Journal of Applied Informatics and Computing (JAIC)
Vol.7, No.2, December 2023, pp. 165~171
e-ISSN: 2548-6861

Comparison of Naive Bayes Method with Support Vector Machine in Helpdesk Ticket Classification

Arief Wibowo[1], Hariyanto[2]
[1,2]Master of Computer Science, Faculty of Information Technology, Universitas Budi Luhur, Indonesia
arief.wibowo@budiluhur.ac.id [1], 22211601097@student.budiluhur.ac.id [2]

Article Info

Article history:

Received 2023-08-23
Revised 2023-10-01
Accepted 2023-10-03

Keyword:

Helpdesk,
Machine Learning,
Naïve Bayes,
Support Vector Machine,
Text Mining.

ABSTRACT

The technical support department or helpdesk department is a unit that requires a quick response in handling its tasks. The company's helpdesk team can consist of several individuals who know specific or specialized issues. Typically, technical problems are handled with an application that can track issues based on tickets. Ticket queue systems are used to facilitate control over the actions of the service or repair provided by the team. Helpdesk applications assist in addressing issues reported by users and then help upper-level management distribute tasks and monitor the helpdesk team's performance, including providing solutions to users' various problems. This research aims to predict the placement of fields that serve assistance based on the corpus users provide in the natural language. Prediction modelling is done using the Naïve Bayes and Support Vector Machine algorithms. The modelling results show that the accuracy rate of helpdesk service prediction with the Naïve Bayes algorithm reaches 82.06%, while the accuracy rate of prediction with the Support Vector Machine algorithm reaches 85.30%.

I. INTRODUCTION

In facing the increasing operational activities of the company and future challenges, the need for required applications is also growing. The Information Technology (IT) field, responsible for the operational aspects of application services, will be affected operationally due to the increase in the number of questions and complaints related to the applications used in the company. This will make IT staff performance less efficient if they have to wait for manual authorization from the support team, which can also cause mistakes and possible delays because the authorization process is still manual [1].

In this regard, management sees the need to create a unique portal as a means to accommodate questions and complaints related to the implementation of work related to the recording of active and retired participant transactions, investment management, income recording or management, including financial reports, related to the use of applications to support business processes. This is why a helpdesk application is created as part of the Dapenbun portal. This helpdesk application aims to facilitate questions about the business process or application usage. Users can submit questions or support requests through access to the portal and select the "create ticket" option in the Helpdesk application.

The IT field is interested in obtaining user information regarding their needed services. Providing labels such as "development" and "maintenance" can help improve the quality of services in the future. One of the challenges is how to perform relevant grouping in addressing user needs. The approach used is text mining/text processing.

In text classification, text mining methods are often used, a variation of data mining aimed at discovering interesting patterns in large amounts of text data. The first stage in text mining involves pre-processing the document collection, such as text classification, information extraction, and term extraction. Then, the results are stored in intermediate representations, such as distribution analysis, clustering, trend analysis, and association rules. Finally, these results can be visualized [2]. Text mining can extract and find helpful information from text data [3]. Several examples of applications related to text classification include sentiment analysis, document classification, spam classification, and document summarization [4].

Previous research has explored similar themes. For instance, a study employed the Naïve Bayes algorithm to classify the graduation status of students at Dian Nuswantoro University, achieving an accuracy rate of 82.08% [5]. Another investigation focused on classifying customer Twitter data in the myTelkomsel category using the Support Vector Machine (SVM) method, specifically within the context of Telekomunikasi Selular, attaining the highest accuracy of 81.56% [6]. Additionally, a study categorized user complaints regarding KAI Access ticket reservations, employing both SVM and Naïve Bayes algorithms. The findings indicated a higher accuracy when utilizing the SVM algorithm compared to Naïve Bayes [7]. Furthermore, research applied text mining to handle customer complaints within a relevant division, using the decision tree method and achieving an accuracy rate of 71.74% [8]. Lastly, a study classified helpdesk queries using the Support Vector Machine method, yielding an accuracy rate of 78% [9].

This research aims to predict a piece of text information about helpdesk problems that will be handled by what field, this is very necessary for service improvement and efficiency in Human Resources (HR), and the algorithms used are Naïve Bayes and Support Vector Machine. Naive Bayes is a machine learning algorithm that uses probabilistic and statistical calculations. This algorithm was first introduced by British scientist Thomas Bayes in 1960 [2], Naive Bayes algorithm is one of the statistical classification methods that can be used to predict the probability of entering a class [3] while Support Vector Machine (SVM) is an algorithm included in the supervised learning model or supervised learning related to data analysis and pattern recognition [4], first introduced by Vapnik, Boser, and Guyon in 1992 at the Annual Workshop on Computational Learning Theory.

The use of these two algorithms is because the Naïve Bayes (NB) algorithm has the advantage of relatively short and easy modelling time, but this algorithm is highly dependent on the amount of data in each class. Thus, if the dataset is limited, the resulting accuracy will decrease [13], while the SVM algorithm is one of the more powerful machine learning classifiers and tends to be more accurate. In addition, SVM has several advantages, namely it is not prone to overfitting when training [14] and provides a globally optimal solution or result that tends to be the same for each test [9], however, SVM has the disadvantage that it can only work optimally when performing binary classification [15].

In this test, the Text Mining method is used to classify helpdesk tickets using the Naïve Bayes (NB) and Support Vector Machine (SVM) methods, combined with feature weighting using term frequency and TF-IDF. With the support of these two methods, the IT field hopes to determine the accuracy level in handling helpdesk tickets based on similarity.

II. Research Method

The research method applied in this study is a quantitative research approach. The research starts with data collection, pre-processing, feature extraction using the TF-IDF method, implementation of classification using the Naïve Bayes and Support Vector Machine (SVM) methods, and model testing. The detailed steps in the research stages can be found in Figure 1.

Data collection is performed from the ticket view located in the helpdesk database. The first step in this process is to gather transaction data from the helpdesk database using a specific query. The required data is then converted into CSV format. After that, the RapidMiner application [16] is used to pre-process the data using the Transform Cases, Tokenize, Remove Duplicate, Filter Stopwords (Dictionary), and Filter Token by Length methods.

Cross-validation is conducted using the Support Vector Machine (SVM) and Naïve Bayes (NB) algorithm. The results of this process include a format that presents accuracy, precision, and recall values from the Performance Vector (Performance-SVM) and Performance Vector (Performance-NB).

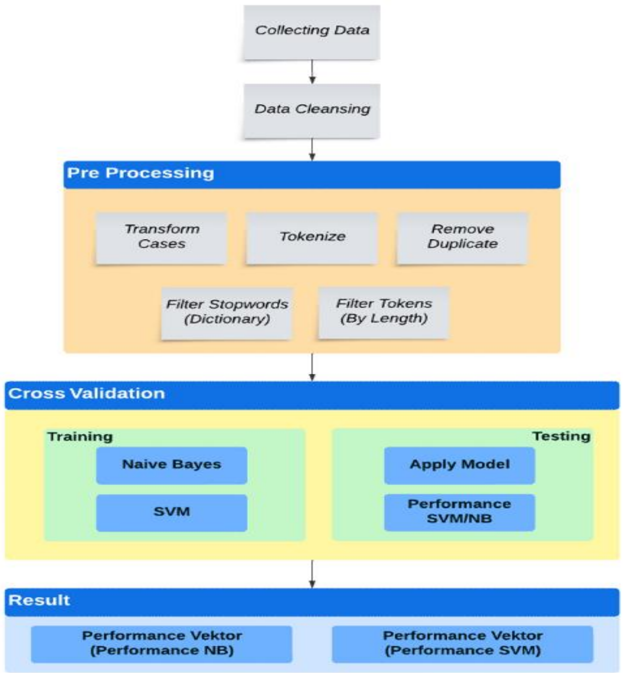

Figure 1. Research Method

III. Result and Discussion

A. Business Understanding

In the business understanding phase, which can also be referred to as the research understanding phase, data mining problems are defined and formulated.

TABLE 1.
RAW DATA

Title	Sub Division
Permintaan Install Scan, Pemindahan Data dan Pemindahan Komputer *(Request for Install Scan, Data Transfer and Computer Transfer)*	Maintenance
video zoom terkait simulasi Business Continuity Planning (BCP) dan Disaster recovery Center (DRC). *(Zoom videos related to Business Continuity Planning (BCP) and Disaster Recovery Center (DRC) simulations.)*	Maintenance
Permintaan Data Report Pensiun Gugur Seluruh Pemberi Kerja Bulan Juni s.d Agustus 2021 *(Request for All Employer Fall Retirement Report Data for June to August 2021)*	Development

In Table 1, the raw data is presented. The "title" attribute represents the user-inputted definition or problem in the helpdesk application during the ticket creation. On the other hand, the "Sub Division" attribute is filled by the team responsible for responding to the ticket.

B. Data Understanding

This phase involves understanding the data that will be used as the research subject as a preparation step before moving on to the following data processing phase. This research utilizes a database used by the helpdesk application. The "title" data is stored in the "ticket" table with a varchar attribute, while the "Sub Division" data is stored in the "sub_dept_clasification" table with a varchar attribute.

C. Data Collection

Figure 2 displays the process conducted in data collection. The data acquisition resulted in 1795 raw data entries. To obtain the required data, a specific query needs to be executed. After successfully retrieving the data, the next step is to convert it into CSV format to facilitate further analysis.

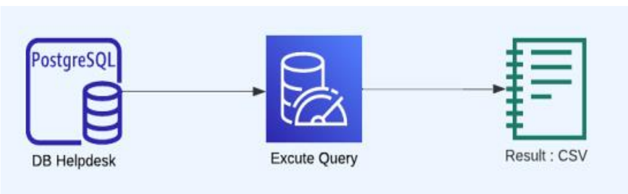

Figure 2. Data Collection Process

D. Text Pre-processing

The first step of pre-processing is data cleaning. Data cleaning ensures the accuracy, consistency, and usefulness of data in a data set. The process is to detect data errors or corrupt data and correct or delete data as needed. Based on the data generated during the collection process, 1,795 raw data entries were obtained. However, upon further examination, it was found that there were test or trial data and data that needed to be more suitable for use. Therefore, data cleansing is necessary, resulting in 1,327 data entries suitable for use in this research.

Figure 3 illustrates text pre-processing steps, an essential stage in text data classification. The goal is to eliminate noise, standardize word forms, and reduce word variations before performing text-mining processes [5].

Figure 3. Pre-processing

Here are the steps typically performed in text preprocessing:

1) Transform Cases*:* The first step in this process is to convert all letters in the sentence or document to either lowercase or uppercase. Through this transformation, the text becomes standardized in terms of capitalization.

2) Tokenizing: This process divides the text into meaningful words, sentences, or other parts. Tokenization aids in text segmentation, allowing for further processing.

3) Filter Stopwords: This step aims to eliminate words that do not carry specific meanings or frequently appear and do not contribute significantly to the analysis. For example, conjunctions and personal pronouns are often considered "stopwords" and are removed.

4) Filter Token by Length: This process involves removing words with a certain number of characters after tokenization. In this research, the minimum character length is three characters, and the maximum is 25 characters. Words with fewer than three or more than 25 characters will be removed.

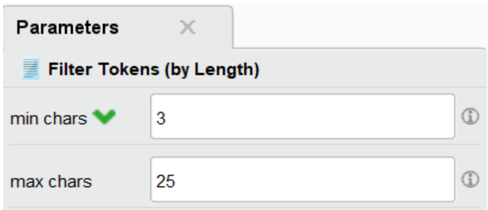

Figure 4. Parameters Filter Tokens (By Length)

Figure 4 displays the parameters that need to be input for text processing.

TABLE 2.
TEXT PROCESSING SAMPLE

Stage	Result
Raw Data	*Mohon Bantuanya Untuk Menggati Foto Do Nopes 4011012 Dengan Data Foto Pensiun Dan Ktp Berikut Dikarenakan Resolusi Foto Kurang Jelas* (Please Help To Replace The Photo Of Do Participant Number 4011012 With The Following Pension Photo Data And ID Card Due To Unclear Photo Resolution)
Transform Case	*mohon bantuanya untuk menggati foto do nopes 4011012 dengan data foto pensiun dan ktp berikut dikarenakan resolusi foto kurang jelas* (please help to replace the photo of do participant number 4011012 with the following pension photo data and id card due to unclear photo resolution)
Filter Stopwords (Dictionary)	*bantuan ganti foto do nopes 4011012 data foto pensiun ktp resolusi foto kurang jelas* (photo replacement assistance do participant number 4011012 retirement photo data ktp unclear photo resolution)
Tokenize	*bantuan, ganti, foto, do, nopes, 4011012, data, foto, pensiun, ktp, resolusi, foto, kurang, jelas* (help, replace, photo, do, participant number, 4011012, data, photo, pension, ID card, resolution, photo, less, clear)
Filter Token By Length	*bantuan ganti foto nopes 4011012 data foto pensiun ktp resolusi foto kurang jelas* (photo replacement assistance participant number 4011012 photo data pension ktp photo resolution less clear)

Meanwhile, Table 2 displays an example of Text Processing results.

E. Cross-validation

The next step is cross-validation using the Naïve Bayes (NB) and Support Vector Machine (SVM) algorithms. The steps involved in this process include the training and testing phases using the previously created models.

After the models are built, testing is conducted by applying the models to the test data. This process involves the Naïve Bayes (NB) and Support Vector Machine (SVM) algorithms to perform classification and provide prediction results.

The results of this process are analysed using evaluation metrics such as accuracy, precision, and recall. Accuracy measures how well the model can correctly classify. Precision measures how many accurate positive predictions the model made out of all optimistic predictions. In the context of classification, precision calculates the number of accurate positive predictions divided by the total number of optimistic predictions made by the model. Conversely, recall is a measure of how well the model can identify all actual positive instances. In classification, recall calculates the number of accurate optimistic predictions divided by the total number of positive instances in the dataset.

Precision and recall are closely related evaluation metrics in measuring the performance of a classification model. Precision emphasizes accuracy in predicting positive results, while recall emphasizes the model's ability to find all positive instances. In this case, performance vectors (performance-SVM) and performance vectors (performance-NB) are created to obtain the percentage values of accuracy, precision, and recall based on the used model (SVM or NB).

By performing this process, information can be obtained about the performance of the Naïve Bayes (NB) and Support Vector Machine (SVM) models in classification based on the generated performance vectors.

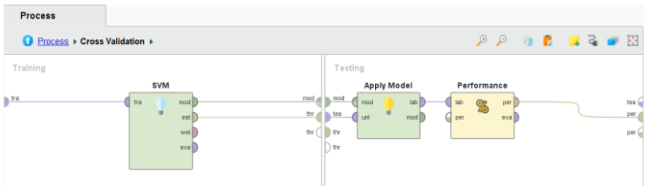

Figure 5. Cross-validation – Support Vector Machine (SVM)

Figure 5 shows the cross-validation process with the Support Vector Machine (SVM) method. In this process, two stages are performed almost simultaneously: the training and testing stages.

The pre-processed data is used to train the classification model in the training stage using the Support Vector Machine (SVM) algorithm.

After the training stage is complete and the data is ready for modeling, the testing stage is next. The model will classify the test data and generate predictions based on the information learned during the training stage. In the testing stage, the model's performance is evaluated by calculating its performance. The model's performance is assessed using evaluation metrics such as accuracy, precision, and recall. These metrics provide insights into how well the model can correctly classify and identify relevant outcomes.

By conducting both stages, an understanding can be gained of how well the SVM model can process and classify data, as well as the model's performance in terms of accuracy, precision, and recall.

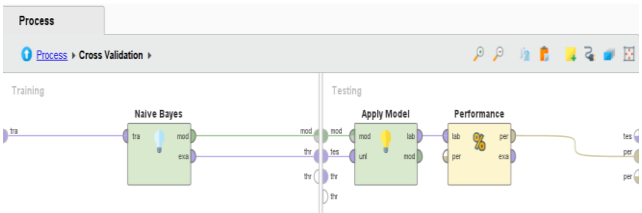

Figure 6. Cross validation – Naïve Bayes (NB)

Figure 6 shows the cross-validation process with the Naïve Bayes method. In this process, two stages are performed almost simultaneously: the training and testing stages. In the training stage, the pre-processed data is fed into the Naïve Bayes (NB) algorithm to build the classification model.

After the training stage is completed, it is followed by the testing stage. In this stage, the unprocessed data is inputted into the previously created model (apply model). The model will classify the test data and generate predictions based on the information learned during the training stage.

In the testing stage, the model's performance is evaluated by calculating its performance. Model performance evaluation involves the use of evaluation metrics such as accuracy, precision, and recall. These metrics provide insights into how well the model can correctly classify and identify relevant outcomes.

By conducting both stages, an understanding can be gained of how well the Naïve Bayes (NB) model can process and classify data, as well as the model's performance in terms of accuracy, precision, and recall.

F. Evaluation

After performing cross-validation in the training and testing stages using the Naïve Bayes algorithm, the model's performance evaluation results are as follows:

- Accuracy: 82.29%
- Precision: 86.84%
- Recall: 79.09%

The accuracy indicates how well the Naïve Bayes model performs overall classification. In this case, the model achieves an accuracy of 82.29%, indicating that approximately 82.29% of the test data is classified correctly.

Precision represents the number of accurate positive results generated by the model. In this case, the model achieves a precision of 86.84%, meaning that approximately 86.84% of the positive classifications given by the model are true positives.

Recall measures how well the model can identify positive results overall. In this case, the model achieves a recall of 79.09%, indicating that the model can correctly identify approximately 79.09% of the total positive instances.

These performance evaluation results provide an overview of how well the Naïve Bayes model can classify data correctly and to what extent the model can recognize relevant outcomes.

accuracy: 82.29% +/- 3.00% (micro average: 82.29%)

	true Development	true Maintenance	class precision
pred. Development	529	149	78.02%
pred. Maintenance	86	563	86.75%
class recall	86.02%	79.07%	

Figure 7. The Accuracy value of Naïve Bayes (NB) algorithm

In Figure 7, the evaluation result of accuracy obtained from the Naïve Bayes algorithm is 82.29%. Additionally, a margin of error of +/- 3.00% indicates the extent to which the results may vary. In this case, the average micro accuracy is 82.29%.

From these results, it can be concluded that the maximum achievable accuracy by the Naïve Bayes algorithm is 85.29%, and the minimum accuracy is 79.29%. Therefore, even though the actual accuracy is 82.29%, there is a possibility that the accuracy can vary within the range of 79.29% to 85.29%.

It is important to note that this margin of error provides an understanding of the reliability of the accuracy results and the degree of variability that may occur in the measurement.

precision: 86.84% +/- 3.41% (micro average: 86.75%) (positive class: Maintenance)

	true Development	true Maintenance	class precision
pred. Development	529	149	78.02%
pred. Maintenance	86	563	86.75%
class recall	86.02%	79.07%	

Figure 8. The precision value of the Naïve Bayes algorithm.

In Figure 8, using the Naïve Bayes algorithm, the obtained precision value is 86.84%. There is also a margin of error of +/- 3.41%, indicating the extent to which the precision results may vary. The average micro precision obtained is 86.75% for the positive class "maintenance."

Based on these results, it can be concluded that the maximum achievable precision value by the Naïve Bayes algorithm is 90.25%, while the minimum precision value is 83.43%. Although the actual precision value is 86.84%, there is a possibility of variation within the range of 83.43% to 90.25%.

It is important to note that the margin of error provides an understanding of the reliability of the precision results and the degree of variability that may occur in the measurement.

recall: 79.09% +/- 4.73% (micro average: 79.07%) (positive class: Maintenance)

	true Development	true Maintenance	class precision
pred. Development	529	149	78.02%
pred. Maintenance	86	563	86.75%
class recall	86.02%	79.07%	

Figure 9. The recall value of the Naïve Bayes algorithm.

In Figure 9, the Naïve Bayes algorithm yields a recall value of 79.09% in the testing. There is also a margin of error of +/- 4.73%, indicating the extent to which the recall results may vary. The average micro recall obtained is 79.07% for the positive class "maintenance."

Using this information, it can be concluded that the maximum achievable recall value by the Naïve Bayes algorithm is 83.82%, while the minimum recall value is 74.36%. Although the actual recall value is 79.07%, there is a possibility of variation within the range of 74.36% to 83.82%.

It is important to note that the margin of error provides an understanding of the reliability of the recall results and the degree of variability that may occur in the measurement.

The Support Vector Machine (SVM) algorithm achieves results as follows:

- Accuracy:85.38%
- Precision: 84.27%
- Recall: 89.47%.

Therefore, based on the performance evaluation using metrics such as accuracy, precision, and recall, the Support Vector Machine (SVM) algorithm demonstrates exemplary performance in data classification. However, it is essential to note that these results depend on the data used, algorithm parameter settings, and specific application context. Careful performance evaluation and selecting appropriate algorithms based on specific needs are essential to ensure optimal classification quality.

The accuracy rate of 85.38% indicates the extent to which this algorithm can correctly classify data. The precision value of 84.27% indicates how well the algorithm can identify relevant data, while the recall value of 89.47% indicates how well the algorithm can identify all relevant data overall.

The combination of accuracy, precision, and recall values provides an overview of the performance of the SVM algorithm in data classification, with high accuracy and an excellent ability to identify relevant data.

accuracy: 85.38% +/- 2.67% (micro average: 85.38%)

	true Development	true Maintenance	class precision
pred. Development	496	75	86.87%
pred. Maintenance	119	637	84.26%
class recall	80.65%	89.47%	

Figure 10. The accuracy value of the Support Vector Machine (SVM) algorithm.

In Figure 10, the Support Vector Machine (SVM) algorithm yields an accuracy value of 85.38% in cross-validation. Within the margin range of +/- 2.67%, the average microaccuracy value is 85.38%. Thus, the maximum achievable accuracy value is 88.05%, while the minimum accuracy value is 82.71%.

This indicates that the SVM algorithm has a consistent and stable level of accuracy in data classification. The average micro accuracy value represents the algorithm's overall performance in classifying data within the margin range. The maximum and minimum accuracy values provide the upper and lower bounds of the algorithm's performance in terms of accuracy.

Therefore, the SVM algorithm performs well in data classification, with a relatively high and stable level of accuracy within the specified margin range.

precision: 84.27% +/- 2.34% (micro average: 84.26%) (positive class: Maintenance)

	true Development	true Maintenance	class precision
pred. Development	496	75	86.87%
pred. Maintenance	119	637	84.26%
class recall	80.65%	89.47%	

Figure 11. Precision value algorithm Support Vector Machine (SVM)

In Figure 11, the Support Vector Machine (SVM) algorithm yields a precision value of 84.27% in cross-validation. Within the margin range of +/- 2.34%, the average micro precision value is 84.26%. Thus, the maximum achievable precision value is 86.61%, while the minimum precision value is 81.93%.

This indicates that the SVM algorithm has a stable and consistent level of precision in data classification. The average micro precision value represents the algorithm's overall performance in classifying data within the margin range. The maximum and minimum precision values provide the upper and lower bounds of the algorithm's performance in terms of precision.

recall: 89.47% +/- 3.18% (micro average: 89.47%) (positive class: Maintenance)

	true Development	true Maintenance	class precision
pred. Development	496	75	86.87%
pred. Maintenance	119	637	84.26%
class recall	80.65%	89.47%	

Figure 12. Recall value algorithm Support Vector Machine (SVM)

In Figure 12, the Support Vector Machine (SVM) algorithm yields a recall value of 89.47% in cross-validation. Within the margin range of +/- 3.18%, the average micro recall value is 89.47%. Thus, the maximum achievable recall value is 92.65%, while the minimum is 86.29%.

These results indicate that the SVM algorithm has a stable and consistent level of recall in classifying data. The average micro-recall value represents the algorithm's overall performance in recognizing and retrieving relevant data within the margin range. The maximum and minimum recall values provide the upper and lower bounds of the algorithm's performance in terms of recall.

Therefore, the SVM algorithm performs well in data classification, with a relatively high and stable level of recall within the specified margin range. This indicates that the SVM algorithm can accurately recognize and retrieve relevant data.

Based on the conducted testing and in-depth analysis, the following findings are discovered:

1) Out of the total 1327 helpdesk tickets, 615 tickets are categorized as Development tickets and 712 tickets are categorized as Maintenance tickets. This indicates a difference in the ticket distribution between the Development and Maintenance categories.
2) Implementing the Support Vector Machine (SVM) algorithm yields an accuracy rate of 85.38%. This means that SVM can classify helpdesk tickets with a high percentage of accuracy.
3) Implementing the Naïve Bayes algorithm yields an accuracy rate of 82.29%. This indicates that Naïve Bayes can also classify helpdesk tickets with good accuracy.

4) In addition to accuracy, the results of implementing the SVM algorithm also show a precision value of 84.27% and a recall value of 89.47%. On the other hand, implementing the Naïve Bayes algorithm has a precision value of 86.84% and a recall value of 79.09%.
5) When comparing the accuracy of previous research with this research, it can be concluded that the use of the Support Vector Machine (SVM) algorithm produces higher accuracy than Naïve Bayes in classifying helpdesk tickets.
6) The importance of the selected data timeframe in this study is also highlighted. A longer data timeframe can result in more diverse training data, thereby improving the accuracy and performance of the classification model.

Based on the test results that have been carried out, in classifying helpdesk tickets, by comparing the use of the Naïve Bayes algorithm with the Support Vector Machine, the results obtained a higher level of accuracy is Support Vector Machine, this result will increase the effectiveness of using the helpdesk application because the determination of the division that will handle it has been done automatically using the Support Vector Machine algorithm.

IV. CONCLUSION

Based on test data in previous studies that produced an average accuracy rate of 81.12%, while in this study using the Support Vector Machine (SVM) algorithm obtained an accuracy of 85.38% much higher when compared to the results of previous studies.

The test results indicate that the Support Vector Machine (SVM) algorithm performs better in the classification of helpdesk tickets than the Naïve Bayes algorithm. Specifically, SVM achieves higher accuracy and better classification capabilities for helpdesk tickets. However, it is essential to note that algorithm performance can vary depending on the dataset used and the appropriate parameter settings. Thorough performance evaluation and algorithm selection are crucial to ensure optimal classification quality in specific application contexts.

Extending the time range used in the training data is necessary to improve the accuracy level of the data. We can gather more diverse data by using a more extended time range. This will help enhance the accuracy of the classification model since more data allows the model to learn patterns and variabilities within the data more effectively.

By expanding the time range of the data used in the data collection process, we can address the issue of limited variation in the training data and improve the model's ability to classify helpdesk tickets with higher accuracy. Involving data from a more comprehensive time range allows the model to learn from more examples and patterns from more significant variations in helpdesk tickets. This can assist the model in recognizing and classifying tickets more accurately.

Therefore, it is recommended to consider extending the time range to collect training data to improve the accuracy level in the helpdesk ticket classification process. However, it is essential to ensure that the collected data remains relevant to the context and changes occurring in the helpdesk application so that the model can learn from data representing the actual conditions.

REFERENCES

[1] Altintaġ and A. C. Tantuğ, “Machine Learning-Based Ticket Classification in Issue-Tracking Systems,” 2014. [Online]. Available: http://worldconferences.net/

[2] R. Feldman and J. Sanger, *The text mining handbook : advanced approaches in analyzing unstructured data*. Cambridge University Press, 2007.

[3] J. Han, M. Kamber, and J. Pei, “Data Mining. Concepts and Techniques, 3rd Edition (The Morgan Kaufmann Series in Data Management Systems),” 2011.

[4] A. Kulkarni and A. Shivananda, *Natural Language Processing Recipes*. Apress, 2019. doi: 10.1007/978-1-4842-4267-4.

[5] Yuda S. Nugroho, “Data Mining Menggunakan Algoritma Naïve Bayes Untuk Klasifikasi Kelulusan Mahasiswa Universitas Dian Nuswantoro.”

[6] S. Prayoginingsih and R. P. Kusumawardani, “Klasifikasi Data Twitter Pelanggan Berdasarkan Kategori myTelkomsel Menggunakan Metode Support Vector Machine (SVM) Studi Kasus: Telekomunikasi Selular,” 2017.

[7] H. T. A. Antonius Yadi Kuntoro, “Klasifikasi Keluhan Pengguna KAI Access Untuk Pemesanan Tiket Dengan Algoritma SVM Dan Naïve Bayes”.

[8] D. D. Saputra, B. Pratama, Y. Akbar, and W. Gata, “Penerapan Text Mining Untuk Assignment Complaint Handling Customer Terhadap Divisi Terkait Menggunakan Metode Decision Tree Algoritma C4.5 (Studi Case : PT. XL Axiata, Tbk),” *CKI On SPOT*, vol. 11, no. 2, 2018.

[9] S. Hilda Kusumahadi, H. Junaedi, and J. Santoso, “Klasifikasi Helpdesk Menggunakan Metode Support Vector Machine,” *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 4, no. 1, pp. 54–60, Jan. 2019, doi: 10.30591/jpit.v4i1.1125.

[10] S. Russell and P. Norvig, “Artificial Intelligence A Modern Approach Third Edition,” 2010.

[11] I. N. T. Wirawan and I. Eksistyanto, “Penerapan Naive Bayes Pada Intrusion Detection System Dengan Diskritisasi Variabel.”

[12] C. Putra and E. Irawati, “Algoritma Support Vector Machine Untuk Mendeteksi Sms Spam Berbahasa Indonesia,” 2015.

[13] Muhammad Yusril Helmi Setyawan, Rolly Maulana Awangga, and Safif Rafi Efendi, *Comparison Of Multinomial Naive Bayes Algorithm And Logistic Regression For Intent Classification In Chatbot*.

[14] D. M. Abdullah, “Machine Learning Applications based on SVM Classification: A Review”, doi: 10.48161/Issn.2709-8206.

[15] D. N. Fitriana and Y. Sibaroni, “Sentiment Analysis on KAI Twitter Post Using Multiclass Support Vector Machine (SVM),” *Accredited by National Journal Accreditation*, vol. 4, no. 2, pp. 846–853, 2020, [Online]. Available: http://jurnal.iaii.or.id

[16] S. Land and S. Fischer, “RapidMiner 5 RapidMiner in academic use.” [Online]. Available: www.rapid-i.com

[17] M. W. Berry and J. Kog, “Text Mining: Applications and Theory,” 2010