A Comparative Study of Naïve Bayes and K-Nearest Neighbors (KNN) Algorithms in Sentiment Analysis of ChatGPT Usage Among Students

Syahli Kurniawan[1], Rahmad Hidayat[1*], and Muhammad Reza Zulman[1]

[1]Informatics Engineering Study Program, Department of Information Technology and Computer, Politeknik Negeri Lhokseumawe, Aceh, Indonesia

[*]Email: rahmad_hidayat@pnl.ac.id

Received on 27-10-2025 | *Revised on 21-11-2025* | *Accepted on 17-12-2025*

Abstract—This study compares the performance of the Naïve Bayes and K-Nearest Neighbors (KNN) algorithms in sentiment analysis of Lhokseumawe State Polytechnic students toward the use of ChatGPT. The comparison is conducted due to the varied results of previous research, where the effectiveness of both algorithms largely depends on the data type and context. The model was developed using 9.800 external data collected from Twitter and Google Play Store, which were processed through text preprocessing and TF-IDF transformation stages, and then tested on 237 student questionnaire data as a case study. The initial evaluation showed that Naïve Bayes achieved an accuracy of 88% with a prediction time of 0,0063 seconds, while KNN recorded an accuracy of 83% with a prediction time of 0,4760 seconds. In the student questionnaire test, Naïve Bayes again outperformed with 79,75% accuracy compared to KNN's 49,37%.

Keywords: ChatGPT, Classification, K-Nearest Neighbors, Naïve Bayes, Sentiment Analysis, Students

I. INTRODUCTION

The development of artificial intelligence (AI) has brought a significant impact on higher education, both in the learning process and in providing adaptive and contextual information sources [1]. One of the AI implementations widely used by students is ChatGPT, a language model developed by OpenAI to generate automated text and answer academic questions [2]. The use of ChatGPT has been proven to enhance learning effectiveness, simplify assignment preparation, and support critical thinking through conversational interactions [3].

Despite its many advantages, the use of ChatGPT also raises several concerns. Dependence on this model has the potential to reduce students analytical thinking and creativity, as well as increase the risk of plagiarism in academic writing [4]. Moreover, ethical aspects and the authenticity of academic work outcomes have become important issues to consider.

In the context of sentiment analysis, the Naïve Bayes (NB) and K-Nearest Neighbors (KNN) algorithms are commonly used methods because of their effectiveness in text classification. However, their performance varies depending on the dataset characteristics. Study [5] found that KNN achieved higher accuracy in classifying Twitter sentiments regarding the relocation of the capital city, while research [6] reported that Naïve Bayes performed better on e-commerce reviews. Other studies [7] and [8] also presented diverse outcomes in different domains, emphasizing that the effectiveness of NB and KNN is highly influenced by data distribution, feature quantity, and usage context.

Based on the reviewed literature, there has been no research specifically comparing both algorithms in the context of higher education, particularly regarding students' perceptions of ChatGPT usage. Therefore, this study aims to (1) analyze students' sentiments toward ChatGPT usage, (2) compare the performance of Naïve Bayes and KNN in terms of accuracy and execution time, and (3) recommend the most optimal algorithm for text-based sentiment analysis in higher education.

II. METHOD

This research was conducted through four main stages: data collection, text preprocessing, model construction, and algorithm evaluation.

A. Dataset

The dataset used in this study consisted of two data sources. The first was external data containing 9.800 reviews, including 2.000 from Twitter and 7.800 from Google Play Store, which were labeled using a lexicon-based approach [9]. The second was questionnaire data from 237 students of Lhokseumawe State Polytechnic regarding ChatGPT usage, also labeled using the lexicon-based method. The research architecture developed for the sentiment analysis system is shown in Figure 1.

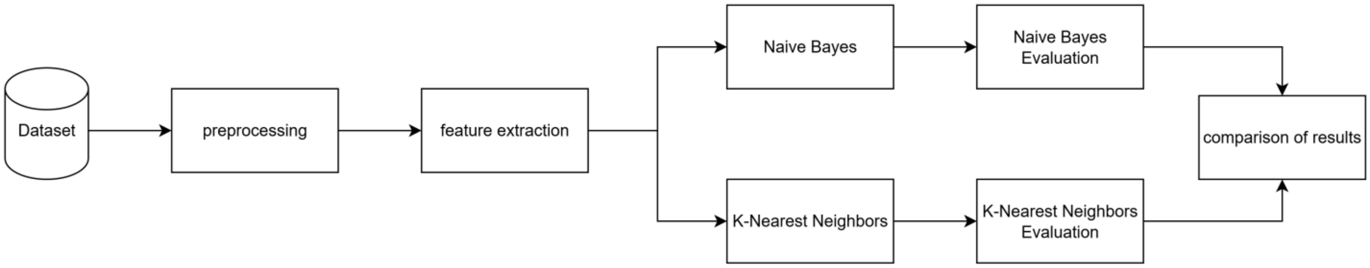

Figure 1. Proposed Model Architecture

This architecture represents the overall workflow of the sentiment analysis system, which begins with text data collection, followed by preprocessing, feature extraction using TF-IDF, and classification using Naïve Bayes and K-Nearest Neighbors (KNN). Each component interacts sequentially to ensure that the input text is transformed into a measurable sentiment output. The same architecture was applied to both data sources—external and questionnaire-based—to maintain consistency in the testing process.

Table I presents a sample of external text data used to train and construct the Naïve Bayes and KNN models. These data were selected to represent diverse user expressions and sentiment variations, ensuring that the models could recognize different linguistic patterns effectively. The outcomes from this training phase were later validated using questionnaire data obtained from students to evaluate the system's real-world performance.

TABLE I
THE EXAMPLE OF EXTERNAL DATA

Index	User_Name	Score	Content
0	Wizard Who Is out Of Spell	1	Can the automatic text deletion system be removed? Every time I want …
1	Pinz aja	4	The application is really good, it can make photos realistic and HD, but there is a small technical ….
2	Evelyn Herta Elisya	3	The app is really good, honestly, it's really helpful for my assignment. But now, ….
3	Tiara Khairunnisa	5	Its performance is excellent, and GPT adapts its language to suit its users, making it more …
4	Andi Saputra	5	Overall, it's okay, compared to other AI, honestly, it's more comfortable to use Chatgpt. It just doesn't ….

The external data were used to train the model, which was then evaluated using the questionnaire data collected from Lhokseumawe State Polytechnic students. An example of the student data used in this study is presented in Table II.

TABLE II
THE EXAMPLE OF STUDENT DATASET

No	Username	Text
1	Susi	I use ChatGPT to help me understand course materials…
2	Dedi Firmansyah	I feel dependent on chatgpt to solve my......
4	Budi Kusuma	Sometimes, because it makes me confused what they are.
5	Dedi Pratama	No
6	Anja_yani	I use chatgpt to help understand....

The results of this student data evaluation will produce a confusion matrix which will be the final conclusion.

B. Text Pre-processing

The data preprocessing stage is a crucial component in text-based research as it determines the quality of information that will be processed during the classification stage. This process aims to transform raw and unstructured data into a more organized format so that it can be effectively processed by machine learning algorithms. The data collected from Twitter, the Google Play Store, and student questionnaires generally contain many non-textual elements such as punctuation marks, emojis, uppercase letters, and meaningless words that can reduce model accuracy if not properly handled. Therefore, a series of data cleaning procedures was carried out to ensure that only relevant and meaningful words were used in the classification process [10].

The preprocessing phase consists of four main stages: case folding, tokenization, stopword removal, and stemming. The case folding stage converts all letters to lowercase so that words with different capitalizations are treated as having the same meaning. Tokenization breaks sentences into individual word units for easier analysis. Subsequently, stopword removal eliminates common words such as "yang" (that), "dengan" (with), and "saya" (I), which do not significantly affect sentiment context. The final stage, stemming, aims to return each word to its root form using the Nazief–Adriani algorithm. Thus, words like "membantu" (helping), "dibantu" (helped), and "bantuan" (assistance) are normalized into a single root word, "bantu" (help). A complete illustration of the text-cleaning stages is presented in Figure 2.

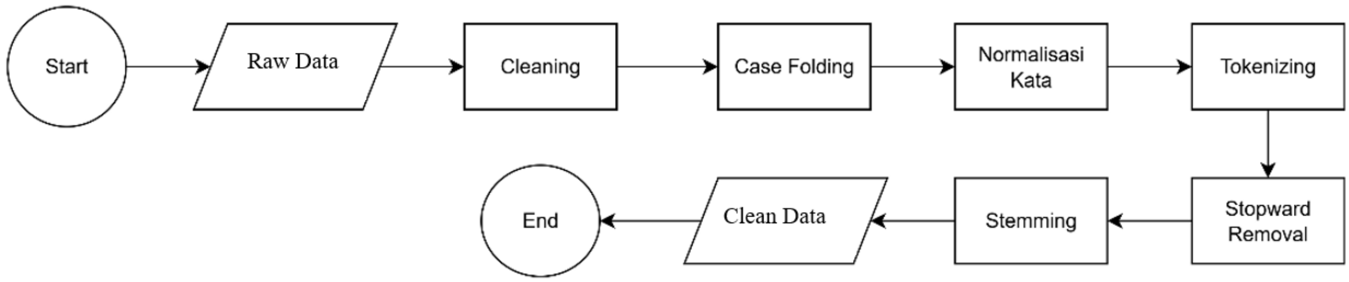

Figure 2. Pre-processing Stages

Feature transformation using the TF-IDF (Term Frequency–Inverse Document Frequency) method was performed after the text cleaning stage was completed. This method converts words into numerical representations based on their level of importance within a document. Words that frequently appear in a particular document but rarely occur in other documents are assigned higher TF-IDF weights. This principle enables the model to emphasize words that truly represent the opinion context within the text corpus. The TF-IDF approach also helps reduce the influence of common words that appear across all documents. An example of the results from the text-cleaning process can be seen in Table III, which shows a comparison between the students' original sentences and the simplified preprocessed outputs.

TABLE III
PRE-PROCESSING RESULTS

No	Text	Clean Text	Label
1	ChatGPT helps me understand lecture materials, complete assignments faster, and find creative ideas…	chatgpt helps understand lecture materials complete assignments faster. ..	Positive
2	Sometimes, this makes me confused about what they are …	sometimes makes confused about what is clear purpose know which …	Negative
3	I feel dependent on ChatGPT to complete assignments. I use ChatGPT to avoid ….	feel dependent on chatgpt to complete tasks use chatgpt to avoid reading long…	Negative

C. Classification Algorithms

This study used two machine learning algorithms: Naïve Bayes (NB) and K-Nearest Neighbors (KNN). Naïve Bayes is a probabilistic algorithm based on Bayes' Theorem, assuming independence among features [11] [12]. Meanwhile, KNN is a distance-based algorithm that classifies data according to the majority class among the nearest neighbors [13][14]. Both algorithms have been proven effective in various text classification studies, though their performance depends on data distribution and feature size. Before we train the Naïve Bayes model, the dataset was split into training and test sets at an 80:20 ratio. The split was performed randomly using the train_test_split(), with stratify=True to maintain a proportional distribution of sentiment labels and random_state=42 to ensure consistent results. Below the complete configurations of Naïve Bayes architecture using in this study:

```
X_train, X_test, y_train, y_test =
 train_test_split(
    df['clean_text'],
    df['label'],
    test_size=0.2,
    stratify=df['label'],
    random_state=42
)

model_nb = MultinomialNB()

# Pelatihan model
start_train_nb = time()
model_nb.fit(X_train_tfidf, y_train)
train_time_nb = time() - start_train_nb

# Prediksi
start_pred_nb = time()
y_pred_nb = model_nb.predict(X_test_tfidf)
pred_time_nb = time() - start_pred_nb
```

In contrast, KNN tends to produce more accurate results when sufficient training data are available, but its prediction time increases significantly as the dataset grows due to the need to compute distances between all data points. We configure out KNN model using 7 neighbors (n_neighbors=7), a cosine similarity metric, and distance-based weights (weights='distance'). A brute-force algorithm is used for neighbor search, and n_jobs=-1 enables parallel processing. Below the complete configurations for KNN using in this study:

```
model_knn = KNeighborsClassifier
(
    n_neighbors=7,
    metric='cosine',
    weights='distance',
    algorithm='brute',
    n_jobs=-1
)
model_knn.fit(X_train_tfidf, y_train)
y_pred_knn = model_knn.predict(X_test_tfidf)
```

D. Model Evaluation

The evaluation of each classification model was conducted systematically to assess its performance and efficiency in sentiment analysis. This process involved analyzing the results using a confusion matrix, from which several standard evaluation metrics — namely accuracy, precision, recall, and F1-score — were derived [15]. These metrics were chosen because they collectively provide a comprehensive view of the model's predictive capabilities, ensuring that the evaluation is not biased toward a single aspect of performance. For instance, while accuracy measures the overall correctness of predictions, precision and recall offer deeper insight into how well the model distinguishes between different sentiment classes. The F1-score, which represents the harmonic mean of precision and recall, is particularly valuable because it balances the trade-off between the two, especially in cases where class distribution is imbalanced.

In addition to performance metrics, execution time was also measured to evaluate the computational efficiency of each model. This measurement is crucial, as sentiment analysis systems are often deployed in real-time applications where prediction speed can significantly influence usability and user experience. A model that achieves high accuracy but requires a long time to produce results may not be practical in real-world scenarios, especially when handling large-scale or streaming text data. Before the evaluation phase, the text data was transformed into numerical feature representations using the TF-IDF (Term Frequency–Inverse Document Frequency) technique. TF-IDF is widely used in text classification tasks because it captures both the frequency of words and their importance within the corpus, resulting in more meaningful feature representations for machine learning models. The configuration parameters of the TF-IDF vectorizer used in this study are summarized in Table IV.

TABLE IV
TF-IDF CONFIGURATION

No	Parameter	Value	Description
1	ngram_range	(1,2)	Uses unigram and bigram as features
2	max_features	3000	Takes 3,000 words/phrases with the highest weights
3	min_df	3	Ignores words appearing in fewer than 3 documents
4	max_df	0,85	Ignores words appearing in more than 85% of documents
5	sublinear_tf	True	Applies logarithmic scaling to term frequency

The configuration choices above were designed to optimize the representation of the text while reducing noise and irrelevant features. For instance, including both unigrams and bigrams allows the model to capture not only individual word meanings but also contextual relationships between consecutive words, which often carry sentiment-relevant information. Meanwhile, limiting the number of features and filtering out overly rare or overly common terms helps prevent overfitting and improves computational efficiency.

Once the TF-IDF vectors were generated, the models were trained and evaluated on the prepared dataset. The criteria for selecting the best-performing model were based on the evaluation indicators listed in Table V.

TABLE V
THE MODEL PERFORMANCE METRICS

No	Evaluation Indicator	Description
1	F1-Score	Main indicator reflecting the balance between precision and recall
2	Accuracy	Percentage of correct predictions over all test data
3	Precision	Measures the correctness of positive predictions
4	Recall	Measures how well the model captures all true positive data
5	Execution Time	Time required by the model to perform prediction (efficiency)

Among these indicators, F1-score was prioritized as the main criterion for model comparison because it provides a more reliable measure of overall performance in sentiment classification, particularly when the distribution of positive and negative classes is not perfectly balanced. However, accuracy remains an important complementary metric for assessing the general performance of the classifier. Precision and recall are valuable for understanding the model's behavior in terms of false positives and false negatives, respectively, while execution time serves as a practical consideration for real-world deployment. Overall, this multi-dimensional evaluation framework ensures a fair and comprehensive assessment of both the predictive effectiveness and computational efficiency of the models. By combining multiple performance metrics with a focus on both accuracy and speed, the selection of the best sentiment classification model can be based on a balanced view of quality and practicality.

III. RESULTS AND DISCUSSION

This study presents a comparative analysis of the performance of the Naïve Bayes and K-Nearest Neighbors (KNN) algorithms in sentiment classification tasks using two different types of datasets: external data (collected from platforms such as Twitter and Google Play Store) and student questionnaire data obtained from the Lhokseumawe State Polytechnic. The evaluation aimed to investigate how each algorithm performs under different data characteristics and text contexts, particularly in terms of accuracy, precision, recall, F1-score, and computational efficiency. By analyzing results from both datasets, a comprehensive understanding of the strengths and limitations of each model can be achieved.

A. Results on External Data

At this stage, two text classification models were developed using the Naïve Bayes (NB) and K-Nearest Neighbors (KNN) algorithms. These models were trained and tested using external textual data collected from social media platforms (Twitter) and application reviews (Google Play Store), consisting of a total of 9,800 samples. The use of external data in this phase aimed to construct and evaluate the baseline performance of both algorithms before applying them to the main case study — sentiment classification of students opinions toward ChatGPT. The experiments were conducted to assess each model's ability to classify sentiment into positive and negative categories. The results revealed a clear difference in performance between the two algorithms. Naïve Bayes consistently achieved higher accuracy compared to KNN, indicating that it was more effective in learning sentiment polarity from large-scale, real-world textual data.

One of the main reasons behind this superior performance lies in the probabilistic foundation of Naïve Bayes, which enables it to process noisy and diverse linguistic data more effectively. Since Naïve Bayes computes the likelihood of each class based on word probability distributions, it can generalize even from incomplete or informal expressions — a common characteristic of user-generated content containing slang, abbreviations, or emotional wording. On the other hand, KNN depends heavily on distance calculations across all feature vectors, which makes it sensitive to irrelevant or redundant words that commonly appear in textual datasets.

The comparison results of both algorithms are summarized in Figure 3, which presents their respective performance scores in terms of accuracy, precision, recall, and F1-score.

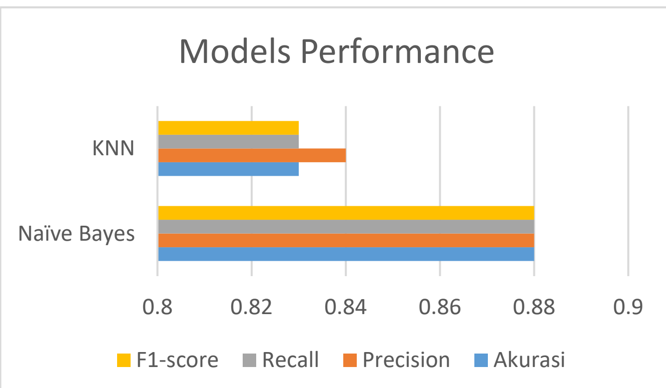

Figure 3. Model Performance

As shown in Figure 3, Naïve Bayes produced consistently higher values across all evaluation metrics. This suggests that the algorithm better captures contextual sentiment indicators and handles sparse TF-IDF features more effectively than KNN. While KNN can perform adequately on smaller and more homogeneous datasets, its dependency on neighbor distance calculations makes it less suitable for large-scale text data. The ability of Naïve Bayes to compute probabilities based on word frequency allows it to generalize more effectively, even when the input data contains noise or inconsistencies commonly found in real-world text.

To gain a deeper understanding of this performance difference, it is important to consider the characteristics of the external dataset itself. The data collected from social media and app reviews tends to include short, informal, and unstructured sentences, often with spelling variations, emojis, or mixed-language phrases. These attributes make the dataset highly heterogeneous and context-dependent. Under such conditions, Naïve Bayes benefits from its probabilistic modeling approach, which evaluates the likelihood of a sentiment class based on term occurrence rather than strict vector similarity. This makes it particularly effective in recognizing sentiment patterns from limited contextual clues, especially when sentences are fragmented or expressed with emotional tone.

Moreover, Naïve Bayes demonstrates strong resilience when dealing with imbalanced datasets, a common trait in opinion data where positive reviews are often more prevalent than negative ones. By calculating class-conditional probabilities, the algorithm can still maintain stability and minimize bias toward dominant classes. This probabilistic mechanism allows Naïve Bayes to better represent sentiment polarity across a wide range of linguistic styles and data distributions. In contrast, KNN tends to misclassify such data, as its performance heavily depends on the density and proximity of training samples in the feature space.

Additionally, the external dataset's high dimensionality due to TF-IDF representation increases the computational complexity for KNN, as it must calculate distances for every new instance across all features. This factor contributes to its slower performance and lower scalability. Naïve Bayes, however, only multiplies precomputed conditional probabilities, enabling it to process text classification tasks much faster while maintaining stable accuracy.

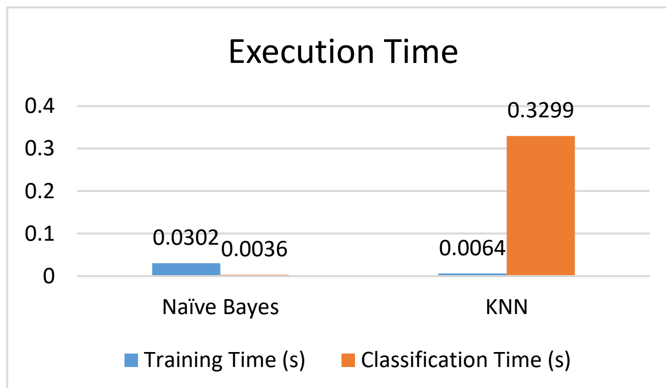

Figure 4. Execution Time

However, accuracy and F1-score alone are not sufficient to evaluate the overall efficiency of a model. In text classification tasks — especially those designed for large-scale or real-time applications such as sentiment monitoring or opinion tracking — execution time becomes an equally critical factor in performance evaluation. To further assess computational efficiency, the training and prediction times of both algorithms were compared, as illustrated in Figure 4. The next section discusses these results in detail and highlights how Naïve Bayes achieves a better trade-off between accuracy and processing speed, making it more practical for real-world implementations.

Figure 4 clearly shows that Naïve Bayes required considerably less time during both training and prediction. The KNN model, in contrast, took longer because every new input had to be compared against all training samples using distance-based similarity computations. Naïve Bayes, on the other hand, relies on precomputed probability values, allowing it to make predictions much faster once the model is trained.

These results confirm that Naïve Bayes is both accurate and computationally efficient when used for large-scale external datasets. The balance between prediction accuracy and processing speed makes it a more robust foundation for sentiment classification tasks. Consequently, the Naïve Bayes model developed in this stage serves as the primary classifier to be applied in the next phase of the study — classifying student sentiment toward ChatGPT using the original questionnaire labels as ground truth. Overall, the external dataset experiment provides valuable insight into how both algorithms behave under real-world textual conditions. The results reinforce that Naïve Bayes not only performs better in terms of classification accuracy but also offers practical scalability for larger and more complex sentiment analysis applications.

B. Results on Student Questionnaire Data

After developing and evaluating the model using external data, the next step was to apply the same classification approach to the student questionnaire dataset. This dataset contained a total of 237 text responses collected from students, where each response had been labeled based on lexicon as positive or negative. These original labels served as the ground truth for validating the performance of both models — Naïve Bayes and K-Nearest Neighbors (KNN).

TABLE V
ACCURACY COMPARISON ON STUDENT DATA ALGORITHM

No	Methods	Accuracy
1	Naïve Bayes	79,75%
2	KNN	49,37%

When evaluated on this dataset, the results consistently followed a similar trend as observed in the external data experiment: Naïve Bayes once again outperformed KNN in classifying student sentiment. The accuracy comparison is presented in Table IV, showing that Naïve Bayes achieved an accuracy of 79,75%, while KNN achieved only 49,37%. The detailed comparison of performance metrics is shown in Figure 5, which illustrates the differences in accuracy, precision, recall, and F1-score between both algorithms.

As seen in Figure 5, Naïve Bayes obtained higher scores across all metrics, demonstrating its superior capability in recognizing sentiment polarity from short and varied student responses. This performance indicates that Naïve Bayes can generalize effectively even when dealing with relatively small and informal datasets. The probabilistic structure of the algorithm allows it to identify subtle contextual cues and interpret variations in student expression styles, such as abbreviations or mixed emotional tones, which are common in opinion-based survey data. Furthermore, these results suggest that Naïve Bayes adapts well to the linguistic diversity typically found in student responses, where opinions are expressed using informal vocabulary or non-standard sentence structures. The model's ability to rely on word frequency and conditional probabilities rather than strict syntactic order makes it less sensitive to grammatical inconsistencies, allowing for stable performance even with limited training samples.

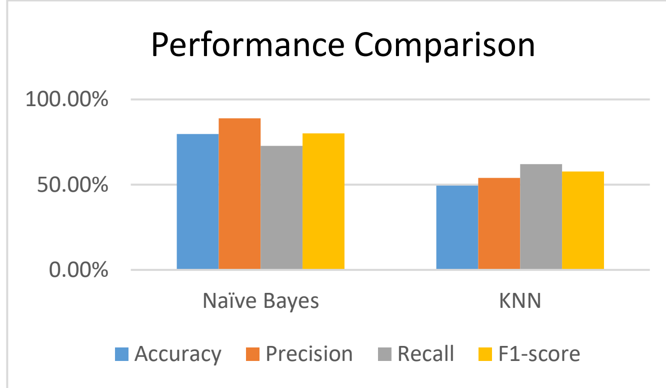

Figure 5. Performance Comparison both Architecture

This adaptability is particularly beneficial in academic contexts, where the language used in feedback or survey data often varies depending on the student's writing style and experience.

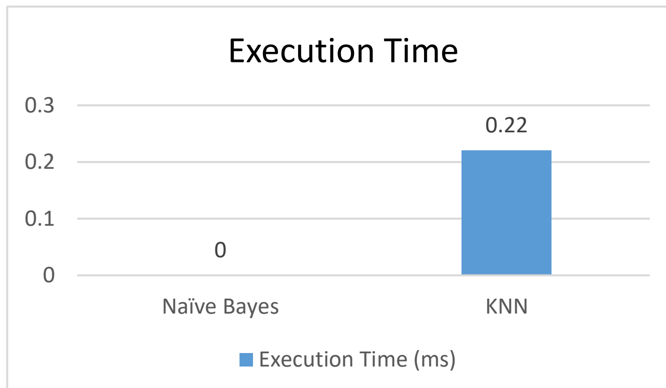

Figure 6. The Models Execution Time

In contrast, KNN tends to struggle with such variations because it treats each text as a high-dimensional vector and depends on distance calculations between samples. When the feature space becomes sparse due to short text length or inconsistent phrasing, the algorithm's accuracy decreases significantly. This limitation reinforces why Naïve Bayes is often preferred for sentiment analysis tasks involving natural, human-written content. To further analyze computational efficiency, execution time was measured for both models. The results, illustrated in Figure 6, show that Naïve Bayes executed much faster than KNN, which required longer processing time due to its distance-based computation method. In practical scenarios, this faster execution is a major advantage, particularly when processing responses from large numbers of students or implementing automated feedback systems in real time. From Figure 6, it is evident that the execution time of KNN was significantly higher, as it calculates the distance between each test instance and all training samples before making predictions. In contrast, Naïve Bayes performed instant predictions using precomputed probability distributions, resulting in a more efficient classification process.

Overall, these findings confirm that Naïve Bayes consistently provides better performance and efficiency when applied to student sentiment classification. The strong alignment between predicted labels and the ground truth demonstrates that the model trained with external data successfully generalized to a smaller, domain-specific dataset. This proves that the Naïve Bayes model is not only accurate but also reliable for identifying student perceptions toward ChatGPT in an educational setting.

C. Student Sentiment Analysis

The sentiment analysis revealed that the majority of students expressed positive sentiments toward the use of ChatGPT in academic activities, reflecting an overall acceptance of AI-assisted learning tools. However, a smaller portion of responses contained negative opinions, primarily related to concerns about ethical implications, overreliance, and potential plagiarism. This sentiment distribution, which was obtained through responses collected via Google Form surveys and automatically labeled using a lexicon-based approach, is illustrated in Figure 7.

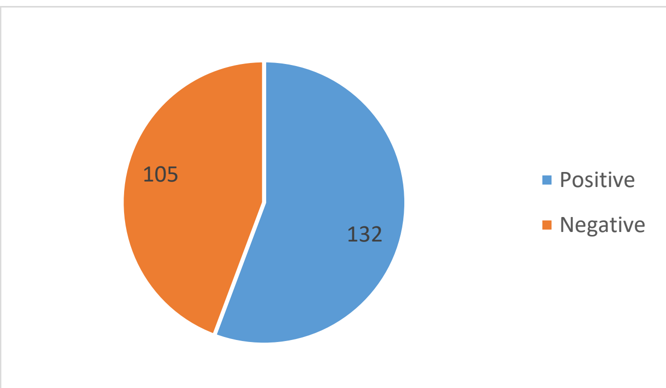

Figure 7. Distribution of Student Sentiment towards ChatGPT

As shown in Figure 7, positive sentiment dominates the dataset, suggesting that most students perceive ChatGPT as a helpful and accessible tool for completing assignments, understanding course materials, and generating creative ideas. Many respondents indicated that ChatGPT enhances learning efficiency by providing quick explanations and alternative perspectives. Nevertheless, a notable minority of students expressed skepticism, emphasizing the potential ethical and academic risks associated with its use. These include dependency on AI-generated answers, reduced analytical thinking, and the possibility of academic dishonesty if students rely excessively on ChatGPT for assignments. Such feedback underscores the importance of balanced guidance from educators, ensuring that AI is used as a supportive aid rather than a replacement for human reasoning.

These findings are consistent with the study in [6], which demonstrated the superior performance of Naïve Bayes in analyzing text-based reviews, especially in distinguishing nuanced opinions. However, they differ from the results reported in [5], where KNN performed better on Twitter-based sentiment analysis. This contrast reinforces that algorithmic performance is highly dependent on the data domain and linguistic characteristics of the dataset. In this case, the academic context—with its formal yet varied language—appears to favor the probabilistic nature of Naïve Bayes. Overall, the sentiment results confirm that students tend to adopt a positive yet cautious attitude toward ChatGPT. The insights from this analysis can be used by educational institutions to develop AI literacy programs, promote ethical use, and encourage critical thinking while integrating generative AI tools into learning environments.

IV. CONCLUSION

This study compared the Naïve Bayes and K-Nearest Neighbors (KNN) algorithms in conducting sentiment analysis of students toward the use of ChatGPT. The experimental results demonstrated that Naïve Bayes consistently outperformed KNN, both on the external dataset (accuracy of 88% with a prediction time of 0,0063 seconds) and on the student questionnaire dataset (accuracy of 79,75%). In contrast, KNN achieved lower accuracy — 83% on the external data and 49,37% on the questionnaire data — while also requiring longer execution times. Based on our experiments, Naïve Bayes is more optimal for opinion-based text classification, both for large and small datasets. Therefore, it is more suitable for use in sentiment analysis within higher education contexts. The sentiment analysis also showed that the majority of students have a positive perception of ChatGPT usage, although ethical concerns and dependency issues remain significant considerations. The limitation of this research lies in the relatively small number of data samples and the use of only two algorithms. Future research is recommended to employ larger datasets and compare the results with other algorithms such as Support Vector Machine (SVM) or deep learning models to obtain more comprehensive results.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to Lhokseumawe State Polytechnic and all student respondents who participated in the questionnaire survey. Special thanks are also extended to the academic supervisors who provided valuable guidance and support throughout the completion of this research.

REFERENCES

[1] W. Suharmawan, "Pemanfaatan Chat GPT Dalam Dunia Pendidikan," *Education Journal : Journal Educational Research and Development*, vol. 7, no. 2, pp. 158–166, Aug. 2023, doi: 10.31537/ej.v7i2.1248.

[2] O. F. Rizki and R. Fernandes, "Pengetahuan dan Pemanfaatan Chat GPT di Kalangan Mahasiswa (Studi Kasus: Mahasiswa Departemen Sosiologi Universitas Negeri Padang)," *Naradidik: Journal of Education and Pedagogy*, vol. 3, no. 3, pp. 222–228, Sep. 2024, doi: 10.24036/nara.v3i3.218.

[3] R. N. Yahya, S. N. Azizah, and Y. T. Herlambang, "Pemanfaatan ChatGPT di Kalangan Mahasiswa: Sebuah Tinjauan Etika Teknologi dalam Perspektif Filsafat," *UPGRADE : Jurnal Pendidikan Teknologi Informasi*, vol. 1, no. 2, pp. 53–59, Feb. 2024, doi: 10.30812/upgrade.v1i2.3481.

[4] D. Dwihadiah, A. Gerungan, H. Purba, P. Studi Pendidikan Jarak Jauh Ilmu Komunikasi, and U. Pelita Harapan, "Penggunaan ChatGPT di Kalangan Mahasiswa dan Dosen Perguruan Tinggi Indonesia".

[5] Syahril Dwi Prasetyo, Shofa Shofiah Hilabi, and Fitri Nurapriani, "Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN," *Jurnal KomtekInfo*, pp. 1–7, Jan. 2023, doi: 10.35134/komtekinfo.v10i1.330.

[6] J. Khatib Sulaiman, A. Shopee Salman Alfaris, and U. Amikom Yogyakarta, "Komparasi Metode KNN dan Naive Bayes Terhadap Analisis Sentimen Pengguna," *Indonesian Journal of Computer Science Attribution*, vol. 12, no. 5, pp. 2023–2766.

[7] J. W. Iskandar and Y. Nataliani, "Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, pp. 1120–1126, Dec. 2021, doi: 10.29207/resti.v5i6.3588.

[8] F. Sholekhah, A. D. Putri, R. Rahmaddeni, and L. Efrizoni, "Perbandingan Algoritma Naïve Bayes dan K-Nearest Neighbors untuk Klasifikasi Metabolik Sindrom," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 2, pp. 507–514, Feb. 2024, doi: 10.57152/malcom. v4i2.1249.

[9] D. Nasien *et al.*, "Perbandingan Implementasi Machine Learning Menggunakan Metode KNN, Naive Bayes, Dan Logistik Regression Untuk Mengklasifikasi Penyakit Diabetes," 2024.

[10] R. Hidayat, A. Harjoko, and A. Musdholifah, "A Robust Image Retrieval Method Using Multi-Hierarchical Agglomerative Clustering and Davis-Bouldin Index," *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 2, pp. 441–453, Apr. 2022, doi: 10.22266/ijies2022.0430.40.

[11] H. B. Tambunan and T. W. D. Hapsari, "Analisis Opini Pengguna Aplikasi New PLN Mobile Menggunakan Text Mining," *PETIR*, vol. 15, no. 1, pp. 121–134, Dec. 2021, doi: 10.33322/petir.v15i1.1352.

[12] F. Sholekhah, A. D. Putri, and L. Efrizoni, "Comparison of Naive Bayes and K-Nearest Neighbors Algorithms for Metabolic Syndrome Classification Perbandingan Algoritma Naïve Bayes dan K-Nearest Neighbors untuk Klasifikasi Metabolik Sindrom," Indones. J. Mach. Learn. Comput. Sci., vol. 4, no. April, pp. 507–514, 2024.

[12] R. Mahendrajaya, G. A. Buntoro, and M. B. Setyawan, "Analisis Sentimen Pengguna Gopay Menggunakan Metode Lexicon Based Dan Support Vector Machine," 2019. [Online]. Available: http://studentjournal.umpo.ac.id/index.php/komputek

[13] B. Mahesh, "Machine Learning Algorithms - A Review," *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381–386, Jan. 2020, doi: 10.21275/art20203995.

[14] I. Syahdana, R. Hidayat, M. Khadafi, and I. Artikel, "A Web-Based Laptop Purchase Recommendation Model Using Natural Language Processing (NLP) on Marketplace Reviews," vol. 4, no. 2, pp. 116–122, 2024, doi: 10.30811/jaise.v4i2.6133.

[15] B. Mahesh, "Machine Learning Algorithms - A Review," *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381–386, Jan. 2020, doi: 10.21275/art20203995.