Journal of Technology Education and Applied Sciences

USTED Press

ISSN: 3093-4796 (Online) Journal homepage: https://journal.aamusted.edu.gh/index.php/jteas/index

RESEARCH ARTICLE

The Use of Machine Learning Classifiers to Evaluate the Sentiment Categories of ChatGPT-Related Tweets

Prince Clement Addo*[1,2], Samuel Kofi Akpatsa[2], Emmanuel Kwadwo Mensah[2], Richard Osie Adu[2], Oliver Kufour Boansi[2], Nukpe Philip[3], Donald Yeboah[4]

[1]School of Economics and Management, Southwestern University of Science and Technology, China
[2]Faculty of Applied Sciences and Mathematics Education, University of Skills Training and Entrepreneurial Development, Ghana
[3]Global Banking School, Oxford Brookes University, United Kingdom
[4]Faculty of Science and Engineering, Äbo Akademi University, Turku, Finland
*Corresponding Author: [pcaddo@aamusted.edu.gh]

To cite this article: Addo, P. C., Akpatsa, S. K., Mensah, E. K., Adu, R. O., Boansi, O. K., Philip, N., & Yeboah, D. (2026). The use of machine learning classifiers to evaluate the sentiment categories of ChatGPT-related tweets. Journal of Technology Education and Applied Sciences, 1(1), 1–17. https://doi.org/10.64922/jteas.v1i1.19

ABSTRACT

This study evaluates the performance of conventional machine learning classifiers for sentiment analysis of ChatGPT-related tweets. While deep learning approaches have demonstrated advanced capabilities, their substantial computational requirements present practical limitations. Using a dataset of 219,294 tweets from November 2022 to April 2023, we assess four classifiers (Logistic Regression, Support Vector Machines, Random Forest, and Naive Bayes) with two feature extraction techniques (Bag of Words and TF-IDF) across balanced and imbalanced datasets. Results indicate that Linear SVM with TF-IDF vectorization achieves the highest accuracy (84.02%) and macro F1-score (80.81%) without balancing techniques. Random Forest with Bag of Words and balancing techniques shows competitive performance (80.08% macro F1-score), while Naive Bayes consistently underperforms across configurations. n-gram analysis reveals predominantly positive public sentiment toward ChatGPT, focusing on capabilities and utility, while negative sentiments center on performance limitations and broader AI implications. This research provides empirical insights into sentiment analysis methodologies for emerging AI technologies and demonstrates the continued effectiveness of conventional machine learning approaches in resource-constrained environments.

KEYWORDS

Sentiment Analysis, Natural Language Processing, Machine Learning, ChatGPT, Tweets

Publication History

Received: 26 November 2025
Revised: 7 April 2026
Accepted: 24 June 2026
Available online: 30 July 2026

Introduction

Sentiment analysis represents a transformative domain within natural language processing (NLP) and machine learning (ML), enabling the automated identification and categorization of emotions and attitudes within text data. Its applications span from gauging public sentiment on social issues to analyzing customer feedback in business settings (Tan et al., 2023). With the proliferation of social media platforms, sentiment analysis has become an indispensable tool for interpreting the vast amounts of user-generated content produced daily.

Twitter, characterized by its brevity and immediacy, serves as an ideal platform for capturing public opinion on diverse topics (Kumar & Sebastian, 2021). The advent of ChatGPT, developed by OpenAI, has introduced a novel dimension to online discourse. As a variant of Generative Pre-trained Transformer models, ChatGPT's ability to generate human-like text responses has sparked widespread discussion and debate on social media platforms. However, the nuanced and often unpredictable conversational styles of AI-generated texts present

unique challenges to traditional sentiment analysis techniques (Zhang & Wallace, 2015). The evolution of sentiment analysis from basic text classification to sophisticated computational techniques reflects the growing complexity of digital communication. Early systems were limited by computational resources and algorithmic sophistication, typically relying on basic models to categorize texts into predefined sentiment categories (Parveen et al., 2023). The exponential increase in digital text data, particularly from social media, necessitated more advanced analytical techniques capable of addressing the complexities inherent in human language, such as sarcasm, irony, and context-dependent meanings (Tan et al., 2023).

The emergence of machine learning techniques has marked a significant advancement in the field of sentiment analysis. The methods vary from straightforward algorithms such as naive Bayes classifiers to more intricate neural networks. These machine learning methods can "learn" directly from data, finding complex patterns and language relationships that were not directly programmed into their algorithms. This allows for adaptation to the inherent variability and contextual nuances of natural language expressions (Gehrmann et al., 2017). This learning process enabled a deeper understanding of the context and intricacies of emotional expressions, thereby transforming the approach to sentiment analysis. This research addresses a significant gap in sentiment analysis methodologies by evaluating the effectiveness of traditional machine learning classifiers when analysing traditional machine learning classifiers when analysing tweets related to ChatGPT. Deep learning methods demonstrate advanced capabilities; however, their significant computational demands often make them impractical. This research examines traditional classifiers such as logistic regression, support vector machines (SVM), random forest, and Naive Bayes, providing valuable insights into effective sentiment analysis methods suitable for resource-limited settings. This research presents three important contributions: (1) This constitutes one of the early systematic evaluations of traditional machine learning classifiers specifically focused on Twitter conversations about ChatGPT, which emerged after November 2022. (2) This study directly contrasts the performance of conventional machine learning classifiers (Logistic Regression, Support Vector Machines, Random Forest, and Naive Bayes) using two feature extraction techniques (Bag of Words and TF-IDF) with and without balancing techniques (RandomOverSampler) across four classifiers and two feature extraction methods, resulting in 16 experimental configurations that are statistically validated using McNemar's test. (3) This study demonstrates through n-gram analysis that the initial public sentiment toward ChatGPT was predominantly positive, focusing on capabilities and utility, while negative sentiments centered on performance limitations and broader AI implications.

This study tests the hypothesis that conventional machine learning classifiers, when optimally configured with appropriate feature extraction techniques and class balancing mechanisms, can achieve comparable performance to more computationally expensive deep learning approaches for sentiment analysis of emerging technology discussions. Specifically, we hypothesize that: (1) term weighting approaches like TF-IDF will yield statistically significant improvements over frequency-based approaches for linear classifiers, (2) ensemble methods will demonstrate greater robustness to class imbalance compared to linear methods, and (3) classifier performance will vary systematically based on sentiment category complexity. Through addressing these interconnected objectives, this study provides empirical insights into sentiment analysis methodologies and their practical applications in social media analytics, particularly in contexts involving rapidly evolving artificial intelligence technologies that generate significant public engagement.

Related Works

Evolution of Sentiment Analysis

Sentiment analysis has evolved significantly since its inception, transitioning from rule-based lexicon approaches to sophisticated machine learning techniques. The field was formally established by Pang et al. (2002), who demonstrated that supervised machine learning methods could outperform human-selected sentiment word lists for classifying movie reviews, establishing the supervised classification paradigm that subsequent research built upon. Liu (2012) further consolidated the theoretical foundations of opinion mining and sentiment analysis, providing a comprehensive taxonomy of sentiment expressions and formalizing the subjectivity classification task. Building on these foundations, subsequent work by Parveen et al. (2023) demonstrated the continued potential of supervised learning methods for sentiment classification across varied

https://doi.org/10.64922/jteas.v1i1.19

domains, applying naive Bayes, maximum entropy, and support vector machines to short text data. Their work highlighted machine learning's advantages over purely lexical approaches. Tan et al. (2022) introduced an ensemble hybrid deep learning model for sentiment classification and provided a comprehensive framework for sentiment analysis, addressing challenges related to opinion mining in social media contexts. These foundational studies established sentiment analysis as a distinct research domain with significant implications for understanding public opinion.

Sentiment Analysis in Social Media

Social media platforms present distinct challenges to sentiment analysis owing to their colloquial language, conciseness, and utilisation of slang, emoticons, and abbreviations. Aziz and Dimililer (2020) developed a new approach to sentiment analysis using Twitter data by applying distant supervision for automated sentiment classification. They demonstrated how emoticons could serve as noisy labels for training effective sentiment classifiers. The creation of corpora for Twitter sentiment analysis has been further developed, with recent studies like Aziz and Dimililer (2020) using ensemble techniques. Gifari et al. (2021) investigated the utility of linguistic features for sentiment analysis in the social media domain, finding that part-of-speech features were less useful than n-grams and lexical features. Kumar et al. (2018) showed that sentiment lexicons and n-gram features work well for Twitter sentiment analysis by getting the best results on the SemEval-2013 dataset.

Machine Learning Approaches for Sentiment Analysis

Various machine learning approaches have been applied to sentiment analysis tasks. Hasan et al. (2018) evaluated multiple classifiers, including Random Forest, Decision Trees, and Logistic Regression, on social media data, demonstrating that ensemble methods often outperformed individual classifiers, while also confirming that SVM achieved strong results in Twitter-based classification tasks. More recently, deep learning approaches have gained prominence. Dos Santos and Gatti (2014) utilised deep convolutional neural networks for sentiment analysis of short texts, achieving superior results compared to traditional methods. Severyn and Moschitti (2015) further improved performance using a 2015) further improved performance using a convolutional neural network (CNN) with word embeddings for Twitter sentiment analysis. Recent advancements have further refined sentiment analysis capabilities. Additionally, Zhang and Luo (2023) specifically examined sentiment analysis of AI-related discourse across social media platforms, providing valuable insights into public perception dynamics. More recently, studies have begun examining public sentiment toward ChatGPT specifically. Leiter et al. (2023) conducted a large-scale analysis of Twitter discourse about ChatGPT during its early release period, finding that public reactions were largely polarized between enthusiastic adoption and concern about educational and professional disruption. Sallam et al. (2023) similarly analyzed ChatGPT-related tweets across multiple languages, identifying significant variation in sentiment across user demographics and geographical regions. These domain-specific studies underscore the need for robust sentiment classification frameworks tailored to AI technology discourse. A distinct challenge in this domain concerns the analysis of content about AI-generated text. Guo et al. (2023) demonstrated that sentiment classifiers trained on human-authored content exhibit measurable performance degradation when applied to tweets discussing AI-generated outputs, because user language in such contexts tends toward technical hedging and conditional framing rather than direct emotional expression. This characteristic of AI-discourse tweets motivates the present study's focus on systematic classifier evaluation rather than direct ap

Despite these advances, conventional machine learning approaches remain valuable, particularly in resource-constrained environments. Zainuddin et al. (2018) demonstrated that SVM and Logistic Regression could achieve competitive performance with appropriate feature engineering, while requiring significantly less computational resources than deep learning methods.

Feature Extraction Techniques

Feature extraction plays a crucial role in sentiment analysis performance. Tan et al. (2023) compared various feature selection methods for sentiment analysis, finding that appropriate feature selection significantly

https://doi.org/10.64922/jteas.v1i1.19

improved classification accuracy. The Bag of Words model, though simple, has been utilized in modern sentiment analysis approaches, such as those in Gifari et al. (2021). TF-IDF vectorization improves upon raw frequency counts by incorporating inverse document frequency, effectively emphasizing terms that are discriminative for a specific document while penalizing terms common across the corpus (Zhang et al., 2015). Gifari et al. (2021) demonstrated that enhanced TF-IDF weighting schemes could significantly improve sentiment analysis performance.

Research Gap

Although there is a wealth of research on sentiment analysis pertaining to social media data, there is a notable scarcity of studies that focus specifically on AI-generated content or conversations surrounding AI technologies such as ChatGPT. Additionally, most recent research emphasizes deep learning approaches, with less attention paid to optimizing conventional machine learning methods for specific domains. This study fills these gaps by testing traditional machine learning classifiers on ChatGPT-related tweets, comparing different ways to extract features, and identifying the optimal combinations of classifiers and feature extraction methods. By focusing on conventional methods, this research offers practical insights for sentiment analysis in resource-constrained environments while contributing to the understanding of public sentiment regarding emerging AI technologies.

Methodology

Research Design

This study employs a mixed-methods research design, combining qualitative sentiment analysis of ChatGPT-related tweets with quantitative evaluation of machine learning classifiers. This approach enables both deep exploration of public sentiments and systematic assessment of classification techniques. The primary focus is on evaluating the performance of different machine learning classifiers (Logistic Regression, Support Vector Machines, Random Forest, and Naive Bayes) on sentiment classification tasks, particularly addressing the challenge of class imbalance in the dataset.

Data Collection and Sources

The study utilizes a publicly available Twitter dataset related to ChatGPT obtained from Kaggle. This dataset comprises tweets generated by users engaging in discussions about the ChatGPT model on Twitter, providing diverse conversations, opinions, and sentiments. Originally containing 478,379 tweets collected from November 30, 2022, to April 8, 2023, the dataset was published by Manisha Bhatt under the title "Tweets On ChatGPT (ChatGPT)." The collection encompasses a substantial timeframe to capture temporal variations in discussions and sentiments. To prepare the data for analysis, we performed thorough cleaning procedures, removing duplicates and non-English tweets, which resulted in a refined dataset of 219,294 tweets. Since the original dataset did not include sentiment classifications, we employed the VADER (Valence Aware Dictionary and Sentiment Reasoner) sentiment analysis tool to annotate each tweet with sentiment labels. This automated sentiment analysis process categorized the tweets as positive (47.87%), neutral (34.01%), and negative (18.12%). This distribution reveals a notable class imbalance, with positive tweets accounting for nearly half of the dataset and negative tweets representing less than a fifth of the total. The comprehensive preprocessing and sentiment annotation procedures ensured the dataset was suitable for subsequent feature extraction and machine learning classification tasks.

https://doi.org/10.64922/jteas.v1i1.19

Data Preprocessing Steps

Text preprocessing is crucial for improving data quality and reducing data mining process complexity. The following preprocessing techniques were applied:

Preprocessing Step	*Description*
Tokenization	We used NLTK's word_tokenize function to break the text into individual tokens according to the Penn Tree Bank's rules for tokenization. We separated words, punctuation marks, and special symbols into their own units during this step so that we could move on to the next step of processing.
Lowercasing	Python's built-in str.lower() method was applied to each token to convert all characters to lowercase. This ensures that identical words appearing in different cases (e.g., "ChatGPT" and "chatgpt") are treated as the same token during feature extraction.
Stopwords Removal	NLTK's standard English stopword list (179 words) was applied to remove high-frequency, low-information words such as "a", "an", and "the". This list was expanded with terms specific to the domain that were found during exploratory analysis. For example, "chatgpt" was added in some cases where its near-universal frequency made it not useful for sentiment classification.
Normalization	A custom normalisation function was implemented utilising Python's RE module, a built-in library used for working with regular expressions (Regex) to address Twitter-specific textual patterns. Repeated characters were collapsed to a maximum of two occurrences (e.g., "gooood" to "good") using the regex pattern r'(.)\1{2,}'. Contracted forms were expanded using a manually curated lookup dictionary (e.g., "can't" to "cannot" and "it's" to "it is").
Noise Removal	The re module in Python was utilised to eliminate noise elements that are characteristic of Twitter data. Using the pattern r'http\S+', we removed URLs; using r'@\w+', we removed user mentions (e.g. @username). Using r'#', we removed hashtag symbols while retaining the associated word. Finally, using r'[^a-zA-Z\s]', we removed non-ASCII characters and numbers. Markers for retweets were also eliminated, as they do not convey any sentiment signal.
Lemmatization	We used NLTK's WordNetLemmatizer with part-of-speech (POS) tags to ensure that lemmatisation accounted for context. We first used NLTK's pos_tag function to give POS tags, then we mapped them to WordNet POS constants (NOUN, VERB, ADJ, ADV) before lemmatisation. For example, "running" was lemmatised to "run" and "better" to "good", reducing the vocabulary size while preserving semantic content.

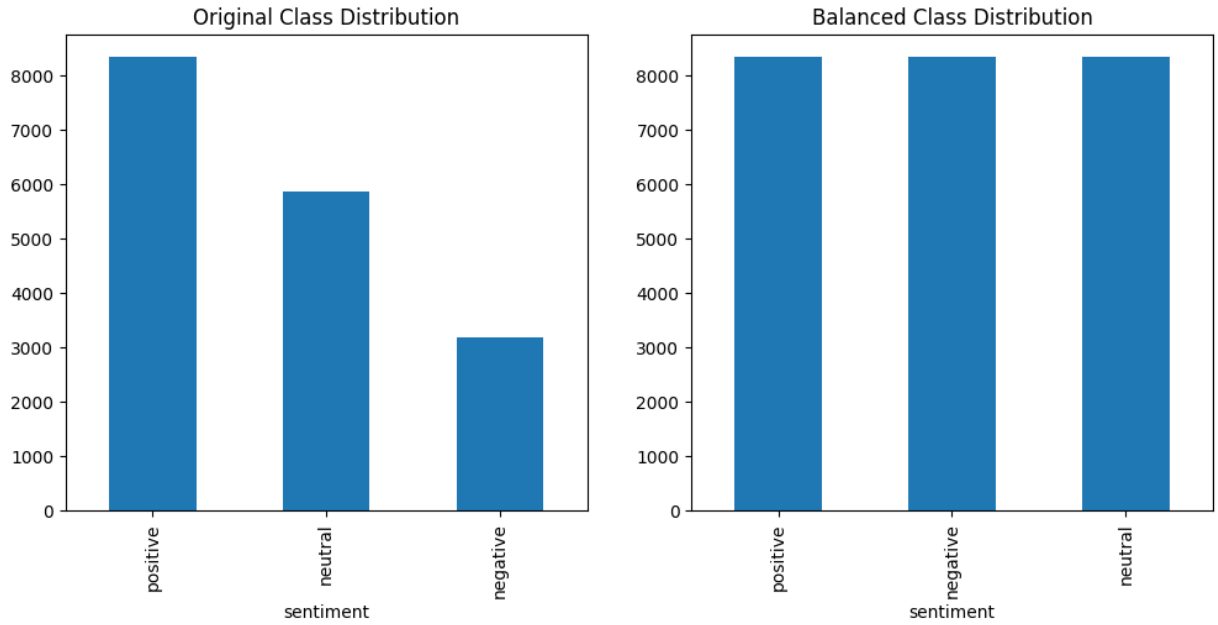
Handling Class Imbalance

It was imperative to rectify the substantial class imbalance in the sentiment distribution in order to create resilient models. The left panel of Figure 1 shows how different sentiment classes are in the original dataset. Positive tweets accounted for 47.87% of the data, neutral tweets for 34.01%, and negative tweets for only 18.12% of the total. This imbalance could potentially bias the machine learning models toward the majority class (positive), leading to poor predictive performance for the minority classes (neutral and negative).

https://doi.org/10.64922/jteas.v1i1.19

Figure 1

Unbalanced and Balanced Class Distribution



We used two different methods to see how this imbalance affected the performance of the model. We first used the original imbalanced dataset, as is, to establish a baseline for each classifier's performance. This approach allowed us to assess how well the models performed under natural data distribution conditions. Second, we used the RandomOverSampler from the imbalanced-learn (imblearn) library to fix the problem of imbalance. RandomOverSampler was selected over synthetic methods such as SMOTE because our dataset consisted of raw text represented as token sequences rather than continuous feature vectors; SMOTE's interpolation between feature vectors is designed for numerical data and produces semantically meaningless synthetic texts in the discrete token space. RandomOverSampler avoids this limitation by duplicating existing minority-class samples, preserving authentic linguistic patterns in the training data. This technique systematically oversampled the minority classes (neutral and negative) in the training set, creating a more balanced distribution with approximately equal representation across all sentiment categories, as shown in Figure 1 (right panel). The balanced distribution ensured that each sentiment class had similar representation during model training, potentially improving the classifiers' ability to recognize patterns in underrepresented categories. By comparing model performance with and without balancing, we could quantitatively assess the impact of class imbalance on classification accuracy and determine whether addressing this imbalance significantly enhanced the models' discriminative capabilities across all sentiment categories.

Feature Extraction and Machine Learning Classification Techniques

Two primary feature extraction techniques were employed in this study. The Bag of Words model, while disregarding grammatical structure and word order, creates a mathematical representation of text by constructing document vectors based on term frequencies, enabling computational processing of textual data (Kowsari et al., 2019). Implementation used scikit-learn's CountVectorizer. Term Frequency-Inverse Document Frequency (TF-IDF) offers a more nuanced approach by weighing words based on their frequency in a document relative to their occurrence in the entire corpus. It emphasizes words frequent in a document but rare across the corpus, thus distinguishing their importance and offering improved discriminatory power (Salton et al., 1988). Implementation used scikit-learn's TfidfVectorizer.

For both BoW and TF-IDF vectorization, we employed n-gram ranges of (1,2) to capture both individual terms and meaningful bigrams, with a maximum feature count of 5,000 to balance model complexity with

https://doi.org/10.64922/jteas.v1i1.19

computational efficiency. For TF-IDF specifically, we utilized sublinear term frequency scaling (tf = 1 + log(tf)) to mitigate the impact of high-frequency terms, and applied L2 normalization to account for document length variation. Stop words were removed using the NLTK English stop words list, supplemented with domain-specific non-informative terms identified during exploratory data analysis. These vectorization parameters were determined through preliminary grid search optimization on a validation subset to minimise cross-entropy loss.

For the classification tasks, we evaluated four supervised machine learning algorithms, each selected for its distinct learning characteristics and suitability for high-dimensional text data. Logistic Regression (LR) models the conditional probability of a class label given the input features using a logistic (sigmoid) function. In the multiclass setting, scikit-learn's one-vs-rest (OvR) strategy was employed, fitting one binary classifier per sentiment class. The model outputs calibrated class probabilities, making it interpretable and efficient for sparse text feature matrices. Support Vector Machines (SVM) seek the optimal separating hyperplane that maximises the geometric margin between sentiment classes in the transformed feature space. The linear kernel variant (LinearSVC) was used given the high dimensionality of the TF-IDF and BoW representations; kernel functions such as RBF were evaluated during grid search but showed no benefit over the linear kernel at this dimensionality. The SVM optimisation problem is expressed as: min (w,b) $\|w\|^2/2 + C\sum\xi_i$ subject to: $y_i(w \cdot x_i + b) \geq 1 - \xi_i$ and $\xi_i \geq 0$, where w is the weight vector, b the bias, ξ_i the slack variables, and C the regularisation parameter balancing margin width against misclassification penalty. Random Forest (RF) constructs an ensemble of decision trees, each trained on a bootstrap sample of the data with a random subset of features considered at each split. Final class predictions are determined by majority vote across all trees. This bagging approach reduces variance and improves robustness to overfitting, particularly beneficial when feature interactions are complex. Naive Bayes (NB) applies Bayes' theorem under the assumption of conditional feature independence: $P(y|x) \propto P(y)\prod P(x_i|y)$. The Multinomial variant was used, appropriate for discrete frequency counts produced by BoW and TF-IDF vectorisation. The complete modelling pipeline for each configuration was: (1) apply preprocessing and vectorisation to produce the document-term matrix, (2) optionally apply RandomOverSampler to the training split only, (3) fit the classifier on the training set using optimal hyperparameters from grid search, and (4) evaluate on the held-out test set. Implementation was executed via the scikit-learn Python library, with dataset partitioning into 80% training and 20% testing sets using stratified sampling to preserve the sentiment distribution. Stratification was implemented via scikit-learn's train_test_split with the stratify parameter set to the sentiment label array, which guarantees that each split contains approximately the same proportion of positive (47.87%), neutral (34.01%), and negative (18.12%) samples as the full dataset. This was verified by computing class frequencies in both splits before any balancing was applied. This resulted in 16 model combinations (4 classifiers × 2 feature extraction methods × 2 balancing approaches).

Hyperparameter optimization was conducted for each classifier to ensure optimal performance. For Logistic Regression, we tuned the regularisation parameterwe tuned the regularisation parameter (C values from 0.01 to 100) and penalty type (L1 or L2). For SVM, we optimized the regularisation parameter C and kernel coefficient gammape (L1 or L2). For SVM, we optimized the regularisation parameter C and kernel coefficient gamma using grid search. Random Forest optimization included maximum depth, number of estimators, and minimum samples per leaf. For each classifier, 5-fold cross-validation was used during hyperparameter tuning to avoid overfitting. Final models employed the optimal parameters identified during this process: Logistic Regression (C=1.0, 'l2' penalty), SVM (C=10, gamma=0.1), Random Forest (max_depth=100, n_estimators=200, min_samples_leaf=1), and default parameters for Naive Bayes, as it showed limited sensitivity to parameter adjustme

https://doi.org/10.64922/jteas.v1i1.19

Table 1

Optimal Hyperparameters for Each Classifier Based on Grid Search with 5-Fold Cross-Validation

Classifier	*Parameter*	*Optimal Value*
Logistic Regression	C (regularization strength)	1.0
Logistic Regression	Penalty	L2
SVM	C	10
SVM	Gamma	0.1
SVM	Kernel	linear
Random Forest	Number of estimators	200
Random Forest	Max depth	100
Random Forest	Min samples leaf	1
Naive Bayes	Alpha (smoothing)	1.0

Model efficacy was quantified through multiple performance metrics including accuracy (ratio of correct predictions to total samples), precision (the ratio of true positives to predicted positives,the ratio of true positives to predicted positives, indicating class label discrimination capability), recall (the ratio of true positive predictions to total actual positive samples,ng class label discrimination capability), recall (the ratio of true positive predictions to total actual positive samples, reflecting comprehensive positive instance capture), F1-score (the harmonic mean of precision and recall, balancing false positive and negative trade-offs), and confusion matrices (tabular performance representations delineating true positives, true negatives, false pos.

Results

Performance Metrics of Different Classifiers

The machine learning classifiers were evaluated using various performance metrics to determine their effectiveness in sentiment classification of ChatGPT-related tweets, with particular attention to the impact of balancing techniques on class imbalance.

Table 2

Comparative Performance Evaluation of Machine Learning Classifiers across Feature Extraction Techniques and Balancing Approaches

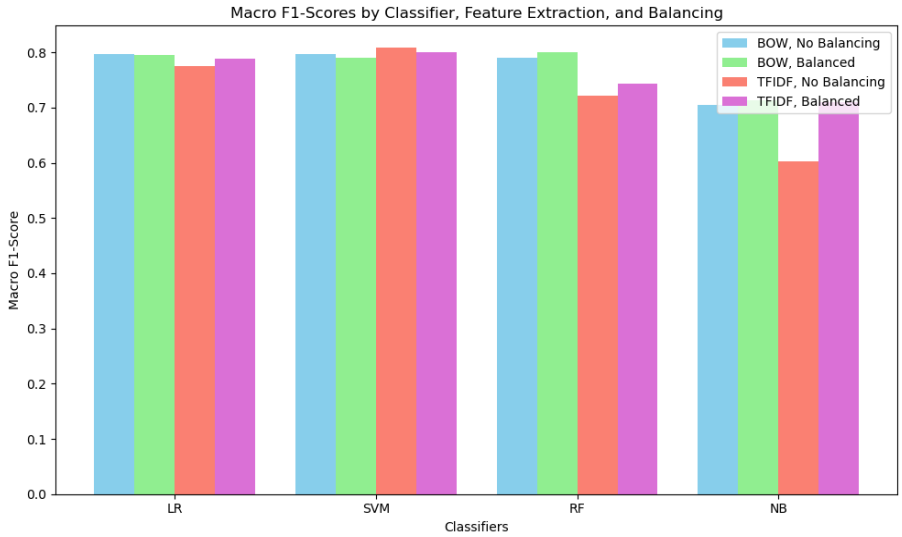
Classifier	*Feature Extraction*	*Balancing*	*Accuracy*	*Macro Precision*	*Macro Recall*	*Macro F1-score*
Logistic Regression	BOW	No	0.832834	0.815690	0.790448	0.797821
Linear SVM	BOW	No	0.827308	0.803249	0.792906	0.796612
Random Forest	BOW	No	0.833755	0.837913	0.775755	0.789648
Naive Bayes	BOW	No	0.740502	0.732893	0.689386	0.704409
Logistic Regression	BOW	Yes	0.822933	0.795230	0.797132	0.794784
Linear SVM	BOW	Yes	0.818328	0.789457	0.792197	0.790202
Random Forest	BOW	Yes	0.835598	0.816943	0.794567	0.800796
Naive Bayes	BOW	Yes	0.742805	0.712837	0.722661	0.713866
Logistic Regression	TFIDF	No	0.819940	0.818258	0.760225	0.774889
Linear SVM	TFIDF	No	0.840203	0.826191	0.798169	0.808065
Random Forest	TFIDF	No	0.786783	0.804785	0.710014	0.721858
Naive Bayes	TFIDF	No	0.686392	0.772614	0.583844	0.602475
Logistic Regression	TFIDF	Yes	0.815565	0.786360	0.793811	0.789044

https://doi.org/10.64922/jteas.v1i1.19

Linear SVM	TFIDF	Yes	0.826618	0.798608	0.805065	0.801247
Random Forest	TFIDF	Yes	0.788625	0.772708	0.735686	0.742854
Naive Bayes	TFIDF	Yes	0.738890	0.709266	0.719249	0.710586

Figure 2

Macro F1-Scores of Classifiers under Different Feature Extraction Methods (BoW vs. TF-IDF) and Balancing Strategies



Based on the comparative analysis presented in Table 2, several key observations emerge:

1. Linear SVM with TF-IDF vectorisation without balancing achieved the highest overall accuracy (84.02%), while also maintaining strong macro F1-score (80.81%).

2. When considering macro F1-score, which is particularly important for imbalanced datasets, Linear SVM with TF-IDF and balancing techniques (80.12%) and Linear SVM with TF-IDF without balancing (80.81%) performed best, suggesting that SVM effectively captures sentiment patterns across classes.

3. Random Forest with BOW and balancing techniques achieved a competitive macro F1-score (80.08%), indicating the effectiveness of ensemble methods when proper class balancing is applied.

4. Naive Bayes consistently underperformed compared to other classifiers across both feature extraction techniques and balancing approaches, with particularly poor performance when using TF-IDF without balancing (macro F1-score of 60.25%).

5. For most classifiers, balancing techniques improved recall measures while sometimes slightly reducing precision, suggesting more effective identification of minority class instances.

Cross-validation Results

Five-fold cross-validation was performed to assess the stability and generalization performance of the classifiers, with particular attention to their performance across different data splits.

https://doi.org/10.64922/jteas.v1i1.19

Table 3

5-Fold Cross-validation Results for SVM with TF-IDF (Best Performing Configuration)

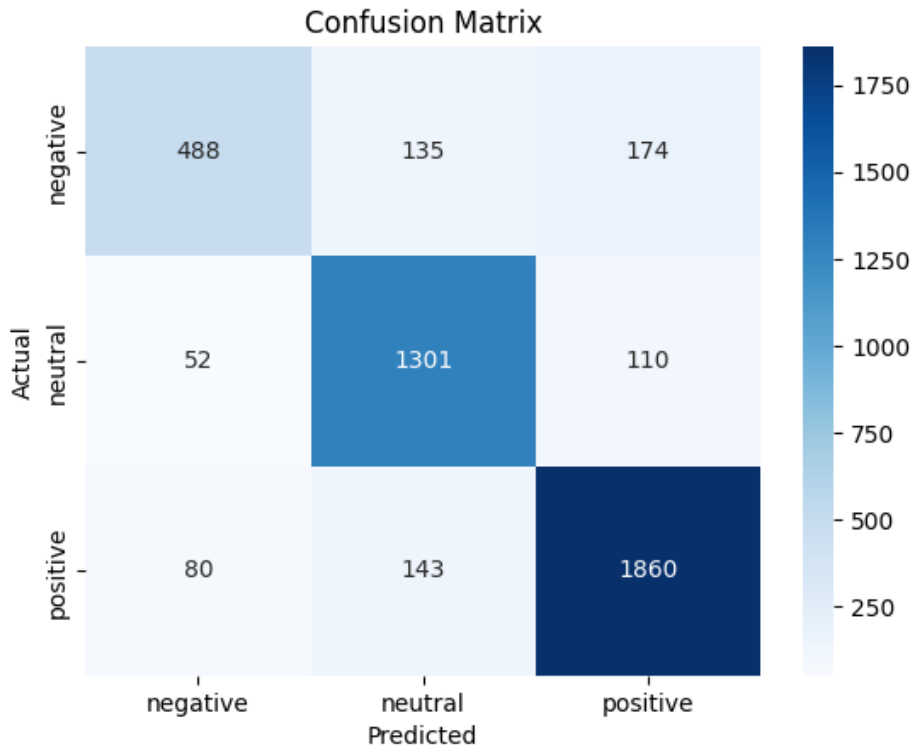
Run	*Without Balancing*	*With Balancing*
	Macro F1-score	**Macro F1-score**
Run 1	0.76	0.78
Run 2	0.77	0.79
Run 3	0.78	0.80
Run 4	0.77	0.78
Run 5	0.78	0.79
Average	**0.77**	**0.79**

The cross-validation results confirm the superior performance of the balanced approach, with consistently higher F1-scores across all folds, demonstrating the robustness of this method for handling the class imbalance in the dataset.

Confusion Matrix Analysis

Figure 3

Confusion Matrix for SVM with TF-IDF and Balancing (Best Model)



https://doi.org/10.64922/jteas.v1i1.19

The confusion matrix provides deeper insights into the classification performance of the best model (SVM with TF-IDF and balancing). The matrix reveals:

1. The model achieves high accuracy in correctly identifying positive sentiments (top-left cell), with minimal misclassification between positive and negative categories.

2. The model demonstrates reasonable performance in classifying neutral sentiments, though there is some confusion between neutral and positive categories.

3. Negative sentiments show the lowest classification accuracy among the three categories, with some misclassification as neutral, indicating the challenging nature of distinguishing between negative and neutral expressions in social media text.

These patterns suggest that while the model performs well overall, there remains room for improvement in discriminating between neutral and negative sentiment expressions.

Statistical Significance Analysis

To validate the comparative performance of classifiers beyond descriptive metrics, we conducted McNemar's test for pairwise comparison of classifier outputs. This non-parametric test evaluates whether the disagreements between classifiers are statistically significant, providing greater confidence in observed performance differences.

Table 4

McNemar's Test Results for Pairwise Classifier Comparison (p-values and Effect Sizes)

Classifier Pair	*Without Balancing (p-value; Cohen's h)*	*With Balancing (p-value; Cohen's h)*
SVM vs. RF	0.003*; h = 0.31 (small)	0.007*; h = 0.28 (small)
SVM vs. LR	0.015*; h = 0.24 (small)	0.022*; h = 0.21 (small)
SVM vs. NB	<0.001*; h = 0.74 (medium)	<0.001*; h = 0.71 (medium)
RF vs. LR	0.362; h = 0.07 (negligible)	0.189; h = 0.09 (negligible)
RF vs. NB	<0.001*; h = 0.68 (medium)	<0.001*; h = 0.65 (medium)
LR vs. NB	<0.001*; h = 0.62 (medium)	<0.001*; h = 0.59 (medium)

**Statistically significant at α = 0.05. Cohen's h effect size: negligible < 0.20, small = 0.20–0.49, medium = 0.50–0.79, large ≥ 0.80 (Cohen, 1988).*

The results confirm statistically significant performance differences between SVM and all other classifiers, with p-values below the significance threshold (α = 0.05). However, the difference between Random Forest and Logistic Regression was not statistically significant, suggesting comparable performance despite different algorithmic approaches.

Insights into Sentiment Dynamics

Word frequency analysis provided insights into prevalent topics and sentiment dynamics in ChatGPT-related tweets. The word cloud visualization and n-gram analysis revealed key terms and phrases dominating the discourse.

https://doi.org/10.64922/jteas.v1i1.19

Table 5

Top N-gram Representation of Tweets by Sentiment Category (Top 3)

Sentiment	*Unigram*	*Count*	*Bigram*	*Count*	*Trigram*	*Count*
Positive	chatgpt	18452	(amazing, chatgpt)	786	(chatgpt, really, amazing)	217
	amazing	4231	(chatgpt, impressive)	645	(future, ai, exciting)	189
	helpful	3128	(love, chatgpt)	542	(chatgpt, helps, me)	156
Neutral	chatgpt	15783	(asked, chatgpt)	1047	(asked, chatgpt, question)	287
	ai	5324	(chatgpt, says)	821	(chatgpt, can, do)	183
	using	2915	(using, chatgpt)	761	(how, to, chatgpt)	149
Negative	chatgpt	7222	(chatgpt, wrong)	423	(chatgpt, not, working)	145
	wrong	1652	(chatgpt, failed)	319	(ai, taking, jobs)	122
	problem	1329	(ai, dangerous)	298	(chatgpt, giving, misinformation)	105

Figure 4

Word-cloud representation of ChatGPT-related tweets.

The word cloud was generated using Python's WordCloud library (v1.8.1) applied to the full preprocessed corpus after stopword removal. Word size is proportional to term frequency across all tweets; terms appearing fewer than 50 times were excluded. The same NLTK English stopword list used during preprocessing was applied, supplemented with the token "chatgpt" to prevent it from dominating the visualization given its near-universal occurrence across all sentiment classes.

The n-gram analysis by sentiment category reveals distinct patterns in how users discuss ChatGPT:

1. **Positive sentiments** are characterized by expressions of admiration ("amazing," "impressive") and utility ("helpful"), suggesting widespread appreciation for ChatGPT's capabilities.

2. **Neutral sentiments** focus on usage patterns and capabilities, with phrases like "asked chatgpt" and "using chatgpt" indicating functional interactions without strong emotional valence.

3. **Negative sentiments** center on performance issues ("wrong," "failed") and broader concerns about AI ("dangerous," "taking jobs"), highlighting areas of user frustration and societal anxiety.

https://doi.org/10.64922/jteas.v1i1.19

These patterns offer valuable insights into the multifaceted nature of public perception toward ChatGPT, spanning from enthusiastic adoption to critical scepticism.

Discussion

Interpretation of Classifier Performance

The performance metrics reveal significant insights into the effectiveness of different machine learning classifiers for sentiment analysis of ChatGPT-related tweets. Linear SVM with TF-IDF vectorisation emerged as the most effective approach, achieving the highest accuracy (84.02%) and macro F1-score (80.81%) without balancing.

The strong performance of Random Forest with BOW and balancing techniques (80.08% macro F1-score) demonstrates the potential of ensemble learning approaches in sentiment classification, particularly when class imbalance is properly addressed. This finding resonates with previous studies that have highlighted ensemble methods' effectiveness in improving classification performance through aggregating multiple decision boundaries (Hasan et al., 2018).

Logistic Regression also demonstrated competitive performance across different configurations, achieving 79.78% macro F1-score with BOW without balancing. This supports existing literature on Logistic Regression's effectiveness in text classification due to its interpretability and ability to model probabilistic relationships between features and sentiment categories (Tan et al., 2023).

Naive Bayes consistently underperformed compared to other classifiers, likely due to its strong assumption of feature independence, which may not hold in complex social media text data where contextual relationships between words significantly influence sentiment expressions (Zainuddin & Selamat, 2015).

The superior performance of SVM with TF-IDF can be theoretically explained through the lens of margin maximization in high-dimensional spaces. Unlike Naive Bayes, which assumes feature independence, SVM determines an optimal separating hyperplane that maximises the geometric margin between sentiment classes, as formalised in the optimisation problem presented in Section 3.4. This margin-maximisation objective, combined with the L2-regularised linear kernel, allows the model to handle the high-dimensional sparsity of TF-IDF representations without overfitting, while the TF-IDF weighting simultaneously elevates discriminative sentiment-carrying terms and suppresses high-frequency vocabulary with limited sentiment value. Together these properties explain the consistent empirical superiority observed across experimental configurations.

The cross-validation results further validate these observations, showing consistent performance across multiple iterations for SVM with TF-IDF. The improved performance with balancing techniques (average macro F1-score of 0.79 compared to 0.77 without balancing) underscores the importance of addressing class imbalance in sentiment analysis tasks, particularly for ensuring equitable performance across all sentiment categories.

Comparison of Feature Extraction Techniques and Balancing Approaches

The comparison between BoW and TF-IDF vectorizers revealed interesting patterns in their impact on classifier performance. For SVM, TF-IDF consistently outperformed BoW, achieving higher accuracy and macro F1-scores. This suggests that the term weighting approach of TF-IDF, which emphasises distinctive terms while downplaying common ones, effectively captures sentiment-indicative features in ChatGPT-related tweets.

Conversely, Random Forest performed better with BoW, particularly when combined with balancing techniques. This pattern suggests that ensemble methods like Random Forest benefit from the straightforward frequency-based representation of BoW, potentially because the ensemble approach already captures complex feature interactions without requiring the additional weighting provided by TF-IDF.

https://doi.org/10.64922/jteas.v1i1.19

The impact of balancing techniques varied across classifiers and feature extraction methods. For SVM with TF-IDF, balancing slightly reduced overall accuracy (from 84.02% to 82.66%) but improved recall measures, resulting in more balanced performance across sentiment categories. This trade-off between overall accuracy and balanced class performance highlights the importance of selecting evaluation metrics aligned with analytical objectives, particularly in applications where minority class detection is crucial.

Insights into Sentiment Dynamics

The n-gram analysis by sentiment category provides valuable insights into the discourse patterns surrounding ChatGPT on Twitter. The dominance of positive sentiment expressions suggests generally favourable reception of the technology, with terms like "amazing," "impressive," and "helpful" reflecting appreciation for ChatGPT's capabilities. This positive sentiment aligns with broader trends in early technology adoption, where novel capabilities often generate initial enthusiasm. Neutral sentiments primarily focused on functional aspects and use cases, suggesting widespread experimentation with and exploration of ChatGPT's capabilities. The prevalence of phrases like "asked chatgpt" and "using chatgpt" indicates integration of the technology into users' information-seeking and problem-solving processes. Negative sentiments, while less prevalent, centered on specific concerns: performance limitations ("wrong," "failed"), accuracy issues ("misinformation"), and broader societal implications ("ai taking jobs"). These concerns mirror ongoing debates about AI technologies' reliability, trustworthiness, and socioeconomic impact, reflecting multifaceted apprehensions about rapidly evolving AI capabilities.

When viewed through the lens of technology adoption theories, these sentiment patterns align with early-stage technology diffusion characteristics. The predominance of positive sentiment reflects reactions typical of early adopters, who tend to emphasise potential benefitstend to emphasise potential benefits while tolerating limitations. The specific positive expressions focusing on capability ("amazing," "impressive") rather than usability suggest the technology is still perceived as novel rather than fully integrated into routine workflows. Meanwhile, negative sentiments centering on accuracy and reliability concerns rather than fundamental objections to the technology itself indicate the "performance gap" often experienced in early deployment phases of innovations. This theoretical contextualisation suggests ChatGPT was, during the study period, navigating the transition from early adopter enthusiasm toward broader acceptance, with sentiment patterns reflecting this transitional position.

Implications and Applications

The findings from this study have significant implications for multiple stakeholders in the AI ecosystem. For developers and organizations deploying conversational AI systems, understanding sentiment patterns provides valuable feedback on user experiences, highlighting both strengths to leverage and limitations to address. The identification of specific negative sentiment triggers (e.g., accuracy concerns) offers actionable insights for targeted improvements.

For researchers in sentiment analysis, our results demonstrate the continued relevance of conventional machine learning approaches, particularly when computational efficiency is a consideration. The superior performance of Linear SVM with TF-IDF, achieving over 84% accuracy without deep learning architectures, suggests that well-optimized traditional methods remain valuable tools in sentiment analysis applications.

For social media analysts and market researchers, the sentiment categorization and n-gram analysis methodologies demonstrated here provide a framework for monitoring public perception of emerging technologies. The approach can be adapted to track sentiment evolution over time, assess response to specific AI capabilities or limitations, and identify emerging concerns or appreciation points in public discourse.

Limitations and Future Research

While this study provides valuable insights into sentiment analysis of ChatGPT-related tweets, several limitations should be acknowledged. The dataset, while substantial, may not capture the full diversity of discussions about ChatGPT across different languages, platforms, and timeframes. Additionally, inherent

https://doi.org/10.64922/jteas.v1i1.19

biases in Twitter's user demographics may skew sentiment distributions compared to broader public opinion. Methodologically, our approach focused on conventional machine learning classifiers without exploring deep learning alternatives such as transformers or bidirectional LSTMs, which might offer performance improvements at the cost of greater computational requirements. Furthermore, the three-category sentiment classification (positive, neutral, negative) may not capture more nuanced emotional expressions such as excitement, frustration, surprise, or skepticism.

Future research could address these limitations by:

1. Expanding the dataset to include multiple languages and platforms, employing more sophisticated data collection techniques, and conducting longitudinal studies to capture temporal variations in sentiment.

2. Implementing more granular sentiment categorisation beyond the positive-neutral-negative framework, potentially incorporating emotion detection to identify specific affective states like curiosity, confusion, or amazement.

3. Exploring hybrid approaches that combine conventional machine learning with selective deep learning components, balancing performance improvements with computational efficiency.

4. Investigating the relationship between sentiment expressions and specific ChatGPT capabilities or limitations, providing more targeted insights for technology improvement.

5. Conducting comparative analyses of sentiment toward different AI models and technologies to contextualize ChatGPT-specific findings within broader AI perception trends.

Conclusion

This study evaluated the effectiveness of machine learning classifiers for sentiment analysis of ChatGPT-related tweets, comparing different feature extraction techniques and assessing the impact of balancing approaches on classification performance. The results demonstrate that Linear SVM with TF-IDF vectorization provides the most effective approach for sentiment classification in this context, achieving 84.02% accuracy without balancing and maintaining strong performance across sentiment categories with balancing techniques (80.12% macro F1-score).

The analysis of sentiment-specific discourse patterns revealed predominantly positive public reception of ChatGPT, with appreciation for its capabilities alongside specific concerns about performance limitations and broader AI implications. These findings contribute to understanding sentiment dynamics in AI-related social media conversations and demonstrate the practical efficacy of conventional machine learning approaches for sentiment classification in resource-constrained environments.

Beyond the practical implications, our findings provide empirical insights into sentiment analysis methodologies. The consistent superiority of discriminative models (particularly SVM) over generative approaches (Naive Bayes) across multiple experimental configurations provides empirical support for theoretical predictions regarding their relative strengths in high-dimensional text classification tasks. Furthermore, the differential impact of feature weighting techniques across classifier types suggests a complex interaction between feature representation and learning algorithm that warrants further theoretical exploration. These results challenge simplistic views of classifier selection and underscore the importance of considering the theoretical alignment between data characteristics, feature extraction techniques, and classification algorithms when developing sentiment analysis systems.

By providing insights into both methodological effectiveness and substantive sentiment patterns, this research offers valuable contributions to sentiment analysis practice and AI technology perception research. While acknowledging limitations in dataset diversity and classification granularity, our approach achieves high

https://doi.org/10.64922/jteas.v1i1.19

accuracy with traditional classifiers, suggesting continued relevance for conventional methods alongside emerging deep learning approaches in sentiment analysis applications.

Acknowledgements

The authors would like to acknowledge all individuals whose constructive comments and suggestions helped to improve the quality of this paper. No external funding was received for the completion of this work.

Author Contribution Statement

Conceptualisation: PCA (lead), EKM (equal)

Data analysis: PCA (lead), SKA (equal), ROA (equal)

Methodology/Frameworks: SKA (equal), NP (equal), DY (equal)

Datasets and Tools: SKA (lead), ROA (equal)

Initial write-up: PCA (lead), SKA (equal), OKB (supporting)

Revision: PCA (equal), EKM (equal), OKB (equal), EAF (equal)

Empirical Validation: PCA (lead), ROA (supporting), SKA (supporting), DY (supporting), NP (supporting), OKB (supporting)

Quality Assessment: PCA (lead), ROA (equal), SKA (supporting), OKB (supporting)

Conflict of Interest Statement

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

AI Disclosure Statement

No Artificial Intelligence (AI) tools were used in the preparation of this manuscript.

References

Aziz, R. H. H., & Dimililer, N. (2020). Twitter sentiment analysis using an ensemble weighted majority vote classifier. In *2020 International Conference on Advanced Science and Engineering (ICOASE)*. https://doi.org/10.1109/icoase51841.2020.9436590

Bordoloi, M., & Biswas, S. K. (2023). Sentiment analysis: A survey on design framework, applications and future scopes. *Artificial Intelligence Review*, *56*(11), 12505–12560. https://doi.org/10.1007/s10462-023-10442-2

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Lawrence Erlbaum Associates. https://doi.org/10.4324/9780203771587

Dey, A., Jenamani, M., & Thakkar, J. J. (2018). Senti-N-Gram: An N-gram lexicon for sentiment analysis. *Expert Systems with Applications*, *103*, 92–105. https://doi.org/10.1016/j.eswa.2018.03.004

Gehrmann, S., Dernoncourt, F., Li, Y., Carlson, E. T., Wu, J. T., Welt, J., John, F. J., Moseley, E. T., Grant, D. W., Tyler, P. D., & Celi, L. A. (2017). *Comparing rule-based and deep learning models for patient phenotyping*. arXiv. https://doi.org/10.48550/arxiv.1703.08705

Gifari, M. K., Lhaksmana, K. M., & Dwifebri, P. M. (2021). Sentiment analysis on movie review using ensemble stacking model. In *2021 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS)*. https://doi.org/10.1109/icadeis52521.2021.9702088

Guo, B., Zhang, C., Wang, Z., Guo, J., & Yu, Z. (2023). *How close is ChatGPT to human experts? Comparison corpus, evaluation, and detection* (arXiv:2301.07597). arXiv. https://doi.org/10.48550/arXiv.2301.07597

https://doi.org/10.64922/jteas.v1i1.19

Hasan, A., Moin, S., Karim, A., & Shamshirband, S. (2018). Machine learning-based sentiment analysis for Twitter accounts. *Mathematical and Computational Applications*, *23*(1), 11. https://doi.org/10.3390/mca23010011

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information*, *10*(4), 150. https://doi.org/10.3390/info10040150

Leiter, C., Zhang, R., Chen, Y., Belouadi, J., Larionov, D., Fresen, V., & Eger, S. (2023). *ChatGPT: A meta-analysis after 2.5 months* (arXiv:2302.13795). arXiv. https://doi.org/10.48550/arXiv.2302.13795

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167. https://doi.org/10.2200/S00416ED1V01Y201204HLT016

Mercha, E. M., & Benbrahim, H. (2023). Machine learning and deep learning for sentiment analysis across languages: A survey. *Neurocomputing*, *531*, 195–216. https://doi.org/10.1016/j.neucom.2023.02.015

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)* (pp. 79–86). https://doi.org/10.3115/1118693.1118704

Parveen, H., Hada, B., Lalwani, P., & Gupta, M. (2023). Sentiment analysis on social media using machine learning techniques: A comprehensive review. *Multimedia Tools and Applications*, *82*(14), 21591–21630. https://doi.org/10.1007/s11042-022-14160-9

Sallam, M., Salim, N. A., Barakat, M., & Al-Tammemi, A. B. (2023). ChatGPT applications in medical, dental, pharmacy, and public health education: A descriptive study highlighting the advantages and limitations. *Narra J*, *3*(1), e103. https://doi.org/10.52225/narra.v3i1.103

Sravya, K., Sowmya, G., Yamini, P., Anusha, P., & Krishna, P. S. (2021). Sentiment analysis on Twitter. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3920078

Tan, K. L., Lee, C. P., & Lim, K. M. (2023). A survey of sentiment analysis: Approaches, datasets, and future research. *Applied Sciences*, *13*(7), 4550. https://doi.org/10.3390/app13074550

Tan, K. L., Lee, C. P., Lim, K. M., & Anbananthen, K. S. M. (2022). Sentiment analysis with ensemble hybrid deep learning model. *IEEE Access*, *10*, 103694–103704. https://doi.org/10.1109/access.2022.3210182

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

Zainuddin, N., & Selamat, A. (2015). Sentiment analysis using support vector machine. In *2014 International Conference on Computer, Communications, and Control Technology (I4CT)* (pp. 333–337). https://doi.org/10.1109/i4ct.2014.6914200

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems* (Vol. 28). https://doi.org/10.48550/arXiv.1509.01626

Zhang, Y., & Wallace, B. C. (2015). *A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification*. arXiv. https://doi.org/10.48550/arxiv.1510.03820

https://doi.org/10.64922/jteas.v1i1.19