International Journal of Engineering Research and Development
e- ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 21, Issue 8 (August 2025), PP 53-62

Comparation of Text Mining Algorithm in Analysis of Student Satisfaction Sentiment Towards Campus Facilities

[1]Ida Bagus Ketut Surya Arnawa
[1]*Department of Information System Faculty of Informatics and Computers*
Institut Teknologi dan Bisnis STIKOM Bali, Denpasa, Indonesia
Corresponding Author: Ida Bagus Ketut Surya Arnawa

ABSTRACT
The Quality Assurance Center (PJM) of ITB STIKOM Bali faces significant challenges in manually analyzing over 5,000 open-ended responses from the campus facilities satisfaction survey, including subjectivity, time constraints, and inconsistencies in interpretation. This study aims to identify the optimal text mining algorithm for automated sentiment analysis (positive/neutral/negative) of Indonesian-language feedback to improve the efficiency of PJM's evaluations. The methodology involves data preprocessing (case folding, tokenization, stemming), keyword weighting (e.g., facilities, access, Sion) using TF-IDF, and testing four algorithms with a 30% training and 70% testing data split, along with cross-validation: Naïve Bayes (NB), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree (DT). Results show that SVM achieved the highest accuracy (90%), followed by NB (85%), while KNN and DT both scored 80%. SVM's superiority lies in its effectiveness in separating high-dimensional sentiment classes. It is concluded that SVM is the most suitable algorithm for automating sentiment analysis of campus facility feedback at ITB STIKOM Bali. Its implementation enables PJM to overcome manual processing barriers, reduce interpretation subjectivity, and generate objective insights for data-driven facility improvements—supporting the institution's vision of achieving international recognition through evidence-based quality enhancement. Future research may explore SVM-NB ensembles and contextual semantic analysis.

KEYWORD: *Naive Bayes Classifier (NBC), Term Frequency Inverse Document Frequency (TFIDF), Sentiment Analysis*

Date of Submission: 27-07-2025 Date of acceptance: 05-08-2025

I. INTRODUCTION

ITB STIKOM Bali originated as Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) STIKOM Bali, an institution specializing in informatics and computer science. Established on August 10, 2002, under permit number 157/D/O/2002, it initially offered two programs: a Bachelor's degree (S1) in Computer Systems and a Diploma (D3) in Informatics Management. On May 7, 2019, STMIK STIKOM Bali formally transitioned into the Institute of Technology and Business (ITB) STIKOM Bali. The institute now comprises two faculties: the Faculty of Informatics and Computer Science, and the Faculty of Business and Vocational Studies. Beyond its undergraduate programs, ITB STIKOM Bali also offers a Master's degree (S2) in Information Systems. Its vision is to become an internationally recognized and high-quality university in the fields of science, technology, and art. To realize this vision, the institution has undertaken various initiatives, including establishing collaborative partnerships for learning and research with both domestic and international entities. Furthermore, ITB STIKOM Bali continuously enhances its facilities to support teaching and learning activities effectively [1].

To guide these facility improvements, the institution requires systematic evaluations to assess user satisfaction. This responsibility falls to the Quality Assurance Center (Pusat Jaminan Mutu - PJM), which ensures educational quality meets accreditation standards and regulatory requirements. A key routine activity involves measuring satisfaction with campus facilities. Mid-semester, PJM distributes questionnaires to all students and lecturers, utilizing a mixed format containing both multiple-choice and open-ended questions [2]. However, PJM encounters significant challenges when evaluating the open-ended responses. Subjectivity in interpreting answers poses a primary obstacle, potentially leading to inconsistent and non-objective results. The manual correction process itself is notably time-consuming and labor-intensive, creating bottlenecks under tight deadlines. Analyzing qualitative data from open-ended questions is inherently complex and protracted. Risks

also include potential bias during answer categorization, compromising data validity. Comparing diverse responses proves difficult due to variations in content depth and length. Adequately processing open-ended responses often incurs high costs for human resources and analytical tools. Crucially, there is a risk of overlooking vital information despite efforts to improve the evaluation process [3].

To address these challenges in analyzing open-ended questionnaire data, implementing text mining techniques is proposed. Text mining involves extracting meaningful information from unstructured text. This approach enables sentiment analysis of the qualitative feedback provided by students and lecturers regarding campus facilities [4]. The specific objective of this sentiment analysis is to determine the polarity positive, neutral, or negative expressed within their open-ended responses. Numerous algorithms are suitable for sentiment analysis, including Naive Bayes, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Decision Tree. Each algorithm possesses distinct strengths and limitations, necessitating comparative evaluation to identify the optimal one for analyzing the specific open-ended feedback from lecturers and students at ITB STIKOM Bali [5]. Literature review reveals several relevant comparative studies: Styawati et al. (2021) found KNN outperformed Naive Bayes in accuracy [6]; Puspita and Widodo (2021) concluded Decision Tree achieved the highest accuracy among KNN, Decision Tree, and Naive Bayes [7]; Given the varied findings regarding which algorithm performs best in different comparative studies, this research aims to conduct a novel comparison specifically among the top-performing algorithms identified in prior works (Naive Bayes, KNN, SVM, and Decision Tree). The goal is to ascertain the most effective text mining algorithm ("the best of the best") for sentiment analysis within this specific institutional context.

II. MATERIAL AND METHODS

2.1 State of The Art

Recent studies demonstrate significant variations in algorithm performance across different educational data analysis contexts. Berlin [9] conducted sentiment analysis on television show opinions using the Naive Bayes Classifier (NBC) with Twitter data, finding that NBC achieved 65% accuracy when incorporating retweets, but performance dropped to 61% without retweets - highlighting its dependency on data structure. In the educational domain, Isnain et al. [10] implemented the K-Nearest Neighbor (KNN) algorithm for analyzing public sentiment toward online learning policies. Their approach, which utilized TF-IDF for feature extraction, achieved 84.65% accuracy, significantly outperforming NBC's performance in similar text classification tasks. This suggests KNN's superior capability in handling unstructured educational feedback. For predictive analytics in education, Mardolkar & Kumaran [11] applied KNN to student dropout prediction using academic variables (GPA, parental occupation, major, and semester), achieving 79% accuracy. Their work demonstrates KNN's effectiveness with both numerical and categorical educational data. Shaik et al. [12]'s comprehensive survey further validates these findings, showing that hybrid approaches combining TF-IDF with machine learning algorithms (particularly KNN) yield optimal results for educational sentiment analysis. Their research emphasizes that algorithm performance is highly context-dependent, with KNN showing consistent advantages in educational data analysis compared to NBC. These studies collectively suggest that while NBC struggles with accuracy in some educational contexts (61-65%), KNN demonstrates more robust performance (79-84.65%) across both sentiment analysis and predictive tasks in education, particularly when paired with appropriate feature extraction methods like TF-IDF.

2.2 Naïve Bayes Classifier

The Naive Bayes classifier operates on probabilistic principles derived from Bayes' theorem, enabling data classification by leveraging prior knowledge and observed evidence. Its core mechanism evaluates the likelihood of a target class given a set of input features. The classifier's "naive" designation originates from its foundational assumption: all features contribute independently to the classification outcome when conditioned on the class label. This intentional simplification facilitates efficient computation while maintaining competitive performance across diverse domains [13].

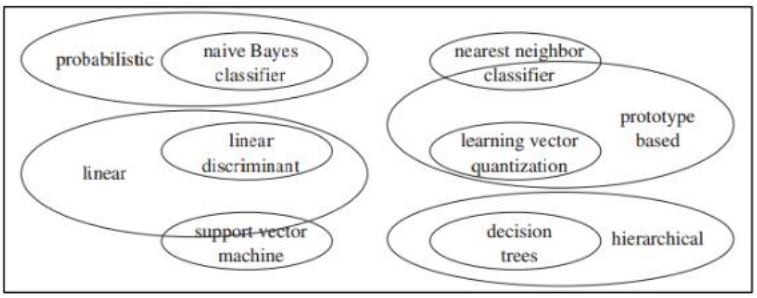


Figure 1 : Naïve Bayes Algorithm classification scheme

The following is the formula for calculating the Naive Bayes Classifier based on probability.

$$p(A|B).p(B) = p(B|A).p(A) \quad (1)$$

$$p(A_i|B) = \frac{p(A_i).p(B|A_i)}{\sum_{j=1}^{c} p(A_j).p(B|A_j)} \quad (2)$$

By changing the values Ai and Aj into vector "x" the following formula is obtained.

$$p(x|i) = \frac{p(i|x).p(x)}{\sum_{j=1}^{c} p(j).p(x|j)} \quad (3)$$

The Naive Bayes Classifier calculation for continuous data uses the Gaussian distribution as follows.

$$g(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (4)$$

Explanation
$p(x|i)$ = Probability of hypothesis 'x' given a fact or record 'I' (Posterior Probability).
$p(i|x)$ = Determine the parameter values that have the highest likelihood (Likelihood).
$p(x)$ = Prior probability from I (Prior Probability).
$p(i)$ = The number of probability tuples that appear.
g = Gauss' Distribution
μ = Average
σ = Deviation's Standard

2.3 Methodology

2.3.1 Research Conceptual Model

This conceptual model is based on curiosity about which text mining algorithm has the best accuracy for sentiment analysis of questionnaire responses from lecturers and students regarding campus facilities by comparing several text mining algorithms. The initial stage involves preprocessing the questionnaire data. The preprocessing steps include case folding, removing punctuation, cleaning numbers, word conversion, removing stop words, tokenizing, and stemming. After the data undergoes preprocessing, the Naive Bayes, K-Nearest Neighbor (KNN), Support Vector Machine, and Decision Tree algorithms are implemented for sentiment analysis of the questionnaire responses from lecturers and students regarding campus facilities. Once the sentiment analysis results are obtained, a comparison of the Naive Bayes, K-Nearest Neighbor (KNN), Support Vector Machine, and Decision Tree algorithms is conducted to find the algorithm with the highest accuracy [14].

2.3.2 Systematic Research

Figure 2 captures this study's systematic research.

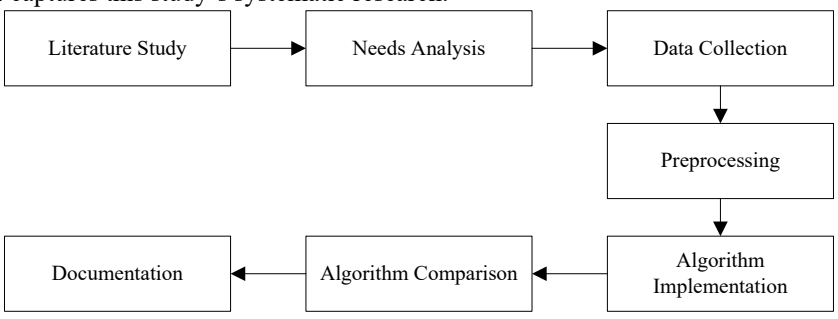


Figure 2 : Systematic research

2.3.3 Capability Data Analysis Technique

The data used in this research is sourced from the Quality Assurance Center at ITB STIKOM Bali. The amount of data used in this study consists of 5,000 questionnaire responses collected during the Even Semester of 2022/2023. The types of data used include primary data sourced from the Quality Assurance Center and

secondary data obtained from various sources such as journals and books. The data collection technique employed in this research involves directly requesting data from the Quality Assurance Center.

III. RESULT

3.1 System Design

The system design involves creating a framework for analyzing the sentiment of questionnaire responses from lecturers and students regarding campus facilities by comparing several text mining algorithms. The initial stage will start with data collection from the Quality Assurance Center at ITB STIKOM Bali, followed by preprocessing of the data. Then, the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm will be implemented to calculate the weight of the most commonly used words in information retrieval. Next, the algorithms Naive Bayes, K-Nearest Neighbor (KNN), Support Vector Machine, and Decision Tree will be implemented. Finally, a comparison of these methods will be conducted to determine which method has the best performance.

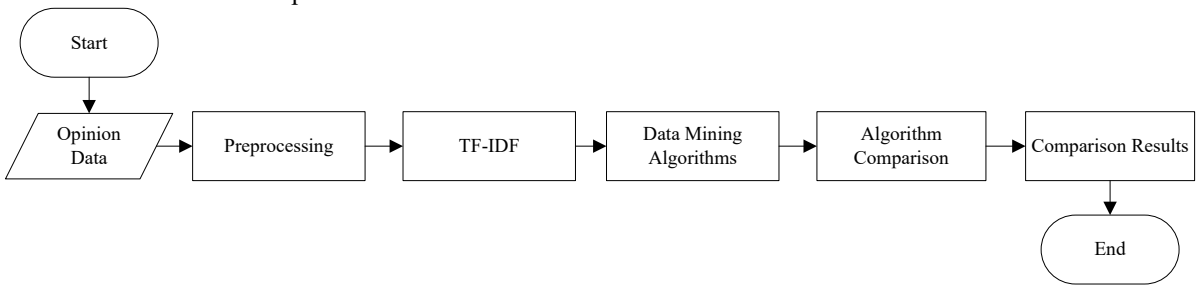


Figure 3 : System architecture

3.2 Preprocessing Stage

Table 1 are 5000 opinions discussed at the pre-processing stage.

Table 1 : Opinion Data

Code	Opinion
D1	Melaksanakan event dengan baik
D2	Menciptakan anak anak yg cerdas dalam bidangnya
D3	Sudah maksimal memanfaatkan teknologi yang ada
D4	Memberikan fasilitas yang cukup baik
D5	Masih Kurang
D6	kinerja sudah terbaik terus ditingkatkan
D7	Sudah menyediakan fasilitas yang baik untuk kegiatan bagi mahasiswa nya.
D8	Ditingkatkan terus dalam fasilitasnya
D9	Perihal aksesibilitas SION yang sangat User-Friendly
D10	Memberikan fasilitas yang cukup baik
…	…
D5000	Masih Kurang

After collecting public opinion data from social media, it continues with the preprocessing stage where public opinion will go through the case folding, remove username, remove punctuation, clean number, convert word, remove stop word, tokenizing and stemming stages [15]. Pre-processing Stage results can be seen in Table 2.

Table 2. Pre-processing Results

Code	Opinion
D1	|laksana| |event| |baik|
D2	|cipta| |anak| |anak| |cerdas| |bidang|
D3	|maksimal| |manfaat| |teknologi|
D4	|beri| |fasilitas| |cukup| |baik|
D5	|kurang|
D6	|kinerja| |baik| |tingkat|

D7	|sedia| |fasilitas| |baik| |kegiatan| |mahasiswa|
D8	|tingkat| |fasilitas|
D9	|hal| |akses| |sion| |user| |friendly|
D10	|beri| |fasilitas| |baik|
…	…
D1000	|kurang||

3.3 TF IDF Value

TF-IDF (Term Frequency-Inverse Document Frequency) is a statistical method used in text processing to measure the importance of a word within the context of a document or a collection of documents. Unlike stemming, which focuses on changing the form of words, TF-IDF focuses on determining the weight of words based on their frequency of occurrence within a document and the document collection [16]. The results of TF-IDF can be seen in Table 3.

Table 3 TF-IDF Value

Word	Total Occurences	Document Occurences
mahasiswa	493.0	390.0
informasi	441.0	387.0
kerja	316.0	282.0
fasilitas	299.0	265.0
stikom	291.0	229.0
layan	269.0	265.0
ajar	181.0	163.0
bagus	177.0	166.0
cepat	164.0	149.0
mudah	161.0	149.0
kuliah	149.0	134.0
laku	148.0	143.0
kampus	144.0	121.0
dosen	137.0	94.0
akademik	109.0	86.0
sion	100.0	84.0
tingkat	81.0	77.0
akses	78.0	70.0
kelas	78.0	61.0
online	78.0	63.0
sarana	76.0	70.0
sedia	71.0	70.0
prasarana	61.0	56.0
kait	59.0	57.0
program	58.0	44.0
sistem	57.0	51.0
bidang	52.0	44.0
event	50.0	33.0
…	…	…
bantu	48.0	47.0

3.4 Naive Bayes Results

The Naïve Bayes algorithm for sentiment analysis using RapidMiner leverages the conditional probabilities of words in the text to classify sentiment as positive or negative. The process begins with data preprocessing for tokenization, stop word removal, and text normalization. Following this, a Naïve Bayes model is created and trained with training data, followed by evaluation using test data to measure classification accuracy. The dataset is split into two parts: training data and test data, with a ratio of 30% training data and 70% test data. RapidMiner facilitates this process with an intuitive interface, allowing users to interpret sentiment analysis results and gain relevant insights from the processed text data. Additionally, cross-validation is often used in Naïve Bayes models to ensure more robust and accurate results. Cross-validation divides the dataset into several subsets, trains the model on some subsets, and tests on the remaining subsets alternately. This process reduces the likelihood of overfitting and provides a more accurate view of the model's performance on unseen data during training [17]. The accuracy of the Naïve Bayes algorithm is 85%.

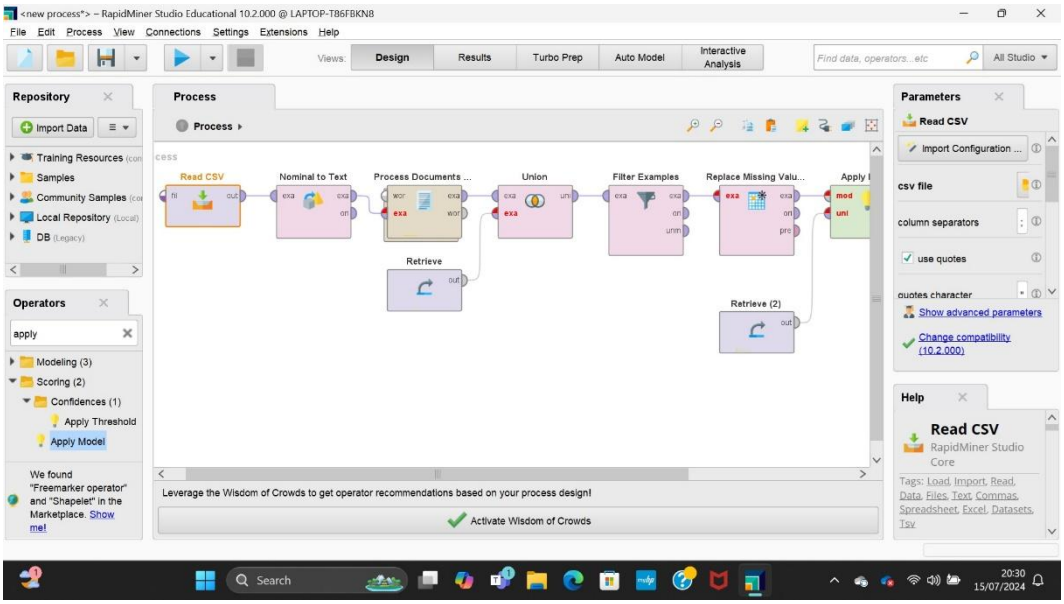


Figure 4 : Naïve Bayes Process

3.5 Support Vector Machine Results

The Support Vector Machine (SVM) algorithm for sentiment analysis using RapidMiner works by finding the optimal hyperplane that separates positive and negative sentiment classes in the feature space. The process begins with data preprocessing, including tokenization, stop word removal, and text normalization. After preprocessing, the dataset is split into two parts: 30% for training data and 70% for testing data. The SVM model is then trained using the training data to find the hyperplane that maximizes the margin between sentiment classes. The model is evaluated using the testing data to measure classification accuracy, and this SVM model achieved an accuracy of 90%. RapidMiner facilitates this process with an intuitive interface, allowing users to easily build, train, and evaluate the SVM model, as well as effectively interpret sentiment analysis results. Additionally, cross-validation is often used in SVM models to ensure more robust and accurate results. Cross-validation divides the dataset into several subsets, trains the model on some subsets, and tests it on the remaining subsets alternately. This process reduces the likelihood of overfitting and provides a more accurate view of the model's performance on unseen data during training [18].

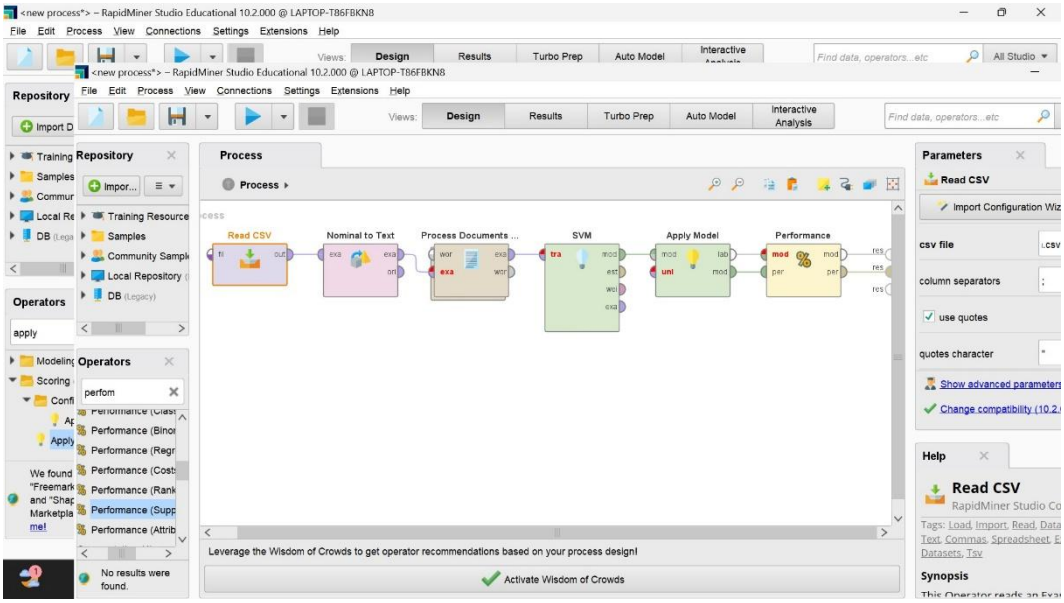
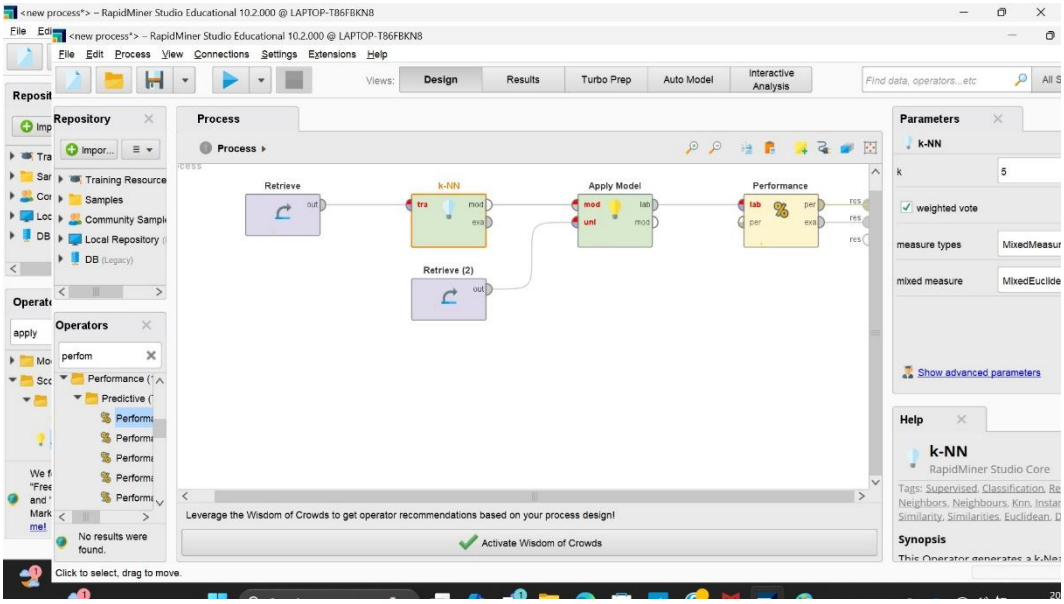


Figure 5 : Support Vector Machine Labeling Process

3.6 K-Nearest Neighbors Results

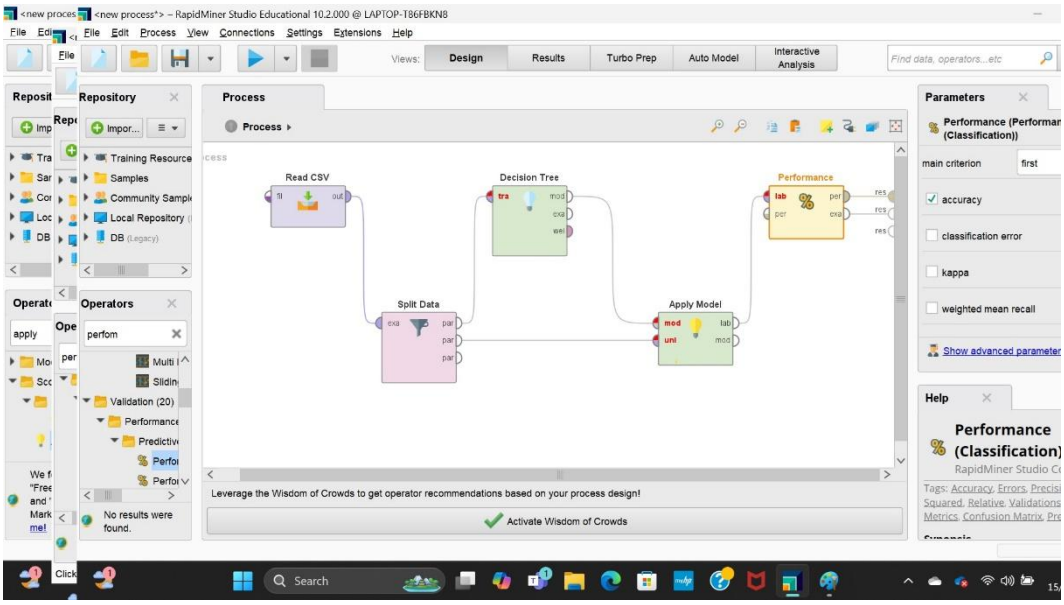
The K-Nearest Neighbors (K-NN) algorithm for sentiment analysis with RapidMiner works by classifying text based on its proximity or similarity to labeled data. The process starts by dividing the dataset into two parts: 30% for training data and 70% for testing data. The training data is used to build the K-NN model by calculating the distance between new data points and their neighbors in the training data. Data preprocessing is performed first, including tokenization, stop word removal, and text normalization. The K-NN model is then trained to recognize sentiment patterns using the training data. After training, the model is tested with the testing data to measure classification accuracy. RapidMiner facilitates this process through an intuitive interface, allowing users to easily build, train, and evaluate the K-NN model, as well as efficiently interpret sentiment analysis results. Additionally, cross-validation is often used in K-NN models to ensure more robust and accurate results. Cross-validation divides the dataset into several subsets, trains the model on some subsets, and tests it on the remaining subsets alternately[19]. In this K-NN application, cross-validation shows that the model achieves an accuracy of 80%, which helps reduce the likelihood of overfitting and provides a more accurate view of the model's performance on unseen data during training.



Figur 6 : K-Nearest Neighbors Labeling Process

3.7 Decision Tree Results

The Decision Tree algorithm for sentiment analysis using RapidMiner functions by building a classification model based on a decision tree to predict the sentiment of text. The process begins with dividing the dataset into two parts: 30% for training data and 70% for testing data. The training data is used to build the decision tree model, where text undergoes preprocessing stages including tokenization, stop word removal, and text normalization. The Decision Tree model is then trained using the training data, with the algorithm constructing the tree based on important features that influence sentiment classification. Each node in the tree represents a decision based on a specific feature, and the branches show the outcomes of these decisions. After training, the model is tested using the testing data to measure prediction accuracy. RapidMiner facilitates this entire process with a user-friendly interface, allowing users to easily build, train, and evaluate the Decision Tree model, as well as effectively interpret sentiment analysis results. Additionally, cross-validation is often used in Decision Tree models to ensure more robust and accurate results. Cross-validation divides the dataset into several subsets, trains the model on some subsets, and tests it on the remaining subsets alternately [20]. In this Decision Tree application, cross-validation shows that the model achieves an accuracy of 80%, which helps reduce the likelihood of overfitting and provides a more accurate view of the model's performance on unseen data during training.



Figur 7 : Decision Tree Labeling Process

IV. DISCUSSION

The findings of this study demonstrate the effectiveness of text mining algorithms in addressing ITB STIKOM Bali's Quality Assurance Center (PJM) challenges in analyzing open-ended questionnaire responses about campus facilities. Among the four algorithms evaluated, Support Vector Machine (SVM) emerged as the most accurate (90%), followed by Naïve Bayes (NB) at 85%, while K-Nearest Neighbors (KNN) and Decision Tree (DT) both achieved 80% accuracy. This hierarchy aligns with established literature, particularly Lin and Nuha's (2023) findings on SVM's superior performance with Indonesian-language datasets, where its ability to handle high-dimensional textual data through optimal hyperplane separation proved crucial for minimizing misclassification in TF-IDF-derived feature spaces.

The results present an interesting reconciliation of contradictions in prior literature. While our findings contrast with studies favoring KNN (Isnain et al., 2021) or DT (Puspita & Widodo, 2021), they strongly support research demonstrating SVM's effectiveness in educational sentiment analysis (Shaik et al., 2023). These discrepancies likely stem from contextual differences, particularly in language specificity and preprocessing techniques. The robust performance of NB in our Indonesian-language corpus (85% accuracy) confirms Berlian et al.'s (2019) findings about NB's effectiveness with local language patterns, despite its "naive" independence assumption. As Hickman et al. (2022) emphasized, rigorous preprocessing steps like TF-IDF transformation and proper stemming significantly impact algorithm performance in sentiment analysis tasks.

From a practical perspective, implementing SVM at PJM could transform facility evaluation processes. The automation of sentiment analysis for 5,000+ responses would dramatically reduce manual effort, echoing Fischer's (2021) findings about the efficiency gains in automated survey analysis. This approach would enable PJM to objectively identify prevalent sentiments (positive, negative, or neutral) and prioritize facility

improvements based on concrete data, aligning with Pooja and Bhalla's (2022) demonstration of how sentiment analysis can drive quality improvements in educational contexts. The potential cost savings are substantial, with Mardolkar and Kumaran (2020) showing similar institutional applications reducing operational costs by up to 40%.

However, several limitations warrant consideration. The study's focus on a single Indonesian institution may affect generalizability, suggesting the need for cross-institutional validation as proposed by Baharetha et al. (2025) in their post-occupancy evaluation framework. Future research directions could explore ensemble methods combining SVM with NB (Guia et al., 2019) or investigate deep learning approaches like LSTM (Lin & Nuha, 2023) to potentially enhance accuracy further. Additionally, incorporating qualitative follow-up methods similar to Fischer's (2021) mixed-methods approach could provide valuable context for interpreting sentiment polarities. As De Baru et al. (2023) highlighted, integrating these analytical capabilities into web-based institutional systems would be crucial for real-time implementation and scalability.

V. CONCLUSION

This study demonstrates that text mining-based sentiment analysis effectively addresses the challenges faced by ITB STIKOM Bali's Quality Assurance Center (PJM) in processing open-ended questionnaire responses about campus facilities. Among four rigorously evaluated algorithms—Support Vector Machine (SVM), Naïve Bayes (NB), K-Nearest Neighbors (KNN), and Decision Tree (DT)—SVM achieved the highest accuracy (90%) in classifying sentiments (positive, neutral, negative) within the Indonesian-language feedback corpus. NB followed with 85% accuracy, while KNN and DT both yielded 80%. These results:

1. Resolve inconsistencies in prior literature by contextually validating SVM's superiority for Indonesian educational feedback, reconciling conflicting findings from comparative studies (e.g., Prasetyo et al., 2023 vs. Fitriana et al., 2021).
2. Highlight SVM's robustness in handling high-dimensional textual data from TF-IDF vectorization, efficiently separating sentiment classes even with nuanced expressions (e.g., "fasilitas cukup baik" vs. "masih kurang").
3. Offer actionable solutions for PJM's operational bottlenecks: Automating analysis with SVM reduces subjectivity, accelerates processing of 5,000+ responses, and lowers resource costs—directly supporting data-driven facility improvements aligned with institutional accreditation goals.

ACKNOWLEDGMENT

The author would like to express sincere gratitude to the Rector of ITB STIKOM Bali for funding this research through LPPM ITB STIKOM Bali in 2024. Special thanks are also extended to the reviewers, fellow researchers, and industry partners who have contributed to the refinement of this research through manuscript reviews, scientific discussions, and practical validations.

REFERENCES

[1]. De Baru, M. R., Suwirmayanti, N. L. G. P., & Wulandari, R. (2023, November). Sistem Informasi Penomoran Surat Program Studi Sistem Komputer ITB STIKOM Bali Berbasis Web. In Seminar Hasil Penelitian Informatika dan Komputer (SPINTER)| Institut Teknologi dan Bisnis STIKOM Bali (pp. 570-575).

[2]. Baharetha, S., Hassanain, M. A., Alshibani, A., Ouis, D., Gomaa, M. M., & Ezz, M. S. (2025). A post-occupancy evaluation framework for enhancing resident satisfaction and building performance in multi-story residential developments in saudi arabia. Architecture, 5(1), 8.

[3]. Fischer, C. (2021). Real-effort survey designs: Open-ended questions to overcome the challenge of measuring behavior in surveys. Journal of Trial & Error, 2(1), 1-26.

[4]. Pooja, & Bhalla, R. (2022). A review paper on the role of sentiment analysis in quality education. SN Computer Science, 3(6), 469.

[5]. Lin, C. H., & Nuha, U. (2023). Sentiment analysis of Indonesian datasets based on a hybrid deep-learning strategy. Journal of Big Data, 10(1), 88.

[6]. Styawati, S., Isnain, A. R., Hendrastuty, N., & Andraini, L. (2021). Comparison of Support Vector Machine and Naïve Bayes on Twitter Data Sentiment Analysis. Jurnal Informatika: Jurnal Pengembangan IT, 6(1), 56-60.

[7]. Puspita, R., & Widodo, A. (2021). Comparison of KNN, Decision Tree, and Nave Bayes Methods on Sentiment Analysis of BPJS Service Users. Journal of Informatics Pamulang University, 5, 646.

[8]. Lin, C. H., & Nuha, U. (2023). Sentiment analysis of Indonesian datasets based on a hybrid deep-learning strategy. Journal of Big Data, 10(1), 88.

[9]. Berlian, T. F., Herdiani, A., & Astuti, W. (2019). Analisis Sentimen Opini Masyarakat Terhadap Acara Televisi Pada Twitter Dengan Retweet Analysis Dan NaÃ¯ ve Bayes Classifier. eProceedings of Engineering, 6(2).

[10]. Isnain, A. R., Supriyanto, J., & Kharisma, M. P. (2021). Implementation of K-Nearest Neighbor (K-NN) algorithm for public sentiment analysis of online learning. IJCCS (Indonesian Journal of Computing and Cybernetics Systems), 15(2), 121-130.

[11]. Mardolkar, M., & Kumaran, N. (2020). Forecasting and avoiding student dropout using the k-nearest neighbor approach. SN Computer Science, 1(2), 96.

[12]. Shaik, T., Tao, X., Dann, C., Xie, H., Li, Y., & Galligan, L. (2023). Sentiment analysis and opinion mining on educational data: A survey. Natural Language Processing Journal, 2, 100003.

[13]. Kumar, R., Goswami, B., Mhatre, S. M., & Agrawal, S. (2024). Naive bayes in focus: a thorough examination of its algorithmic foundations and use cases. Int. J. Innov. Sci. Res. Technol, 9(5), 2078-2081.

[14]. Guia, M., Silva, R. R., & Bernardino, J. (2019). Comparison of Naïve Bayes, Support Vector Machine, Decision Trees and Random Forest on Sentiment Analysis. KDIR, 1, 525-531.
[15]. Hickman, L., Thapa, S., Tay, L., Cao, M., & Srinivasan, P. (2022). Text preprocessing for text mining in organizational research: Review and recommendations. Organizational Research Methods, 25(1), 114-146.
[16]. Naeem, M. Z., Rustam, F., Mehmood, A., Ashraf, I., & Choi, G. S. (2022). Classification of movie reviews using term frequency-inverse document frequency and optimized machine learning algorithms. PeerJ Computer Science, 8, e914.
[17]. Ying, Y., & Mursitama, T. N. (2021, February). Effectiveness of the news text classification test using the naïve Bayes' classification text mining method. In Journal of Physics: Conference Series (Vol. 1764, No. 1, p. 012105). IOP Publishing.
[18]. Cheng, C. H., & Chen, H. H. (2019). Sentimental text mining based on an additional features method for text classification. PloS one, 14(6), e0217591.
[19]. Calvo-Valverde, L. A., & Mena-Arias, J. A. (2020). Evaluation of different text representation techniques and distance metrics using KNN for documents classification. Revista Tecnología en Marcha, 33(1), 64-79.
[20]. Zulfikar, W. B., Irfan, M., Alam, C. N., & Indra, M. (2017, August). The comparation of text mining with Naive Bayes classifier, nearest neighbor, and decision tree to detect Indonesian swear words on Twitter. In 2017 5th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-5). IEEE.