IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 12, No. 3, September 2023, pp. 1500~1507
ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i3.pp1500-1507

Lexicon-based sentiment analysis for Kannada-English code-switch text

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]Abiba Systems, CTO and RACE Labs, REVA University, Bangalore, India

Article Info

Article history:

Received Oct 6, 2022
Revised Nov 30, 2022
Accepted Dec 21, 2022

Keywords:

Code-switch text
Lexicon based
Natural language processing
Sentiment analysis

ABSTRACT

Sentiment analysis is the process of computationally recognizing and classifying the attitudes conveyed in each text towards a particular topic and product. which is either positive or negative. Sentiment analysis is one of the interesting applications of natural language processing and which is used to analyze the social media. Text in social media is casual and it can be written either in code-switch or monolingual text. Several researchers have implemented sentiment analysis on monolingual text, though sentiments can be expressed in code-switch text. Sentiment analysis can be applied through deep learning, machine learning, or a Lexicon-based approach. Machine learning and deep learning methods are time-consuming, computationally expensive, and need training data for analysis. Lexicon-based method does not require training data and requires less time to find the sentiments in comparison with machine learning and deep learning. In this paper, we propose the Lexicon-based approach (NBLex) to analyze the sentiments expressed in Kannada-English code-switch text. This is the first effort that targets to perform sentiment analysis in Kannada-English code-switch text using the Lexicon-based approach. The proposed approach performed with better Accuracy of 83.2% and 83% of F1-score.

Corresponding Author:

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

1. INTRODUCTION

Social media websites such as Twitter, YouTube and Facebook. are open for online users to convey their sentiments about movies, products and services. online every day. Sentiment analysis is the process of detecting sentiments such as positive and negative from text content. Plutchik [1], proposed "A Psychoevolutionary Theory of Emotions" such as two sentiment classes *i.e.*, *Positive* and *Negative* with eight elementary emotions i.e., *Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise*, and *Trust*.

In India, most internet users are multilingual or bilingual. According to the Times of India article (Nov 7th, 2018), 52% of Indians are bilingual and 18% are multilingual. This multilingualism allows the users to use different vernaculars in social media communication. The ability to exchange language is termed code-switching or code-mixing [2]. Code-mixing is the common phenomena on social media [3]–[5].

Monolingual and code-switch text can be used in social media communication. To demonstrate, **Ex1** stands for monolingual text, where both scripting and source vernaculars are the same (English), and **Ex2** stands for code-switch text, where scripting and source vernaculars are different (scripting language is English and the source language is in Kannada). Sentiments can be conveyed through code-switch text or

monolingual text. In **Ex1,** the user conveys Positive sentiment and in **Ex2,** the user conveys Negative sentiment. Internet users are comfortable **to** communicate on social media with code-switch text since they have an exemption from grammatical and linguistic rules.
Ex1: "Abdul kalam is the always best & brilliant President of India"
Ex2: "Avrgi enu arta haguthe, bidi guru"
Translation: "What can they understand, leave it, Sir"

Code-switch text is out-of-vocabulary (OOV) and it is very common in social media text [6]. Some researchers are implementing sentiment analysis in the code-switch text by using machine translation (translating into the English language). This translation process requires greater effort to improve performance. As evident, machine learning and deep learning approaches are incapable of handling OOV words and these are computationally expensive and need more training data. Due to these limitations, the Lexicon-based approach is used for sentiment analysis instead of machine translation, machine learning, and deep learning techniques.

The Lexicon-based method uses a list of pre-labeled words which are categorized into *Positive* and *Negative* sentiments [7]. Lexicon-based methods are of two types, namely corpus-based and dictionary-based. In the corpus-based approach, Lexicons are constructed based upon the statistics (co-occurrence) of a word or semantic of a word. In the dictionary-based method, first limited words are collected (seed words) with sentiment labels. The next step is to use the Bootstrap technique to collect all synonyms of seed words from dictionary and add these synonyms to the seed list. Some of the well-known Lexicons are Affective Norms for English Words (ANEW) [8], WordNet-Affect Lexicon [9], and *NRC* Word-Emotion Association Lexicon [10].

The dictionary-based method is not a desired method for code-switch text since it has a restricted vocabulary and social media text is OOV. Hence, the corpus-based method can handle these drawbacks since there is no boundary for vocabulary. The study proposes a corpus-based Lexicon method to analyze sentiment in Kannada-English code-switch text.

Palanisamy *et al.* [11] build two types of Lexicons for sentiment analysis (Category specific Lexicon and common Lexicon) by using serendio-taxonomy. It has a collection of positive, negative, stop words, phrases, and negation. Adding word sense disambiguation improves the performance of the proposed approach. Guan *et al.* [12] proposed two steps for generating a Lexicon in the Chinese language. In the first step, they trained one-word vectors from distributed information, and in the second step, used renovated word vectors with a similarity-based approach. Desai *et al.* [13] proposed a technique by adding shallow parsing with a sentiment Lexicon. The advantage of this technique is, that it is very accurate and spontaneously creates a structure to exclude the cost of physically tagging data. However, it does not permit the inside arrangement of the basic word, or specifying a value in a sentence.

Jin *et al*. [14] proposed a technique by adding user-based features with global Lexicon features to perform sentiment analysis in a short social media text. Further study on optimal combination methods and feature fusing strategies will improve the accuracy of the model. Park *et al.* [15] used a dictionary-based approach to outline thesaurus for sentiment analysis by using three online dictionaries (Co-occurrence words, a list of synonyms-antonyms, and seed words). The proposed approach takes more time since it uses three dictionaries and also informal words, and occasionally informal words are not considered. Xiong *et al.* [16] developed a domain-Specific sentiment-lexicon by identifying the correlation among sentiment words (global information, local information, and constraint information) and it has given good adaptability with a semi-supervised approach.

Vu *et al.* [17] introduced a Lexicon method for sentiment analysis. It is a new and effective method since it combines the most popular Lexicon methods such as the SentiWN and LIU. Ashna *et al.* [18] developed a dictionary-based sentiment Lexicon for Malayalam movie reviews. This method reached 90% accuracy at the document level and 87.5% at the sentence level. One limitation of this approach is the different collection of phrases and idioms, which is since unigrams are used. This limitation can be overcome by using bigrams and trigrams. Awwad *et al.* [19] suggested a hybrid stemming method to enhance lexicon-based sentiment analysis (*MPQA II*, *MPQA*, *HRMA*, and HarvadA) at both the document level and sentence level. An increasing number of stemmers can help to improve the accuracy of the model.

Rezapour *et al.* [20] analyzed the effect of combining manually annotated hashtags with sentiment Lexicon and achieved a 7% improvement in accuracy. *POS* tagging can help improve the accuracy of this method. Yadav *et al.* [21] studied the importance of a domain-specific Lexicon for sentiment analysis. They built a domain-specific lexicon by introducing a bigram algorithm with a proposed strategy for developing some new corpora. Mowlaei *et al.* [22] developed a Lexicon using a genetic algorithm and this fits aspect-level problems. They have used two Lexicons, the first one is the intensity Lexicon (*NRC* Hashtag, *AFINN*, and Sentiment 140) and another one is the polarity Lexicon (Liu opinion Lexicon). However, the stemming and ordering of phrases are not considered while generating the Lexicon.

Agarwal *et al.* [23] used the *NRC* emotion Lexicon to predict how fans' sentiment changes over time. Sohangir *et al.* [24] compared Lexicon methods (*VADER*, SentiWordNet) with machine learning methods (Naïve Bayes, support vector machine (SVM), logistic regression) and found that the Lexicon method is faster than machine learning methods. Yuan *et al.* [25] suggested a method for the Chinese sentiment Lexicon using the Word2Vec tool and studied the sentiment words. Increasing the corpus size helps improve accuracy. Chathuranga *et al.* [26] generated a sentiment Lexicon for sentiment analysis in the Sinhala language using a semi-automated structure using a corpus-based approach.

Chang *et al.* [27] proposed a method using the skip-gram variant for mapping word spaces and generated language Lexicons with a smaller number of resources. Further improvement can be possible by increasing the vocabulary size and the accuracy of Lexicons. Taj *et al.* [28] suggested a dictionary-based lexicon approach (WordNet Lexicon dictionary) for sentiment analysis in BBC news articles. However, the proposed approach has limited word coverage. Yin *et al.* [29] built an automatic sentiment Lexicon (*FCP-Lex*) using CPchunks and reduced the ambiguity of words and obtained high-quality corpora. However, further study on Chinese natural language processing problems such as word embedding, semantic disambiguation, and word segmentation can be concentrated. Abd *et al.* [30] proposed a Lexicon-based sentiment analysis system on IMDb movie review dataset. The proposed system remains showed better accuracy unfluctuating, if the size of dictionary is altered. Sallam *et al.* [31] proposed a collaborative filtering system based on sentiment analysis on Arabic book dataset to provide recommendations. The proposed system reduced the average error values in terms of root mean squared error (RMSE) and mean absolute error (MAE).

Pamungkas *et al.* [32] performed sentiment analysis (SentiWordNet) by translating Bahasa Indonesia code-switch data into English and achieved 68% of accuracy. There are some limitations while translating from one language to another language like variation of slang, non-standard language, ambiguity, and thwarted expectation phenomenon. Karamollaoglu *et al.* [33] performed sentiment analysis on Turkish messages by extracting equivalent English words for Turkish words using an online dictionary. This process leads to misinterpretation and ambiguity of sentiment conveyed in sentences. Rodzman *et al.* [34] demonstrated that Lexicon-based methods resolve the limitations of machine learning techniques for sentiment analysis in code-switch data. Lexicon-based methods give more accurate results in comparison with Naïve Bayes for domain-specific Malay documents. However, adding more data in the dictionary and relating phrase levels for optimal results.

Pratama *et al.* [35] implemented various combinations of Lexicon resources with machine translations for sentiment analysis. The Google Translator with SentiWordNet combination is giving high accuracy (72%). The proposed method must be tested on large dataset whether the classifier contributes a superior performance or even contributes an inferior performance. Tsamis *et al.* [36] performed Lexicon-based sentiment analysis on bilingual (English and Greek) languages. They used *MPQA* Lexicon and Greek sentiment Lexicon to find the sentiment. Based on the above literature study no researchers have been done on Lexicon-based sentiment analysis in Kannada-English code-switch data till now. This is the first study to implement sentiment analysis in Kannada-English code-switch data by using the Lexicon approach. A Kannada-English code-switch sentiment Lexicon (*NBLex*) has been generated.

2. METHODOLOGY

2.1. Lexicon generation

In this section, the focus is on how the Kannada-English code-switch corpus can be created and annotated and on *NBLex* Lexicon process. Here, the study considerer the text as a code switch even if one word differs from the monolingual condition. The study ensures that all the comments in our corpus are written in English script, as shown in **Ex2**. Since the goal is to perform sentiment analysis in Kannada-English code-switch text, comments that do not follow the code-switch nature primarily are not considered.

Initially, the researchers gathered 7194 Kannada-English code-switch comments from YouTube.com based on different areas like social events, movie reviews, celebrities and politics. A pre-processing task is carried out to eliminate noisy data like special characters, symbols and digits as they do not have any significance in creating Lexicon since these words do not convey any sentiment polarity. Manual labeling of sentiment (*Positive* and *Negative*) is carried out as we do not have any programmed tagging techniques for Kannada-English code-switch text.

Algorithm 1: Building Frequency Dictionary

```
Freq_Dicti (D, T, L)
        Input:
        D - Dictionary with keys (word, label tuple) and frequency.
        T - list of comments.
        L - a list equivalent to sentiment (1 or 0).
        Output:
```

```
        D - Dictionary with keys and frequency.
        for t, i in (T, L)
                for w in t
                        p = (w, i)
                        if p in D
                                D[p] = D[p] + 1
                        else
                                D[p] = 1
 return D
```

Building the frequency dictionary is the first step in Lexicon creation, where keys are tuples (word, label) and values are frequency. The label value is either 1 or 0 and frequency is an integer value. Algorithm 1 depicts the complete steps for building a frequency dictionary. Further, we compute the positive and negative frequency of the word. Next step is, we need to calculate the total number of *Positive* words, *Negative* words, and total number of words from the corpus. Algorithm 2 depicts the complete steps for *NBLex* Lexicon creation. Further, we compute the Positive and Negative probability for each of the word using (1). Where, *pp* is the positive probability, *pf* is a positive frequency, *Np* is the total number of positive words, and *N* is the total number of words. We are using (2), to calculate negative probability of the word.

$$pp = \frac{pf + 1}{Np + N} \tag{1}$$

$$np = \frac{nf + 1}{Nn + N} \tag{2}$$

Where *np* is the negative probability, *nf* is the negative frequency, *Nn* is the total number of negative words. In the end, we calculate the score for every word using (3). Where *S* is the score of the word (any real number).

$$S = \log \frac{pp}{np} \tag{3}$$

Algorithm 2: *NBLex* Lexicon Construction

```
Nblex_score (D, T, L)
        Input:
        D - Dictionary with keys (word, label tuple) and frequency.
        T - List of comments.
        L - Sentiment label (1 or 0).
        Output:
        S - Real value for each word in the dictionary.
        S = {}
        /* Compute T (total number of words) */
        V = set p[0] for p in D.keys()
        T = len(V)
        /* Compute total number of positive and negative words */
        Np = Nn = 0
        for p in D.keys()
                if p[1] > 0 then
                        Np += D[p]
                else
                        Nn += D[p]
        for w in V
                /* Compute Positive and Negative frequencies */
                pf = lookup(D, w, 1)
                nf = lookup(D, w, 0)
                /* Compute Positive and Negative probability */
                pwp = (pf + 1) / (Np + T)
                pwn = (nf + 1) / (Nn + T)
                #Compute Score
                S[w] = log (pwp) - log (pwn)
 return S
```

2.2. Sentiment prediction

The proposed model is implemented by gathering a total of 1,799 comments from YouTube.com based on different domains like movie reviews, politics, celebrations and social events, not restricted to any one specific domain. During the pre-processing task, removed noisy data such as special characters, symbols, and digits, to increase the accuracy of sentiment analysis. Sentiment annotation (Positive and negative) has been done by linguistic experts as we conversed in the above section since we do not have any programmed labeling system for Kannada-English code-switch text. Figure 1 shows the steps for sentiment analysis process on Kannada-English code-switch text. Once the Lexicon is developed, the score for each comment in the test dataset is to be calculated. Scores for each word can be calculated using (4).

$$ST = \sum_{i=0}^{n} S(w) \quad (4)$$

Where, *ST* refers to sentence score and S(w) is the value/score of each word. Once scores are calculated, then the sentiment prediction task is carried out using (5). If the score of comment is greater than 0, then the sentiment predicted is positive, otherwise it is negative. Algorithm 3 shows the entire procedure for sentiment prediction task using *NBLex* Lexicon.

$$P = \begin{cases} 1\ (positive), if\ ST > 0 \\ 0\ (negative), otherwise \end{cases} \quad (5)$$

Algorithm 3: Sentiment Analysis

```
Input: text, t /* text - Input data
 t - Collection of tokens with scores */
Output: Sentiment /* Positive or Negative */
/*Calculating score of text (Text comment) */
Sent_Score (text, t)
{
 S = 0 /* S is the variable to record the score of text */
 for i in text /* compute score for each word in the text */
 if i in t then
 S += S.get(i)
 return S
}
/* Finding the Sentiment of test_data */
Sent_Analysis (test_data)
{
 P = [] /* empty list */
 for k in test_data /* compute the score for each comment in the test_data */
 if Sent_Score(k, S) > 0 then
        Ps = 1 (Positive)
        else
        Ps = 0 (Negative)
 P.append(Ps)
}
```

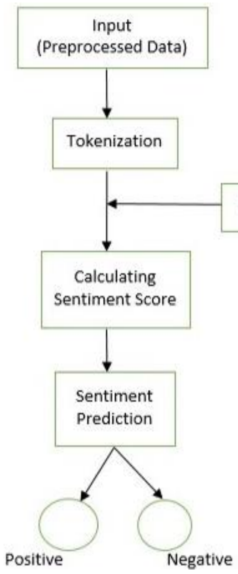

Figure 1. Sentiment analysis process

3. RESULTS AND DISCUSSION

In this section, the study compares the results of the Lexicon-based approach with other models such as Naïve Bayes (machine learning) and bidirectional long short-term memory neural network (BiLSTM) (deep learning). Initially, the total corpus (8993) is divided (80:20) into training (7194) and text (1799) datasets to train and test for both the Naïve Bayes and BiLSTM models. The performance parameters such as Accuracy, Precision, Recall, and F1-score are evaluated with Naïve Bayes, BiLSTM, and *NBLex* Lexicon models.

3.1. Naïve Bayes

In this experiment, we performed two kinds of vectorizations like bag-of-words (BOW) and term frequency and inverse document frequency (TF-IDF) for sentiment analysis on the test dataset. Both BOW and TF-IDF approaches produce the same outcomes. Table 1 shows the Accuracy, Precision, Recall, and F1 scores of both approaches. BOW and TF-IDF both approaches are providing the same values for all the parameters (Accuracy, precision, recall and F1-score).

3.2. BiLSTM

The deep learning approach (BiLSTM) is applied to test datasets to perform sentiment analysis. Table 2 shows the accuracy, precision, recall, and F1-score of BiLSTM. On comparing Naïve Bayes and BiLSTM it is observed that Naïve Bayes performs better in terms of all parameters such as accuracy, precision, recall, and F1-score.

3.3. NBLex

The study carried out the Lexicon-based analysis to predict the sentiment from the test dataset. Table 3 shows the accuracy, precision, recall, and F1-score of *NBLex* Lexicon approach. Table 4 shows the precision, recall, and F1-score for all the three approaches *i.e.*, *NBLex*, Naïve Bayes, and BiLSTM. The Lexicon-based approach produces better results in terms of precision, Recall and F1-score in comparison with Naïve Bayes and BiLSTM.

In Table 5, the accuracy of three approaches is compared (*NBLex*, Naïve Bayes, and BiLSTM). From Figure 2 it is observed that the Lexicon-based sentiment prediction approach produces better results in terms of Accuracy with 83.2% in comparison with Naïve Bayes and BiLSTM. The main reason for more accuracy in NBLex is that the corpus-based Lexicon approach is better in dealing with OOV in code-switch text, also there is no limit for vocabulary.

Table 1. Accuracy, precision, recall and F1-score of BOW and TF-IDF

Machine learning vectorization	Accuracy	Precision	Recall	F1-score
BOW	80.7	0.80	0.83	0.81
TF-IDF	80.7	0.80	0.83	0.81

Table 2. Accuracy, precision, recall and F1-score of BiLSTM

Deep learning approach	Accuracy	Precision	Recall	F1-score
BiLSTM	75.7	0.75	0.77	0.75

Table 3. Accuracy, precision, recall and F1-score of NBLex

Lexicon	Accuracy	Precision	Recall	F1-score
NBLex	83.2	0.82	0.85	0.83

Table 4. Comparative analysis of precision, recall and F1-score

Approach	Precision	Recall	F1-score
NBLex	**0.82**	**0.85**	**0.83**
Naïve Bayes	0.80	0.83	0.81
BiLSTM	0.75	0.77	0.75

Table 5. Comparison of accuracy

Approach	Accuracy
NBLex	**83.2**
Naïve Bayes	80.7
BiLSTM	75.7

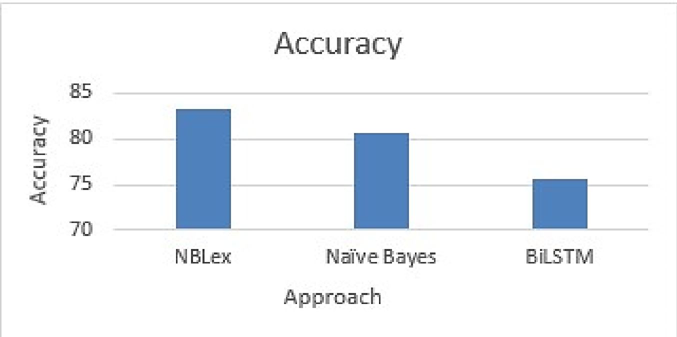

Figure 2. Accuracy comparison of *NBLex*, Naïve Bayes and BiLSTM

4. CONCLUSION

This work proposes a Lexicon-based approach for sentiment analysis in Kannada-English code-switch text. This approach performs better in comparison with other existing models like Naïve Bayes (machine learning) and BiLSTM (deep learning). The proposed approach (*NBLex*) has achieved 83.2% of Accuracy and 83% of F1-score for Kannada-English code-switch text. We strongly believe that, the Lexicon method is an alternative to perform sentiment analysis in code-switch data. Best of our knowledge, this is the first work that aimed on sentiment analysis in Kannada-English code-switch text. In the future, we are planning to handle sarcastic text for predicting sentiment and emotion, since most of the users are expressing positive and negative sentiments or emotions in a single comment.

ACKNOWLEDGEMENTS

We thank REVA University for supporting to carry out this research.

REFERENCES

[1] R. Plutchik, "A psychoevolutionary theory of emotions," *Social Science Information*, vol. 21, no. 4–5, pp. 529–553, 1982, doi: 10.1177/053901882021004003.
[2] J. Lipski, "Code-switching and the problem of bilingual competence," *Aspects of bilingualism*, pp. 250–264, 1978, doi: 10.1017/S1366728917000335.
[3] M. Gysels, "French in urban Lubumbashi Swahili: Codeswitching, borrowing, or both?," *Journal of Multilingual and Multicultural Development*, vol. 13, no. 1–2, pp. 41–55, 1992, doi: 10.1080/01434632.1992.9994482.
[4] N. Poulisse, "Duelling languages: Grammatical structure in codeswitching," *International Journal of Bilingualism*, vol. 2, no. 3, pp. 377–380, 1998, doi: 10.1177/136700699800200308.
[5] P. Muysken, "Bilingual speech: A typology of code-mixing," *Cambridge University Press*, vol. 11, 2000.
[6] T. Baldwin, P. Cook, M. Lui, A. MacKinlay, and L. Wang, "How noisy social media text, how diffrnt social media sources?," *6th International Joint Conference on Natural Language Processing, IJCNLP 2013-Proceedings of the Main Conference*, pp. 356–364, 2013.
[7] S. M. Mohammad, "From once upon a time to happily ever after: Tracking emotions in mail and books," *Decision Support Systems*, vol. 53, no. 4, pp. 730–741, 2012, doi: 10.1016/j.dss.2012.05.030.
[8] M. M. Bradley and P. J. Lang, "Affective Norms for English Words (ANEW): Instruction manual and affective ratings," *IEEE Internet Computing*, vol. 12, no. 5, pp. 44–52, 2008, [Online]. Available: http://dionysus.psych.wisc.edu/methods/Stim/ANEW/ANEW.pdf%5Cnhttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Affective+Norms+for+English+Words+(+ANEW+):+Instruction+Manual+and+Affective+Ratings#0%5Cnhttp://scholar.google.com/scholar?hl=en&bt.
[9] C. Strapparava and A. Valitutti, "WordNet-Affect: An affective extension of WordNet," *Proceedings of the 4th International Conference on Language Resources and Evaluation, LREC 2004*, pp. 1083–1086, 2004.
[10] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word-emotion association lexicon," *Computational Intelligence*, vol. 29, no. 3, pp. 436–465, 2013, doi: 10.1111/j.1467-8640.2012.00460.x.
[11] P. Palanisamy, V. Yadav, and H. Elchuri, "Serendio: Simple and practical lexicon based approach to sentiment analysis," **SEM 2013- 2nd Joint Conference on Lexical and Computational Semantics*, vol. 2, pp. 543–548, 2013.
[12] X. Guan, Q. Peng, J. Zhang, and X. Zhang, "Renovating word vectors to build Chinese sentiment lexicon," *2015 IEEE International Conference on Information and Automation, ICIA 2015 - In conjunction with 2015 IEEE International Conference on Automation and Logistics*, pp. 2977–2982, 2015, doi: 10.1109/ICInfA.2015.7279798.
[13] J. M. Desai and S. R. Andhariya, "Sentiment analysis approach to adapt a shallow parsing based sentiment lexicon," *ICIIECS 2015 - 2015 IEEE International Conference on Innovations in Information, Embedded and Communication Systems*, 2015, doi: 10.1109/ICIIECS.2015.7193160.
[14] Z. Jin, Y. Yang, X. Bao, and B. Huang, "Combining user-based and global lexicon features for sentiment analysis in twitter," *Proceedings of the International Joint Conference on Neural Networks*, vol. 2016-Octob, pp. 4525–4532, 2016, doi: 10.1109/IJCNN.2016.7727792.
[15] S. Park and Y. Kim, "Building thesaurus lexicon using dictionary-based approach for sentiment classification," *2016 IEEE/ACIS 14th International Conference on Software Engineering Research, Management and Applications, SERA 2016*, pp. 39–44, 2016, doi: 10.1109/SERA.2016.7516126.
[16] G. Xiong, Y. Fang, and Q. Liu, "Automatic construction of domain-specific sentiment lexicon based on the semantics graph," *2017 IEEE International Conference on Signal Processing, Communications and Computing, ICSPCC 2017*, vol. 2017-Janua, pp. 1–6, 2017, doi: 10.1109/ICSPCC.2017.8242562.
[17] L. Vu and T. Le, "A lexicon-based method for sentiment analysis using social network data," *Int'l Conf. Information and Knowledge Engineering (IKE'17)*, p. 146, 2017.
[18] M. P. Ashna and A. K. Sunny, "Lexicon based sentiment analysis system for Malayalam language," *Proceedings of the International Conference on Computing Methodologies and Communication, ICCMC 2017*, vol. 2018-Janua, pp. 777–783, 2018, doi: 10.1109/ICCMC.2017.8282571.
[19] H. Awwad and A. Alpkocak, "Using hybrid-stemming approach to enhance lexicon-based sentiment analysis in Arabic," *Proceedings - 2017 International Conference on New Trends in Computing Sciences, ICTCS 2017*, vol. 2018-Janua, pp. 229–235, 2017, doi: 10.1109/ICTCS.2017.26.
[20] R. Rezapour, L. Wang, O. Abdar, and J. Diesner, "Identifying the overlap between election result and candidates' ranking based on hashtag-enhanced, lexicon-based sentiment analysis," *Proceedings - IEEE 11th International Conference on Semantic Computing, ICSC 2017*, pp. 93–96, 2017, doi: 10.1109/ICSC.2017.92.
[21] S. Yadav and M. Sarkar, "Enhancing sentiment analysis using domain-specific lexicon: A case study on GST," *2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2018*, pp. 1109–1114, 2018, doi: 10.1109/ICACCI.2018.8554600.
[22] M. E. Mowlaei, M. S. Abadeh, and H. Keshavarz, "Lexicon generation using genetic algorithm for aspect-based sentiment analysis," in *2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES)*, Jun. 2018, vol. 13, no. 1, pp. 000133–000138,

doi: 10.1109/INES.2018.8523902.
[23] A. Agarwal and D. Toshniwal, "Application of lexicon based approach in sentiment analysis for short tweets," *Proceedings on 2018 International Conference on Advances in Computing and Communication Engineering, ICACCE 2018*, pp. 189–193, 2018, doi: 10.1109/ICACCE.2018.8441696.
[24] S. Sohangir, N. Petty, and Di. Wang, "Financial sentiment lexicon analysis," *Proceedings - 12th IEEE International Conference on Semantic Computing, ICSC 2018*, vol. 2018-Janua, pp. 286–289, 2018, doi: 10.1109/ICSC.2018.00052.
[25] Z. Yuan and L. Duan, "Construction method of sentiment lexicon based on word2vec," *Proceedings of 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference, ITAIC 2019*, pp. 848–851, 2019, doi: 10.1109/ITAIC.2019.8785471.
[26] P. D. T. Chathuranga, S. A. S. Lorensuhewa, and M. A. L. Kalyani, "Sinhala sentiment analysis using corpus based sentiment lexicon," *19th International Conference on Advances in ICT for Emerging Regions, ICTer 2019 - Proceedings*, 2019, doi: 10.1109/ICTer48817.2019.9023671.
[27] C. H. Chang, M. L. Wu, and S. Y. Hwang, "An approach to cross-lingual sentiment lexicon construction," *Proceedings - 2019 IEEE International Congress on Big Data, BigData Congress 2019 - Part of the 2019 IEEE World Congress on Services*, pp. 129–131, 2019, doi: 10.1109/BigDataCongress.2019.00030.
[28] S. Taj, B. B. Shaikh, and A. Fatemah Meghji, "Sentiment analysis of news articles: A lexicon based approach," *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies, iCoMET 2019*, 2019, doi: 10.1109/ICOMET.2019.8673428.
[29] F. Yin, Y. Wang, J. Liu, and L. Lin, "The construction of sentiment lexicon based on context-dependent part-of-speech chunks for semantic disambiguation," *IEEE Access*, vol. 8, pp. 63359–63367, 2020, doi: 10.1109/ACCESS.2020.2984284.
[30] D. H. Abd, A. R. Abbas, and A. T. Sadiq, "Analyzing sentiment system to specify polarity by lexicon-based," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 1, pp. 283–289, Feb. 2021, doi: 10.11591/eei.v10i1.2471.
[31] R. M. Sallam, M. Hussein, and H. M. Mousa, "Improving collaborative filtering using lexicon-based sentiment analysis," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 2, pp. 1744–1753, 2022, doi: 10.11591/ijece.v12i2.pp1744-1753.
[32] E. W. Pamungkas and D. G. P. Putri, "An experimental study of lexicon-based sentiment analysis on Bahasa Indonesia," *Proceedings - 2016 6th International Annual Engineering Seminar, InAES 2016*, pp. 28–31, 2017, doi: 10.1109/INAES.2016.7821901.
[33] H. Karamollaoglu, I. A. Dogru, M. Dorterler, A. Utku, and O. Yildiz, "Sentiment analysis on Turkish social media shares through lexicon based approach," *UBMK 2018 - 3rd International Conference on Computer Science and Engineering*, pp. 45–49, 2018, doi: 10.1109/UBMK.2018.8566481.
[34] S. B. Bin Rodzman, M. Hanif Rashid, N. K. Ismail, N. Abd Rahman, S. A. Aljunid, and H. Abd Rahman, "Experiment with lexicon based techniques on domain-specific Malay document sentiment analysis," *ISCAIE 2019 - 2019 IEEE Symposium on Computer Applications and Industrial Electronics*, pp. 330–334, 2019, doi: 10.1109/ISCAIE.2019.8743942.
[35] B. T. Pratama, E. Utami, and A. Sunyoto, "A comparison of the use of several different resources on lexicon based Indonesian sentiment analysis on app review dataset," *Proceeding - 2019 International Conference of Artificial Intelligence and Information Technology, ICAIIT 2019*, pp. 282–287, 2019, doi: 10.1109/ICAIIT.2019.8834531.
[36] K. Tsamis, A. Komninos, and J. Garofalakis, "Leveraging social media linguistic features for bilingual Microblog sentiment classification," *10th International Conference on Information, Intelligence, Systems and Applications, IISA 2019*, 2019, doi: 10.1109/IISA.2019.8900674.

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, 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 & 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

J.B. 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