International Journal on Advances in Engineering Technology and Science
Volume: 5, Issue: 1, 2024, e-ISSN: 2455-3131
DOI: 10.5281/zenodo.10719323

Performance Analysis using Machine Learning for Code Mixed Languages in Sentiment Analysis

Shruti Mathur[1], Gourav Shrivastava[2]

[1]Research Scholar, [2]Associate Professor
Sanjeev Agrawal Global Educational University, Bhopal

Abstract- **Social media podiums like Twitter, Facebook, and Instagram have gained a lot of attention these days and have become one of the most prominent platforms to communicate, share thoughts and voice opinions. Detection of human emotions like happiness, sadness, anger, sarcasm etc. in textual communications has, therefore, become very important Sarcasm is a way of communication that creates gap between the anticipated meaning and the genuine meaning comprehended from the conversation. Communication and human relations over social media sites like Facebook, Twitter circles around a lot of sarcasm and debates. Sarcasm detection is an important processing problem which is needed to understand the human and machine communication better. Code mixing, as the name suggests, alludes to blending various dialects or more than one language in a solitary expression or a sentence. For a multilinguistic country like India, code mixing has become a very common practice on social media platforms since the pandemic since it is easier for the users to use their native language along with expressing their feelings. This paper aims to understand the gap between the emotion and the contextual meaning by using different machine learning approaches for Sarcasm Detection of code-mixed Hi-En dataset. The algorithms used in this paper are Bernoulli Naïve Bayes, Logistic Regression and Support Vector Machine. SVM outperforms all the used algorithms giving an accuracy of 87.36%.**

Keywords: **Code-mixed language; sarcasm detection; Natural language processing.**

I. INTRODUCTİON

Code-mixed data is a result of including two or more languages in the same document or text. Code-mixing refers to the practice of using more than one language or linguistic variety in a single conversation or sentence. It is a common phenomenon in multilingual societies and is often used by bilingual or multilingual speakers to express themselves more effectively. Due to the unstructured nature of social media platform users can easily switch between languages easily. Code-switching is the fluent use of two or more languages at the same time, without any apparent hesitation. Code-mixing can occur at different levels, ranging from a single word or phrase to an entire sentence or discourse. It can be used for various purposes, including to convey emphasis, express solidarity, or indicate social status or identity. Code-switching is common in many languages and other cultures, especially among speakers of different ethnic groups and socioeconomic backgrounds. In this paper the focus is on linguistic techniques that extract code-mixed text from unstructured data streams.

Sentiment analysis of two or more language es is an important task to determine the sentiment polarity of code-mixed dataset of pairs of Indian languages like Hindi-English, Bengali-English, Tamil-English, etc [1]. The goal is to find out whether a dataset is positive or negative in terms of its sentiment in terms of the textual data. The method will be carried out using natural language processing techniques on corpora with larger degree of bilingual input than previous approaches. Depending upon the sentiment polarity of sentence/phrases/words the sentiments are classified as positive, negative, neutral, mixed feelings. There is an array of sentiments communicated in conversations like joy, sorrow, fear, anxiety, sarcasm etc. It is relatively easier to detect simple sentiments like joy or sorrow but code-mixing gets really complex while detecting more deep and complex sentiments like sarcasm. In addition, sarcasm is a very difficult art form, even for experienced writers. The message implied by the literal meaning of words, such as "You good singer" may be in direct contrast to their actual literal meaning. For example, if you want to convey that an individual was just rude and insulting with their tone of voice, but they really aren't talking sarcastically. This can be done through careful word selection and tone of voice. Unlike the simpler sentiments, sarcasm detection, in a standalone text, is a non-trivial task due to its below-the-surface semantics. Hence, code-mixing in such scenarios becomes very difficult in order to convey the intentional ambiguity in communication.

Code mixing is an orthogonal concept of code segmentation and it involves selecting a subset of the source code to be processed. In machine learning terminology it is referred as a feature extraction. Code mixing is used predominantly in applications such as language identification, linguistic processing, question answering, sentiment analysis, entity extraction and attribute extraction.

For computers to be able to interpret natural language, it must first be learned by the computer. The problem of machine learning is how to make computers understand human language. For instance, in statistical machine translation (SMT), a series of text is fed into a system which learns from examples shown to it. This approach has been widely applied to summarization, named entity recognition and part-of-speech tagging [2].

Types of code mixing-

- Inter Sentential/Extra Sentential - this occurs at the margins of the sentence. E.g. - Kudos! Aap jeet gaye.
- Intra Sentential- this occurs within the sentence. E.g. - I like having home cooked rotis for lunch

© 2016-2024, IJAETS

Sentiment analysis, also known as opinion mining or emotion analysis, is a piece of NLP that is utilized to perform different errands like dissecting the web patterns, film evaluations or even the tone of the discussion, or an assessment. There are two primary ways to deal with opinion examination subjectivity investigation and feeling grouping. Subjectivity examination is worried about identifying conclusions or feelings, while opinion arrangement is worried about sorting those suppositions into various polarities or rankings. These polarities incorporate positive, negative and impartial feelings. Also, sentiments/ opinions are also described through emoticons, idioms and abbreviations.

With the advent of amplified use of social media stages like Twitter, Facebook, and Instagram etc. for routine communication between people across the globe, it has become imperative to detect, analyse and study the linguistic emotions in these textual chat communications. Linguistic expressions like humour, anger, aggression, sarcasm etc. also need to be detected in these textual communications. The automatic detection of these human emotions has found an important place in today's research. Automatic detection of these emotions means use of a computational method to detect the occurrence of these feelings. This paper focuses on the detection of such emotion which is frequently a part of communication and is very difficult to detect i.e., Sarcasm. Sarcasm is a tool that can be used in many industries, including web development, to communicate the exact opposite of what you mean. As a communication tool, sardonic or ironic humour use sarcasm to communicate the exact opposite of your actual meaning. Example: "You are getting close to your weight-loss goal", said the dietician with heavy sarcasm, as she sees a box of empty sweets. Sarcasm being a sentiment, is very difficult to detect in code-mixed language like Hindi-English and is a wide area of research these days. Code-mixed text is text that has been translated from one language to another. In Twitter, when someone uses code-mixed text, they are trying to express their opinion or communicate something. Code-mixed text is easy and effective to use, making it globally acceptable.

This paper reviews the code-mixed language pair for sarcasm detection on social media. The paper will specifically emphasize on use of Hindi-English language mix for sarcasm detection. A code-mix dataset consists of phrases and words from two are more languages in a single sentence. It is generally used by people who understand and use more than one language to communicate in their daily life. Nowadays globally people are not monolingual rather they communicate with a mixture of several languages [3]. Code-Mix language is widely seen on social media and with so many social media applications available the amount of code-mix data available is tremendous. India being a multi-linguistic society, understanding of code-mixed text has become an important area of research especially relating to social media [4]. However, this data has various combinations of languages especially in India where 22 different languages exist.

Sarcasm detection is a relatively specialised area of NLP research, a particular application of sentiment analysis where the focus is on purely on sarcasm and not the overall sentiment. Consequently, the goal of this sector is to determine whether a particular text is satirical or not. Sarcasm detection is a significant processing challenge which is required for better comprehension and to act as an interface for two-way communication between machines and humans. Being able to recognize the contradiction is the fundamental issue that it highlights in sentence processing and hence, it serves as a sub-task of text classification task in NLP [5].

The sarcasm over social media can be of various types. Social media is an open platform to let people express their thoughts and feelings over something or someone. An individual may like-dislike, agree or disagree with another person's tweets, posts, or opinions. It is however easier for humans to understand the gap between the underlying sarcastic nature of the comment, but the machine brain cannot easily differentiate. This is an addition to the numerous machine learning as well as natural language processing approaches that have been used to address the problem. The paper focuses on code-mix data for the Hindi-English dataset abbreviated as the Hi-En dataset.

Table 1. An Example of a Table

Tweet: "Ha ha ha, really great job, yaar. Tune dikha diya that even a clock that doesn't work is right twice a day." **Translation:** "Ha ha ha, really great job, man. You proved that even a clock that doesn't work is right twice a day."	Sarcasm
Tweet: "Kamal ka kaam kiya hai, dil se mehnat ki hai" **Translation:** "You have done an amazing job, worked hard."	Non-Sarcasm
Tweet: "Very well done, shayad apne aapko Einstein samajh rakha hai." **Translation:** "Very well done, perhaps you have thought yourself to be Einstein."	Sarcastic
Tweet: "Bahut hi unique kaam kiya hai, koi bhi nahi kar sakta hai." **Translation:** "Very unique work done, no one else can do it."	Sarcastic
Tweet: "In dino economy achi chal rahi h." **Translation:** "These days the economy is doing well."	Non-Sarcastic

Twitter messages, or tweets, are typically identified by hash tags, such as #mad, #driving, #happy, etc. These hash tags were employed to create a distinctive collection of organically occurring critical, sarcastic, and supportive tweets. [4]

In Sentiment analysis, machine learning is generally used to analyse text from polarity. Sentiment analysis programs are developed and trained extensively to detect complex nature

of human languages like the context of statements, figures of speech, hidden emotions like humour, anger, sarcasm, with reasonable accuracy. Deep learning can be used to extract non-linear features from natural language data. It is known for its use in image recognition, speech recognition, and geolocation systems. Humans have been using deep learning to overcome challenges in natural language processing since the 1970s. This article is a brief overview the various models for sentiment analysis in code- mixed language data sets using machine learning and deep learning approaches. Sentiment analysis is the science of analysing mappings from a set of texts to classify them into positive and negative sentiment.

II. RELATED WORK

(Ansari et al., 2018) throws light on using plain machine learning algorithms for sentiment analysis of texts English-Hindi and English-Marathi. The dataset used in this paper contains 1200 Hindi and 300 Marathi samples collected from social media including YouTube comments, tweets and chats. The proposed methodology includes 6 steps which are Language Identification, Sentiment Scores Tagging, Supervised Learning Methods and lastly the Output as Sentiments [6]. Among all the algorithms applied direct linear SVM has the maximum F1 score. (Swami et al., 2018) provides collection of English-Hindi mixed language tweets which contain both sarcastic and non-sarcastic tweets. It offers linguistic annotation at the token level and tweet level for the presence of sarcasm [7]. For training the algorithms, developing them and evaluating the performance of language identification and sarcasm detection on code-mixed data, this corpus can be used. (Sentamilselvan et al., 2021) shows 1) Support Vector Machine was used on the dataset because it has a high classification accuracy, particularly for the sarcasm detection problem. 2) Naïve Bayes is one of the classifiers that can be applied to detect sarcasm in spoken conversations. 3) Decision tree can also be used to classify sarcasm in spoken conversations in the SemEval 2018 dataset [8]. Support Vector Machine gave the most elevated precision for the dataset. On another dataset, random Forest algorithm calculation gave the best outcomes with the exactness of 76%.

(R. Srinivasan et al., 2021) throws light on class imbalance distribution in code-mixed Tamil-English data for sentiment analysis. Oversampling techniques namely Synthetic Minority Over-Sampling (SMOTE) and Adaptive Synthetic (ADASYN) is used to solve the class imbalance problem [9]. An enhanced spell-checking algorithm is used for sentence classification. Levenshtein distance metric is used to normalize the words with spelling variations. TF- IDF is used for feature extraction. Different machine learning algorithms that were experimented includes Random Forest Classifier, Logistic Regression, XGBoost classifier, SVM and Naive Bayes. For evaluation of the algorithm's macro average F1 score is used. According to this paper the performance of Logistic Regression is superior to other algorithms. (Aditya Bohra et al., 2018) addresses hate speech detection in code-mixed text. Using the Twitter Python API, tweets were analysed by choosing specific hashtags and key phrases related to politics, rioting, and other public demonstrations [10]. A total of 1,12,718 tweets were retrieved in JSON format. After pre-processing, a dataset of 4574 code-mixed Hindi-English tweets was created. Different feature vectors were used to train supervised machine learning model. These vectors included Character N-Grams, Word N-Grams, Punctuations, Negation Words and Lexicon. Support Vector Machine algorithms performs the best with an accuracy of 71.7%. (Akshita Aggarwal et al., 2020) has five different deep learning models. The corpus was scraped from twitter. The dataset included 100k Hindi- English code-mixed tweets with 49% of it being sarcastic and rest 51% non-sarcastic tweets [11]. Data pre-processing was performed in order to remove any sort of extra URLs, hashtags, mentions and other punctuations. Two types of word embeddings were used that is Word2Vec and FastText. The paper also presented a comparison of traditional Machine Learning algorithms for sarcasm detection. From which RBF Kernel SVM had the maximum accuracy of 71.23%. The paper concludes with attention based Bi-directional LSTM being the best performer with an accuracy of 78.49%. (Anbukkarasi S et al., 2020) Bi-LSTM algorithm is used for classification of text which processes the data both in forward and backward direction [12]. Along with this transformers-based models can be used to achieve more accurate results then LSTM and traditional machine learning algorithms such as SVM, random forest, etc. (Bedi et al.,2021) uses Attention based multi-modal classification model. This paper presents a unique dataset named MaSac that is extracted from the hindi series Sarabhai vs Sarabhai. MaSac is a qualitative multi-modal dataset for sarcasm detection and humor classification in code mixed conversations [13]. To extract speech signals Google Speech API- based automatic speech recognition tool is used. (Diwan et al., 2021) approximately 600 hrs of speech dataset were created for the research from different sources and different domains. The variabilities in each language were preserved. Robust Multilingual ASR was built by considering graphemes and phonemes in six different languages [14]. Code-switching ASR designed for 2 major pairs Hindi-English and Bengali English. The dataset generated is huge as compared to existing publicly available dataset in this category and can be further used for future research. (Ramandeep et al., 2021) talks about how sentiment analysis has wide range of applications in e-commerce, recommendation systems, review analysis, etc. Sarcasm detection, multipolarity, word ambiguity are the major issues in sentiment analysis [15]. The two major ways of sentiment analysis discussed in this paper includes machine- learning based and lexicon-based techniques.

III. PROPOSED MODEL

Code mixed Sentiment Analysis system is used to detect the sentiments in a code-mixed sentence. Code-mixed text, extracted from social media platforms, serves as an input followed by the data-pre-processing step [16]. Pre-processing steps involve data processing which involves a series of processes wherein the code-mixed data would be converted into individual tokens, the language would be identified, text would be normalized, transliteration would be carried out, abbreviations would be expanded and the emoticons would be replaced by their actual alternate text. Finally, stop words,

punctuations would be removed by lemmatization. The main processes involved in here are:

POS Tagging is a type of Named Entity Recognition, i.e., the process of identifying named entities or words (or phrases) in text. A variety of techniques can be used to perform POS tagging: rule-based methods that use partial/full matches between training/test data; grammar-based methods; statistical techniques like clustering and hidden Markov models; and methods that rely on statistical properties of certain knowledge bases such as dependency parser or ngrams.

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. Negation Handling is an automatic way of determining the scope of negation. 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 and make them negative. NLP Negation Handling is an automated way of determining the scope of negation. Examples of question words include: What? who, why? which? whose? The NLP has a negation handling mechanism which detects the negation automatically. It can spot the polarity of a sentence and identify the scope of negation in that sentence. Approach for sentiment detection: Various methods like Lexicon based methods using SentiWordNet and HindiSentiWordNet and also Machine Learning and Deep Learning methods using classifiers and Neural Networks are used individually as well as in combination to detect the sentiments present in the text.

Machine learning and deep learning methodologies for Sentiment Analysis:

In Sentiment analysis, machine learning is generally used to analyze text from polarity. Sentiment analysis programs are developed and trained extensively to detect complex nature of human languages like the context of statements, figures of speech, hidden emotions like humor, anger, sarcasm, with reasonable accuracy [17]. There have been different models proposed for sentiment examination in English language informational indexes utilizing data sets using Machine learning and Deep learning approaches.

Deep learning is a type of machine learning that allows computers and artificial intelligence to learn from large amounts of data. It is a broad discipline which covers many techniques, algorithms, and structures for neural networks. These models have been used in state-of-the-art technology such as natural language processing, computer vision, speech recognition, stock market prediction and medical diagnosis [18]. Deep learning models can be used for speech recognition and natural language processing applications, image segmentation, prediction and forecasting patterns and trends. Theoretical and practical research on reinforcement learning methods, multi-agent systems, nonlinear dynamics and machine learning is geared toward the development of theoretical knowledge that is useful in human intelligence. The goal is to increase the richness of this body of knowledge by leveraging its predictive power toward uncovering key insights into the nature of intelligence.

Recently, attention has been drawn to issues like sentiment analysis and sarcasm detection using machine learning and neural networks. Research has been done for monolingual dataset and certain code-mix dataset like Tamil-English, Bengali English and mainly the English dataset [19].

However, very few research has been done for sarcasm detection for Hi-En dataset. Our approach is to research on various algorithms, compare accuracy metrics and thus provide insight on the most suitable algorithm that can be implemented. It is aimed to do sarcasm detection on code-mix dataset of Hindi-English by using machine learning algorithms as well as convolutional neural networks and then compare the results.

The machine learning algorithms that will be implemented are Bernoulli Naïve Bayes, Logistic Regression and Support Vector Machines. The accuracy metrics that will be measured for analysis are F1 Score, Precision and Recall.

Deep learning techniques like Long-Short Term Memory and Recurrent Neural Networks will also be will be implemented in the future.

IV. DESİGN AND IMPLEMENTATİON

Dataset collection

The dataset consists of tweets and comments from Twitter with over 115000 samples. The dataset is in JSON format. It has 3 parameters which are article_link, headline and is_sarcastic. In the is_sarcastic parameter, 0 means that the statement is non-sarcastic and 1 means sarcastic.

Table 2 gives the overview of total number of tweets and count of sarcastic and non-sarcastic tweets in our training dataset.

Further the model assigns ‘Sarcastic’ class to 1 and ‘Not sarcastic to 0. When an input is provided from the testing dataset features of the input text are compared with the training dataset and accordingly output is produced.


```
{
  "article_link": 7,
  "is_sarcastic": 0,
  "headline": "for a sec i thought this was a rally in rawalpindi"
},
{
  "article_link": 8,
  "is_sarcastic": 1,
  "headline": "bhaai ko sab pata hain"
},
{
  "article_link": 9,
  "is_sarcastic": 0,
  "headline": "kiya dhoni k liye as a captan e tne safal rahe hain
  uska ek bahot bada karan rahe hain"
},
```


Fig. 1. Dataset

Table 2. Training Dataset

Category	Tweet Count
Total Tweets	115000
Sarcastic	57633
Non-Sarcastic	57367

Performance analysis of different algorithms

Without being specifically designed to do so, computers are now capable of learning new tasks through machine learning. Machine learning can be used to analyse text for polarity in sentiment analysis. The three models that are used in this paper are discussed below.

- Bernoulli Naïve Bayes

Bernoulli Naive Bayes is a probabilistic classifier. It assumes that each feature is independent of other features and can't influence their values. It uses decision tree algorithm to find the probability of a certain class, which correspond to different potential labels (example: label= "male", probability="0.75"). Naïve Bayes is a probabilistic classification algorithm that assumes that, given all available data, each feature can be said to exhibit independent or different behavior for the class.

The concept of Bernoulli Naïve Bayes follows the concept of probability called the Bernoulli distribution as shown below.

$$p(x) = P[X = x] = \begin{cases} q = 1 - p & x = 0 \\ p & x = 1 \end{cases} \quad (1)$$

One of the reasons for using Bernoulli Naïve Bayes is that it based on Bernoulli Distribution and accepts only binary values such as true or false, yes or no, 0 or 1 and so on. In this paper, sklearn is used to calculate the accuracy and f1 score of the model.

- Logistic Regression

Logistic regression is a type of multivariate statistical model that aims to predict a target variable based on several independent variables. The dependent variable in this case would be categorical.

Logistic regression is used to model and make predictions on the likelihood of an outcome, such as whether a person will take certain medication. This technique can be used in many different situations, making it an important machine learning tool. In this course you will learn everything from how to implement logistic regression and what works well in practice, through to advanced techniques for building robust predictive models. In Logistic Regression, an independent variable x is assigned a value between 0 and 1 for each single observation z that has been observed. The Logistic regression equation can be obtained from the Linear Regression equation as follows:

$$\log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \ldots + b_nx_n \quad (2)$$

- Support Vector Machine

Support Vector Machine (SVM) is a popular machine learning algorithm used for classification, regression, and outlier detection. In particular, SVM is widely used for classification tasks because of its ability to handle high-dimensional data and its effectiveness in separating data into different classes. SVM works by finding the best hyperplane (a line or a plane in high-dimensional space) that separates the data points into different classes. The hyperplane is chosen such that it maximizes the margin, which is the distance between the hyperplane and the closest data points from each class. In order to achieve this result, an inter-class hyperplane is generated. The algorithm selects a hyperplane at random, gets its separating vectors, and then finds the optimal parameters to maximize their expected separation. This algorithm works by first partitioning data into a training set, and then using some other method (such as linear or polynomial regression) to generate a classification model for each class. For example, if you're given 10 features (or features for each training set) that are numeric values and you want to build a classifier that predicts whether a new piece of data belongs to training set 2 or not, then your first step would be to create two feature vectors for each training set that have squared differences of zero. While the SVM algorithm solves linear classification problems, it has proved effective in a wide range of applications. These include subspace selection, clustering and retrieval, language modelling, text categorization and many more. The new data points are then classified based on which side of the boundary they fall. In the case of sarcasm detection, SVM can be used to classify a given text as sarcastic or not by training the model on a labelled dataset of text examples. In this paper, sklearn linear SVM library is used for implementing our SVM based models.

V. RESULT AND ANALYSİS

In this paper three different approaches to split the data into training and testing sets. The first approach includes 80- 20 split with 80% of data used for training and 20% used for testing. The reason for splitting the dataset into 80:20 ratio is that it provides a good balance between having enough data for the model to learn from, while still reserving enough data for evaluating the model's performance.

The other two splitting approaches have been summarized in Table 3 below. The accuracy of a model is defined as the ratio of the number of correct predictions to the total number of predictions.

Accuracy = (#***Correct Predictions***) / (#***Total Predictions***) (3)

Table 3: Accuracy Table

Train-Test Split	Accuracy Obtained		
	Bernoulli Naïve Bayes	Logistic Regression	Support Vector Machine
Training Set: 80% Testing Set: 20%	0.79	0.86	0.87
Training Set: 60% Testing Set: 40%	0.79	0.86	0.87
Training Set: 50% Testing Set: 50%	0.78	0.86	0.87

It can be inferred from the Table 3 that all the three algorithms have very close performance results. The performance decreases when the train-test split is made 50-50. Support Vector Machine gives the best result at an 80-20 split.

The F1 Score is used to evaluate the performance of different algorithms from which it is that SVM performs the best out of the three models.

$$F1\ Score = 2(True\ Positives) / (2(True\ Positives) + False\ Positive + False\ Negative) \quad (4)$$

VI. CONCLUSİON

The number of people using social media to express their views has exponentially increased over the past decade and tasks like opinion mining and sentiment analysis have gained greater attention and importance. Sarcasm over social media makes performing these tasks yet another challenge. In our project, we have used a Hindi-English code-mixed dataset for sarcasm detection. This paper has presented a baseline supervised classifier using three different machine learning techniques that is developed using the same dataset. This study deals with the problem of sarcasm detection on social media, which faces the additional challenge of expressing their views in Hindi-English code-mixed dataset with high accuracy and robustness. It is presented that a supervised classifier using three machine learning techniques that is developed using the same dataset and compared them by giving advantages and disadvantages of each. For both logistic regression and support vector machine, it is found that support vector machine gives highest accuracy followed by logistic regression whereas Bernoulli Naïve Bayes gives lesser accuracy than base model classifier. Social media used for expressing views and messages has become a common practice nowadays. Majorly sarcasm is one of the most challenging tasks when it comes to analyzing social media interactions. The use of machine learning techniques like decision trees or SVM based classifiers is growing in popularity over the past decade as they are capable of capturing more complexity and information than hand crafted classifiers ever could. A code-mixed dataset consisting of Hindi-English text pairs is used, both positive and negative examples, to develop a baseline supervised classifier using three different machine learning methods - Support Vector Machine (SVM), Logistic Regression (LogiReg) and Bernoulli Naïve Bayes (BNA). It is further found that Support Vector Machine gave highest accuracy followed by Logistic Regression whereas Bernoulli Naïve Bayes gave the least accuracy because BNA does not consider conditional probability and often underestimates the performance level that can be achieved with other techniques.

REFERENCES

[1] Patra, Braja & Das, Dipankar & Das, Amitava. (2018). Sentiment Analysis of Code-Mixed Indian Languages: An Overview of SAIL_Code- Mixed Shared Task @ICON-2017.

[2] Ming Zhou, Nan Duan, Shujie Liu, Heung-Yeung Shum, Progress in Neural NLP: Modelling, Learning, and Reasoning, Engineering, Volume 6, Issue 3, 2020, Pages275-290, ISSN2095-099, https://doi.org/10.1016/j.eng.2019.12.014. (https://www.sciencedirect.com/science/article/pii/S2095809919304928).

[3] Pradhan, R., Sharma, D.K. An ensemble deep learning classifier for sentiment analysis on code-mix Hindi–English data. Soft Comput (2022). https://doi.org/10.1007/s00500-022-07091-y.

[4] Kumar, Rajesh & Singh, Pardeep. (2017). Bilingual Code-Mixing in Indian Social Media Texts for Hindi and English. 10.1007/978-981-10- 5780-9_11.

[5] Sarsam, Samer & Al-Samarraie, Hosam & Alzahrani, Ahmed & Wright, Bianca. (2020). Sarcasm detection using machine learning algorithms in Twitter: A systematic review. International Journal of Market Research. 62. 10.1177/1470785320921779.

[6] Ansari, Mohammed Arshad, and Sharvari Govilkar. "Sentiment analysis of mixed code for the transliterated hindi and marathi texts." International Journal on Natural Language Computing (IJNLC) Vol 7 (2018).

[7] Swami, Sahil, Ankush Khandelwal, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. "A corpus of english-hindi code-mixed tweets for sarcasm detection.", arXiv preprint arXiv:1805.11869 (2018).

[8] Sentamilselvan, K., Suresh, P., Kamalam, G. K., Mahendran, S., & Aneri, D. (2021, February). "Detection on sarcasm using machine learning classifiers and rule-based approach." IOP Conference Series: Materials Science and Engineering (Vol. 1055, No. 1, p. 012105). IOP Publishing.

[9] R. Srinivasan C. N. Subalalitha, "Sentimental analysis from imbalanced code-mixed data using machine learning approaches", Springer, 20 March 2021

[10] Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed S. Akhtar, Manish Shrivastava, "A Dataset of Hindi-English Code-Mixed Social Media Text for Hate Speech Detection", NAACL HLT 2018, New Orleans, Louisiana, June 6, 2018

[11] Akshita Aggarwal, Anshul Wadhawan, Anshima Chaudhary and Kavita Maurya, "Did you really mean what you said: Sarcasm Detection in Hindi-English Code-Mixed Data using Bilingual Word Embeddings", 2020 EMNLP Workshop W-NUT.

[12] Anbukkarasi S, Varadhaganapathy S, SA-SVG@Dravidian-CodeMix-FIRE2020: Deep Learning Based Sentiment Analysis in Code-mixed Tamil-English Text, CEUR Workshop Proceedings, December 16-20, 2020

[13] Bedi, Manjot, et al. "Multi-modal sarcasm detection and humor classification in code-mixed conversations." IEEE Transactions on Affective Computing (2021).

[14] Diwan, Anuj, et al. "Multilingual and code-switching ASR challenges for low resource Indian languages." arXiv preprint arXiv:2104.00235 (2021).

[15] Ramandeep Kour, Gurdeep Singh Josan, "Sentiment Analysis of CodeMixed Text: A survey", IJCST Vol. 12, Issue 2, April - June 2021.

[16] Thara, S., Poornachandran, P. Social media text analytics of Malayalam–English code-mixed using deep learning. J Big Data 9, 45 (2022). https://doi.org/10.1186/s40537-022-00594-3.

[17] Ahmad, Gazi Imtiyaz, et al. "Machine Learning Techniques for Sentiment Analysis of Code-Mixed and Switched Indian Social Media Text Corpus-A Comprehensive Review." International Journal of Advanced Computer Science and Applications 13.2 (2022).

[18] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions.

[19] J Big Data 8, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8

[20] Proceedings of the 2020 EMNLP Workshop W-NUT: The Sixth Workshop on Noisy User-generated Text, pages 7–15 Online, Nov 19, 2020. 2020 Association for Computational Linguistics.