# Sentiment Analysis of Codemixed Text: A Survey

1Ramandeep Kour, 2Gurpreet Singh Josan
1,2Dept. of Computer Science, Punjabi University, Patiala, Punjab, India

## ABSTRACT
In recent times, sentiment analysis has grown in popularity as one of the most active research fields in information retrieval and text mining. Sentiment analysis is essential in many real-world applications such as e-commerce, review analysis, recommendation systems, analysis of current trends, political campaigns, etc. Sentiment analysis becomes more difficult in the situation when data is noisy and collected from social media. India is a multilingual country of multilingual people; those are non-native English speakers who communicate in multiple languages. The switching from one language to another language is called code-switching or code-mixing, which depends on the type of mixing. In a multilingual society, such content is often a composition of different languages. This phenomenon of mixing the vocabulary and syntax of two or more languages (code-mixing) makes the processing of such content significantly harder. This paper presents the survey on sentiment analysis on code mixed data. We have also discussed applications of and challenges Sentiment Analysis on code mixed text. Data acquisition, data preprocessing and Tokenization are some of the steps involved in the methodology.

**KEYWORDS**-*Sentiment Analysis, Machine Learning, Lexicon-based approach, Codemixed.*

## I. INTRODUCTION
**(A.)Sentiment analysis** is a part of natural language processing that can be used for a variety of tasks such as analyzing movie ratings, user modeling, curating web trends, and determining the tone of a text, voice, or opinion. There are several names for sentiment analysis, opinion extraction, opinion mining, sentiment mining, subjectivity analysis, and emotion analysis, review mining, all of which perform slightly different tasks **[19]**. Balahur & Turchi, 2014 **[3]** proposed the two main approaches in Sentiment Analysis first is subjectivity analysis and the second is sentiment classification. Subjectivity analysis is concerned with detecting opinions or sentiments, whereas sentiment classification is concerned with categorizing those opinions into different polarities or rankings. Balahur & Turchi, 2014 **[25]** concentrated on categorizing text as positive, negative, or neutral. Others classify data at different levels of granularity, such as highly positive, positive, neutral, negative, or highly negative (S. Bhattacharjee, et al., 2015). Our goal is to identify the text and then establish the label (or labels) that describe the sentiment of the text, such as positive, neutral, or negative. Consider the following scenario: *"I like the new pen!"* → Positive, *"I'm not sure if I like the new pen"* → Neutral and *"The don like the design of new pen!"* → Negative. Sentiment analysis, also known as opinion mining, identifies, extracts, and evaluates subjective data that defines a product, organization, service, topic, or person's positive, negative, or neutral opinions, issues, reviews, feelings, appraisals, or attitude. Sentiment Analysis is the domain of using software to comprehend feelings, and it's a must-know for developers and business leaders in the modern workplace. Sentiment analysis can be done mainly at three levels: document level, sentence level, entity, or aspect level. Moralwar & Deshmukh, 2015 **[24]** presents the different levels of opinion analysis i.e. document level, sentence level, feature level, word level and phrase level. The data source for review collection and approaches for sentiment classification. Most work has been done on product reviews downloaded from Amazon

**Document-level:** Document-level identifies whether a whole opinion document summarizes a positive or negative sentiment, (Pang, Lee, and Vaithyanathan, 2002; Turney, 2002). For example, given the review of a product, the system describes whether the review is a positive or negative opinion about the product. This task is known as *document-level sentiment classification*. The analysis of the level assumes that each document defines the opinion on a single entity (e.g., a single product).

**Sentence level:** This level work on a sentence, which describes whether each sentence summarizes a positive, negative, or neutral opinion. Neutral means no opinion. This is closely related to *subjectivity classification* (Wiebe, Bruce, and O'Hara, 1999), which distinguishes sentences (called *objective sentences*) and defines the factual information about sentences (called *subjective sentences*) that summarize the subjective views and opinions. There is we note that thing the subjectivity is not equivalent to objective sentences that can imply opinions, e.g., "*We bought the car last month and the windshield wiper has fallen off.*"

**Entity and Aspect level:** This level performs the finer-grained analysis. Earlier this level was called *feature level* (*feature-based opinion mining and summarization*) (Hu and Liu, 2004) **[13]**. Instead of looking at documents, paragraphs, sentences, clauses, or phrases, the aspect level directly defines the opinion itself. Instead of looking at documents, paragraphs, sentences, clauses, or phrases, the aspect level directly defines the opinion itself. It describes that the opinion expresses the *sentiment* (positive or negative) and a *target* (of opinion). For example, the sentence "*although the services are not good, but I still love this hotel*" expresses a positive tone, but we cannot say this is an entirely positive sentence. The above sentence is positive for the *hotel* (emphasized), but negative for its *services* (not emphasized).

**(B).Code-Mixed:** India is a multilingual country with multilingual people, who are non-native English speakers who communicate in more than one language. In linguistics, code-switching refers to the use of more than one language or variety in a communication **[33]**. Code-switching is a method of combining two or more languages in a single discussion. According to Hymes (1974), Code-switching is "a frequent phrase for the alternate use of two or more languages, variants of a language, or even speech styles." The process of switching between languages is referred to as code-switching or code-mixing, depending on the method of mixing. Within the same speech event, code-switching is defined as the mixing of words, phrases, and sentences from two different grammatical systems across sentence borders. (Bhatt, 1997) defines the alternate use of two linguistic systems inside a clause. According to (Hamers

and Blanc, 2000), 'Code-mixing' and 'code-switching were considered as signs of incompetence. Kim, 2006 **[15]** Presents that bilingual phenomena are ordinary in the area of bilingualism. When data is noisy and gathered from social media, sentiment analysis becomes more complicated. However, (Khnert, Yim, Nett, Kan, and Duran, 2005) remark that an alternative view is to recognize the cultural, social, and communicative validity of the mixing of two traditionally isolated linguistic codes as a third legitimate code. These phenomena may influence bilingual's language positively. The purpose is to indicate the positive factors of code-mixing and code-switching for language education by discussing societal factors related to the reasons and motivations for these phenomena. The majority of previous research in this field has concentrated on a single language, especially English. However, as the world becomes more globalized and the number of people using the Internet grows, it is becoming more popular to see posts written in several languages, making the Sentiment Analysis method much more difficult and challenging. Furthermore, people prefer to combine languages in one sentence in unstructured content like Twitter messages. In reality, the trend of using multiple languages in a single sentence has emerged, and such mixed language has rarely been a topic of SA in the past. It is important to provide a different method or methodology to cater to this type of data since some details in another language may be missed if the research is performed only for that language. The use of code-mixed language in everyday conversations stems from the fact that certain multilingual speakers prefer to communicate in their native language rather than English.

**(C).Sentiment Analysis of Code Mixed languages:** Sentiment analysis is useful in many real-world applications such as review analysis, recommendation systems, and so on. Sentiment analysis becomes more difficult in the situation when data is noisy and collected from social media. India is a multilingual country and multilingual people; those are non-local English speakers who use more than one language to communicate with each other. The switching between the languages is called code-switching or code-mixing, which depends on the type of mixing. **[29]**.In the mixing languages to find the code-mixing behavior at the word level is fairly common than the sentence level. Here we mainly focus on the combination of English with Hindi (Hinglish), which is the 4th most spoken language in the world. In the Hinglish model, just a single English word appreciation has been utilized, however more noticeably for the Hindi words - rather than using the Devanagari content, English phonetic composing is a well-known practice in India.

For example: "U saw caste and religion in them… nation saw talent and trusts in them!! Problem is tum kisi par ykeen mat karna!! " In the above example, some words are from the English language and some words are from the Hindi language however they all are written in English. The social media texts like blogs, micro-blogs (e.g. Twitter), and chats (e.g. WhatsApp and Facebook messages) have made numerous new opportunities for information access and language technology, but it has additionally presented many new challenges making it one of the current prime research areas. Although current language technologies are essentially worked for English, when non-native English speakers use social media they combine English with other languages. They type half of the messages in English and half in other than English on Twitter. Code-mixing represents several unseen difficulties to NLP tasks like word-level language identification, part-of-speech tagging, dependency parsing, machine translation, and semantic processing. (Choudhary et al.,) **[9]** Proposed novel methodology called Sentiment Analysis of Code-Mixed Text (SACMT) to order sentences into their corresponding sentiment - positive, negative, or neutral, using contrastive learning. (Patra et al., 2018) **[27]** presents the sentiment identification task from Hindi-English (HI-EN) and Bengali-English (BN-EN) code-mixed datasets. For data collection of different languages from Twitter API is used.

## II. Applications of Sentiment Analysis

(Liu, 2012) **[19]** Focused on Sentiment Analysis applications and challenges. Nowadays, the natural language processing community shows more interest in the Sentiment Analysis system. The lifestyle of people has changed by using the internet, now they are more express their reviews and opinions on any product, and this tendency helped the researchers in getting user-generated content easily.

The major applications of sentiment analysis are the following:

**A. Purchasing Product or Service**

While buying an item or service, taking the right choice/decision is never a difficult task. By this method, people can evaluate the other's customer's opinion and experience about the product or service and they help to compare the competing brands. Now people don't want to believe in external consultants. The sentiment analysis extracts people opinion from the large collection of unstructured content on the internet and then analyze it and after this present to them in a highly structured and understandable manner.

**B. Quality Improvement in Product or Service**

With the help of sentiment analysis, the manufacturers collect the opinion as well as the favorable opinion about their product or service because they can improve the quality of their product or service.

**C. Marketing Research**

Sentiment analysis techniques can be used in marketing research. The sentiment analysis techniques help to analyze the recent trend of customers about products or services. SA also helps to analyze the recent attitude of peoples towards some new government policy. These all results can be contributed to collective intelligent research.

**D. Recommendation Systems**

By classifying the people's sentiments into positive and negative, the system can say which opinions get recommended and which ones should not get recommended.

**E. Opinion Spam Detection**

Since the internet is available to all, so anybody can put anything on the web, this expanded the chance of spam content on the internet. People may write spam content to mislead individuals. Sentiment analysis can help to classify the internet content into' spam' content and 'not spam' content.

**F. Policy Making**

With the help of opinion mining, policy makers can take an individual's point of view about some policy and they can use this information in creating new policies.

**G. Decision Making**

People's sentiments and experiences also help in the decision-making process. Sentiment analysis gives the people's opinion that can be effectively utilized for decision making.

## III. Challenges

Sentiment Analysis is a very challenging task **[38]**. Following are some of the challenges faced in Codemixed Sentiment Analysis-

### A. Sarcasm Detection

Sarcasm can occur in user-generated text like Facebook comments, tweets, etc. In the sentiment analysis very difficult to detect sarcasm without having a good understanding of the particular topic, the specific environment, and the situation. In the sarcastic text, people using positive words to express the negative sentiment, because using these facts sarcasm is easily cheating with the sentiment model. Very difficult to successfully train the sentiment analysis models when the continuous variation occurs in the word that is used in the sarcastic sentence.

Example of Sarcasm:

| Situation | Sarcastic Remark |
|---|---|
| In the situation, when something bad happens: | Aaj mujhe exactly yhii chahiye tha! |
| At the point when you expected that something should occur, especially after warning somebody about it: | Well, kya surprise hai |

### B. Negation Detection

The polarity of phrases, words, and even sentences is reversed in the negation. Researchers use various linguistic rules to find whether negation is occurring or not, but it's also important to find the range of the words that are affected by the negation words. In the negated sentence has no fixed size for the scope of affected words. For example, in a sentence like "The show was not interesting," the scope is just the following word after the negation word. But for another sentence like "I don't consider this film a comedy movie," the impact of the negation word "not" is until the end of this sentence. If a positive or negative word falls inside the scope of negation, it can change the original meaning of the word in this case the opposite polarity will be returned. To deal with negation sentences the simplest approach is to use the most state-of-the-art opinion analysis techniques is checking as negated all the words from a negation cue to the next punctuation token.

### C. Word ambiguity

Word ambiguity is another challenge that is faced when we are working on a sentiment analysis problem. In the problem of word ambiguity, there is impossible to define the polarity in advance because the polarity for some words is strongly dependent on the full sentence context. In some existing methods, the lexicon-based sentiment analysis approaches are very popular. The sentiment lexicon defines the sentiment words with their polarity values and some public sentiment lexicons are also available on the internet like SentiWordNet, General Inquirer, and SenticNet, among others. It is impossible to develop a universal sentiment lexicon that defines the polarity for every word because the word polarity varies in different areas.

For example:

1. "The story is unpredictable."
2. "The steering wheel is unpredictable."

The above two examples show how context affects opinion word sentiment. The word polarity of "unpredictable" is predicted to be positive in the first scenario. The polarity of the identical word is negative in the second.

### D. Multipolarity

Sometimes, we analyze the given document, sentence, or unit of text to express the multipolarity. In these cases, the total result of the analysis can be misleading and sometimes hide valuable information about all the numbers. In the picture when authors talk about the different things, people, products, or companies in the review or article. Because of this, some subjects will be criticized and some praised. Here, the total opinion polarity will be missing about the key information. Because of missing the key information, it is necessary to extract all the entities or words in the sentence with assigned opinion labels and if we needed then calculated the polarity.

### E. Non-Grammatical Constructs

This non-grammatical type construct doesn't follow English grammar or neither follows any other particular Hindi grammar. For example: *'Bhai jaan why you don't ask anything…… yaar…. Bhai jaan bolo na'.* As a result, these types of difficulties were encountered in the Codemixed text.

### F. Phonetic Similarity of various words in the participant languages

This is the most challenging problem in Codemixed Sentiment Analysis. For example: *'man'* meaning *'an adult human male'* in English and same as *'man'* meaning *'mind'* in Hindi. So in the same sentence "man" can have different meanings in different languages.

## IV. Methodology

For the Sentiment Analysis task following steps are involved such as:

### A. Data Acquisition

The process of collecting and arranging data for analysis is known as data acquisition. The procedure of collecting, measuring, and evaluating correct insights for research using established approved procedures is referred to as data collection. Based on acquired data, a researcher might evaluate their hypothesis. Regardless of the subject of research, data collecting is typically the first and most significant phase. Depending on the information needed, the methodology of data gathering varies by topic of study.

### B. Data Preprocessing

The data is collected from different sources that are in raw format, which is not feasible for analysis. So data preprocessing is a method that is used to convert the raw data into a clean dataset. Need of Data Preprocessing: The data must be formatted properly to achieve better outcomes from the used model in Machine Learning applications. Some Machine Learning models require data in a specific format; for example, the Random Forest technique does not tolerate null values, therefore null values must be handled from the original raw data set to run the algorithm.

### C. Tokenization

We cannot fill raw text directly into deep learning models. For this, we use a tokenizer, splitting a text into a list of tokens is

known as tokenization. A sentence is a token in a paragraph, while a word is a token in a sentence. For instance, the word "This is a pen" can be tokenized as "This," "is," "a," "pen." Tokenization can be done using a variety of methods and libraries.

### D. Analyzed Output

The reviews are the input and tokens are the output of the data preprocessing phase. The token is used as input to various algorithms, which produce a rating as an output. These ratings can be represented in a variety of ways, including pie charts, bar graphs, and other graphs.

## V. Sentiment Classification Techniques

Sentiment analysis can be performed using three methods: machine learning and lexicon-based and hybrid-based approaches.

machine learning techniques. (Dang et al., 2020)such as Twitter or Facebook, has become a powerful means of learning about the users' opinions and has a wide range of applications. However, the efficiency and accuracy of sentiment analysis is being hindered by the challenges encountered in natural language processing (NLP **[10]** Introduced the sentiment analysis can be divided into two groups: (1) traditional models and (2) deep learning models. (Malik & Kumar, 2018) **[21]** Introduced the nave Bayes classifier, maximum entropy classifier, and support vector machines are examples of traditional machine learning approaches. (Long et al., 2018)and which one cannot. The answer will not only help understand the mechanism of hydroxylation but can also benefit the development of new drugs. In this paper, we proposed a novel approach for predicting hydroxylation using a hybrid deep learning model integrating the convolutional neural network (CNN [20] Introduced CNN, DNN, and RNN are some of the deep learning models that can be utilized for

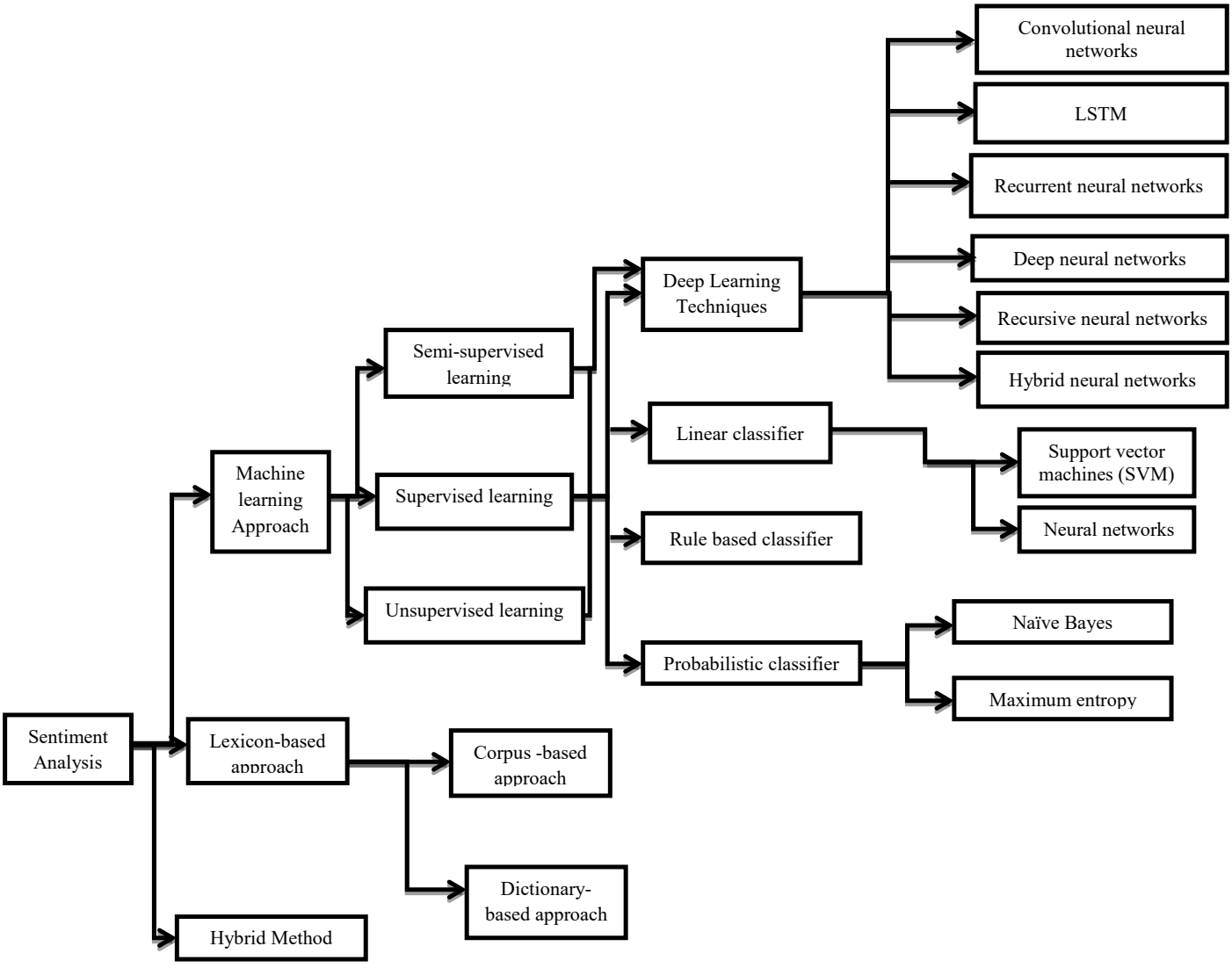


Figure 1: Sentiment Analysis techniques **[10]**.

**Machine learning-based techniques:** Machine learning is classified as supervised learning, semi-supervised learning, and unsupervised learning, and it uses machine learning algorithms. The expected output must be compared to the real output in supervised learning. On the other hand, unsupervised learning uses prior knowledge and data to increase its accuracy it does not require any desired output. Semi-supervised machine learning combines the advantages of both supervised and unsupervised sentiment analysis.

**Lexicon-based techniques:** (Salas-Zárate et al., 2017) **[29]** introduced the lexicon-based approach splits the entire text into lexemes and processes them. Corpus-based and dictionary-based approaches are two types of lexicon-based approaches. The polarity of the sentence is determined using a corpus-based method as negative, positive, or neutral. K-nearest neighbors (k-NN), conditional random field (CRF), and hidden Markov models (HMM) are some of the techniques used in corpus-

based sentiment analysis. The dictionary-based approach is a mathematical method of determining how sentences affect the reader's feelings. A dictionary of terms, such as those contained in SentiWordNet and WordNet, is used to classify sentiment. The **hybrid approaches** combine lexicon-based and machine-learning-based approaches **[30].**

## VI. Related Work

**Machine learning-based**

In this survey paper various machine learning techniques are described such as Naive Bayes (NB), Support Vector Machines (SVM), Maximum Entropy (MaxEnt), BERT, etc. Two SVM-based classifiers were introduced by (Mohammad et al., 2013) **[23]** to detect sentiments in tweets and text messages, using a variety of surface-form and semantic features.(Tang et al., 2015) **[35]** introduced deep learning-based techniques for a variety of sentiment analyses, including word embedding, sentiment classification, and opinion extraction.(Ansari et al., 2017) **[2]** Attempted to conduct sentiment analysis on "tweets" using various machine learning algorithms. They use various machine learning algorithms to conduct sentiment analysis using the extracted features. Before training preprocesses the tweets to make them suitable for feeding into models. They implemented several machine learning algorithms like Naive Bayes, Maximum Entropy, Decision Tree, Random Forest, XGBoost, SVM, Multi-Layer Perceptron, Recurrent Neural networks, and Convolutional Neural Networks to classify the polarity of the tweet. **(Mandal & Das, 2018) [22]** The author analyzed the results of various experiments carried out on a movie reviews dataset having this code-mixing property for sentiment identification of two languages, English and Bengali, both typed in Roman script. They tested various machine learning algorithms trained on code-mixed data and have achieved the maximum accuracy of 59.00% using the Naïve Bayes (NB) model and 72.50% using the Support Vector Machine (SVM) model. They conclude that when the train and test data are of a similar type, SVM performs well, whereas NB performs better when the train and test data are of a slightly different type. **(Lal et al., 2019) [17]** Introduce hybrid architecture for the task of sentiment analysis of English-Hindi code-mixed data. They use two different Bidirectional Long Short Term Memory (BiLSTM) Networks, one that looks at the overall sentiment of the sentence and the other utilizes an attention mechanism to focus on the individual sentiment-bearing sub-words. This combined with a Feature Network consisting of orthographic features and monolingual trained word embeddings achieve the state-of-the-art results - 83.54% accuracy and 0.827 F1 scores on a benchmark dataset. **(Baruah et al., 2020)we present the results that the team IIITG-ADBU (codalab username {`}abaruah{'} [6]** The author talked about the detection of sentiment in code-mixed Hindi-English tweets. For this objective, they conducted two sets of tests. The multilingual BERT classifier was used in the first experiment, and SVM classifiers were used in the second set of tests. Character n-gram features were used to train the SVM classifier using cleaned data. The best F1 score was 0.678 for the character-based SVM classifier and the multilingual BERT classifier was obtained an F1 score of 0.342 in the test set.**(Sultan et al., 2020) [32]** They proposed a Transfer Learning-based model that fine-tunes XLM-RoBERTa, a transformer-based multilingual masked language model. They used two TF-IDF feature vectors as an input for the entire text; the first is a word-level TF-IDF vectorizer, while the second is a character-level TF-IDF vectorizer. Combine the outputs of two vectorizer into a single vector that will be used as the model's input. They present the results of studies conducted on Spanish-English (Spanglish) data. The system achieves a 75.9% average F1-Score on the test set.**(Kumar et al., 2020) [16]** Presented the architecture of the convolutional neural net (CNN) that helps in the classification of positive and negative tweets. Using the XLM-R embeddings, they first trained a CNN model on the provided dataset. On neutral data points, the CNN model appeared to be confused, but it worked well on positive and negative tweets. The self-attention-based LSTM, which takes as input the output of the XLM-R encoder that, helps in the classification of neutral tweets. For sentiment analysis of code-mixed tweets, they used Hindi-English (Hinglish) and Spanish-English (Spanglish) datasets and achieved F1 scores of 0.707 on Hindi-English and 0.725 on the Spanish-English dataset.**(Advani et al., 2020)** Introduced the feature engineering approach to sentiment analysis for the Hindi-English and Spanish-English code-mixed text and extracted different hand-engineered features. They created a classifier with a collection of hand-engineered lexical, sentiment, and metadata features that can distinguish between "positive," "negative," and "neutral" sentiment. The term frequency-inverse document frequency (TF-IDF) scores were used to weighted n-gram features and built a Logistic Regression classification model using Scikit-learn library. They achieved a weighted F1 score of 0:65 for the "Hinglish" task and 0:63 for the "Spanglish" tasks.**(Ver & Soares, 2020) [36]** Developed an approach based on a set of four models MultiFiT, BERT, ALBERT, and XLNET. After retrieving prediction values from these models, the final classification algorithm ensemble calculates an average of all softmax values from these four models. They predict the sentiment of Hindi-English code-mixed of a tweet and the system got 72.7% on the F1 score.**(Braaksma et al., 2020) [8]** Proposed a custom model that fine-tunes in two steps: once with a language modeling objective, and with a task-specific objective. They used English BERT-base and Spanish BERT-base monolingual models with the same 'base' architecture for the Spanish-English (Spanglish) datasets. The performance of sentiment classification is improved by a two-step fine-tuning method and the large multilingual XLM-RoBERTa model achieves the best-weighted F1-score with 0.537 on development data and 0.739 on test data.**(Parikh et al., 2020) [26]** Introduced an ensemble of Logistic Regression, Random Forest, and BERT. In this experiment, they followed two approaches, the first involved feature extraction using TF-IDF and using Logistic Regression or Random Forests to classify tweets as positive, negative or neutral. The second included BERT-based transfer learning. The statistical-based feature extracting using TD-IDF was able to better predict the "Neutral" class than BERT, for "Positive" and "Negative" class BERT gave better predictions and achieved an F1-score of 0.693 on the Hindi-English Codemixed Tweet dataset.**(Garain et al., 2020) [11]** The author used feature extraction algorithms in combination with traditional machine learning algorithms like SVR and Grid Search to classify the Hindi-English code-mixed sentences to their respective sentiment class. They convert the given tweets into a sequence of words and then run the Grid Search Cross-Validation algorithm on the processed tweet. They achieved an f1-score of 66.2% on English-Hindi code-mixed sentences**(Banerjee et al., 2020) [4]** Proposed a Recurrent Convolutional Neural Network for code-mixed sentiment analysis that combines both the recurrent neural network (RNN) and the convolutional network (CNN) deep neural networks to better capture the semantics of the text. They used a Hindi-English dataset and the system obtained 0.69 F1 scores on the given test data.**(Singh & Lefever, 2020) [31]** Introduced two

| References | Focus Area | Technique | Dataset (Language Pairs) | F1-Score |
|---|---|---|---|---|
| **(Baruah et al., 2020)we present the results that the team IIITG-ADBU (codalab username {`} abaruah{'}** | Sentiment Classification | BERT classifier, SVM classifiers | Hindi-English | BERT - 0.678<br>SVM – 0.342 |
| **(Zhu et al., 2020)English-Spanish** | Sentiment Classification | Multinomial Naive Bayes and Sub-word LSTM | Hindi-English and English-Spanish | Hinglish= 0.647<br>Spanglish= 0.682 |
| **(Bao et al., 2020)especially in multilingual societies like India. Detecting the emotions contained in these languages, which is of great significance to the development of society and political trends. In this paper, we propose an ensemble of pesudo-label based Bert model and TFIDF based SGDClassifier model to identify the sentiments of Hindi-English (Hi-En** | Sentiment Classification | BERT model and TFIDF based SGD Classifier model | Hindi English | 0.686 |
| **(Sultan et al., 2020)** | Sentiment Classification | Transfer Learning-based XLM-RoBERTa Model | Spanish-English | 0.759 |
| **(Kumar et al., 2020)** | Sentiment Classification | CNN and self-attention-based LSTM Model | Hindi-English and Spanish-English | Hinglish= 0.707<br>Spanglish= 0.725 |
| **(Advani et al., 2020)** | Sentiment Classification | Feature engineering approach | Hindi-English and Spanish-English | Hinglish = 0.65<br>Spanglish= 0.63 |
| **(Ver & Soares, 2020)** | Sentiment Classification | MultiFiT, BERT, ALBERT, and XLNET | Hindi-English | 72.7% |
| **(Braaksma et al., 2020)** | Sentiment Classification | XLM-RoBERTa model | Spanish-English | 0.739 |
| **(Parikh et al., 2020)** | Sentiment Classification | Ensemble of Logistic Regression, Random Forest, and BERT | Hindi-English | 0.693 |
| **(Javdan et al., 2020)** | Sentiment Classification | NBSVM | Hindi-English and Spanish-English | Hinglish= 0.706<br>Spanglish=0.751 |
| **(Garain et al., 2020)** | Sentiment Classification | Feature extraction algorithms with SVR and Grid Search | English-Hindi | 66.2% |
| **(Banerjee et al., 2020)** | Sentiment Classification | RNN and CNN | Hindi-English | 0.69 |
| **(Singh & Lefever, 2020)** | Sentiment Classification | Uses pre-trained English embedding's | Hindi-English | 70.52%. |
| **(Wang, 2020)** | Sentiment Classification | BERT | Hindi-English | 0.730 |

| | | | | |
|---|---|---|---|---|
| **(Gopalan et al., 2020)** | Sentiment Classification | BERT models and feedforward neural networks | Hindi-English | 71.3% |
| **(Bear & Constantina, 2020)** | Sentiment Classification | TueMix - trained with : TF-IDF n-grams | Hindi-English | 0.685 |
| **(Zaharia et al., 2020)** | Sentiment Classification | Multilingual BERT, XLM-RoBERTa | Hindi-English and Spanish-English | Hinglish= 0.6850<br>Spanglish= 0.7064 |
| **(Leon, 2020)** | Sentiment Classification | Word-embeddings trained on code-switched tweets | Spanish and English | 0.722 |
| **(Lal et al., 2019)** | Sentiment Classification | BiLSTM | English- Hindi | 0.827 |
| **(Mandal & Das, 2018)** | Sentiment Classification | Naïve Bayes (NB) model and SVM model | English and Bengali | NB =59.00%<br>SVM = 72.50% |

approaches for sentiment analysis of the Hindi-English code mixed dataset. The first technique uses cross-lingual embedding's that are created by combining Hindi-English and pre-trained English Fast Text word embedding's in the same space. The second method uses pre-trained English embedding's that are incrementally retrained using Hindi-English tweets. On the held-out test data, the results show that the second approach performs better, with an F1-score of 70.52%. **(Wang, 2020) [37]** Proposed the Bidirectional Encoder Representation from Transformers (BERT) to perform the task of sentiment analysis of code-mixed tweets. They preprocess the data first, then use the Fine-tune approach to reach a specific network after the BERT model Layer and reduce network overfitting with Multi-task and Adam algorithm. The model achieves an averaged 0.730 F1 score on Hindi-English Codemixed tweets. **(Bear & Constantina, 2020) [7]** Introduced a logistic regression algorithm-TueMix trained with three feature components: TF-IDF n-grams, monolingual sentiment lexicons, and surface features - with a neural network approach. TueMix outperformed neural network systems and the addition of the linguistic features beyond the TF-IDF n-grams are enhancing their model. They create a sparse feature matrix of word and character n-grams using sklearn's TF-IDF vectorizer. Result in a weighted F1-score of 0.685 on the Hindi-English Codemixed dataset.**(Zaharia et al., 2020) [39]** Introduced the multilingual BERT approach achieves promising performance on the Hindi-English task and multilingual Transformer-based model, XLM-RoBERTa for the Spanish-English task. They achieved an average F1-score of 0.6850 on Hindi-English and 0.7064 on the Spanish-English dataset.**(Leon, 2020) [18]** Introduced the word embeddings trained on code-switched tweets, especially those that combine Spanish and English tweets. They train their embeddings using the CBoW algorithm rather than skip-gram as it is better suited to smaller datasets. They used them to train a sentiment classifier that 0.722 on F1 scored.

**Lexicon-based**

To determine polarity, **(Taboada et al., 2011) [34]** Proposed a lexicon-based technique that matches opinion terms in a sentiment dictionary with data. They assign sentiment scores to opinion words, which describe how positive, negative, or objective the words in the dictionary. **(Hu et al., 2004) [13]** Presented the dictionary-based approach's key strategy A small set of opinion words with known orientations is manually collected**.** In contextual advertising, **(Qiu et al., 2010) [28]** Employed a dictionary-based technique to identify sentiment sentences. They presented an advertising approach that would improve ad relevancy and user experience. They presented a rule-based strategy to deal with topic word extraction and consumer attitude identification in advertising keyword extraction using syntactic parsing and sentiment dictionaries. They contributed to automotive forums web forums. Their findings showed that the proposed method for ad keyword extraction and ad selection is effective.

**Hybrid-based**

Hybrid approaches combine lexicon-based and machine-learning-based approaches **(Shekhawat et al., 2020) [30]**. To identify the sentiments of code-mixed Hindi-English and English-Spanish data, researchers**(Zhu et al., 2020)English-Spanish [40]** presented an ensemble model using word n-grams-based Multinomial Naive Bayes (MNB) and sub-word-level representations in LSTM (Sub-word LSTM). They generate word-based uni-gram and bi-gram features of the sentence and then input them into the MNB classifier. They used LSTM deep learning model and employ middle-level representations of the sub-word learned by the filter during the convolution process. This task aims to assess the polarity of the text by categorizing it as positive, negative, or neutral. They tested the system on Hindi-English and English-Spanish code-mixed social media data sets and in the Hindi-English task, the F1 score was 0.647, while in the English-Spanish task, it was 0.682.

**(Bao et al., 2020)especially in multilingual societies like India. Detecting the emotions contained in these languages, which is of great significance to the development of society and political trends. In this paper, we propose an ensemble of pesudo-label based Bert model and TFIDF based SGDClassifier model to identify the sentiments of Hindi-English (Hi-En [5]** identify the sentiments of Hindi English (Hi-En) code-mixed data, the author proposed an ensemble of pseudo-label-based Bert model and TFIDF based SGD Classifier model. The ensemble model combines the strengths of rich semantic information from the Bert model and word frequency information from the probabilistic n-gram model to predict the sentiment of a given code-mixed tweet. This Model

achieves 0.686 F1 scores. **(Javdan et al., 2020) [14]** They used different preprocessing techniques and proposed to use various methods that vary from NBSVM to more complex deep neural network models. They used the NBSVM model, which is a combination of Nave Bayes and linear models like Support Vector Machine (or logistic regression), and the calculation ratio of log Nave Bayes used as a feature for Support Vector Machine. They used the TF-IDF matrix with character n-gram features to model as input. They applied this method over both of the Hindi-English and Spanish-English Codemixed datasets and achieved an F1 score of 0.751 for the Spanish-English sub-task and 0.706 over the Hindi-English sub-task. **(Gopalan et al., 2020) [12]** introduced the two main approaches: first is using transfer learning to fine-tune pre-trained BERT models and the second is training feed-forward neural networks to practice bag-of-words representations. The best-performing fine-tuned model gives 63.9% accuracy and the best-performing bag-of-words model gives 60.0% accuracy. Then they used bagging to create a BERT classifier and fine-tuned BERT gives better performance with the Hindi-English dataset. They obtained a 71.3% F1 score on Hindi-English Codemixed tweets.

Table1: Comparisons of various Sentiment Classification Techniques (2020)

## VII. Conclusion

Sentiment analysis is useful in many real-world applications such as review analysis, recommendation systems, and so on. Sentiment analysis becomes more difficult in the situation when data is noisy and collected from social media. India is a multilingual country of multilingual people; those are non-local English speakers who use more than one language to communicate with each other. The switching from one language to another language is called code-switching or code-mixing, which depends on the type of mixing. This survey paper presented an overview of the recent updates in Sentiment Analysis of Codemixed Text. We also highlight NLP techniques and Machine Learning (ML), Lexicon and hybrid approaches to process the code-mixed data. It discusses the applications and challenges faced by researchers in this field.

## VIII. References


[1] Advani, L., Lu, C., & Maharjan, S. (2020). *C1 at SemEval-2020 Task 9 : SentiMix : Sentiment Analysis for Code-Mixed Social Media Text using Feature Engineering*. 1227–1232.

[2] Ansari, A. F., Seenivasan, A., Anandan, A., & Lakshmanan, R. (2017). *Twitter Sentiment Analysis CS5228 Project Group Report*. *1*, 15. https://github.com/abdulfatir/twitter-sentiment-analysis/blob/master/docs/report.pdf

[3] Balahur, A., & Turchi, M. (2014). Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. *Computer Speech and Language*, *28*(1), 56–75. https://doi.org/10.1016/j.csl.2013.03.004

[4] Banerjee, S., Ghannay, S., Rosset, S., Vilnat, A., Rosso, P., & Val, D. (2020). *LIMSI UPV at SemEval-2020 Task 9 : Recurrent Convolutional Neural Network for Code-mixed Sentiment Analysis*. 1281–1287.

[5] Bao, W., Chen, W., Bai, W., Zhuang, Y., Cheng, M., & Ma, X. (2020). Will_go at {S}em{E}val-2020 Task 9: An Accurate Approach for Sentiment Analysis on {H}indi-{E}nglish Tweets Based on Bert and Pesudo Label Strategy. *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 1348–1353. https://www.aclweb.org/anthology/2020.semeval-1.182

[6] Baruah, A., Das, K., Barbhuiya, F., & Dey, K. (2020). {IIITG}-{ADBU} at {S}em{E}val-2020 Task 9: {SVM} for Sentiment Analysis of {E}nglish-{H}indi Code-Mixed Text. *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 946–950. https://www.aclweb.org/anthology/2020.semeval-1.121

[7] Bear, E., & Constantina, D. (2020). *TueMix at SemEval-2020 Task 9 : Logistic Regression with Linguistic Feature Set for Sentiment Analysis of Code-Mixed Social Media Text*. 1316–1321.

[8] Braaksma, B., Scholtens, R., Suijlekom, S. van, Wang, R., & Üstün, A. (2020). FiSSA at SemEval-2020 task 9: Fine-tuned for feelings. *ArXiv*, 1239–1246.

[9] Choudhary, N., Singh, R., Bindlish, I., & Shrivastava, M. (n.d.). *Sentiment Analysis of Code-Mixed Languages leveraging Resource Rich Languages*.

[10] Dang, N. C., Moreno-García, M. N., & de la Prieta, F. (2020). Sentiment analysis based on deep learning: A comparative study. *ArXiv*.

[11] Garain, A., Mahata, S. K., & Das, D. (2020). *JUNLP at SemEval-2020 Task 9 : Sentiment Analysis of Hindi-English code mixed data using Grid Search Cross Validation*. 1276–1280.

[12] Gopalan, V., Hopkins, M., & Bert, F. (2020). *Reed at SemEval-2020 Task 9 : Fine-Tuning and Bag-of-Words Approaches to Code-Mixed Sentiment Analysis*. 1304–1309.

[13] Hu, M., Liu, B., & Street, S. M. (2004). *Mining and Summarizing Customer Reviews*.

[14] Javdan, S., Shangipour ataei, T., & Minaei-Bidgoli, B. (2020). IUST at SemEval-2020 Task 9: Sentiment Analysis for Code-Mixed Social Media Text using Deep Neural Networks and Linear Baselines. *ArXiv*, 1270–1275.

[15] Kim, E. (2006). Reasons and Motivations for Code-Mixing and Code-Switching. *Issues in EFL*, *4*(1), 43–61. http://originalresearch.blog.uns.ac.id/files/2010/04/reasons-and-motivations-for-code-mixing-and-code-switching-by-eunhee-kim.pdf

[16] Kumar, A., Agarwal, H., Bansal, K., & Modi, A. (2020). BAKSA at SemEval-2020 task 9: Bolstering CNN with self-attention for sentiment analysis of code mixed text. *ArXiv*, 1221–1226.

[17] Lal, Y. K., Kumar, V., Dhar, M., Shrivastava, M., & Koehn, P. (2019). De-mixing sentiment from code-mixed text. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student*

*Research Workshop*, 371–377. https://doi.org/10.18653/v1/p19-2052

[18] Leon, F. A. L. De. (2020). *CS-Embed at SemEval-2020 Task 9 : The effectiveness of code-switched word embeddings for sentiment analysis*. 922–927.

[19] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–184. https://doi.org/10.2200/S00416ED1V01Y201204HLT016

[20] Long, H., Liao, B., Xu, X., & Yang, J. (2018). A hybrid deep learning model for predicting protein hydroxylation sites. *International Journal of Molecular Sciences*, *19*(9). https://doi.org/10.3390/ijms19092817

[21] Malik, V., & Kumar, A. (2018). Analysis of Twitter Data Using Naive Bayes Algorithm. *International Journal on Recent and Innovation Trends in Computing and Communication*, *6*(4), 120–125. http://www.ijritcc.org

[22] Mandal, S., & Das, D. (2018). Analyzing roles of classifiers and code-mixed factors for sentiment identification. *ArXiv*, *1*.

[23] Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. **SEM 2013 - 2nd Joint Conference on Lexical and Computational Semantics*, *2*(SemEval), 321–327.

[24] Moralwar, S. B., & Deshmukh, S. N. (2015). *International Journal of Computer Sciences and Engineering Engineerin g Open Access Different Approaches of Sentiment Analysis*. *3*.

[25] Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of the 7th International Conference on Language Resources and Evaluation, LREC 2010*, *January 2010*, 1320–1326. https://doi.org/10.17148/ijarcce.2016.51274

[26] Parikh, A., Bisht, A. S., & Majumder, P. (2020). {IRL} ab_{DAIICT} at {S}em{E}val-2020 Task 9: Machine Learning and Deep Learning Methods for Sentiment Analysis of Code-Mixed Tweets. *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 1265–1269. https://www.aclweb.org/anthology/2020.semeval-1.169

[27] Patra, B. G., Das, D., & Das, A. (2018). Sentiment Analysis of Code-Mixed Indian Languages: An Overview of SAIL_Code-Mixed Shared Task @ICON-2017. *ArXiv*, *March*.

[28] Qiu, G., He, X., Zhang, F., Shi, Y., Bu, J., & Chen, C. (2010). Expert Systems with Applications DASA : Dissatisfaction-oriented Advertising based on Sentiment Analysis q. *Expert Systems With Applications*, *37*(9), 6182–6191. https://doi.org/10.1016/j.eswa.2010.02.109

[29] Salas-Zárate, M. D. P., Medina-Moreira, J., Lagos-Ortiz, K., Luna-Aveiga, H., Rodríguez-García, M. Á., & Valencia-García, R. (2017). Sentiment Analysis on Tweets about Diabetes: An Aspect-Level Approach. *Computational and Mathematical Methods in Medicine*, *2017*. https://doi.org/10.1155/2017/5140631

[30] Shekhawat, S. S., Shringi, S., & Sharma, H. (2020). Twitter sentiment analysis using hybrid Spider Monkey optimization method. *Evolutionary Intelligence*, *0123456789*. https://doi.org/10.1007/s12065-019-00334-2

[31] Singh, P., & Lefever, E. (2020). *LT3 at SemEval-2020 Task 9 : Cross-lingual Embeddings for Sentiment Analysis of Hinglish Social Media Text*. 1288–1293.

[32] Sultan, A., Gaber, A., Salim, M., & Hosary, I. El. (2020). WESSA at semeval-2020 task 9: Code-mixed sentiment analysis using transformers. *ArXiv*, 1342–1347.

[33] Swtcihing, C. (n.d.). *Code switching and mixing ( Communication in Learning Language )*. 123–135.

[34] Taboada, M., Brooke, J., & Voll, K. (2011). *Lexicon-Based Methods for Sentiment Analysis*. *September 2010*.

[35] Tang, D., Qin, B., & Liu, T. (2015). Deep learning for sentiment analysis: Successful approaches and future challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *5*(6), 292–303. https://doi.org/10.1002/widm.1171

[36] Ver, M., & Soares, S. (2020). *Deep Learning Brasil - NLP at SemEval-2020 Task 9 : Sentiment Analysis of Code-Mixed Tweets Using Ensemble of Language Models*. 1233–1238.

[37] Wang, P. (2020). *MeisterMorxrc at SemEval-2020 Task 9 : Fine-Tune Bert and Multitask Learning for Sentiment Analysis of Code-Mixed Tweets*. 1294–1297.

[38] Xia, R., Zong, C., & Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, *181*(6), 1138–1152. https://doi.org/10.1016/j.ins.2010.11.023

[39] Zaharia, G., Vlad, G., Cercel, D., Rebedea, T., & Chiru, C. (2020). *UPB at SemEval-2020 Task 9 : Identifying Sentiment in Code-Mixed Social Media Texts using Transformers and Multi-Task Learning*. 1322–1330.

[40] Zhu, Y., Zhou, X., Li, H., & Dong, K. (2020). Zyy1510 Team at {S}em{E}val-2020 Task 9: Sentiment Analysis for Code-Mixed Social Media Text with Sub-word Level Representations. *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 1354–1359. https://www.aclweb.org/anthology/2020.semeval-1.183