Exploring Contextual Embeddings for Twitter Sentiment Analysis: A Comparative Study of NLP (Natural Language Processing)

Ishu[1], Mohammad Arman[2], Mohd Shoyeb Sheakh[3]

[1] Data-Science Department, Noida Institute of Engineering and Technology, Greater Noida, India
[2] Data-Science Department, Noida Institute of Engineering and Technology, Greater Noida, India
[3] Data-Science Department, Noida Institute of Engineering and Technology, Greater Noida, India

Abstract

Sentiment analysis or Opinion Mining is a field of natural language processing ***(NLP)*** *that plays an important role to understand and analyse information by identifying thoughts or emotions in text. In this research paper, we compare various natural language processing techniques for Twitter sentiment analysis; particularly, we focus on the success rate contextual embeddings exhibit. Overall, machine learning models such as Naive Bayes Classifier, Logistic Regression, CatBoost Classifier, and XGBoost form the bag-of-words representations of the text. Still, they can also be conditioned on other methods such as TF-IDF or the text itself. Our findings provide valuable insights into how the performance of sentiment analysis depends on the bag-of-words representation of the model. Using BOW, in particular, BOW bag-of-words concatenation technique, we managed to produce the following success rate: Naive Bayes 60%, Logistic regression 84%, CatBoost Classifier 85%, and XGBoost, 83%. Furthermore, using BOW bag-of-words as a representation, we managed to reach the following success rate with XGBoost 83%. (86%) keeping competitive advantages in the case of the majority of the rest of the classifiers. Further, the present study discloses the potential of vectorization techniques in isolation: the implementation of only TF-IDF vectorization presents different accuracies in the event of Naive Bayes as 50%, Logistic Regression as 76%, CatBoost Classifier as 83%, and XGBoost as 87%. To sum up, the provided evidence reveals the opportunities of the application of contextual embeddings as applied in the Twitter sentiment analysis, enhancing future studies in the NLP developed field.*

Keywords : ***Sentiment Analysis, Opinion Mining, Natural Language Processing(NLP), Machine Learning, Neural Networks, TF-TDF, BOW(Bag of Words), Deep Learning, Python programming Language.***

1.Introduction

Natural Language Processing **(NLP)** has emerged as a powerful tool for extracting meaningful insights from textual data across various domains. Within the use of NLP, sentiment analysis has gained significant attention due to its wide-ranging applications in understanding human sentiment, opinion, and emotion expressed in text. Sentiment analysis, also known as opinion mining, involves the computational study of subjective information to determine the underlying sentiment polarity, whether it be positive, negative, or neutral. The simple view is that the sentences of a text are first analysed in terms of their syntax, this provides an order and structure that is more open and responsible to an analysis in terms of semantics, or literal meaning; and this is followed by a stage of pragmatic* analysis whereby the meaning of the text in context is determined.

In this paper we tried to excel in analysing the sentiments (emotions) underlying in the text or opinions of the people, that are written as feedbacks, reviews or comments, we describe a unified NLP sentiment analysis system that achieves good performance on multiple benchmark tasks by discovering its own internal representations. **Sentiment analysis (SA)[11, 12],** a subset of **natural language processing (NLP),** is crucial for gauging public opinion on products, services, or topics. With the rise of online platforms like blogs, product reviews, and social media, users express their opinions through **User Generated Content (UGC)**. This content, both positive and negative, is readily accessible online, particularly in the form of electronic **UGC (E-UGC).** E-UGC provides detailed insights for product reviews, with businesses. Many **businesses to customer (B2C)** incorporate features like video and photo uploads to enhance customer feedbacks.

Main aim or the goal of sentiment analysis is to identify positive or negative overall attitudes or opinions towards a brand, product or service based on text comments

2. Related work

Duc-Hong Pham and Anh-Cuong Le**[3]** use a hybrid approach, The initial model, word2vec, establishes word embeddings within a semantic framework, serving as input for our proposed model. Concurrently, the second model focuses on generating advanced representations for sentences or documents, demonstrating superior efficacy in natural language processing tasks compared to traditional spelling-based methods.**[3,10,12]**

Md. Rakibul Hasan, Maisha Maliha, and M. Arifuzzaman**[2]** employed an NLP-based pre-processing framework to effectively filter tweets. They subsequently integrated the Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) models for sentiment analysis.**[2,10]**

Varsha Sahayak, Vijaya Shete, and Apashabi Pathan**[4,3]** employ Support Vector Machines (SVMs) for sentiment analysis, contributing to automated sentiment classification in Twitter data. They also investigate Naive Bayes, recognized for its efficiency in text categorization.**[4]**

Coletta, L. F., da Silva, N. F., Hruschka, E. R., & Hruschka, Jr., E. R. **[1,8,9],** employ traditional classifiers such as Naive Bayes, Maximum Entropy, and Support Vector Machines (SVM) for sentiment analysis in tweets. Their study emphasizes leveraging SVM as the primary classifier and introduces the C3E-SL cluster ensemble algorithm to enhance sentiment analysis accuracy through integrated classifier and clustering techniques.**[1]**

Rajak, Panda, and Kumar (2024)**[5]** investigated sentiment analysis on Twitter during the COVID-19 pandemic, emphasizing the integration of text information and sentiment dictionaries. Their study underscores the significance of incorporating domain-specific knowledge, such as sentiment dictionaries, to improve sentiment analysis accuracy in social media data.

3. Proposed Work: Twitter Sentiment Analysis

This section represents our proposed work and focuses on a strategy to sentiment analysis on Twitter data. The architectural overview describing an overall process design of sentiment analysis of Twitter data(comments) Analysis.

Now a days as the result of increasing growth of the Digital era, all the activities are now inclined to the internet. This leads to the rapid growth in the Web information due to the huge amount of the user generated content, this even involves the indulging of the social media interactions like posts, blogs, articles, forums, polling, etc. for promotion or advertising of the products or services in the world. Social media Application Like **Facebook, Instagram, Twitter (now X), LinkedIn** are the most common platforms that serves the purpose of the social Activities and interactions.

So here comes the task of Sentiment analysis of the text content or the opinions of the people, generated during the social interactions, which are to be analysed in order to know the emotion of the text - how that activity is impacting people positively or either negatively.

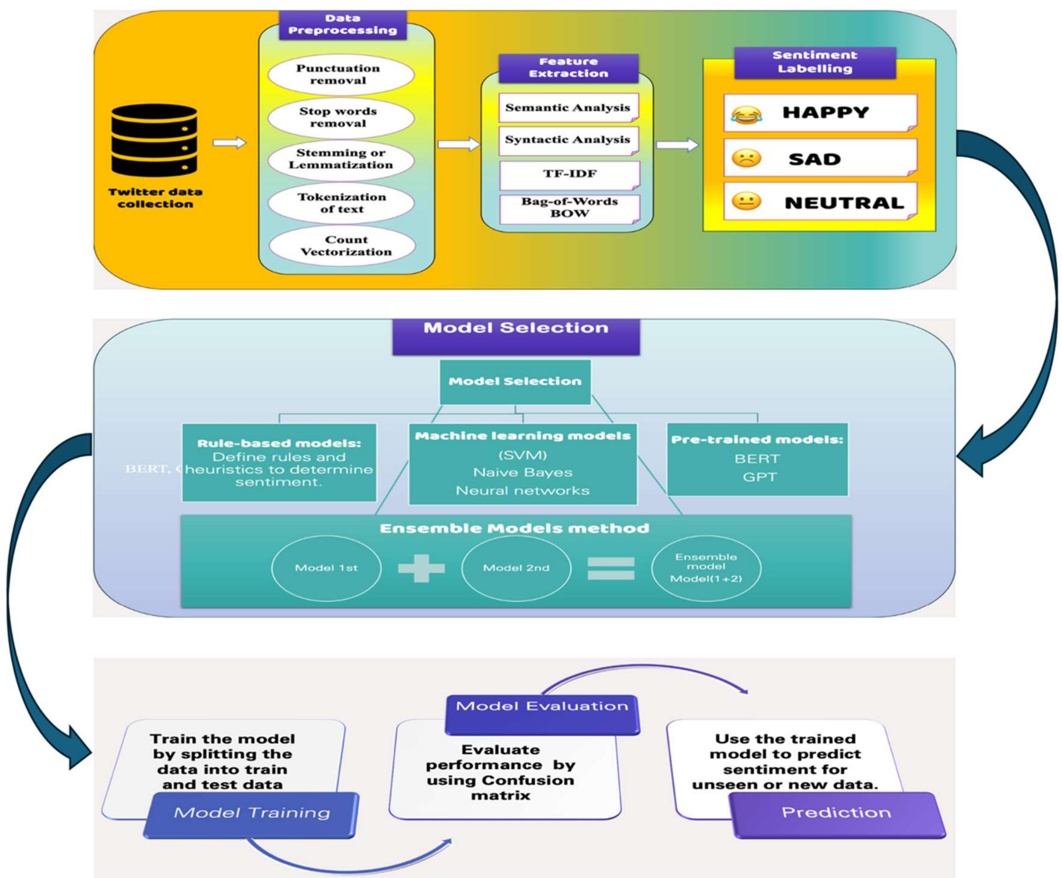


Fig.1 Process of Sentiment Analysis Using NLP (Natural Language processing)

3.1. Sentiment Analysis Study of Methodology/Steps/Applications

Here we are going to study the process of Sentiment Analysis of text from the corpus to understand people's feelings or opinions expressed in text. Here are the steps involved in performing the sentiment analysis.

Aim of the paper is to present a detailed process and methodology for conducting Twitter sentiment analysis. The gathering the textual data from relevant sources, such as social media platforms, customer reviews, surveys, or other text-rich datasets. In our scenario we are using the dataset of 1.6 Million Tweets for analysing. Data pre-processing is the most important step in data analysis. This step involves multiple processes that are inclined on each other in order to clean the data and make it suitable for fitting the model. Data Cleaning: Cleaning the Dataset refers to the removing of null values and also to remove the special characters and punctuation marks from the text, it also includes the removal of irrelevant data or information present in the textual data, hence the noiseless data makes it efficient to analyse.Stop Words removal: Stop-words basically needed to be processed out

Data Pre-processing

Text cleaning: removing special characters, URLs, mentions, etc. Tokenization: breaking text into words or phrases Stop word removal: eliminating common words with little semantic value

Stemming or lemmatization: reducing words to their base form Handling emojis, slang, and abbreviations

Example: Pre-processing a tweet by removing URLs, mentions, and stop words

Feature Extraction

Bag-of-Words (Bow) model: representing text as a vector of word frequencies TF-IDF (Term Frequency-Inverse Document Frequency): weighting words based on their importance Word embeddings (e.g., Word2Vec, GloVe): capturing semantic relationships between words N-grams: considering sequences of adjacent words.

Example: Extracting TF-IDF features from pre-processed tweets

Classification algorithms (e.g., Naive Bayes, Support Vector Machines, Random Forests) Deep learning models (e.g., Recurrent Neural Networks, Convolutional Neural Networks) Ensemble methods: combining multiple models for improved performance Consideration of model complexity, scalability, and interpretability. Example: Training a Support Vector Machine classifier on TF-IDF features

Selection of evaluation metrics (accuracy, precision, recall, F1-score, etc.) Cross-validation: assessing model performance on different subsets of data Handling imbalanced datasets Comparison with baseline models or human annotators. Example: Evaluating the SVM classifier using 10-fold cross-validation and F1-score. Interpretation and Analysis

Visualization techniques (word clouds, bar charts, heatmaps, etc.) for exploring sentiment distribution

Identifying influential factors (keywords, topics, user demographics) Qualitative analysis of misclassified instances Drawing insights and implications from the results

Example: Visualizing sentiment distribution using a word cloud and analysing influential topics

3.2. Machine learning models

3.2.1Naïve Bayes Classifier.

The Naive Bayes classifier is a probabilistic machine learning algorithm that is based on Bayes' theorem, with the "naive" assumption that all features are independent of each other given the class. Despite its simplicity, Naive Bayes is often surprisingly effective in practice, especially for text classification tasks like spam filtering or sentiment analysis.

4. Methodology:

This section delineates the methodology employed to evaluate the efficacy of our proposed deep learning and machine learning techniques for sentiment analysis of extracted tweets. Python was selected as the programming language owing to its suitability for data analytics tasks, facilitated by its extensive range of modules and libraries, including NumPy, Pandas, Matplotlib, and Seaborn, which are well-suited for data analysis purposes. Furthermore, we conducted comparative analyses with alternative classifiers to enhance the performance of our approach.

4.1. Dataset (Data gathering)

The Dataset was basically downloaded/gathered from the Kaggle website, about the tweet datasets that are available and even the Data was gathered through the twitter tweets web scrapping module named as ***snscrape or tweepy*.**[17, 20] The dataset contained 160k tweets that were tested through machine learning and deep learning algorithms for the purpose of sentiment analysis. As the dataset gathered was in raw format so we added the attribute(column) names to the dataset namely as ***target_rating*, *id*, *username*, *text*** (tweets or opinions for analysis). Then we transformed the target_rating into sentiment/opinion classifications to discern between positive and negative sentiments as follows:

Target_rating = 0 to 100 : Negative, Target_rating = 100 to 1000 : Negative, Target_rating = 1000 to above : positive, these labels namely positive, negative, and neutral are then converted to [1,0,-1].

And then the features that are relevant for the analysis are the selected for better efficiency. These classification labels, along with the features created are then fed to the various machine learning and deep learning models that are further deployed for the purpose of Sentiment Analysis of public opinions.

Id	Text tweets	category
1.	the larger blame increasing pollution congestion delhi has min modi minister was adamant his decision increasing metro fare which was opposed govt this money making decision has costed delhi metro lost lakh commuters	1
2.	modi the kid boosted gang thieves through window his only job and open the door that his gang can come loot bjp doesnt expect come back power they are using the time remaining leach much resources out india they can	1
3.	how can the election commission india enforcing the mcc not the modi code conduct but moral code mute spectator this nonsense the governor shame	0
4.	whats going controversial amendments environmental laws procedures documents reveal modi and javadekars war indias environment via	1
5.	dont underestimate that mlas potential might become indias next prime minister read bjp parrots modi line asks isn' sonia italy beti sept 2002	0
6.	exactly point those who reach positions power are too paranoud about upsetting the applecart only those with unconditional islamic support base show the raw guts act thats why khanchris also preffers islamic support its not merit based support	-1

***Table.1 A sample of dataset containing tweets and its created labels*.**

S.No.	Sentiment Type	Label used	Total Value Counts
1.	Positive Tweets	1	72250
2.	Neutral Tweets	0	55213
3.	Negative Tweets	-1	35510

***Table.2 Table containing total value counts of three classes of tweets*.**

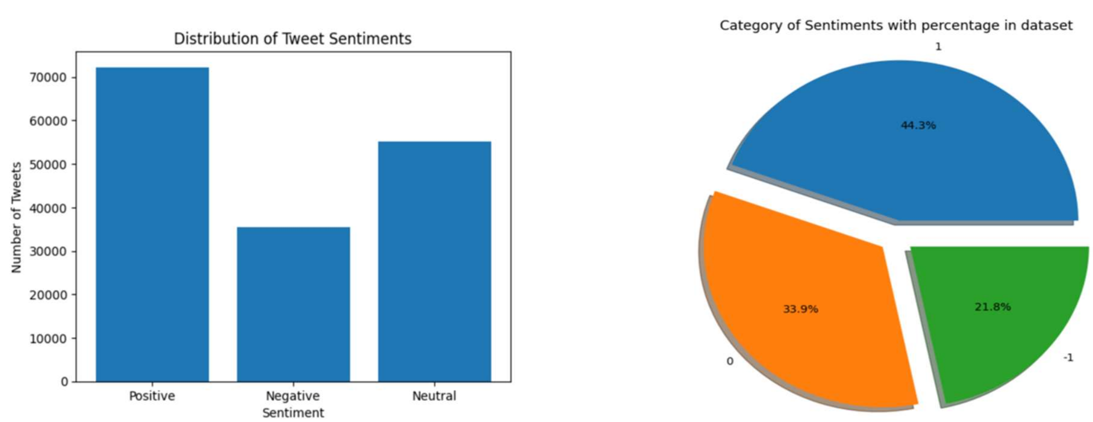


Fig.2 Bar plot and Pie chart Showing the Distribution of the Category of tweets.

4.2. Formation of the Attributes and Training/Testing Datasets

After the data collection here comes the step of **Data Preprocessing** where we needed to transform the data, eliminating the redundancy of data, replacing the missing values, converting people's (tweets)opinion text into analyzable formats for to get best efficiency during analysis process.
Following are the steps that are considered for preprocessing of the data:

4.2.1 Data Cleaning: Removing the null values, and all the punctuation marks or symbols used in textual data, hence noiseless data makes the analysis efficient.

4.2.2 Stop-Words Removal: Stop words are the words in any language that are very common in use but they don't add much information to the text, so they are excluded to reduce the size of data and to increase the efficiency eg: "the", "a", "an","what".

4.2.3 Case Conversion: Converting the textual data to lower case characters and removing the internet slangs from that and the hashtags are important, so they remain unchanged ex: *#naturephotography*, *#lovelife* etc.

4.2.4 Tokenization: Breaking down text into smaller units, such as words or phrases.
4.2.5 Stemming or Lemmatization: Reducing words to their root or base form to standardize variations. ***eg:*** *●Fastest : Fast ●Standardisation : Standard*.

4.2.6 Named Entity Recognition (NER): Identifying the entities in the corpus documents by text summarization (e.g., names, locations, organizations) in the text.

4.2.7 Text Clustering and Classification: Grouping similar documents or assigning predefined categories to texts.

The next subsequent phase concerns about feature extraction, for the machine Learning or deep learning approach we do need to extract features from the dataset. Feature extraction techniques used are **TF-IDF, Bow** as they are used to convert the textual data to numerical vectors.

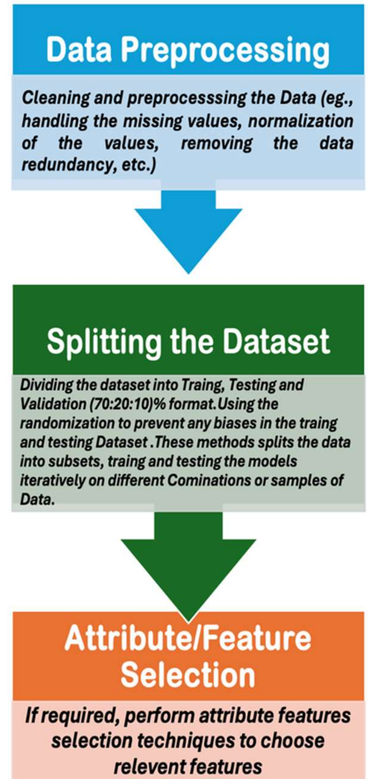


Fig.2 Data preprocessing and Attribute Selection

4.2.8 *Vectorization:*

- **TF-IDF:** TF-IDF, stands for Term Frequency-Inverse Document Frequency, it is a statistical method that is mostly used in NLP(natural language processing) for to retrieve the information. It measures that how important a particular word is in a document with respect to the collection of documents (corpus). Words that are basically present in the textual documents are converted to importance numbers with the help of text vectorization.

  In TF-IDF **TF** is the ratio of frequency of a particular term "**t**" present in the document "**d**" to the total number of words in document "**d**".

$$TF(t,d) = \frac{frequency\ (t,d)}{total\ word\ count(d)} \qquad (1)$$

  IDF determines the importance of the word or a term in all the documents present in the corpus, total number of documents containing that term in corpus. If a term is rare so it has high IDF score so it depicts that term is very important(*technical words, jargons*) but if a term has low IDF score it depicts that term is common in the corpus(e.g.: it, we, they, the, etc.).

$$IDF(t,d) = \left(\frac{total\ number\ of\ documents(d)in\ corpus}{total\ number\ of\ documents\ in\ corpus\ containing\ term(t,d)}\right) \qquad (2)$$

  TF-IDF is the term we get after multiplying the **TF** and **IDF** scores.

$$TF\text{-}IDF = tf_{(t,d)} * log(idf_{(t,d)}) \qquad (3)$$

- **Bow:** Bow stands for Bag of Words, here the text of document is broken into the tokens by the process called tokenization and all the punctuations and stop words are excluded from the text, then in each document the value of each element represents the frequency of the word in that document. Once the vocabulary is built then the vectorization is done.

BoW

1. **Hello and welcome friends**
2. **Arman, Ishu and Shoaib love teaching**
3. **Arman, Ishu and Shoaib love to play cricket**

D.No.	Hello	and	welcome	friends	Arman	Ishu	and	Shoaib	love	teaching	to	play	cricket
1.	1	1	1	1	0	0	0	0	0	0	0	0	0
2.	0	0	0	0	1	1	1	1	1	1	0	0	0
3.	0	0	0	0	1	1	1	1	1	0	1	1	1

- **Concatenating TF-IDF & BoW:** The method we used for the vectorization purpose is the concatenation of both TF-IDF and BoW method for to get a better feature selection as input and for processing and analyzing, optimized analysis and to boost the performance of various machine learning algorithms used respectively.

TF-IDF features

Doc.No	cricket	friends	hello
1.	0.000000	0.000000	0.000000
2.	0.420241	0.000000	0.000000

+

BoW features

Doc.No	cricket	friends	hello
1.	0.0	0.0	0.0
2.	1.0	0.0	0.0

Doc.No	cricket	friends	hello	cricket	friends	hello
1.	0.000000	0.000000	0.000000	0.0	0.0	0.0
2.	0.420241	0.000000	0.000000	1.0	0.0	0.0

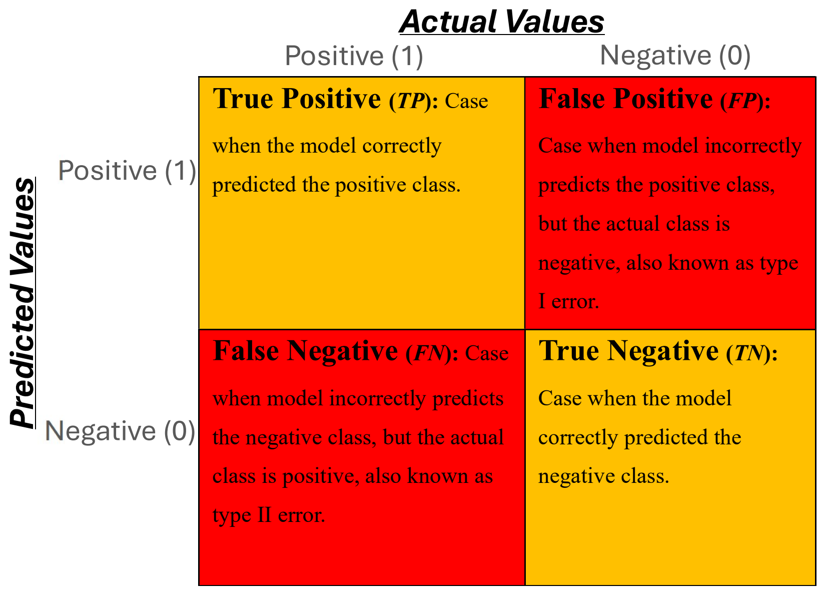
features by Concatenation of TF-IDF and Bow

These Vectorization techniques forms a numerical individual matrix that subsequently partitioned into two distinct datasets respectively as the test dataset and training dataset, so that various machine learning and deep learning model can be trained and tested on those dataset and to analyse the model's performance on the testing datasets.

4.3. Confusion Matrix / Result Analysis:

Confusion matrix basically is a fundamental tool that can be used by importing the confusion_matrix from sklearn.metrics library in python it is used to visualize the model performance in a table like format by comparing the predictions of the model with the actual values.

Confusion matrix usually consists of four components respectively as listed below:



These four values are used for the purpose of analyzing the performance of model as to calculate various performance measures: Accuracy, Recall, Precision, F1 Score that provides the insights about how well our model is fitted and is performing, hence confusion matrix is useful to analyze that where our model is lacking or making errors and helps us to fine-tune the model for better performance.

4.3.1 Accuracy: accuracy is the total fraction of correctly predicted classes versus the total predictions done by model, maximum value of accuracy can be 1 and minimum value can be 0,
accuracy is represented as:

$$Accuracy = \frac{TP+T}{TP+TN+FP+F} \tag{4}$$

4.3.2 Precision: precision is the ratio of total positive class correctly predicted or retrieved to the total of positive classes predicted both correctly and incorrectly.

$$Precision = \frac{TP}{TP \quad FP} \tag{5}$$

4.3.3 Recall: Also known as true positive rate (TPR)

$$Recall = \frac{TP}{TP+FN} \tag{6}$$

4.3.3 F1 Score: F1 score is the harmonic mean of precision and recall, it maintains the balance between precision and recall, value lies between[0,1] formula is inscribed below.

$$F1\ Score = \frac{2*(precision * recall)}{(precision + recall)} \tag{7}$$

5. Results and Insights

Our findings provide valuable insights into how the performance of sentiment analysis depends on the bag-of-words representation of the model. Using BOW, in particular, BOW bag-of-words concatenation technique, we managed to produce the following success rate: Naive Bayes 60%, Logistic regression 84%, CatBoost Classifier 85%, and XGBoost, 83%. Furthermore, using BOW bag-of-words as a representation, we managed to reach the following success rate with XGBoost 83%. (86%) keeping competitive advantages in the case of the majority of the rest of the classifiers. Further, the present study discloses the potential of vectorization techniques in isolation: the implementation of only TF-IDF vectorization presents different accuracies in the event of Naive Bayes as 50%, Logistic Regression as 76%, CatBoost Classifier as 83%, and XGBoost as 87%. To sum up, the provided evidence reveals the opportunities of the application of contextual embeddings as applied in the Twitter sentiment analysis, enhancing future studies in the NLP developed field.

Result of machine learning models using tf-idf and bow concatenation

S.No.	Model	Accuracy	Class	Precision	Recall	F1-Score
1.	Naïve Bayes Classifier	0.60	**-1**	0.70	0.28	0.40
			0	0.86	0.34	0.49
			1	0.54	0.95	0.69
			macro avg	0.70	0.52	0.53
			weighted avg	0.69	0.60	0.56
2.	Logistic Regression	0.84	**-1**	0.81	0.65	0.72
			0	0.82	0.91	0.86
			1	0.87	0.87	0.87
			macro avg	0.81	0.81	0.82
			weighted avg	0.84	0.84	0.84
3	Cat Boost Classifier	0.85	**-1**	0.91	0.61	0.73
			0	0.78	0.97	0.87
			1	0.89	0.86	0.87
			macro avg	0.86	0.81	0.82
			weighted avg	0.86	0.85	0.84
4.	XG Boost Model	0.837	**-1**	0.83	0.59	0.69
			0	0.74	0.96	0.84
			1	0.89	0.82	0.85
			macro avg	0.82	0.79	0.79
			weighted avg	0.83	0.82	0.81

Result of machine learning models using Bow

S.No.	Model	Accuracy	Class	Precision	Recall	F1-Score
1.	Naïve Bayes Classifier	0.62	**-1**	0.64	0.37	0.47
			0	0.77	0.39	0.52
			1	0.57	0.91	0.70
			macro avg	0.66	0.55	0.56
			weighted avg	0.66	0.62	0.59
2.	Logistic Regression	0.81	**-1**	0.79	0.58	0.67
			0	0.74	0.91	0.82
			1	0.88	0.84	0.86
			macro avg	0.80	0.78	0.78
			weighted avg	0.81	0.81	0.80
3	Cat Boost Classifier	0.83	**-1**	0.90	0.60	0.72
			0	0.74	0.98	0.84
			1	0.91	0.84	0.87
			macro avg	0.85	0.80	0.81
			weighted avg	0.85	0.83	0.83
4.	XG Boost Model	0.86	**-1**	0.90	0.69	0.78
			0	0.79	0.98	0.87
			1	0.92	0.86	0.89
			macro avg	0.87	0.84	0.85
			weighted avg	0.87	0.86	0.86

Result of machine learning models using TF-IDF

S.No.	Model	Accuracy	Class	Precision	Recall	F1-Score
1.	Naïve Bayes Classifier	0.50	**-1**	0.89	0.02	0.03
			0	0.83	0.16	0.27
			1	0.47	0.99	0.64
			macro avg	0.73	0.39	0.32
			weighted avg	0.68	0.50	0.38
2.	Logistic Regression	0.76	**-1**	0.87	0.45	0.60
			0	0.74	0.82	0.78
			1	0.76	0.88	0.81
			macro avg	0.79	0.72	0.73
			weighted avg	0.78	0.86	0.75
3	Cat Boost Classifier	0.83	**-1**	0.90	0.60	0.72
			0	0.74	0.98	0.84
			1	0.91	0.84	0.87
			macro avg	0.85	0.80	0.81
			weighted avg	0.85	0.83	0.83
4.	XG Boost Model	0.87	**-1**	0.91	0.69	0.79
			0	0.80	0.98	0.88
			1	0.92	0.86	0.89
			macro avg	0.87	0.85	0.85
			weighted avg	0.88	0.87	0.86

6. References

[1] Coletta, L. F., da Silva, N. F., Hruschka, E. R., & Hruschka, E. R. (2014, October). Combining classification and clustering for tweet sentiment analysis. In *2014 Brazilian conference on intelligent systems* (pp. 210-215). IEEE.

[2] Hasan, M. R., Maliha, M., & Arifuzzaman, M. (2019, July). Sentiment analysis with NLP on Twitter data. In 2019 international conference on computer, communication, chemical, materials and electronic engineering (IC4ME2) (pp. 1-4). IEEE.

[3] Pham, D. H., & Le, A. C. (2018). Learning multiple layers of knowledge representation for aspect based sentiment analysis. Data & Knowledge Engineering, 114, 26-39.

[4] Sahayak, V., Shete, V., & Pathan, A. (2015). Sentiment analysis on twitter data. International Journal of Innovative Research in Advanced Engineering (IJIRAE), 2(1), 178-183.

[5] Rajak, A., Panda, R. N., & Kumar, A. (2024, February). Combining Text Information and Sentiment Dictionary for Sentiment Analysis on Twitter During Covid. In 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT) (Vol. 5, pp. 298-302). IEEE.

[6] Rashid, A., & Huang, C. Y. (2021). Sentiment Analysis on Consumer Reviews of Amazon Products. International Journal of Computer Theory and Engineering, 13(2)

[7] Haque, T. U., Saber, N. N., & Shah, F. M. (2018, May). Sentiment analysis on large scale Amazon product reviews. In 2018 IEEE international conference on innovative research and development (ICIRD) (pp. 1-6). IEEE.

[8] Saif, H., He, Y., & Alani, H. (2012). Semantic sentiment analysis of Twitter. In The Semantic Web–ISWC 2012: 11th International Semantic Web Conference, Boston, MA, USA, November 11-15, 2012, Proceedings, Part I (pp. 508-524). Springer Berlin Heidelberg.

[9] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. J. (2011, June). Sentiment analysis of twitter data. In Proceedings of the workshop on language in social media (LSM 2011) (pp. 30-38).

[10] AlQahtani, A. S. (2021). Product sentiment analysis for amazon reviews. International Journal of Computer Science & Information Technology (IJCSIT) Vol, 13.
[11] Liu, Wei Song, Jingli Lu.[2014] Feature based sentiment analysis for product reviews, Hanshi Wang, Lizhen, *JOURNAL OF SOFTWARE*, 9(2): 274-279
[12] S. S. A. Q. Mahlawi, "Structured data extraction from emails," *2017 International Conference on Networks Advances in Computational Technologies (NetACT)*, vol. 37, pp. 323–328, 2017.
[13] S. Rosenthal, N. Farra, and P. Nakov, "Semeval-2017 task 4: Sentiment analysis in twitter," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 502–518.
[14] Seyed Rajab Nikhashemi, Laily Paim, Saeideh Sharifi, [2013]The Effectiveness of E-Advertisement towards Customer Purchase Intention : Malaysian perspective, IOSR Journal Of Business Management e-ISSN: 2278-487X, Vol-10, Issue 3 (May-Jun2013 PP 93-104
[15] G Angulakshmi, R ManickaChezian. [2014] An Analysis on Opinion Mining: Techniques and Tools, International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 7, July, ISSN (Online) : 2278-1021 ISSN (Print) : 2319-5940 pp-7483-7487
[16] Melville, Wojciech Gryc,[2009] Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification", KDD‟09, Paris, France. Copyright 2009 ACM 978-1-60558-495- 9/09/06.
[17] D. Gruhl, R. Guha, R. Kumar, J Novak, and A Tomkins. [2005] ―The Predictive Power of Online Chatter,‖ Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery in Data Mining (KDD), pp. 78-87,.
[18] Hongwei Wang Wei Wang. [2014] Product Weakness Finder- and Opinion – Aware System through Sentiment Analysis, *Industrial Management and Data Systems*, 114(8):1301-1320.
[19] Antony M. Hooper Maria Uriyo, [2015]"Using Sentiment Analysis to Review Patient Satisfaction Data Located on the Internet, *Journal of Health Organization and Management*, 29(2):221-233.
[20] Alen Yee Long chong Boyin et al,[2016] Predicting Online Product Sales Via Online Reviews, Sentiments And Promotion Strategies", *International Journal Of Operation And Production* Management, 36(4):358-383.
[21] Erik Cambria, Bjö rn Schuller, Catherine Havasi, [2013] New avenues in Sentiment analysis and opinion mining" , Knowledge based approaches to Concept level sentiment Analysis, *IEEE Computer Society* pp- 15-21
[22] Ghulam Shabir , Yousef Mahmood Yousef Hameed , Ghulam Safdar , Syed Muhammad Farouq Shah Gilani,(2014), The Impact of Social Media on Youth: A Case Study of Bahawalpur City, , *Asian Journal of Sciences & Humanities,* Vol. 3(4).