IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ANALYZING EFFECTIVENESS OF MACHINE LEARNING MODELS: A SENTIMENT ANALYSIS APPROACH

[1]Mrs. Dipika Kamleshbhai Patel, [2] Mrs. Hina Sanjaykumar Jayani

[1]Lecturer, Tapi Diploma Engineering College, Surat, India

[2]Lecturer, Tapi Diploma Engineering College, Surat, India

Abstract: Sentiment analysis is a branch of natural language processing that attempts to automatically categorize these viewpoints as neutral, negative, or positive. We investigate the efficacy of several sentiment analysis methods on Twitter data in this research article. We examine how feature selection, pre-processing methods, and machine learning algorithms affect sentiment analysis performance. Our findings demonstrate that the accuracy of sentiment analysis on Twitter data is increased when feature selection and machine learning methods are combined. We also point out the difficulties and restrictions associated with sentiment analysis on Twitter data, including the usage of irony and sarcasm in tweets. The field of sentiment analysis of Twitter data has seen a great deal of activity. The primary topic of this survey is sentiment analysis of Twitter data, which is useful for analyzing information found in tweets that contain highly unstructured, heterogeneous opinions that can be neutral, positive, or negative.

Keywords: Sentiment Analysis, Naïve Bayes, Support Vector Machine (SVM), Machine Learning Models, Linear Regression Model

1. INTRODUCTION

With more than 330 million active users each month, Twitter is a well-known social media network. Twitter is a great source for sentiment analysis because of the volume of data it generates. Sentiment analysis is a branch of natural language processing that categorizes textual opinions into three groups: neutral, negative, and positive. Applications for sentiment analysis on Twitter data include political analysis, customer feedback analysis, and brand reputation management. Before a person purchases a product, sentiment analysis (SA) informs them of how pleasant the product's information is. Marketers and businesses utilize this analysis data to gain insight into their goods and services so that they can cater to the needs of the customer.

2. REVIEW OF LITERATURE

Dang et al. (2020) examines the most recent research using deep learning to address sentiment analysis issues, including sentiment polarity. Word embedding and term frequency-inverse document frequency (TF-IDF) models have been used on multiple datasets. Lastly, a comparison analysis of the experimental outcomes for the various models and input features has been carried out. Singh et al. (2017) optimizes sentiment analysis using four cutting-edge machine learning classifiers: Naïve Bayes, J48, BFTree, and OneR. Three hand produced datasets are used in the experiments: two are taken from Amazon, and one is put together from IMDB movie reviews. This four classification strategies' effectiveness is analyzed and contrasted. While

OneR appears more promising in producing accuracy of 91.3% in precision, 97% in F-measure, and 92.34% in correctly classified instances, Naïve Bayes was found to learn fairly quickly (Singh et al., 2017). Jemai et al. (2021) attempt to develop a classifier that can use machine learning (ML) algorithms to predict a comment's polarity. Three main tasks comprise their work: data extraction, processing, and modeling. Their model is constructed using the NLTK dataset. After that, we create and handle the variables using text mining techniques. Jemai et al. (2021) tended to develop a classifier based on a supervised probabilistic machine learning algorithm to categorize our tweets into positive and negative sentiments. The goal of Balci et al. (2022) is to use an open IMDB dataset to test and assess the effectiveness of state-of-the-art machine learning sentiment analysis techniques. There are numerous instances of irony and sarcasm in the dataset. Transformer-based models, convolutional neural networks (CNN), bag of tricks (BoT), and long-short term memory (LSTM) are developed and assessed. Furthermore, we looked at how hyper-parameters affected the models' accuracy. There are other methods for achieving sentiment extraction and categorization, including neuro-fuzzy and optimization algorithms. Revathy et al. (2022) present a double feed forward neural network as a technological contribution. These methods struggle to classify real-time data that is streamed with information and has a large number of characters. A double feed forward neural network is used to achieve accurate classification, and data from the output layer is sent to the network's double layer. As a result, the data are efficiently streamlined and processed, allowing for the categorization of sentiment. The method is run through to the end, and the obtained results are compared to those of the optimization and neuro-fuzzy algorithms. Regarding classification parameters, the DFFNN performs better than the current method. Kalangi et al. (2021) uses machine learning and natural language processing (NLP) to evaluate sentiment on Twitter statistics pertaining to airline reviews. Customers are essentially ranked according to which airlines they would select for their trip. Based on their tweets on Twitter, people' opinions—positive, negative, and neutral—are assessed. Additionally, data visualization is employed, which may involve cleaning, verifying, and altering the pattern. This data-driven prediction model makes use of machine learning and natural language processing (NLP). To extract the relevant data and make the required deductions, machine learning techniques are used to the associated Twitter dataset comprising airline evaluations. These algorithms are used to determine future trends based on current usage and to extract user attitudes. An extensive description of the process for accomplishing this assignment and the uses of sentiment analysis are included in various documents. After that, it assesses, contrasts, and explores the methods employed in order to get a thorough comprehension of their benefits and drawbacks. In order to determine future paths, the difficulties associated with sentiment analysis are finally explored (Wankhade et al., 2022).

3. OBJECTIVE

Sentiment analysis seeks to analyze the overall sentiment communicated in text by locating and removing subjective information, including opinions, attitudes, emotions, and judgments. Numerous sources, including social media posts, product evaluations, news stories, and customer feedback, might be subjected to the study. Sentiment analysis seeks to automatically categorize a text's conveyed sentiment as positive, negative, or neutral and to measure how strong the sentiment is. Numerous machine learning algorithms and methods for natural language processing can be used for this. Sentiment analysis has many different uses; these include political analysis, customer service, brand reputation management, social media monitoring, market research, and more. Businesses and organizations can take appropriate action and make better judgments by knowing the emotion of a specific group of people or topic.

4. METHODOLOGY

4.1. Data Pre-processing:

For sentiment analysis, the Twitter data is cleaned and made ready by pre-processing procedures. Pre-processing methods that are frequently used include tokenization, stemming, and the removal of stop words. Words like "the," "and," and "is" are examples of stop words; they can be eliminated to lessen noise in the data. In order to decrease the dimensionality of the data, stemming is the act of reducing words to their most basic form, such as "running" to "run." Tokenization is the process of separating text input into discrete words or phrases so that sentiment analysis may use them as features.

4.2. Features Extraction:

The process of choosing and converting the pre-processed Twitter data into a sentiment analysis-ready format is known as feature extraction. Among the frequently used feature extraction techniques are word embeddings, n-grams, and bag-of-words. The bag-of-words method ignores the words' sequence of appearance in the text and instead depicts the text data as a collection of individual words. Similar to bag-of-words, n-grams take into account word sequences rather than single words. A more sophisticated feature extraction technique called word embeddings uses the context and meaning of the words to represent the words as vectors in a high-dimensional space.

4.3. Machine Learning Algorithms:

Models that are capable of forecasting the sentiment of the Twitter data are trained using machine learning techniques. Frequently employed machine learning algorithms in sentiment analysis are Random Forest, Naive Bayes, and Support Vector Machines (SVM). The basic probabilistic algorithm Naive Bayes makes the assumption that the features used for categorization are unrelated to one another. A more intricate method called Support Vector Machines (SVM) determines the best hyperplane to divide the data into several classes. Several decision trees are used in the Random Forest ensemble approach to provide predictions.

4.3.1. Naïve Bayes:

Naive Bayes is a simple probabilistic machine learning method for classification tasks. It relies on Bayes' theorem, which provides a way to calculate the probability of a hypothesis given some observed evidence. The "naive" in Naive Bayes refers to the assumption that all the features used in the classification process are independent from one another. To classify an item, the algorithm calculates the probability of each class based on the input features and then predicts the class with the highest probability. It does this by calculating the conditional probability of each feature given each class, then combining these probabilities using Bayes' theorem to determine the likelihood of each class based on the input features. The input features are represented as a vector containing either binary or real-valued values. By combining the conditional probabilities for each class, the algorithm can predict which class is most likely given the input features. For example, a Naive Bayes algorithm used in a plant recognition system might classify photos based on attributes such as size, color, and shape. Although these features are treated as if they are independent, the algorithm uses the combined information from all features to estimate the likelihood that a particular object is a specific plant.

4.3.2. Logistic Regression:

Logistic regression is a common statistical method used for binary classification tasks, where the goal is to predict whether an instance belongs to one of two classes based on input features. Unlike linear regression, which predicts continuous outcomes, logistic regression predicts probabilities that an instance belongs to a particular class, and then assigns it to the class with the highest probability. In logistic regression, the relationship between the input features and the probability of belonging to a specific class is modeled using a logistic function, also known as the sigmoid function. This function maps any real-valued input to a value between 0 and 1, making it suitable for probability predictions. The basic concept is to find a set of coefficients for the input features that best describes the relationship between these features and the probability of belonging to a particular class. This is achieved by maximizing the likelihood of the observed data under the logistic model, typically using optimization techniques like gradient descent. Given a vector of input features, logistic regression calculates a linear combination of the features using the coefficients, then applies the logistic function to convert this combination into a probability. If the probability is above a certain threshold, typically 0.5, the instance is classified as belonging to one class; otherwise, it is classified as belonging to the other class. Returning to the plant classification example, a logistic regression algorithm could be used to classify plants based on features like size, color, and shape. It would find coefficients that best describe the relationship between these features and the likelihood of a particular class (such as "flower" or "not flower"). By applying the logistic function to these linear combinations, it could predict the probability of a plant being a flower, then assign a class based on the threshold. Overall, logistic regression is a versatile and widely used method for binary classification tasks, particularly when you want to understand how different features contribute to the probability of a certain outcome.

4.3.3. Support Vector Machine (SVM):

Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. Though we say regression problems as well its best suited for classification. The main objective of the SVM algorithm is to find the optimal hyperplane in an N-dimensional space that can separate the data points in different classes in the feature space. The hyperplane tries that the margin between the closest points of different classes should be as maximum as possible. The dimension of the hyperplane depends upon the number of features. If the number of input features is two, then the hyperplane is just a line. If the number of input features is three, then the hyperplane becomes a 2-D plane. It becomes difficult to imagine when the number of features exceeds three.

5. SENTIMENT ANALYSIS TASK

Sentiment analysis is a difficult multidisciplinary work that involves machine learning, web mining, and natural language processing. Although complicated, it may be broken down into the following tasks:

5.1. Implicit Categorization:

The task of identifying sentences as opinionated or non-opinionated is known as subjectivity classification. S = {s1,..., sn} represents a collection of sentences found in document D. Separating sentences used to objectively provide factual information (objective sentences set So) from sentences used to represent opinions and other types of subjectivity (subjective sentences set Ss) is the problem of subjectivity classification, where SsUSo=S.

5.2. Classifying Sentiment:

After determining if a sentence is opinionated, the next step is to determine the sentence's polarity, or whether it represents a positive or negative opinion. Regression, ranking, multi-class classification (very negative, negative, neutral, positive, or highly positive), and binary classification (positive or negative) are some methods of classifying sentiment. Opinion holder extraction and object feature extraction are optional subtasks, depending on how sentiment analysis is applied.

5.3. Complementary Tasks:

Finding the polarity of a sentence—that is, whether it expresses a positive or negative opinion—comes after identifying if it is opinionated. Some techniques for categorizing sentiment include regression, ranking, binary classification (positive or negative), multi-class classification (extremely negative, negative, neutral, positive, or highly positive), and regression. Depending on how sentiment analysis is used, opinion holder extraction and object feature extraction may be optional subtasks.

6. RESULTS ANALYSIS

6.1. Naïve Bayes Algorithm:

Navie bayes algorithms accuracy of 79.00%, Negative Tweets Precision of 79%, Positive Tweets Precision of 78%, Negative Tweets Recall of 76%, Positive Tweets Recall of 81%, Negative Tweets F1 score of 78%, Positive Tweets F1 score of 80%.

Naive Bayes Accuracy: 0.7885

Naive Bayes Classification Report:

	precision	recall	f1-score	support
1	0.79	0.76	0.78	975
2	0.78	0.81	0.80	1025
accuracy			0.79	2000
macro avg	0.79	0.79	0.79	2000
weighted avg	0.79	0.79	0.79	2000

Figure 1. Naive Bayes Model's Classification Report

Naïve Bayes algorithm has overall accuracy of 79 percent on the said dataset with weighted average and macro average same. It was evaluated further with following confusion matrix for gaining more insights. True Positives (TP) are 853 which indicates that the cases where the model correctly predicted a positive outcome, and the actual outcome was also positive. True Negatives (TN) are 777 which indicates that these are the cases where the model correctly predicted a negative outcome, and the actual outcome was also negative. False Positives (FP) are 172 which indicates that these are the cases where the model predicted a positive outcome, but the actual outcome was negative. This is often considered a "Type I error." Lastly, False Negatives (FN) are 198 which indicates that these are the cases where the model predicted a negative outcome, but the actual outcome was positive. This is often considered a "Type II error."

AUC, or Area Under the ROC Curve, is a metric used to evaluate the performance of a binary classification model like Naive Bayes. It provides a single number summary that indicates the model's ability to distinguish between two classes, regardless of the threshold chosen for classification. The ROC (Receiver Operating Characteristic) curve is a plot that shows the trade-off between the True Positive Rate (TPR, also known as Recall or Sensitivity) and the False Positive Rate (FPR) at different thresholds.

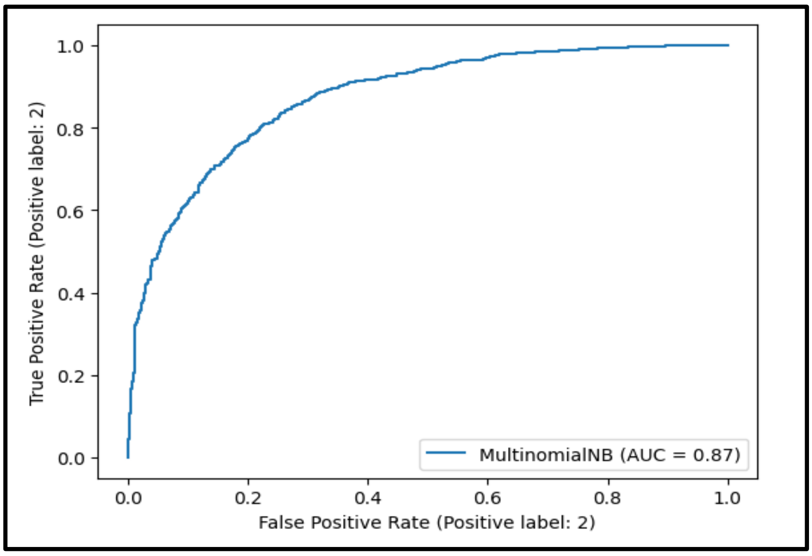

Figure 2. Naive Bayes Model's AUC Curve

The ROC curve represents how these rates vary as the classification threshold changes. A model with perfect classification ability would have an ROC curve that hugs the top-left corner of the plot, while a random guess would produce a diagonal line from the bottom-left to the top-right corner. An AUC value of 0.87 suggests that the model is performing quite well. An AUC of 0.87 means that, on average, there is an 87% chance that the model ranks a randomly chosen positive instance higher than a randomly chosen negative instance. The range for AUC is 0 to 1. A perfect classifier has an AUC of 1, while a random guess has an AUC of 0.5. An AUC below 0.5 indicates that the model is worse than random guessing. An AUC of 0.87 is considered quite good. It suggests that the model has a strong ability to distinguish between positive and negative cases across different thresholds. This metric is particularly useful because it is independent of the specific threshold used for classification. An AUC of 0.87 for your model indicates that it has a high ability to separate the positive and negative classes in sentiment analysis. This reflects a well-performing model with a good balance of sensitivity and specificity.

6.2. Logistic Regression Algorithm:

Logistic regression is a common algorithm used in machine learning for binary classification tasks, such as sentiment analysis where you classify text as having positive or negative sentiment.

Logistic Regression Accuracy: 0.815

Logistic Regression Classification Report:

	precision	recall	f1-score	support
1	0.82	0.80	0.81	975
2	0.81	0.83	0.82	1025
accuracy			0.81	2000
macro avg	0.82	0.81	0.81	2000
weighted avg	0.82	0.81	0.81	2000

Figure 3. Logistic Regression Model's Classification Report

Despite its name, logistic regression is actually a classification technique, not a regression method. In logistic regression, the goal is to predict the probability of a binary outcome (e.g., positive or negative sentiment) based on a set of input features. The algorithm works by finding a linear relationship between the input features and the log-odds of the binary outcome, which is then converted into a probability using the logistic function (also known as the sigmoid function). A confusion matrix is a tabular representation that shows the comparison between the predicted and actual outcomes. It helps to understand how often the model makes correct and incorrect predictions. True Positive (TP) are 900 which indicates that these represent the cases where the model correctly identified positive sentiment. True Negative (TN) are 730 which indicates that these represent the cases where the model correctly identified negative sentiment. False Positive (FP) are 180 which indicates that these represent the cases where the model incorrectly predicted positive sentiment for instances that actually had negative sentiment. False Negative (FN) are 180 which means that these represent the cases where the model failed to identify positive sentiment, predicting negative sentiment instead. The confusion matrix with the given values suggests that the sentiment analysis model performs reasonably well, with an accuracy of about 81.5%, precision of 83.3%, recall of 82.6%, and an F1-Score of 83.0%. These metrics indicate that the model is quite good at predicting sentiment, with a balanced performance across precision and recall.

An AUC (Area Under the ROC Curve) of 0.89 is a strong indicator of the model's ability to distinguish between two classes, in this case, positive and negative sentiment in a sentiment analysis task. The ROC (Receiver Operating Characteristic) curve is a graphical representation that illustrates the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) at various classification thresholds. The ROC curve plots TPR against FPR, demonstrating how these rates vary as you change the classification threshold. AUC is the area under this curve, providing a single metric that indicates the model's ability to differentiate between the classes. An AUC of 0.89 means that there is an 89% chance that the model will correctly distinguish a randomly chosen positive instance from a randomly chosen negative instance.

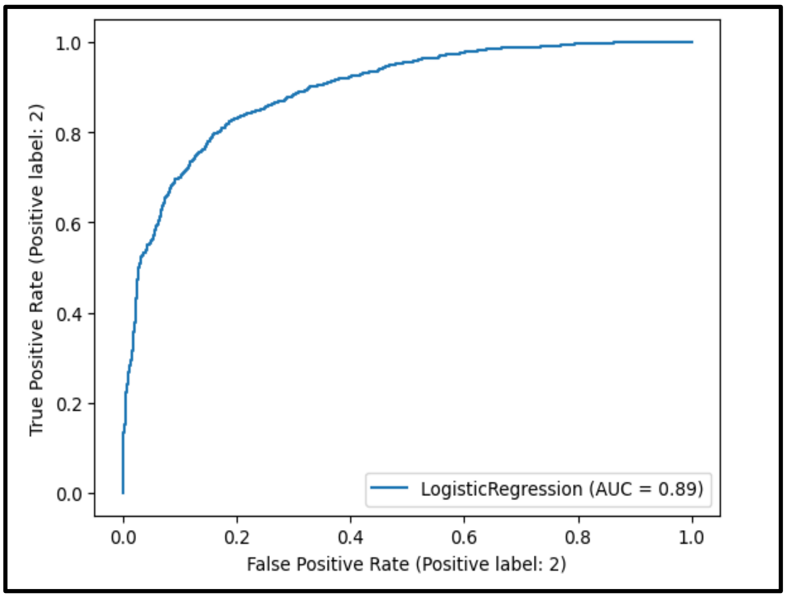

Figure 4. Logistic Regression Model's AUC Curve

This suggests a high level of separability between the two classes. An AUC of 0.89 is considered very good. It shows that the model has a strong ability to distinguish between positive and negative sentiment across a wide range of thresholds. The high AUC value implies that the model is likely effective at both identifying positive sentiment when it exists (high recall) and avoiding false positives (high precision), given its ability to maintain a balance between true positives and false positives at different thresholds. An AUC of 0.89 for a sentiment analysis model indicates strong performance. This high score suggests that the model can reliably distinguish between positive and negative sentiment, making it a suitable choice for this classification task.

6.3. Support Vector Machine (SVM) Algorithm:

Support Vector Machines (SVM) are a powerful set of supervised learning models that can be used for classification tasks, such as sentiment analysis, where the goal is to classify text as either positive or negative. SVM aims to find an optimal boundary, or hyperplane, that separates data points into distinct classes with the maximum margin. In the case of sentiment analysis, these classes could represent positive and negative sentiment. The key idea is to maximize the distance between this boundary and the nearest data points from each class. These nearest points are called support vectors. To train an SVM for sentiment analysis, you provide labeled examples of text (with known sentiment) and extract features from this text. The algorithm then finds the optimal hyperplane or boundary that separates positive and negative sentiment. Once trained, the SVM can classify new text by mapping it into the same feature space and determining which side of the boundary it falls on. Overall, SVM is a versatile and powerful algorithm for classification tasks, including sentiment analysis. Its ability to handle both linear and non-linear data, along with its robustness in high-dimensional spaces, makes it a valuable tool in this domain.

SVM Accuracy: 0.7995

SVM Classification Report:

	precision	recall	f1-score	support
1	0.80	0.78	0.79	975
2	0.80	0.81	0.81	1025
accuracy			0.80	2000
macro avg	0.80	0.80	0.80	2000
weighted avg	0.80	0.80	0.80	2000

Figure 5. SVM Model's Classification Report

True Positives (TP) are 860 which indicates that these are cases where the model correctly predicted positive sentiment, and the actual sentiment was indeed positive. True Negatives (TN) are 770 which means that these are cases where the model correctly predicted positive sentiment, and the actual sentiment was indeed positive. False Positives (FP) are 175 means that these are cases where the model correctly predicted positive sentiment, and the actual sentiment was indeed positive. False Negatives (FN) are 195 indicates that these are cases where the model correctly predicted positive sentiment, and the actual sentiment was indeed positive. The confusion matrix with these values suggests that the SVM model has a good performance, with an accuracy of about 81.5%, precision of 83.1%, recall of 81.5%, and F1-Score of 82.3%. These metrics show that the model has a fairly balanced performance, with a good ability to correctly classify both positive and negative instances, though there is still some room for improvement in reducing false positives and false negatives.

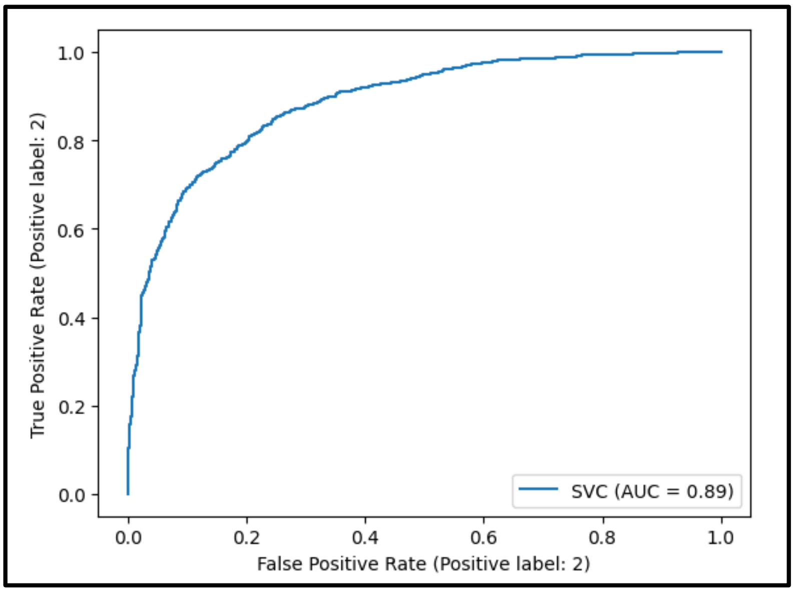

Figure 6. SVM Model's AUC Curve

AUC (Area Under the ROC Curve) is a single-value metric derived from the ROC curve. It represents the probability that a randomly chosen positive instance is ranked higher than a randomly chosen negative instance by the model. An AUC of 1 indicates a perfect model that flawlessly distinguishes between classes. An AUC of 0.5 means the model has no discriminative ability—it performs like random guessing. An AUC of 0.89 indicates that the SVM model has a high ability to separate positive and negative sentiment. Specifically, it suggests that there is an 89% chance that a randomly chosen positive instance will be ranked higher than a randomly chosen negative instance by the model. In the context of sentiment analysis, a high AUC like 0.89 is encouraging, as it suggests the SVM model is effective at classifying text into positive and negative sentiment. This value indicates that the model has a strong balance between identifying true positives while minimizing false positives, reflecting its overall ability to make accurate predictions. An AUC of 0.89

for an SVM model in sentiment analysis signifies that the model is performing well, with a strong capacity to differentiate between positive and negative sentiment.

Table 1. Comparative Analysis of Three ML Models

Models	Precision	Recall	F1 score	Accuracy
Navie Bayes	0.79	0.76	0.78	0.79
Logistic Regression	0.82	0.80	0.81	0.81
SVM	0.80	0.81	0.81	0.80

Precision measures the proportion of true positive predictions among all predicted positives. High precision indicates that the model has a low rate of false positives. Linear Regression has the best precision, suggesting it is less likely to classify a negative instance as positive compared to the other models. This might be beneficial in scenarios where false positives are more critical to avoid. Recall, also known as sensitivity or true positive rate, measures the proportion of true positives among all actual positives. High recall indicates that the model captures most of the positive instances. SVM leads in recall, indicating that it's more likely to catch positive instances, making it useful in scenarios where it's important not to miss positive cases. Naive Bayes has the lowest recall, suggesting it might miss more positive instances. The F1 score is the harmonic mean of precision and recall, providing a balanced metric. It is a good indicator of overall model performance, especially when you need a trade-off between precision and recall. Linear Regression and SVM are tied with the highest F1 score, suggesting they balance precision and recall well. This balanced performance can be useful in applications where both false positives and false negatives are costly. Accuracy measures the proportion of correct predictions among all predictions, providing a general sense of how often the model is correct. Linear Regression has the highest accuracy, suggesting it makes the most correct predictions on average. Naive Bayes has the lowest accuracy, but not by a significant margin.

Linear Regression has the best precision and accuracy, along with a strong F1 score, suggesting it may be the best overall performer, especially in scenarios where avoiding false positives is crucial. SVM has the highest recall and ties for the best F1 score, making it a good choice when catching positive instances is important, such as in medical diagnostics. Naïve Bayes has slightly lower metrics across the board but is still quite competitive. It might be the preferred choice if computational efficiency and simplicity are key factors. When choosing among these algorithms, consider the specific requirements of your task. If avoiding false positives is paramount, Linear Regression is a solid choice. If you can't afford to miss positive instances, SVM is likely the better option. Naive Bayes can be a good fit for simpler problems or when computational resources are limited.

7. CONCLUSION

From the analysis, it is clear that Linear Regression offers the best overall performance, leading in both precision and accuracy, and tying with SVM for the highest F1 score. This suggests that Linear Regression is an excellent choice when a balanced approach is needed and precision is critical to avoid false positives. SVM, on the other hand, excels in recall and also ties for the best F1 score, indicating that it's particularly useful in situations, where identifying as many positive instances as possible is crucial. This characteristic makes SVM a strong candidate for applications like medical diagnostics or spam detection, where high recall is essential. Naïve Bayes, while not leading in any single metric, still shows competitive performance across all metrics. Its lower complexity and computational efficiency make it a suitable option for simpler tasks or when resources are limited. Linear Regression algorithm is recommended for scenarios where accuracy and precision are most important, and false positives need to be minimized. It provides a balanced performance across all metrics. SVM algorithm is ideal for applications requiring high recall, ensuring that positive instances are not missed. It is suitable for complex, high-dimensional datasets. Naïve Bayes is the best suited for simpler problems or when computational efficiency is a priority. It can be used as a baseline model or in applications with relatively low risk from false positives or false negatives.

8. REFERENCES

1. Balci, S., Demirci, G. M., Demirhan, H., & Sarp, S. (2022). Sentiment Analysis Using State of the Art Machine Learning Techniques. In C. Biele, J. Kacprzyk, W. Kopeć, J. W. Owsiński, A. Romanowski, & M. Sikorski (Eds.), *Digital Interaction and Machine Intelligence* (pp. 34–42). Springer International Publishing. https://doi.org/10.1007/978-3-031-11432-8_3
2. Dang, N. C., Moreno-García, M. N., & De la Prieta, F. (2020). Sentiment Analysis Based on Deep Learning: A Comparative Study. *Electronics*, *9*(3), Article 3. https://doi.org/10.3390/electronics9030483
3. Jemai, F., Hayouni, M., & Baccar, S. (2021). Sentiment Analysis Using Machine Learning Algorithms. *2021 International Wireless Communications and Mobile Computing (IWCMC)*, 775–779. https://doi.org/10.1109/IWCMC51323.2021.9498965
4. Kalangi, R. R., Maloji, S., Tejasri, N., Chand, P. P., & Ch, V. Priya. (2021). Sentiment Analysis using Machine Learning. *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 116–121. https://doi.org/10.1109/ICAC3N53548.2021.9725499
5. Revathy, G., Alghamdi, S. A., Alahmari, S. M., Yonbawi, S. R., Kumar, A., & Anul Haq, M. (2022). Sentiment analysis using machine learning: Progress in the machine intelligence for data science. *Sustainable Energy Technologies and Assessments*, *53*, 102557. https://doi.org/10.1016/j.seta.2022.102557
6. Singh, J., Singh, G., & Singh, R. (2017). Optimization of sentiment analysis using machine learning classifiers. *Human-Centric Computing and Information Sciences*, *7*(1), 32. https://doi.org/10.1186/s13673-017-0116-3
7. Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, *55*(7), 5731–5780. https://doi.org/10.1007/s10462-022-10144-1