Journal of Information Technology Management

Print ISSN: 2008-5893

Online ISSN: 2423-5059

Preprocessing of Aspect-based English Telugu Code Mixed Sentiment Analysis

Arun Kodirekka*

*Corresponding Author, Y.S.Rajasekhar Reddy University College of Engineering & Technology Acharya Nagarjuna University, Andhra Pradesh, India. E-mail: karun014@gmail.com

Ayyagari Srinagesh

Department of Computer Science and Engineering, RVR & JC College of Engineering, Andhra Pradesh, India. E-mail: asrinagesh@gmail.com

Abstract

Extracting sentiments from the English-Telugu code-mixed data can be challenging and is still a relatively new research area. Data obtained from the Twitter API has to be in English-Telugu code-mixed language. That data is free-form text, noisy, lexicon borrowings, code-mixed, phonetic typing and misspelling data. The initial step is language identification and sentiment class labels assigned to each tweet in the dataset. The second step is the data normalization task, and the final step is classification, which can be achieved using three different methods: lexicon, machine learning, and deep learning. In the lexicon-based approach, tokenize each tweet with its language tag. If the language tag is in Telugu, transliterate the roman script into native Telugu words. Words are verified with TeluguSentiWordNet, and the Telugu sentiments are extracted, and English SentiWordNets are used to extract sentiments from the English tokens. In this paper, the aspect-based sentiment analysis approach is suggested and used with normalized data. In addition, deep learning and machine learning techniques are applied to extract sentiment ratings, and the results are compared to prior work.

Keywords: English-Telugu code-mixed data; Natural language processing; Telugu Senti Wordnet; Machine learning; Deep learning.

Journal of Information Technology Management, 2023, Vol. 15, Special Issue, pp. 150-163
Published by University of Tehran, Faculty of Management
doi: https://doi.org/ 10.22059/jitm.2023.91573
Article Type: Research Paper

Received: December 22, 2022
Received in revised form: January 15, 2023
Accepted: February 20, 2023
Published online: March 15, 2023

Introduction

India has 22 official languages and is a multilingual nation with many unofficial languages. The English language is essential in all government and private sectors. Mixing the Telugu language with the English lexeme is convenient in regular conversations. In the 21st century, most people are using multiple languages in online conversations, leading to the rise of code-mixed data in huge sizes (Gundapu & Mamidi, 2020; Arun & Srinagesh, 2020). Mixing two or more languages in a word or sentence is called code-mixed or code-switched data. Code-mixed data can be extracted from Twitter, Facebook, Whatsapp, and YouTube are examples of social media sites. Code-mixed data is in the form of informal text, unstructured and noisy data like shortcut words, misspellings, and overlapping the words of one language into other languages (Padmaja et al., 2021; Ghosh et al., 2017; Malgaonkar et al., 2017).

Sentiments are extracted from code-mixed text, which is a more difficult task than English text. Many researchers are working on code-mixed data. Sentiment analysis is an essential part of natural language processing. It is used in decision-making, market analysis, recommendation systems and review analysis (Kodirekka & Srinagesh (2022)). The initial step in extracting sentiments is preprocessing, which can remove unwanted data. The next step is Tokenize each token's code-mixed data and language identification and assigns the parts-of-speech tagging to each valid token. Classification can be implemented in three ways: first, using the lexicon-based approach; second, using a machine learning approach; and third, using deep learning techniques (Kusampudi et al., (2021)).

In this paper, sentiment analysis for English-Telugu code-mixed data is performed. Extracting sentiments from data with mixed English and Telugu code is complex. One of the low resources mixed language combinations is English-Telugu data. Positive, negative, and neutral objects are members of the sentiment class. Aspect-based sentiments are extracted from noisy data.

Literature Review

Extracting the opinions from the Telugu-English code-mixed data and the language-distinguishing proof is a significant challenge. In this paper, word-level language identification was the proposed task. Various verified methods are Naive Bayes classification, Hidden Markov model, conditional Random Field, and Random Forest Classifier. A good f1-score of up to 91% was given for the word-level language recognition for the code-mixed data were conditional random field and hidden Markov models. Along with the language identification, Technical Domain Identification for the Telugu language with multichannel lSTM-CNN method and ICON 2020 data were used in this paper. It acquires an accuracy of up to 69.9% (Gundapu & Mamidi, 2021).

Code-mixed data is increasing in social networks, and the code-mixed text is noisy and free-form multilingual. The rapidly growing area of research is automated code-mixed text. This research mainly focuses on extracting the sentiments at the aspect-based level. English-Telugu bilingual roman script movie-related tweets are the input to the system, which were collected from the Twitter API. Preprocessing techniques were used to clean the data and replace slang words. Identification of named entities and language are at the aspect level. Romanized words from Telugu were transliterated into native alphabets. The Telugu text was classified into sentiment objects like positive and negative. The sentiment score accuracy was 79.9% (Padmaja et al., 2020).

The preprocessing of the English-Telugu movie tweets is the main focus of this work. The text was bilingual in Telugu and English and boisterous. Initially, data was cleaned, and sentiment class labels were assigned. There were two approaches to extracting sentiments from code-mixed tweets: the linguistic and machine-learning approaches. In the linguistic approach, tweets were tokenized. If Telugu was the language of the tokens, the roman script was transliterated into Telugu, and TeluguWordNet was then used to extract the sentiments from the Telugu words. This approach acquired an accuracy of up to 66.82%. The second approach was machine learning with the features like unigrams, N-grams, and Skip grams. The SVM model acquired 76.33% accuracy for training and test data.

Social network users have increased in recent years in multilingual parts of the world like India, and people are using code-mixed conversations in their regular communications. Code-mixed text can mix with Telugu-English and Tamil_English language. They classify the extremities of the code-mixed text into negative and positive opinions. This paper used two approaches to extract the sentiment classifications: lexicon bases and machine learning approaches. Naive Bayes and SVM methods were used in machine learning. These two approaches achieve good accuracy, up to 82% and 85% (Saikrishna & Subalalitha, 2022). English-Telugu tweets were composed in the roman script, which was acquired from the Twitter API. The collected dataset was more noisy and free-form text. Each word of the tweet was annotated with a language tag and sentiment tag assigned to the tweets. The machine learning technique was implemented to assign sentiment classification labels with several features, such as N-grams, emoticons, negation words, and recurring letters (Padmaja et al., 2021). Code-mixed is defined as the mixing of vocabulary and syntax of multiple languages. In this paper, they proposed SemEval 2020 Task 9 on sentiments extracted from the code-mixed data. Here two approaches were implemented, the first is word level embedding, and the second is FastText word embodying for morphology and semantics. For enhanced performance, the LSTM module employs these two techniques (Srinivasan & Subalalitha, 2021).

Methodology

Parts-of-Speech (POS) are an essential feature in natural language processing. POS tagging announces the accurate syntactic corrections on monolingual text. This research contains a report on the automated POS tagging of text using Telugu-English code-mixed data acquired from social media sites like Facebook (Naidu, et al., 2017). Social networks support multilingual conversations, and multilingual text causes code-mixed data. The code-mixed data is unstructured, noisy and free-form data with informal transliterations and misspellings. It was pretty challenging to obtain sentiment scores from Telugu-English code-mixed data. The Code-mixed Telugu-English tweets dataset was newly introduced. The dataset is annotated with a sentiment class label and language label for each token in the tweet. Annotation was done manually with the professionals in Telugu-English languages. The accuracy reported on this dataset is 80.22% by using unsupervised data normalization with the Multilayer Perception model (Gundapu et al., 2021).

Dataset: The English-Telugu code-mixed dataset extracts sentiments from the Twitter API. In this process, three steps are required: one is collecting the dataset, the second is normalizing the dataset, and the last is sentiment annotation of the dataset. The dataset is collected from the Twitter API or YouTube API, and these APIs' user comments or reviews were collected with different aspects of data. The dataset is tokenized into sentences and words. Hyperlinks, URLs, and sentences containing less than five words can be removed from the dataset. Dataset annotation plays a significant key role in this analysis. Here two types of annotations were done: Language identification and sentiment annotation in Table 1. The language tags for each word were included in English (EN), Telugu (TE), Named Entities (NE), and Universal (UNIV). The sentiment classification tags of each word are positive, negative and neutral.

Table 1. CMTET dataset

Polarity	Count
Positive	7,925
Negative	7,713
Neutral	4,219
Total	19,857

Two annotations are done manually with five language professionals in the English-Telugu language. The coherent Kappa scored 92.3% for the sentiment tags and 94.7% for language tag identification. The dataset is openly available and prepared to be used in code-mixed English-Telugu sentiment analysis.

Methodology: Sentiment extraction from the English-Telugu code-mixed corpus is the main objective of this paper. The following approaches, like lexicon-based approach, machine learning and deep learning approaches, are used to extract the sentiments from English-Telugu data which is a very complex task.

Lexicon-Based Approach: Lexicon-based approach depends on dictionaries used to define the sentiments from the sentence. Figure 1 explains the process of extracting the sentiments from the code-mixed English-Telugu corpus.

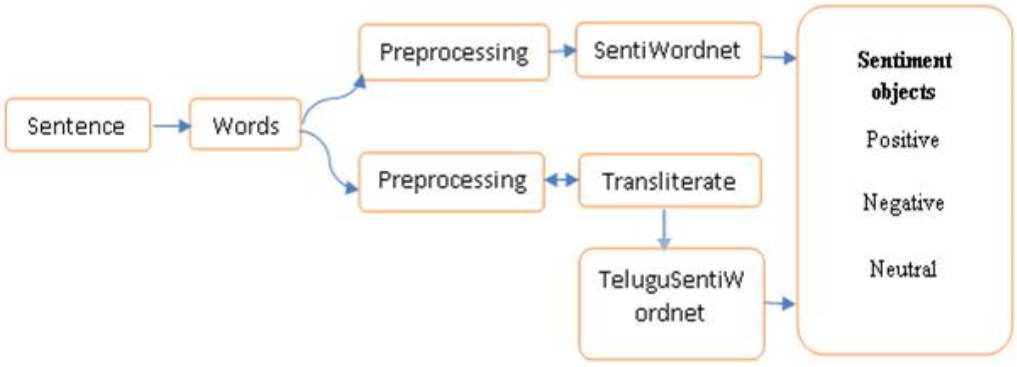

Figure 1. Lexicon based sentiment analysis

The sentences are tokenized into individual words and the language tag is identified at the word level. Words of the sentence are partitioned into Telugu (TE), English (EN) and Universe (UNIV) separately. For the English (EN) language tokens, the following preprocessing tasks can be done before the classification.

a) Correction of Misspellings words
b) Replacement of Emphasize words with dictionary words
c) Remove slang/acronyms words
d) Remove stop words

Misspellings: Informal data contains a lot of noisy data. In noisy data, spelling mistakes have the highest probability. The most efficient and fastest spelling correction tool available in the Python platform is Symspellpy, which is being used in this effort to address the problem.

Emphasize words: In social networks, people use very expressive words like cooooooool, thaaaaaanks, ooooooooooook. These words are not helpful in this analysis. Suitable regular expressions are used to reduce the repetition of characters into two. It becomes cool, thank, ook, but some words need spelling corrections. Then again, spelling corrections are done to normalize the words. Finally, all emphasized words become everyday words cooooooool → excellent, thaaaaaanks → thanks, and ooooooooook →ok.

Remove slang words or acronyms: Slang words and acronyms are not used in this process,

so they are removed from the sentence.

Removing stop words: The stop words are repeatedly used in sentence formation, but these words are not required in the analysis. The natural language toolkit in python supports stop words in English. All stop words were removed from the sentence.

Sentiment classification: Sentiments are extracted from every word which is preprocessed. The sentiments are classified as: positive, negative, and neutral. There are many dictionaries available for the English language to extract sentiments.

Transliteration: If the language tag is in Telugu (TE), the Telugu words represented in the roman script must be transliterated into Telugu native script. This transliteration process is supported in python by using the indic-transliteration package. All roman script words can transliterate into Telugu words with the encoding of ITRANS (Indian languages transliteration).

Preprocessing of Telugu roman script: Once Telugu roman script is transliterated, and then the word is verified with TeluguWordNet, which is available in the pyiwn python package. The size of the TeluguWordNet is 37269 words. If words are matched, then the next task is sentiment classification. If the words are not matched, then preprocessing task is required. The preprocessing task steps are as follows:

Repeated character reduction: English-Telugu code-mixed text faces the emphasized words in the conversation. These types of words are expressive words like the English language emphasized words. Using regular expressions in python, repeated words more than two times are reduced to only two times and then verified with TeluguWordNet. Example: “chaaaaaaaala” → “chaala”.

Insert vowels: Vowels are needed to insert mainly at the last character of the word, after the consonants in the word, and sometimes vowels can be repeated.

Example 1: Insert vowels, at last, the position "bagunnar, chal, manchild". All these words end with consonants, but Telugu words should end with vowels {'a',' e',' i',' o',' u'}. The result becomes "bagunnara, chala, manchild".

Example 2: Insert vowels after the consonants if required.

“Chdu”-->”chudu”, “ekkda”-->”ekkada”.

Example 3: Insert repeated vowels in the word if it is required "Miru" —>"miiru"; Transliteration and preprocessing are interdependent tasks. Every modification must be transliterated and verified in TeluguWordNet. This process will repeat until proper modification of the word is done.

TeluguSentiwordNet: TeluguSentiWordNet extracts the sentiments from the Telugu words. The dictionary size is 7,664 words: positive words are 2136, negative words are 4076, neutral words are 359, and ambiguous words are 1096 (Das et al., 2012; Thamaraimanalan et al., 2021). Telugu words are the output of the transliteration and preprocessing from the roman script. Words are verified with Telugu sentiwordnet, which assigns the sentiment classification object. Sentiments are extracted from the code-mixed data using the lexicon-based approach, with a collection of dictionaries and resources to obtain accuracy.

Machine learning Approach: The Dataset CMTET is the supervised dataset. Sentiment class labels are annotated for each tweet in the dataset. The dataset is divided into a training set with 70% and a testing set with 30% of the data. The dataset is cleaned and preprocessed before applying machine learning. The features are extracted from the dataset using n-grams and TF-IDF and then passed to the sentiment analysis.

The following machine-learning methods are applied to the dataset

a) Multinomial naive Bayes
b) Logistic regression
c) Random Forest
d) Support vector machine
e) Multi-level perceptron algorithm

Multinomial Naive Bayes Algorithm (MNB): MNB classification algorithm is used for text classifications in natural language processing (Faris et al., 2019). The Bayes Theorem is utilized for MNB classification.

$$P(A|B)= P(A)*P(B|A)/P(B)$$

A is the sentiment class label, and B is the code-mixed text. P(A|B) is the posterior probability class label given that text word occurring. P(A), P(B) probability occurrence of A and B. P(B|A) prior probability of B occurring on A.

Logistic regression: The algorithm for classifying the data and predicting binary outcomes is logistic regression. Logistic regression is the extension of linear regression where categorical outcomes are required (Divyapushpalakshmi et al., 2021).

Random Forest algorithm: The ensemble learning algorithm RF can be applied to classification and regression techniques. From noisy data, it is exploited to extract opinions (Bahrawi (2019)). The random classifier forest increases the prediction accuracy, which combines several decision trees in different subsets of the input dataset.

Support vector machine: Support vector machine is the best suitable algorithm for textual data, especially to extract sentiments from average English data and code-mixed data SVM algorithms get good accuracy. SVM works with kernel functions like linear, radial basis, and

sigmoid functions.

Multilayer perceptron algorithm: Multilayer perceptron is a type of feed-forward of a neural network model. Input layer, an output layer, and one or more hidden layers are present in this model. One of the finest models for sentiment extraction is the multilayer perceptron method.

Deep learning approach: Sentiment scores are extracted from the Telugu-English code-mixed dataset by using deep learning approach models. Three types of deep learning algorithms are as follows.

a) Artificial Neural Networks (ANN)
b) Convolutional Neural Networks (CNN)
c) Recurrent Neural Network (RNN)

Artificial Neural Networks (ANN): ANN is a group of multiple perceptron's at each layer. There are three layers of architecture: input, output and hidden layers. This ANN is also called MLP (Multilayer perceptron).

Convolution Neural Networks (CNN): CNN is used in image processing and textual analysis like sentiment analysis. It has a three-step procedure like Word embedding, 1D convolution and max-pooling, which is used to extract sentiments using CNN. Word embedding is used to convert the textual tokens into a numerical vector, 1D convolution can minimize the n-dimensionality into a one-dimensional vector, and max-pooling is the method to acquire maximum values.

Recurrent Neural Network (RNN): Recurrent neural networks are neural networks utilized in modelling sequence data. RNN can predict what will occur next using two inputs: current and past data. RNN is short-term memory. LSTM is the extension of RNN with an extended memory that can store long experiences for a long time.

Results and Discussion

Deep learning technique, machine learning approach, and lexicon-based approaches are the three models used for sentiment extraction from the English-Telugu code-mixed data.

Dataset: In this work, English-Telugu code-mixed benchmark dataset is used. The dataset is collected from Twitter, YouTube and movie topics. Every word is assigned to a language tag and a sentiment tag, which is assigned for each sentence. The sample dataset is provided in Figure 2.

```
NTL: We need Mr chari 's review on master
en en univ univ univ en en en


NEG: worst government . #YSRCP chala chethha ga paripalana chesthumdhi .
en en univ univ te te te te te univ


NEG: bayya nuvvu emina cheppu kani bagoledu ani cheppaku entha kastapadi thestharu
te te te te te te te te te te


POS: Dube gadini vadilesi manchhi Pani chesaru @RCBTweets 👍
ne te te te te te univ univ


POS: I came to watch thyview 's review crying after watching dil bechara . Movie felt very emotional . Did not cry for any movie but this
movie made me feel personal and made me cry for hours . Miss you sushant . And last quote in movie `` we do n't get to decide to when born
or when die , but we can decide how to live our life `` this quote is like made for sushant , and the word 'seri ' man literally cried .
Love you 300 sushant .
en en en en univ univ en en en en te te univ en en en en univ en en en en en en en en en te en en en en te en en en en univ en en en univ
en en en en en univ en en en en en en en en en en en en univ en en en en en en en en univ en en en en te en en univ en te en univ univ
en en en univ en en univ en univ


NTL: Bhaya enti bhaya review ela ichavv chaalaa anukuna mve gurinchi
ne te te en te te te te ne te
```

Figure 2. A sample of code-mixed data set with language and sentiment annotation

Lexicon-Based Approach: Sentiments are extracted from code-mixed data using a lexicon-based model. The dataset requires the normalization of text. Each word in the dataset was annotated with the language and sentiment tag assigned for the sentence. Tokenize the tweets and then transliterate them from the roman text into Telugu words. Telugu words are compiled into TeluguSentiWordNet to extract the sentiments. If the language of the token is in English, sentiments are extracted from the English dictionary. If the data is noisy, miss spelling, repeated characters and some roman scripts are not appropriately transliterated; all these are solved with the aspect-based code-mixed English-Telugu sentiment analysis (ACMS) by using a lexicon-based model that is compared with the current work. The proposed work acquires better accuracy with the precision, recall, f1_scores Table 2 and Figure 3.

Table 2. Benchmark sentiment model compared with existing lexicon model

Class	TPTECML (Padmaja et al., 2021)			ACMS		
	Precision	Recall	F_measure	Precision	Recall	F_measure
Positive	0.807	0.810	0.809	0.84	0.85	0.84
Negative	0.627	0.360	0.457	0.80	0.72	0.74
Neutral	0.360	0.556	0.437	0.65	0.58	0.52

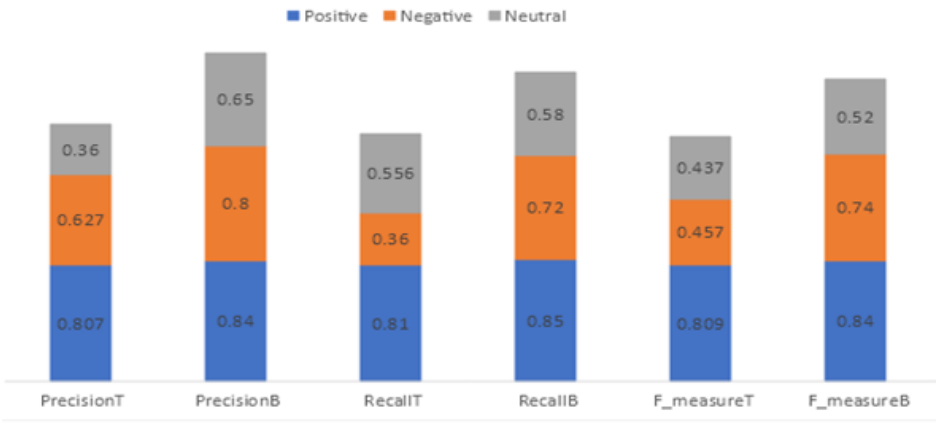

Figure 3. Proposed lexicon-based model ACMS with TPTECML

Figure 3 contains the comparison of existing systems and the proposed system performance measures: precision, recall, F_measureT are related to the existing system and precision, recall, and F_measureB are related to the proposed system.

Machine learning approach: The machine learning approach is used to extract the sentiment scores more accurately. Initially, preprocessing task is implemented in the code-mixed data to prepare the data in a required format. Data cleaning, spelling correction, and reduced repeated words and noisy text are done in this process. The dataset is partitioned into two parts with a ratio of 70:30 for the training and test sets. Training sets and test sets are applied in the machine learning algorithms.

Table 3. ACMS model compared with existing CMTET machine learning model

Model	Parameter	CMTET (Kusampudi et al., 2021)			ACMS		
		Precision	Recall	f1_score	Precision	Recall	f1_score
MN	Accuracy	76.66	67.48	67.80	77	75	76
LR	Accuracy	76.52	75.86	76.13	79	79	79
RF	Accuracy	75.81	75.81	75.67	76	76	74
SVM	Accuracy	74.98	73.61	74.05	79	79	79
MLP	Accuracy	78.08	78.8	78.31	79	79	79

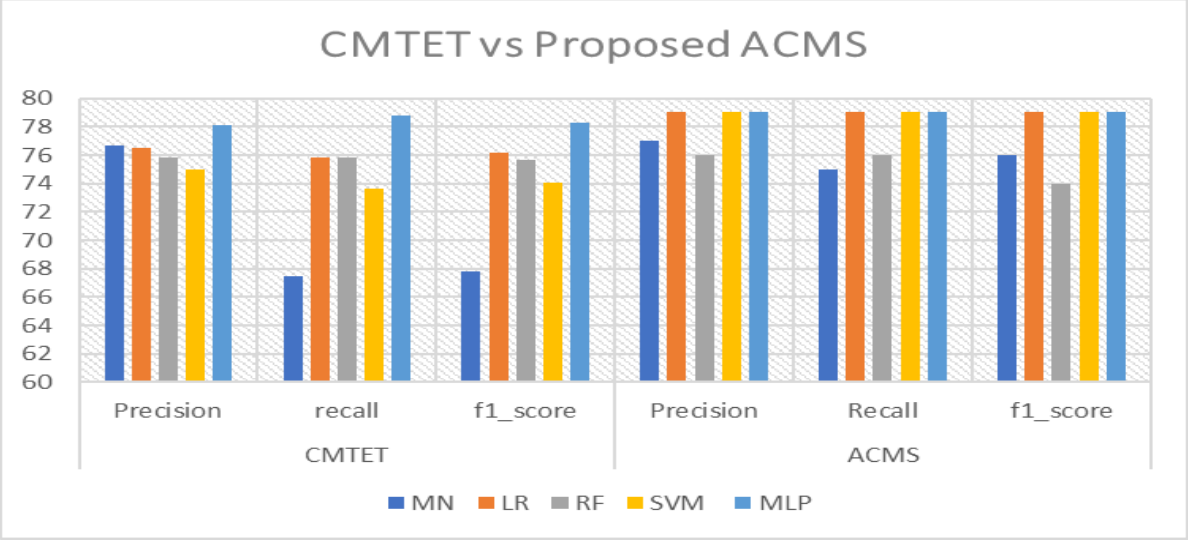

Figure 4. Proposed machine learning model ACMS with existing model CMTET

The machine learning algorithms are specified in the table, which is used to verify the accuracy, precision, recall, and f1_score performance measures. Aspect-based code-mixed English-Telugu sentiment analysis (ACMS) is the proposed method. The proposed method acquires better accuracy results when compared with the existing CMTET work. Observation of the resultant Table 3 ACMS method got better accuracy; out of that, SVM and MLP outperformed. Figure 4 shows the comparison of the proposed machine learning model ACMS with existing model CMTET.

Deep learning approach: The deep learning approach extracts the sentiment scores from the code-mixed dataset. Here, three algorithms are used in deep learning methods. The data set is partitioned into a 70:30 ratio before applying the algorithms. The preprocessed data were subjected to the implementation of CNN, Simple RNN, Bidirectional RNN, Simple LSTM, and Bidirectional LSTM algorithms. The bidirectional LSTM method draws good accuracy up to 81.54% based on the simulated values listed in Table 4. Figure 5 shows the comparison of ACMS with other conventional approaches.

Table 4. ACMS model with Deep learning approach

ACMS with Deep Learning		
Model	Loss	Accuracy
CNN	76.22	74.56
Simple RNN	59.39	78.185
Bidirectional RNN	73.55	77.92
Simple LSTM	56.51	78.98
Bidirectional LSTM	49.75	81.54

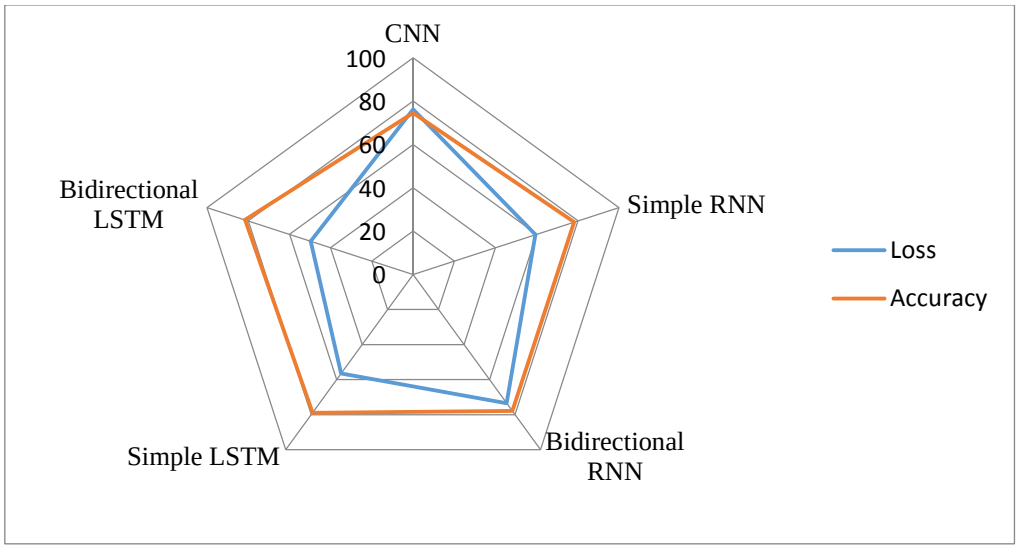

Figure 5. ACMS with deep learning approach

Conclusion

In this research, sentiment scores are obtained from code-mixed samples of the English-Telugu dataset. Normalization of the dataset is implemented on noisy data, misspellings, and free-form data. Using the proposed method, the ACMS lexicon-based approach achieved an accuracy of up to 84%. SVM and MLP algorithms in a machine learning-based approach achieved excellent accuracy. Deep learning method Bidirectional LSTM record also achieved good accuracy. The transliteration process generates wrong Telugu words, which contain misspellings not defined in Telugu dictionaries. The future work is to improve the accuracy of preprocessing English-Telugu code-mixed data.

Conflict of interest

The authors declare no potential conflict of interest regarding the publication of this work. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article

References

Arun, K., & Srinagesh, A. (2020). Multilingual Twitter sentiment analysis using machine learning. International Journal of Electrical & Computer Engineering (2088-8708), 10(6).

Bahrawi, N. (2019). Sentiment analysis using random forest algorithm-online social media based. Journal of Information Technology and Its Utilization, 2(2), 29-33.

Das, A., & Gambäck, B. (2012, July). Sentimantics: conceptual spaces for lexical sentiment polarity representation with contextuality. In Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis (pp. 38-46).

Divyapushpalakshmi, M., & Ramalakshmi, R. (2021). An efficient sentimental analysis using hybrid deep learning and optimization technique for Twitter using parts of speech (POS) tagging. International Journal of Speech Technology, 24(2), 329-339.

Farisi, A. A., Sibaroni, Y., & Al Faraby, S. (2019, March). Sentiment analysis on hotel reviews using Multinomial Naïve Bayes classifier. In Journal of Physics: Conference Series (Vol. 1192, No. 1, p. 012024). IOP Publishing.

Ghosh, S., Ghosh, S., & Das, D. (2017). Sentiment identification in code-mixed social media text. arXiv preprint arXiv:1707.01184.

Gundapu, S., & Mamidi, R. (2020). gundapusunil at SemEval-2020 Task 9: Syntactic semantic lstm architecture for sentiment analysis of code-mixed data. arXiv preprint arXiv:2010.04395.

Gundapu, S., & Mamidi, R. (2020). Word level language identification in english telugu code mixed data. arXiv preprint arXiv:2010.04482.

Gundapu, S., & Mamidi, R. (2021). Multichannel LSTM-CNN for Telugu technical domain identification. arXiv preprint arXiv:2102.12179.

Habibi, S. (2016). Smart innovation systems for indoor environmental quality (IEQ). Journal of building engineering, 8, 1-13.

Indra, S. T., Wikarsa, L., & Turang, R. (2016, October). Using logistic regression method to classify tweets into the selected topics. In 2016 international conference on advanced computer science and information systems (icacsis) (pp. 385-390). IEEE.

Kodirekka, A., & Srinagesh, A. (2022). Sentiment Extraction from English-Telugu Code Mixed Tweets Using Lexicon Based and Machine Learning Approaches. In Machine Learning and Internet of Things for Societal Issues (pp. 97-107). Springer, Singapore.

Kusampudi, S. S. V., Sathineni, P., & Mamidi, R. (2021, September). Sentiment Analysis in Code-Mixed Telugu-English Text with Unsupervised Data Normalization. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021) (pp. 753-760).

Malgaonkar, S., Khan, A., & Vichare, A. (2017, September). Mixed bilingual social media analytics: case study: Live Twitter data. In 2017 international conference on advances in computing, communications and informatics (ICACCI) (pp. 1407-1412). IEEE.

Naidu, R., Bharti, S. K., Babu, K. S., & Mohapatra, R. K. (2017, March). Sentiment analysis using telugu sentiwordnet. In 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET) (pp. 666-670). IEEE.

Padmaja, S., Bandu, S., & Fatima, S. S. (2020). Text processing of Telugu–English code mixed languages. In Advances in Decision Sciences, Image Processing, Security and Computer Vision (pp. 147-155). Springer, Cham.

Padmaja, S., Fatima, S., Bandu, S., Nikitha, M., & Prathyusha, K. (2020). Sentiment Extraction from Bilingual Code Mixed Social Media Text. In Data Engineering and Communication Technology (pp. 707-714). Springer, Singapore.

Padmaja, S., Nikitha, M., Bandu, S., & Sameen Fatima, S. (2021). Feature Impact on Sentiment Extraction of TEnglish Code-Mixed Movie Tweets. In Smart Computing Techniques and Applications (pp. 487-493). Springer, Singapore.

Saikrishna, K. S. B. S., & Subalalitha, C. N. (2022). Sentiment Analysis on Telugu–English Code-Mixed Data. In Intelligent Data Engineering and Analytics (pp. 151-163). Springer, Singapore.

Srinivasan, R., & Subalalitha, C. N. (2021). Sentimental analysis from imbalanced code-mixed data using machine learning approaches. Distributed and Parallel Databases, 1-16.

Thamaraimanalan, T., RA, L., & RM, K. (2021). Multi biometric authentication using SVM and ANN classifiers. Irish Interdisciplinary Journal of Science & Research (IIJSR).

Thara, S., & Poornachandran, P. (2022). Social media text analytics of Malayalam–English code-mixed using deep learning. Journal of big Data, 9(1), 1-25.

Bibliographic information of this paper for citing:

Arun, Kodirekka & Ayyagari, Srinagesh (2023). Preprocessing of Aspect-based English Telugu Code Mixed Sentiment Analysis. *Journal of Information Technology Management*, 15 (Special Issue), 150-163. https://doi.org/ 10.22059/jitm.2023.91573