IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CODE-MIXED SENTIMENT ANALYSIS FOR MULTI-LINGUAL[HI-EN] SOCIAL MEDIA TEXT

[1]Sushrut Kuchik, [2]Rutuja Pujare, [3]Anupama Rai, [4]Romax Rajeev and [5]Dr. Sharvari S. Govilkar

[1,2,3,4]Student,[5]Head of Department
Department of Computer Engineering
Pillai College of Engineering, University of Mumbai, New Panvel, India

Abstract: India has a great diversity of languages. People use English and colloquial languages to put forward their ideas and views on social media platforms like Twitter and Facebook. This gives rise to code mixed text. Sometimes, emoticons are also used along with text to express the emotions and to indicate sarcasm. Analyzing this complex multilingual code-mixed text becomes an enormous challenge for researchers who work in the domain of Text Analytics. The research work regarding sentiment analysis of code-mixed text, sarcasm detection and negation handling has been done separately and there has been no cumulative model to do the same. The proposed research work will use a hybrid approach, consisting of Lexicon based methodology and Unsupervised Deep Learning algorithms to extract sentiments from multilingual code-mixed text from social media especially for the language pair: English-Hindi.

***Index Terms* - Code Mixed Text, Social Media Text, Natural Language Processing, Sentiment Analysis, Lexicon Approach, Supervised and Unsupervised Learning.**

I. INTRODUCTION

Natural Language Processing or NLP is a field in the study of languages where computers interpret the language used by humans, i.e. natural language. The input to the machine can be either in speech or in text format and the task of the machine would be to comprehend the input and generate appropriate output. Natural language processing has two main approaches: using Lexicon approach and using Machine Learning. Lexicon Approach involves various methods to break sentences into tokens, then try to understand the meaning of the tokens with the help of various approaches. Machine Learning approach involves feature extraction of POS, bigrams etc. and training the model with the dataset.

Code-mixing is the mixing of two or more languages or language varieties in a single sentence. It is a type of script in which both native and English is written in Roman script. People mainly use Code-Mixed text on social media, to assert their opinions or to put forward their ideas in an effective way and an easier way. In a country having linguistic diversity like India, analysing such scripts is a huge challenge in the field of NLP for detecting sentiment, sarcasm, irony etc. Additionally, wide use of emoticons in today’s time calls for including them in such analysis procedures.

An example of code mixed text is as follows:

> Air Pollution level in Delhi doubled within hours on Sunday due to use of #firecrackers to celebrate #Dusshera2020!!!!
> I think its STUPIDITY to grt levels....ke pehle firecrackers jalao, and agle din air pollution pe baatein aur protest karo. Extraña!!
> Humein haq nahi hai complain karne ka if we burn crackers. Log bohoooot hypocritic hai.
> U knw Delhi is the only city where u hve to step inside 2 get sm fresh air □□□.
> Politicians ko air pollution ke issue ko address karna chahiye but vo apne “IMP” kaamo mein busy hai□□□♂□♂□♂

In this paper, we propose a hybrid model approach consisting of lexicon and machine learning algorithms to analyze sentiments effectively, identify sarcasm and provide more accurate results in production as compared to the existing systems. We also aim to include emoticon analysis and word sense disambiguation in the course of preprocessing the text.

II. RELATED WORK

One of the earliest works on code mix text was started by Sharma et al.[1] proposed a model that does word-level language identification in the code- mix script of Hindi and English, automatic transliteration of Romanized English language words and judged the sentiment of the script as positive or negative. These authors also worked on additional parts in language identification, figuring out ambiguous words and recognizing sounds. Also, A. Nagvenkar et al.[2], mainly worked on in which 'sense' a word is used in Hindi and English by defining Cross Lingual Word Semantic (CLWS) similarity as a task to find the semantic similarity between two words across languages. They presented a system that computes CLWS similarity between two languages : source(English) and scarce(Hindi). Additionally, D Singh et al.[3], contributed in the field of NLP majorly when they proposed detecting Multiword Expressions for Hindi Language using Word Embeddings and WordNet-based features. Focussing on identifying the co-occurrence of the two words using word bags, they concluded the research that the WordNet-based approaches performs better than Word Embedding based approaches.

Use of attention mechanisms was proposed by Y. Lal et al.[17] using two Bi-LSTM models called Collective and Specific Encoders. This method worked well with compound sentences, which was a drawback with the normal LSTM approach, with a good accuracy. D. Gupta et al.[5] worked on code-mixed data from Government portals. Cosine similarity and CNN Models were used to analyze sentiments. B. M. Badr et al.[4] inferred that POS tags improve classification performance when they are attached to the unigrams. Also, skip gram features are capable of capturing informative sentiment terms than bigram features. K. Ravi et al.[6] recommended using k-fold cross validation for improving the accuracy and increasing the number of classes of classification. Also, they M. Zhang et al.[7] developed a DNN Model for sarcasm detection. They inferred that DNN improves accuracy for sarcasm detection with different error distributions compared with discrete manual features.

P. V. Veena et al.[8] used trigram and 5-gram for classification purposes which produced an accuracy of 93%. Importance of preprocessing of input text is emphasized for improving accuracy. A. Pravalika et al.[9] used a hybrid system consisting of lexicon-based approach and machine learning approach for sentiment analysis. Trie data structure is used for English and Hindi Dictionaries. Importance of including emoticons in the process of polarity checking is highlighted by P. Yadav et al.[10]. Factors affecting sentiment analysis and pros of feature selection techniques are to be considered before proceeding with the implementation. S. Ghosh et al.[11] extended the preprocessing step by expanding the abbreviations. Also many new methods like counting multiple repetitions of a character in a word, counting the number of switchings done between languages and counting the number of uppercase words were employed in a bid to increase the accuracy of the model.

S. Swami et al.[13] worked on EN-HI code-mixed dataset of tweets to check for the presence of sarcasm and irony by annotating a token with a language tag. Supervised classification system was developed using Random forest classifiers for word n-grams and performing 10-fold cross validation and achieved an average F-score of 78.4. Also, P. Mishra et al.[14] for identifying the sentence level sentiment polarity of the code-mixed dataset on HI-EN and BN-EN. Voting Classifier for BN-EN and SVM with (2,6) char n-grams. Shalini K et al.[15] on BEN-EN, HI-EN corpus and considered data from Facebook comments in KAN-EN. They experimented with the distributed representation of text through Doc2Vec, Fasttext, CNN and Bi-LSTM models where Fasttext representation outperformed for all the 3 datasets. A new clustering algorithm from R. Singh et al.[12] used the similarity function between two vectors to resolve various occurrences of a particular word because of change in spelling, thereby improving the accuracy of sentiment analysis and POS Tagging.

Attention based deep learning technique was used in S. Mukherjee et al.[16] which does fusion of character and word feature to develop a robust classifier for code-mixed text of Hindi-English. Here a single sentence of a sequence of 200 characters and also space characters. A.Saini [18] implemented Rule-based technique using ML model GradientBoostingRegressor model which is an ensemble model and uses decision trees internally. Three different datasets with 6 classes were considered and gave good accuracy for the datasets.

So far, we have identified the following gaps. Firstly, language identification is difficult due to Word Sense Ambiguity of code mix text. Additionally, emoticons were rarely used for sentiment analysis. In some papers, deep learning models have fared better, while in some cases, the same type of models have fared lower than expectations. Also, conflict of opinion results in misclassification of text. Uppercase and repeating characters theoretically hold sentiment values, but this was not seen during tests. Finally, handling Scriptio Continua text, i.e. handling texts in which constraint on maximum number of characters is mentioned, where users tend to combine words to fit more text.

III. CODE MIXED SENTIMENT ANALYSIS SYSTEM

The system overview is presented in this section. The input to the system would be Code Mix text from social media platforms like Twitter and Facebook which will be extracted from Twitter using tweepy and from Facebook using the python-facebook-api.In order to achieve better domain results, we combined ML based techniques and the Lexicon Based approach to build Hybrid domain systems to inherit advantages and eliminate disadvantages.

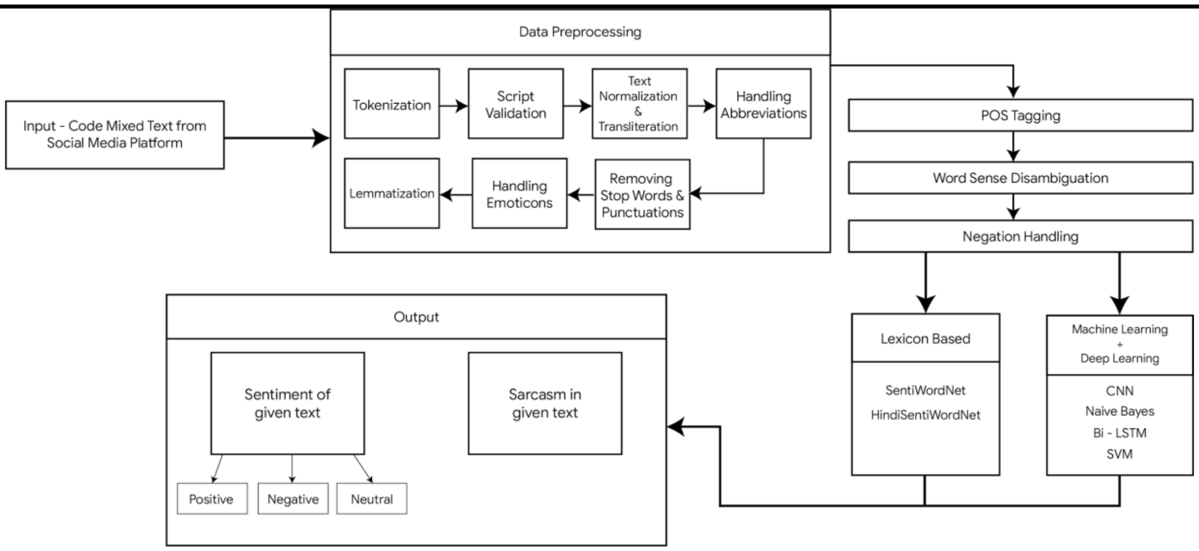


Fig. 3.2 Proposed system architecture

The text preprocessing step would be a standard procedure. We would be converting the code mix text into individual tokens. Then we would identify the language and normalize the text followed by transliteration. Abbreviations would be expanded and emoticons would be replaced by their actual alternate text. Stop words and punctuations will be removed followed by lemmatization in the final phase of preprocessing. The main procedures would involve:

1. **POS Tagging:** POS (part-of-speech) tagger marks up a word in a corpus to its corresponding part of a speech tag based on its context and definition. There are different techniques for POS Tagging like Rule-Based Methods, Transformation Based, Stochastic based. Single best tag is given to every word.

Input: ['Politicians', 'air', 'pollution', 'issue', 'address', 'but', 'important', '□□□□', 'busy', <sad><sad>']
Output: [‘Politicians/NOUN’, ‘air/NOUN’, ‘pollution/NOUN’, ‘issue/NOUN’, ‘address/NOUN’, ‘but/CNJ’, ‘important/ADJ’, ‘□□□□/NOUN’, ‘busy/ADJ’].

2. **Word Sense Disambiguation:** Word Sense Disambiguation is the solution to ambiguity which arises due to different meanings of words in different contexts which are spelled the same.

Input: [‘Politicians/NOUN’, ‘air/NOUN’, ‘pollution/NOUN’, ‘issue/NOUN’, ‘address/NOUN’, ‘but/CNJ’, ‘important/ADJ’, ‘□□□□/NOUN’, ‘busy/ADJ’].
Output: [‘Politicians/NOUN’, ‘air/NOUN’, ‘pollution/NOUN’, ‘issue/NOUN’, ‘address/VERB’, ‘but/CNJ’, ‘important/ADJ’, ‘□□□□/NOUN’, ‘busy/ADJ’].

3. **Negation Handling:** Negations are those words which affect the sentiment orientation of other words in a sentence. Negation words invert the polarity of the sentence or words affected by it. Negation handling is an automatic way of determining the scope of negation. Examples of negation words include not, no, never, cannot, shouldn’t, wouldn’t, etc.

Input: □□□□ □□□ □□□□ □□ complain □□□□ □□ if we burn crackers.
Output: □□□□[0] □□□[0] □□□□[-1] □□[0] complain[-1] □□□□[0] □□[0] if[0] we[0] burn[0] crackers[0].

4. **Hybrid Technology:** In this approach, Lexicon based methods using SentiWordNet and HindiSentiWordNet are used along with Machine Learning methods using classifiers and Neural Networks.

A. Lexicon based methods work with SentiWordNet database consisting of a token and the sentiment value of that token. SentiWordNet is a kind of database or a lexical resource for determining the sentiment of a particular word. It contains the words in a language along with its part of speech, polarity meaning whether the word is positive, negative or neutral. Additionally, the words’ synonyms and glossary are present.

Example of a SentiWordNet entry:

Table 3.2 SentiWordNet Structure

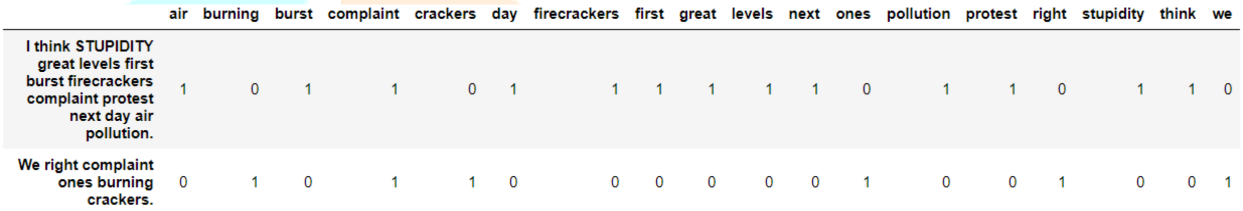
POS,ID	PosScore	NegScore	Synset	Glossary
a,00641158	0.5	0	polite#1	showing regard for others in manners, speech etc.
a,0064516	0.375	0.12	likely#4	within the realm of credibility; "not a very likely excuse"

For each token, an associated positive score and negative score is attached, along with its POS tag.

B. Machine Learning methods recognize mathematical patterns from data and use that knowledge to classify sentiments accurately. Bag of words model (BoW) is a way of extracting features from text for use in modeling in machine learning algorithms. It is a representation of text that describes the occurrence of text that describes the occurrence of words within a document.

Input: "I think STUPIDITY great levels first burst firecrackers complain protest next day air pollution", "We right complaint ones burning crackers"

Output:



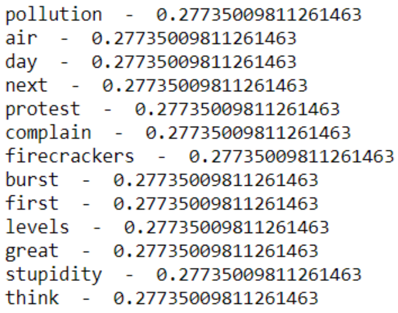
	air	burning	burst	complaint	crackers	day	firecrackers	first	great	levels	next	ones	pollution	protest	right	stupidity	think	we
I think STUPIDITY great levels first burst firecrackers complaint protest next day air pollution.	1	0	1	1	0	1	1	1	1	1	1	0	1	1	0	1	1	0
We right complaint ones burning crackers.	0	1	0	1	1	0	0	0	0	0	0	1	0	0	1	0	0	1

Fig 3.3 Bag of Words Representation

TF-IDF is a short form of Term Frequency – Inverse Document Frequency. It consists of two parts, Term Frequency which summarizes how often a given word appears within a document and Inverse Document Frequency that downscales words that appear a lot across documents. The scores have the effect of highlighting words that are distinct (contain useful information) in a given document.

Input: "I think STUPIDITY great levels first burst firecrackers complain protest next day air pollution", "We right complaint ones burning crackers"

Output:

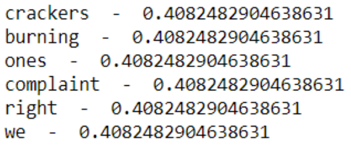


```
pollution  -  0.27735009811261463
air  -  0.27735009811261463
day  -  0.27735009811261463
next  -  0.27735009811261463
protest  -  0.27735009811261463
complain  -  0.27735009811261463
firecrackers  -  0.27735009811261463
burst  -  0.27735009811261463
first  -  0.27735009811261463
levels  -  0.27735009811261463
great  -  0.27735009811261463
stupidity  -  0.27735009811261463
think  -  0.27735009811261463
```

Fig 3.4 TF-IDF Values

5.. Expected Output:

Air Pollution level in Delhi doubled within hours on Sunday due to use of #firecrackers to celebrate #Dusshera2020!!!!
I think its STUPIDITY to grt levels....ke pehle firecrackers jalao, and agle din air pollution pe baatein aur protest karo. Extraña!!
Humein haq nahi hai complain karne ka if we burn crackers. Log bohoooot hypocritic hai.
U knw Delhi is the only city where u hve to step inside 2 get sm fresh air □□□.
Politicians ko air pollution ke issue ko address karna chahiye but vo apne "IMP" kaamo mein busy



```
crackers  -  0.4082482904638631
burning  -  0.4082482904638631
ones  -  0.4082482904638631
complaint  -  0.4082482904638631
right  -  0.4082482904638631
we  -  0.4082482904638631
```

hai□□□♂□♂□♂
Sentiment value: -0.5
Sarcasm: Detected

IV. CONCLUSION

The study of code-mixed sentiment analysis is presented in the aforementioned sections. Previous works related to code-mixed sentiment analysis are studied thoroughly and analyzed for further improvements. The existing architecture is explained along with some shortcomings. Proposed architecture seeks to eliminate those shortcomings using a hybrid approach. The hybrid approach consists of two methods - Lexicon and ML based methods. We have done a cumulative study of code mixed text, sarcasm detection and negation handling while the earlier research was pertaining only to these individual fields. The cumulative study would thereby open a wider area of applications in the field of NLP.

V. ACKNOWLEDGMENT

We would like to thank our Guide and Head of the Department, Dr Sharvari Govilkar for being ever supportive and helpful. Ma'am always pushed us to do better and showed immense faith in us.

In the end we would like to thank our parents for pushing us to achieve our best and motivating us to strive for excellence. We couldn't have gotten this far without their support.

REFERENCES

[1] S. Sharma, P. Srinivas and R C. Balabantaray, "Sentiment Analysis of Code – Mix Script", in *International Conference on Computing and Network Communications (CoCoNet'15), Trivandrum, India (2015)*

[2] A. Nagvenkar, J. Pawar, P. Bhattacharyya, "Let Sense Bags Do Talking: Cross Lingual Word Semantic Similarity for English and Hindi", 2015

[3] D. Singh, S. Bhingardive, K. Patel and P. Bhattacharyya. "Detection of Multiword Expressions for Hindi Language using Word Embeddings and WordNet-based Features", In the *Proceedings of the 12th International Conference on Natural Language Processing* (2015)

[4] B. M. Badr and S. S. Fati "Using Skipgrams, Bigrams, and Part of Speech Features for Sentiment Classification of Twitter Messages", in *Proceedings of the 12th International Conference on Natural Language Processing*, 2015

[5] D. Gupta, A. Lamba, A. Ekbal and P. Bhattacharyya, "Opinion Mining in a Code-Mixed Environment: A Case Study with Government Portals", in *Proceedings of the 13th International Conference on Natural Language Processing*, 249 - 258, December 2016.

[6] K. Ravi and R. Vadlamani "Sentiment classification of Hinglish text" , in *3rd International Conference on Recent Advances in Information Technology (RAIT)*, 2016

[7] M. Zhang "Tweet Sarcasm Detection Using Deep Neural Network", in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 2449–2460, Osaka, Japan, December 11-17 2016*, 2016

[8] Veena P V, Anand Kumar M, Soman K P. "An Effective Way of Word-level Language Identification for Code-mixed Facebook comments using Word-Embedding via Character-embedding" in *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017

[9] A Pravalika, Vishvesh Oza, N P Meghana and S Sowmya Kamath "Domain-specific Sentiment Analysis Approaches for Code-mixed Social Network Data", in *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2017

[10] P. Yadav and D. Pandya "SentiReview: Sentiment Analysis based on Text and Emoticons" in *International Conference on Innovative Mechanisms for Industry Applications (ICIMIA 2017)*, 2017

[11] S. Ghosh, S. Ghosh and D. Das "Sentiment Identification in Code-Mixed Social Media Text", July 2017

[12] R. Singh, N. Choudhary and M. Shrivastava, "Automatic Normalization of Word Variations in Code-Mixed Social Media Text" in *19th International Conference on Computational Linguistics and Intelligent Text Processing*, March 2018.

[13] S. Swami, A. Khandelwal, V. Singh, S. S. Akhtar and M. Shrivastava "A Corpus of English-Hindi Code-Mixed Tweets for Sarcasm Detection" in *19th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing-2018)*, May 2018.

[14] P. Mishra, P. Danda and P. Dhakras "Code-Mixed Sentiment Analysis Using Machine Learning and Neural Network Approaches", August 2018.

[15] Shalini K, B. Ganesh HB, A. Kumar M and Soman K P "Sentiment Analysis for Code-Mixed Indian Social Media Text With Distributed Representation" in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (2018)*, 2018.

[16] S. Mukherjee, V. Prasan, A. Nediyanchath, M. Shah, N. Kumar "Robust Deep Learning Based Sentiment Classification of Code-Mixed Text", December 2019.

[17] Y. K. Lal, V. Kumar, M. Dhar, M. Shrivastava and P. Koehn, "De-Mixing Sentiment from Code-Mixed Text" in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop - 371–377,* July 2019.

[18] A. Saini, "A Novel Code-Switching Behavior Analysis in Social Media Discussions Natural Language Processing" in *2019 IEEE International Conference on Big Data (Big Data),* 2019.