via character-embedding," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2017, pp. 1552–1556, doi: 10.1109/ICACCI.2017.8126062.
[17] D. Mave, S. Maharjan, and T. Solorio, "Language identification and analysis of code-switched social media text," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, 2018, pp. 51–61, doi: 10.18653/v1/W18-3206.
[18] S. Gundapu and R. Mamidi, "Word level language identification in English Telugu code mixed data," *arXiv preprint arXiv:2010.04482*, 2018.
[19] S. Mandal and A. K. Singh, "Language identification in code-mixed data using multichannel neural networks and context capture," in *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, 2018, pp. 116–120, doi: 10.18653/v1/W18-6116.
[20] K. Singh, I. Sen, and P. Kumaraguru, "Language identification and named entity recognition in hinglish code mixed tweets," in *Proceedings of ACL 2018, Student Research Workshop*, 2018, pp. 52–58, doi: 10.18653/v1/P18-3008.
[21] S. D. Das, S. Mandal, and D. Das, "Language identification of Bengali-English code-mixed data using character and phonetic based LSTM models," in *Proceedings of the 11th Forum for Information Retrieval Evaluation*, Dec. 2019, pp. 60–64, doi: 10.1145/3368567.3368578.
[22] I. Chaitanya, I. Madapakula, S. K. Gupta, and S. Thara, "Word level language identification in code-mixed data using word embedding methods for Indian languages," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2018, pp. 1137–1141, doi: 10.1109/ICACCI.2018.8554501.
[23] S. Shekhar, D. K. Sharma, and M. M. Sufyan Beg, "An effective cybernated word embedding system for analysis and language identification in code-mixed social media text," *International Journal of Knowledge-based and Intelligent Engineering Systems*, vol. 23, no. 3, pp. 167–179, Oct. 2019, doi: 10.3233/KES-190409.
[24] B. S. S. Lakshmi and B. R. Shambhavi, "An automatic language identification system for code-mixed English-Kannada social media text," in *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, Dec. 2017, pp. 1–5, doi: 10.1109/CSITSS.2017.8447784.
[25] J. James *et al.*, "Language models for code-switch detection of te reo Māori and English in a low-resource setting," in *Findings of the Association for Computational Linguistics: NAACL 2022*, 2022, pp. 650–660, doi: 10.18653/v1/2022.findings-naacl.49.

BIOGRAPHIES OF AUTHORS

Ramesh Chundi received the B.Sc. degree in computer science and MCA degree from Sri Venkateswara University, India, in 2004 and 2007, respectively. Currently pursuing Ph.D. degree in computer science and Applications from REVA University, India. His research interests include natural language processing (NLP), aritificial intelligence (AI), machine learning, deep learning, data analytics, and data mining. He can be contacted at email: chundiramesh@gmail.com.

Vishwanath R. Hulipalled is a Professor in the School of Computing and IT, REVA University, Bangalore, Karnataka, India. He completed BE, ME and Ph.D. in Computer Science and Engineering. His area of Interests includes machine learning, natural language processing, data analytics and time series mining. He has more than 24 years of academic experience and research. He authored more than 50 research articles in reputed journals and conference proceedings. He can be contacted at email: vishwanth.rh@reva.edu.in.

Jay Bharthish Simha is the CTO of ABIBA Systems and Chief Mentor at RACE Labs, REVA University. He completed his BE (Mech), M.Tech (Mech), M.Phil.(CS) and Ph.D.(AI). His area of interest includes fuzzy logic, soft computing, machine learning, deep learning, and applications. He has more than 20 years of industrial experience and 4 years of academic experience. He has authored/co-authored more than 50 journal/conference publications. He can be contacted at email: jay.b.simha@reva.edu.in and jay.b.simha@abibasystems.com.