# Sentiment Classification of IT Service Feedback via TF-IDF

**Samidi*[1], Devy Fatmawati[2]**
[1,2]Universitas Budi Luhur; Jl. Ciledug Raya, Jakarta Selatan, Indonesia
[1,2]Program Studi Ilmu Komputer, Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta
e-mail: ***[1]samidi@budiluhur.ac.id**, [2]2111601353@student.budiluhur.ac.id

***Abstract***

*Handling user complaints and feedback is a key strategy of Pusintek, the Ministry of Finance of the Republic of Indonesia, to enhance user satisfaction. The challenge faced is the difficulty in accurately analyzing feedback due to differences in comments and categories chosen by users, which requires manual category correction. This study aims to automate feedback comment categorization using classification algorithms. Specifically, Naïve Bayes, Support Vector Machine (SVM), and K-Nearest Neighbors (K-NN) algorithms were applied to 11,108 user feedback records. The CRISP-DM framework was used, with dataset preparation involving sentiment analysis techniques (cleansing, case folding, normalization, filtering, and tokenization) and Term Frequency-Inverse Document Frequency (TF-IDF) weighting. Accuracy values for each algorithm were evaluated. Results show that the SVM algorithm performed the best, achieving an accuracy of 94.10% and consistently delivering the highest precision, recall, and f1-score across all sentiment categories. This research contributes to the development of an automatic feedback classification system that improves categorization accuracy, minimizes manual intervention, and optimizes user feedback analysis. It is expected to enrich the understanding of text classification and natural language processing techniques and open up opportunities for further research.*

.

***Keywords***: Classification, Feedback, Naïve Bayes, SVM, K-NN

**Abstrak**

Penanganan keluhan dan feedback pengguna merupakan strategi utama Pusintek, Kementerian Keuangan RI, untuk meningkatkan kepuasan pengguna. Tantangan yang dihadapi organisasi adalah kesulitan dalam menganalisis feedback secara akurat karena perbedaan komentar dan kategori yang dipilih pengguna, sehingga memerlukan koreksi manual pada kategori.. Penelitian ini bertujuan untuk mengotomatiskan kategorisasi komentar *feedback* menggunakan algoritma klasifikasi. Secara khusus, diterapkan algoritma *Naïve Bayes*, *Support Vector Machine* (SVM), dan *K-Nearest Neighbors* (K-NN) pada sejumlah 11.108 *records* komentar *feedback* pengguna. Kerangka kerja CRISP-DM digunakan sebagai metode dalam pelaksanaan penelitian, dimana persiapan dataset dilakukan melalui teknik analisis sentimen *(cleansing, case folding, normalization, filtering, dan tokenization)* serta pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF). Nilai akurasi masing-masing algoritma kemudian dievaluasi. Hasilnya menunjukkan bahwa algoritma SVM mencapai kinerja tertinggi, dengan nilai akurasi 94,10%, secara konsisten memberikan nilai presisi, *recall*, dan *f1-score* terbaik di semua kategori sentimen. Penelitian ini berkontribusi pada pengembangan sistem klasifikasi feedback otomatis yang tidak hanya meningkatkan akurasi kategorisasi feedback, tetapi juga meminimalkan intervensi manual dan mengoptimalkan proses analisis feedback pengguna. Penelitian ini diharapkan memberikan kontribusi dalam memperkaya pemahaman tentang penerapan teknik-teknik terbaru dalam klasifikasi teks dan pemrosesan bahasa alami, serta membuka peluang untuk penelitian lebih lanjut.

***Kata kunci***: Klasifikasi, Feedback, Naïve Bayes, SVM, K-NN

## 1. INTRODUCTION

One of the strategies implemented by service providers to improve the quality of services provided is through complaint-handling activities. Successful complaint handling, particularly on social media, can predict an increase in repurchase intention. This is influenced by the role of trust in the company and the users' propensity to trust the service provider. [1]. Good complaint handling can of course increase organizational value, increase service user satisfaction, and provide information for the organization to improve the services provided [2].

For user satisfaction analysis needs, service providers must manage the feedback received. Over 2 (two) years, namely 2021 and 2022, feedback data was recorded containing comments received in the amount of 17,931 records. Based on the results of interviews conducted, service providers experienced problems in analyzing existing feedback. The problem that occurs is the discovery of data where the comments entered do not match the feedback category chosen by the user so corrections must be made manually for categories that do not match, where there is quite a large difference between the initial data and the correction data.

Sentiment analysis is a technique or method that can be used to determine the sentiment contained in a comment or opinion by extracting data from the comment or opinion, and then understanding and processing the existing text data [3]. Sentiment analysis enables the extraction of emotional insights from text, providing valuable information about current opinions, and making it an essential tool for understanding user perspectives effectively.[4]. Sentiment analysis can be performed by text mining. Text mining is a technique used to extract unstructured text data so that it can become useful information. Text mining, as noted by Zong, Xia, and Zhang (2021), plays a crucial role in understanding public opinion by processing and analyzing textual data to extract meaningful information. This process involves identifying key components within the text, such as keywords that represent the main topics, opinions that reflect sentiments or attitudes, and user inputs that provide additional context or details. Through this approach, text mining enables researchers and decision-makers to gain deeper insights into public perspectives and trends effectively.[5].

Classification methods can be used to perform sentiment analysis. Classification is a technique used to describe and categorize data into predefined classes. It is also widely applied to predict outcomes based on patterns and relationships within the data, making it a key method in sentiment analysis and other data-driven applications. [6]. Many classification techniques, including Naïve Bayes, Support Vector Machine (SVM), K-Nearest Neighbor (K-NN), Neural Network, and Decision Tree, can be used for sentiment analysis. However, of the five classification algorithms, the Naïve Bayes, SVM, and K-NN algorithms are the algorithms most widely used in previous research. This is because these three algorithms produce higher accuracy values than other classification algorithms in classifying sentiment. The Naïve Bayes algorithm is capable of generating real-time predictions even when working with a small dataset. Additionally, it can efficiently scale to handle larger datasets, making it a reliable choice for sentiment analysis tasks like those applied to Twitter data. [7]. Likewise, the SVM algorithm has advantages in processing data that has many features and high dimensions, and the solution provided is global for each experiment [8], [9]. This is proven by research conducted by [10], [11], [12], [13], [14] that the SVM algorithm produces the highest accuracy value than other algorithms. Meanwhile, for the K-NN algorithm, the optimal accuracy value can be obtained by choosing the right k parameters. Other research conducted by [9] used standard parameter values provided by each algorithm in classifying the sentiment analysis carried out.

Text pre-processing plays a crucial role in the dataset transformation and normalization phase. This phase significantly affects the outcomes of the subsequent classification process, highlighting the importance of proper data preparation for accurate results.[15]. The use of TF-IDF weighting in classification was carried out by [8] to classify helpdesks according to their respective job descriptions using the SVM algorithm. Based on the previous research above, this research will use classification algorithms, namely Naïve Bayes, SVM, and K-NN, which are preceded by TF-IDF weighting after preprocessing.

Based on the explanation above, to make it easier for service providers to classify the categories of feedback submitted by users, researchers are motivated to examine the problems that occur. The Naïve Bayes, SVM, and K-NN classification algorithms are used for classification, TF-IDF weighting and sentiment analysis were the steps that came before them. The reason for maintaining the classification algorithm above and the TF-IDF weighting is the success of research conducted by previous researchers so that it can be applied to research conducted by researchers. The parameters that will be used in the classification algorithm are standard provided by the Python library such as parameters c and epsilon in the SVM algorithm and parameter k in the K-NN algorithm. Of the three algorithms, accuracy, precision, recall, and f1-score values will be evaluated for each algorithm so that the best classification algorithm can be obtained in determining the feedback category for Pusintek ICT service users.

This research is important because it can help service providers in classifying feedback comments submitted by service users. So that the analysis carried out on feedback submitted by users can be used to improve the services provided and to increase service user satisfaction. It is hoped that the results of this research will be able to contribute to the progress of science and technology by proving the performance of the classification algorithm which is preceded by processing text mining and weighting using TF-IDF in sentiment analysis which is then carried out in the classification stage of a comment text so that it can be used as a reference in research multi-class classification. The process of carrying out this research will use existing standards in data mining or data mining, namely the Cross-Industry Standard Process for Data Mining (CRISP-DM) as carried out by [16], [17].

## 2. RESEARCH METHODS

This research is a descriptive and quantitative analysis type with the research stages depicted in Figure 1.

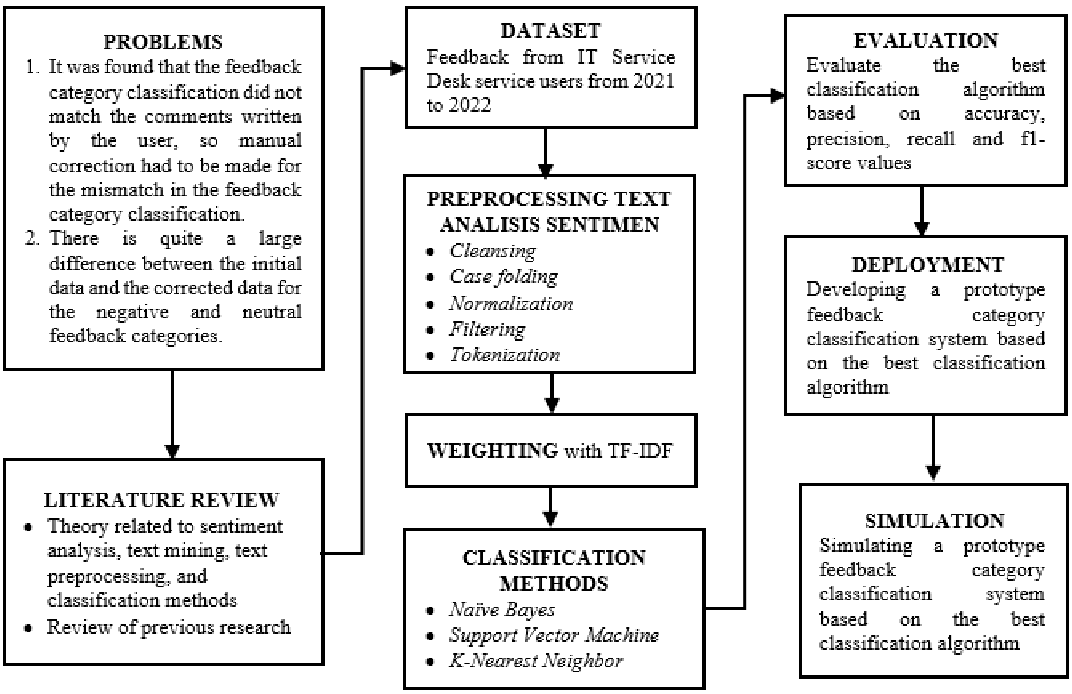


*Figure 1. Research Conceptual Framework*

The steps of the research process are shown in Figure 1, which begins with problem identification and ends with a literature review. From the results of the dataset collection, data processing was carried out using text preprocessing for sentiment analysis and weighting with TF-IDF. Next, the results of the processed data are classified using 3 (three) selected classification algorithms, namely Naïve Bayes, SVM, and K-NN which are then evaluated to produce the algorithm with the best performance. A system prototype was developed to carry out data processing so that the classification carried out could be simulated.

### *2.1. Problem Identification*

An understanding of the background of the problems faced by the research object is very necessary to determine the problem formulation and research objectives. The process began with field studies through observations of the feedback management process and interviews with related parties. The problem identified from the results of the field study was the analysis of feedback data provided by users. Where users when filling in feedback sometimes do not synchronize the selected categories with the comments they enter, which has an impact on the analysis carried out by Pusintek.

### *2.2. Literature Review*

A literature review was carried out to obtain references related to theory, methodology, and problem-solving solutions that have been carried out by other researchers previously with themes relevant to this research. What can be concluded from the literature review carried out is the use of classification algorithms, namely Naïve Bayes, SVM, and K-NN to analyze user sentiment towards ICT service management.

Based on the application of Bayes' theorem, which presumes strong independence, the Naïve Bayes algorithm is a straightforward probability-based prediction method. The algorithm treats each feature as independent from the others, simplifying the computation, and assuming that all features are conditionally independent given the class label. Despite this "naïve" assumption, the algorithm is widely used in various domains due to its efficiency and effectiveness, especially in text classification tasks like spam filtering and sentiment analysis [18]. It is computationally efficient and performs well even with large datasets, though its accuracy may decrease when features are highly correlated. Depending on the type of data, many Naïve Bayes variants are employed, including Gaussian, Multinomial, and Bernoulli Naïve Bayes. Recent studies have also improved the basic algorithm, such as incorporating feature weighting to enhance performance in specific applications like human activity classification and complaint prediction [19].

Equation (1) used for the Naïve Bayes algorithm is as follows.

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \tag{1}$$

Based on equation (1) above, P(H|X) is the probability of event H based on condition X, and P(X|H) is the probability of event.

The SVM algorithm is an algorithm that aims to separate two classes in the input space by finding the optimal hyperplane [20]. SVM works by identifying a line or plane that best divides the classes, creating a clear distinction between them. This hyperplane is defined by support vectors, which are the data points closest to the margin, and the algorithm seeks to maximize the distance between the hyperplane and these nearest data points [21]. SVM is particularly effective for classification problems, especially when dealing with high-dimensional data, and can provide accurate results even with a limited amount of data [22]. Figure 2 [21] illustrates how to separate the +1 and -1 classes to find the optimal hyperplane.

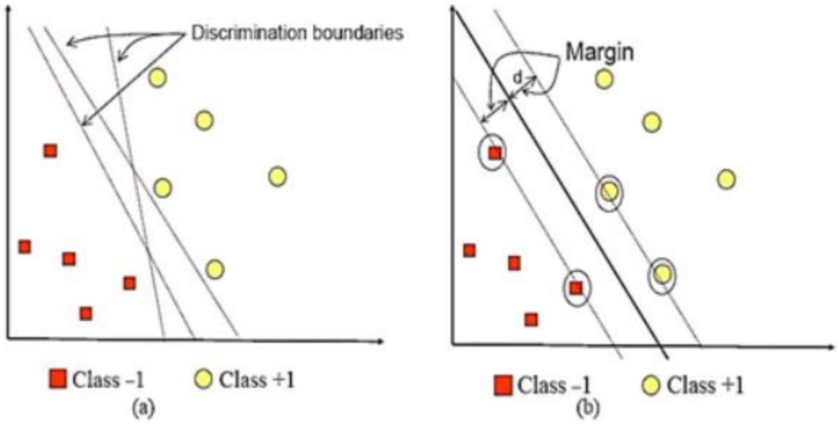


*Figure 2. The best hyperplane*

As can be seen in Figure 2 which depicts several patterns as members of two classes, namely (-1 and +1). Pattern class -1 is depicted with a red box and pattern class +1 is depicted with a yellow circle. Problems in classification occur in the process of separating the two classes with a dividing line. Figure 2(a) depicts the less than optimal line separation and Figure 2(b) depicts the optimal line separation which finds the margin between the two classes. The K-NN algorithm is an algorithm that is often used to classify objects that have neighbors with the closest distance or have the smallest difference in value from the object [23]. The algorithm works by calculating the distance between a test point and a set of labeled data points, then assigning the class label based on the majority class of the closest neighbors. The most common distance metric used is Euclidean distance, though other metrics like Manhattan or Minkowski can also be applied. K-NN is simple and intuitive, it does not require an explicit training phase but rather relies on the entire dataset for classification during the prediction phase [24]. The value of k in the k-Nearest Neighbors (kNN) algorithm is critical for determining classification results, as it represents the number of nearest neighbors used to assign a label to a test sample. If k is too large, the model may become overly general and overlook local patterns, while if k is too small, it may be overly sensitive to noise, leading to overfitting. In Fig. 3, when k is set to 5, both test samples are assigned the '+' class. While the first test sample's label is reasonable, the second should be assigned the '-' class, which would be the case if k were 3. This demonstrates that the optimal k value should be data-driven, varying based on the data distribution, and potentially different for each test sample. This principle also applies to kNN regression and missing value imputation [25].

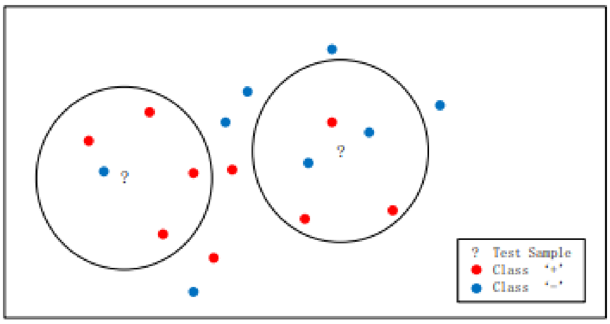


*Figure 3. An example of a K-NN classification task with k = 5 [25]*

The performance of K-NN heavily depends on the choice of the parameter "K," which determines the number of neighbors to consider, and the distance metric used. Recent studies have explored various enhancements to the basic K-NN algorithm [26]. Equation (2) for the K-NN algorithm is as follows.

$$D(x,y) = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} \quad (2)$$

Information:

$D$ = The Euclidean distance between two data points $x$ (test data) and $y$ (sample data). This value represents how similar or dissimilar the two points are based on their features.
$n$ = The number of dimensions or features in the data. For instance, if the data points include attributes like height, weight, and age, $n$ is the total number of attributes.
$i$ = data variables
$x_i$ = The i-th component of vector $x$, representing the value of the test data in dimension $i$.
$y_i$ = The i-th component of vector $y$, representing the value of the sample data in dimension $i$.
$x$ = test data
$y$ = sample data

### *2.3. Dataset*

This stage begins with initial data collection activities as research material. The dataset used is feedback comment data from service users at the IT Service Desk of the Indonesian Ministry of Finance during the period 2021 to 2022 with a total of 17,931 records. The dataset was obtained by the second author, Devy Fatmawati, by exporting it from CRM Deskpro, an application used by ICT service users to provide feedback on the services they receive, directly from the office where she works.

### *2.4. Preprocessing Text*

At this stage, text preprocessing is carried out to transform text data so that it becomes more structured so that it is more meaningful and easier to understand by going through a series of stages [18] which include cleansing, case folding, normalization, filtering, and tokenization, which is then continued with weighting and using TF-IDF.

### *2.5. Classification Methods*

After preprocessing the data, further processing will be carried out using a predetermined classification algorithm. Using the Python programming language, the model with the classification algorithm is implemented. To train the model, a dataset is needed as training data. Using the composition between the two, the dataset is randomly divided into training and test data.

### *2.6. Evaluation*

The evaluation stage involves comparing the three algorithms' f1-score, accuracy, precision, and recall scores to determine which method performs the best. The method used to evaluate the model in this research is the confusion matrix. Four metrics are utilized in the confusion matrix to measure the performance of binary classification and display the results [21]:

a. True Positive (TP) denotes a positive correlation between the actual value and the predicted result;
b. True Negative (TN) denotes a negative correlation between the actual value and the predicted result;
c. False Positive (FP) denotes a negative correlation between the actual value and the predicted result; and
d. False Negative (FN) denotes a positive correlation between the actual value and the predicted result.

The confusion matrix is defined as a summary of the prediction results following categorization, following the previously provided explanation. The number of correct and incorrect predictions is mapped to each actual value. Table 1 displays the confusion matrix visualization.

*Table 1. Confusion Matrix*

| | | Actual Value | |
|---|---|---|---|
| | | **Positive** | **Negative** |
| **Predicted Value** | **Positive** | TP | FP |
| | **Negative** | FN | TN |

### 2.7. Deployment

At this stage, a system prototype was developed to classify service user feedback regarding the fulfillment of services provided by Pusintek. The result of this stage is a system prototype that is created based on the processes that have been carried out previously and the preparation of reports on the activities carried out. Prototype development is supported by the use of hardware in the form of a laptop with an Intel Core i7 processor, 32 GB memory, and 512 GB hard disk capacity as well as software in the form of Windows 11 Professional OS, Microsoft Excel, Python programming language and Streamlit as an application to create a prototype in the form of a web application.

## 3. RESULT AND DISCUSSION

### 3.1. Business Understanding

Over 2 (two) years, namely 2021 and 2022, 17,931 feedback data were recorded which contained comments. Based on information obtained from interviews, there are obstacles faced in analyzing existing feedback. The problem that occurs is that it is discovered that the comment data entered does not match the selected feedback category and requires manual correction of the inappropriate category. Feedback categories are divided into 3 (three), Neutral, Positive, and Negative. The details of the initial feedback data are Positive 15,739, Negative 1097, and Neutral 1095. After manual correction, there was a change in the detailed numbers for each category, namely Positive became 16,089 or an increase of 2.28%, the Negative category increased to 1457 or an increase of 34.48% and the Neutral category decreased to 385 or a decrease of 64.66%. The difference in the number of categories will of course have an impact on the analysis of existing feedback data.

### 3.2. Data Understanding

The data used is feedback data from service users during the period 2021 to 2022 who have comments with a total of 17,931 feedback data. The data that has been collected into one will be processed as research material for the application of the classification algorithm method. The information that will be utilized in this study is user-submitted feedback comments, as shown in Table 2 as an example.

*Table 2. Example of a Feedback Comment*

| No | Comment | Category |
|---|---|---|
| 1 | cuma kelamaan harus nunggu 3 hari | Positive |
| 2 | belum selesai prosesnya | Positive |
| 3 | terima kasih atas bantuan...prosesnya sangat cepat | Neutral |
| 4 | Terimakasih.. saya sangat terbantu sekali.. responnya sangat cepat... | Neutral |
| 5 | Terima kasih banyak admin. Sangat membantu ^_^ | Negatif |
| 6 | Sangat membantu | Negative |
| 7 | sangat terbantu. terima kasih | Positive |
| 8 | Sangat cepat Responnya. | Positive |
| 9 | Cukup | Neutral |
| 10 | Biasa saja | Neutral |
| 11 | bikin pusing aja | Negative |
| 12 | bingung jadinya... bagaimana bisa akun email kemenkeu saya bisa dipakai orang lain | Negative |

### *3.3. Data Preparation*

At this stage, text preprocessing is carried out to transform text data so that it becomes more structured so that it is more meaningful and easier to understand by going through a series of stages which include [27] cleansing, case folding, normalization, tokenization, and filtering which is then continued with word weighting.

The first stage is cleansing, where the data obtained will be cleaned from characters such as ("!\"#$%&()*+-./:;<=>?@[\]^_`{|}~\n,). If depicted in the flowchart for this stage it is as follows.

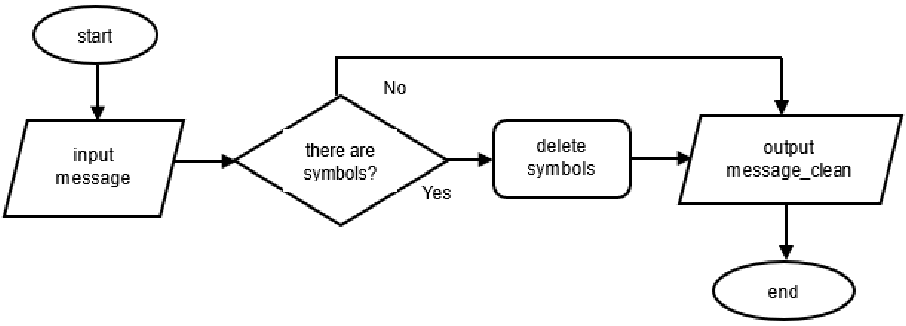


*Figure 3. Cleansing Flowchart*

Figure 3 illustrates the process of cleaning data from symbols or characters. The initial data from feedback comments will be checked to see if there are symbols, if there are, the symbols will be deleted. Table 3 displays the data produced during the purification stage.

*Table 3. Cleansing Process*

| **Input Data** | **Output Data** |
|---|---|
| Admin responsif dan solutif. Terimakasih | Admin responsif dan solutif  terimakasih |

The second step, known as case folding, involves uniformizing the letter forms across the document. Capital letters will be converted to lowercase and rendered uniform from A to Z, and digits will no longer be used as delimiters. As seen in the flowchart for this step, it looks like this.

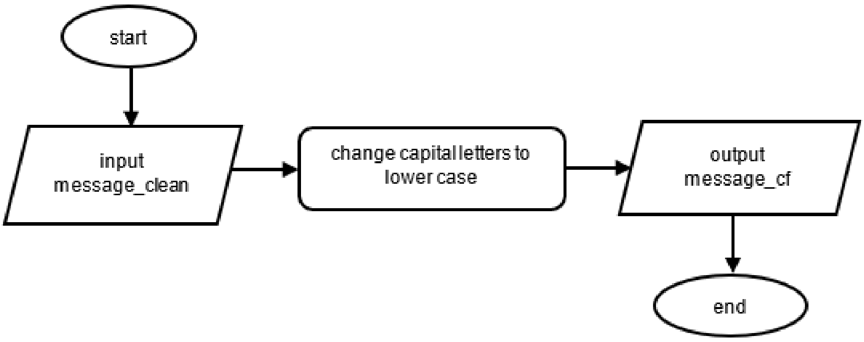


*Figure 4. Case Folding Flowchart*

Figure 4 illustrates the process of standardizing letters into lowercase letters and removing numbers for each existing feedback. Initial data from the results of the cleansing process will be adjusted using capital letters to lowercase, and numbers will be deleted. Table 4 displays the data generated from the case folding stages.

*Table 4. Case Folding Process*

| **Input Data** | **Output Data** |
|---|---|
| Admin responsif dan solutif  terimakasih | admin responsif dan solutif  terimakasih |

The third stage is normalization which aims to normalize sentences so that slang sentences become normal sentences and remove repeated letters such as "adaaa" to become "ada". If depicted in the flowchart for this stage it is as follows.

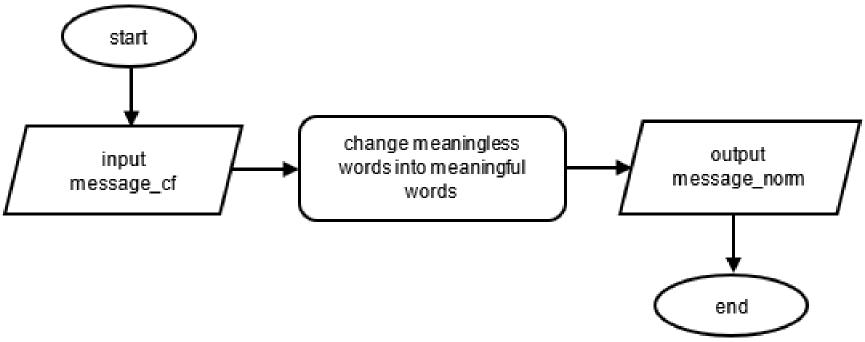


*Figure 5. Normalization Flowchart*

Figure 5 depicts the normalization process. The initial data from the results of the case folding process will change words from meaningless to words that have meaning. Table 5 displays the data resulting from the normalization stage.

*Table 5. Normalization Process*

| Input Data | Output Data |
|---|---|
| admin responsif dan solutif  terimakasih | admin responsif dan solutif terimakasih |

The fourth stage is filtering, where at this stage important words are taken from the normalization results. This process can be done with a stoplist algorithm. If depicted in the flowchart for this stage it is as follows.

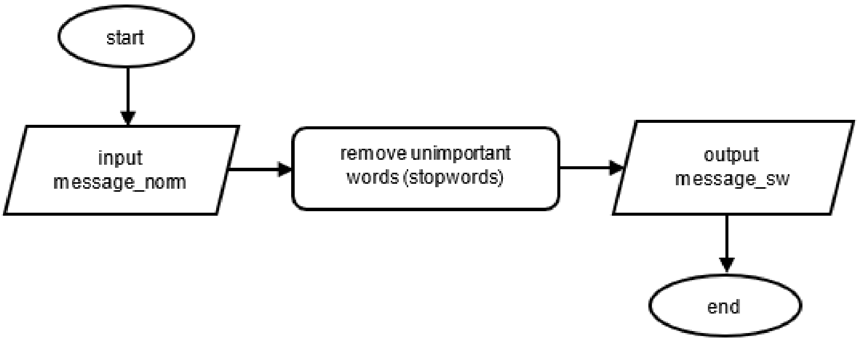


*Figure 6. Filtering Flowchart*

Figure 6 depicts the filtering process. Initial data from the results of the normalization process will remove unimportant words. Table 6 displays the data resulting from the filtering stage.

*Table 6. Filtering Process*

| Input Data | Output Data |
|---|---|
| admin responsif dan solutif  terimakasih | admin responsif solutif terimakasih |

The fifth stage is tokenization, where at this stage the process of cutting the input string is carried out based on each word that makes it up. If depicted in the flowchart for this stage it is as follows.

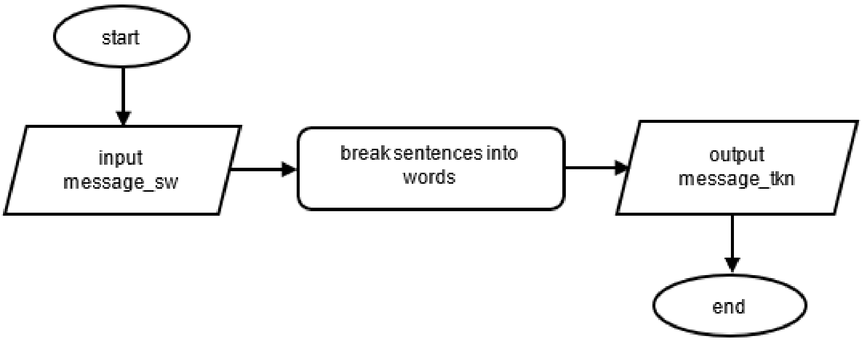


*Figure 7. Tokenization Flowchart*

Figure 7 depicts the tokenization process. Initial data from the results of the filtering process will break down sentences into words. Table 7 displays the data generated from the tokenization stage.

*Table 7. Tokenization Process*

| Input Data | Output Data |
|---|---|
| admin responsif solutif terimakasih | ['admin', 'responsif', 'solutif', 'terimakasih'] |

After carrying out the tokenization stage, the number of datasets from 17,931 records becomes 11,108 records which will be labeled using the Lexicon method which is divided into 3 (three) classes, namely positive, negative, and neutral. At this stage, the Part-Of-Speech Tagging process is carried out, where the data will be labeled for each word according to its word class such as adjectives, adverbs, verbs, and so on. If depicted in the flowchart for the labeling stages it is as follows.

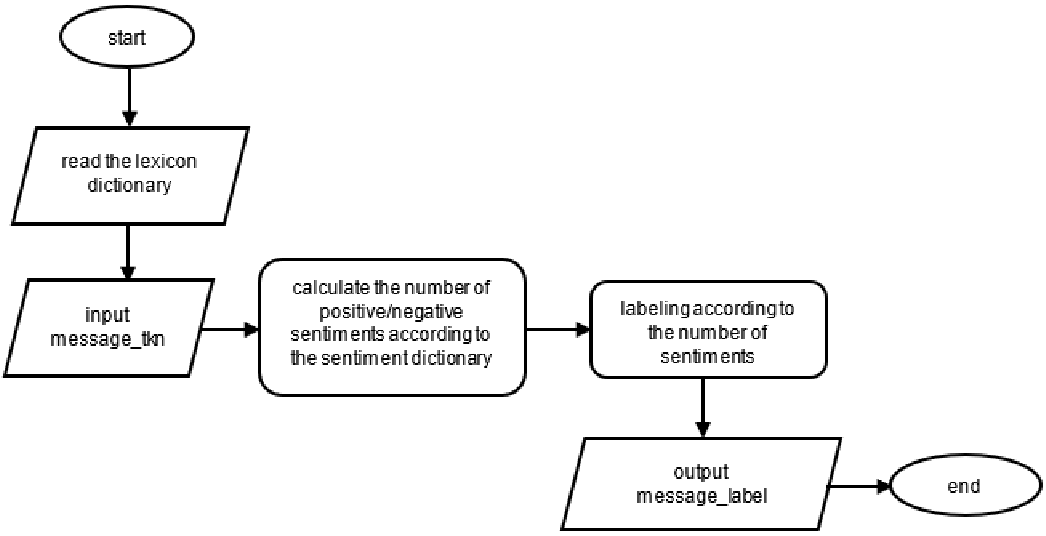


*Figure 8. Labeling Flowchart*

Figure 8 illustrates the labeling process using a lexicon dictionary. Initial data from the results of the tokenization process will be calculated by calculating the number of positive and negative sentiments for each word according to the lexicon dictionary, followed by labeling according to the number of sentiments for each existing feedback. Table 8 displays the data resulting from the labeling stage.

*Table 8. Labeling Process*

| No | Comment | Category | Labeling Result |
|---|---|---|---|
| 1 | cuma kelamaan harus nunggu 3 hari | Positive | **Neutral** |

| | | | |
|---|---|---|---|
| 2 | belum selesai prosesnya | Positive | **Negative** |
| 3 | terima kasih atas bantuan...prosesnya sangat cepat | Neutral | **Positive** |
| 4 | Terimakasih.. saya sangat terbantu sekali.. responnya sangat cepat... | Neutral | **Positive** |
| 5 | Terima kasih banyak admin. Sangat membantu ^_^ | Negative | **Positive** |
| 6 | Sangat membantu | Negative | **Positive** |
| 7 | sangat terbantu. terima kasih | Positive | Positive |
| 8 | Sangat cepat Responnya. | Positive | Positive |
| 9 | Cukup | Neutral | Neutral |
| 10 | Biasa saja | Neutral | Neutral |
| 11 | bikin pusing aja | Negative | Negative |
| 12 | bingung jadinya....bagaimana bisa akun email kemenkeu saya bisa dipakai orang lain | Negative | Negative |

From the labeling results in Table 8, there are several feedback categories filled in by users that do not match the results of the sentiment analysis carried out. To ensure that the feedback categories are by the sentiment analysis carried out previously, the next stage carried out is TF-IDF weighting. At this stage, the process of extracting text data into a numerical data matrix is carried out. In this research, the TfidfVectorizer library from the Sci-Kit Learn module was used to carry out the feature extraction and weighting process. If depicted in the flowchart for the weighting stages it is as follows.

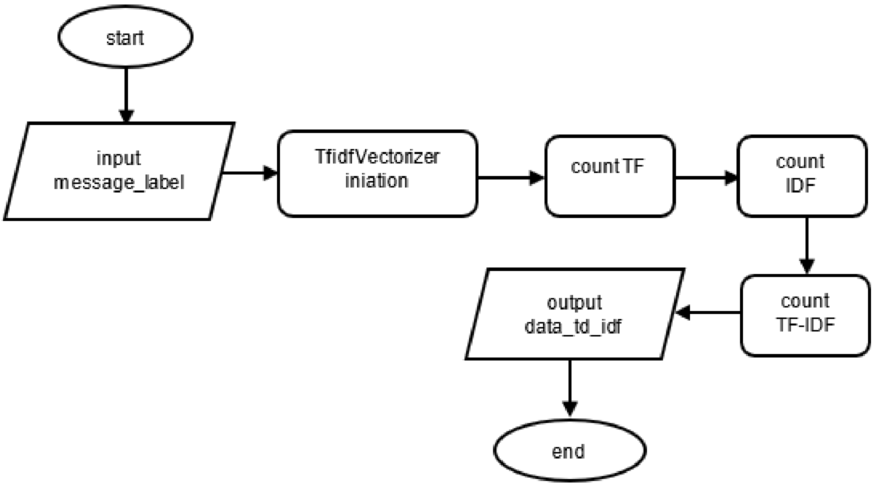


*Figure 9. Weighting Flowchart*

Figure 9 shows how to use TF-IDF for weighting. The TF-IDF weighting stage begins by initiating using TfidfVectorizer on the labeled data. The next stage is the TF calculation by calculating the frequency of words (terms) that appear in one document divided by the number of words (terms) in the document. After the TF calculation is carried out, the stage is continued with the IDF calculation to reduce the weight of the word (term) if the word (term) appears in many documents. The TF-IDF calculation is the next stage carried out to combine the TF value of a word (term) with the IDF value of the word (term) and is continued by changing the TF-IDF weighting results into a data frame. The output of the TF-IDF weighting is a dataset that was originally a text message changed into a numeric data matrix.

### *3.4. Modeling*

At this modeling stage, the three classification algorithms are initiated, namely Naïve Bayes, SVM, and K-NN using the Python programming language from libraries such as scikit-learn. The dataset used is 11,108 records. Next, the dataset is split between training data and test data with a trial comparison of 90:10; 80:20, and 70:30 for each algorithm. The function used to split the dataset in this step is the train_test_split function from scikit-learn. Next, after dividing the dataset, the model performance is evaluated to find the best algorithm performance and the functions used to evaluate model performance in Scikit-Learn are accuracy_score, precision_score, recall_score, and f1_score.

### *3.5. Evaluation*

In the testing stage, a comparison of the trial results of the three classification algorithms, namely Naïve Bayes, SVM, and K-NN, was carried out. Testing on each algorithm was carried out with 3 different compositions between training data and test data, namely 90:10; 80:20, and 70:30. Based on the results of the trial comparison of the composition between training data and test data, the results of each algorithm were obtained for the accuracy, precision, recall and f1-score values as shown in Table 9.

*Table 9. Comparison Results*

| **Algorithm** | | **Data Comparison Experiment** | | |
|---|---|---|---|---|
| | | **90:10** | **80:20** | **70:30** |
| *Naïve Bayes* | *Accuracy* | 89.83% | 89,15% | 89.62% |
| | *Precision* | 92.42% | 92.01% | 92.45% |
| | *Recall* | 89.83% | 89.15% | 89.62% |
| | *F1-Score* | 90.86% | 90.23% | 90.64% |
| SVM | *Accuracy* | 93.88% | **94,10%** | 93,97% |
| | *Precision* | **94.47%** | 94.32% | 94.23% |
| | *Recall* | 93.88% | **94.10%** | 93.97% |
| | *F1-Score* | 94.13% | **94.20%** | 94.09% |
| K-NN | *Accuracy* | 87.13% | 86.72% | 86.68% |
| | *Precision* | 86.01% | 85.41% | 85.71% |
| | *Recall* | 87.13% | 86.72% | 86.68% |
| | *F1-Score* | 86.02% | 85.54% | 85.47% |

From the data above, the results show that the SVM algorithm has the highest accuracy value in classifying Pusintek service user feedback, where with the three experiments that have been carried out, the highest accuracy value obtained is 94.10%, the highest precision value is 94.47%, the recall is 94.10%. 94.10%, and f1-score of 94.20%.

Meanwhile, if the results of this research are compared with previous research conducted by [12] to analyze sentiment related to gadgets based on aspects using Naïve Bayes, SVM, and K-NN, the accuracy results are 79.52% for Naïve Bayes, 94.4 % for SVM, and 77.78% for K-NN. [14] also used a classification algorithm to find out the public's response regarding the Peduli Protect application and obtained SVM results of 76.5%, Naive Bayes of 72.3%, and K-NN of 59.08%. From these two studies, the results obtained were the same as those carried out by the researchers, namely that the SVM algorithm was better for classifying the objects studied. However, this is different from research conducted by [7] to assess public sentiment towards the 2019 Indonesian presidential candidate and the results obtained were that Naïve Bayes had the highest accuracy with a value of 80.90% compared to KNN which only had an accuracy value of 75.58% and the SVM accuracy value is 63.99%.

### *3.6. Deployment*

The final stage of CRISP-DM is deployment, namely developing an application prototype from the best algorithm. The prototype development was carried out using the Python programming language and Streamlit as the interface, to make it easier to read the data results that have been processed by Python programming with the application prototype display as follows.


*Figure 10. Application Prototype*

Figure 10 depicts one of the menu displays on the system prototype, where in the menu you can select the composition between training data and test data to determine the accuracy of the selected algorithm.

## 4. CONCLUSION

According to the findings of the conducted research, following the completion of the data preparation procedure, the number 17,931 in the existing dataset was reduced to 11,108. The preprocessing stages include cleansing, case folding, normalization, tokenization, filtering, and weighting. The preprocessing results are then used to implement the Naïve Bayes, SVM, and K-NN classification algorithms to analyze sentiment in comments from Pusintek ICT service users. Sentiment analysis is categorized into 3 (three) sentiment labels, namely Negative, Neutral, and Positive. Testing is carried out with 3 (three) compositions between training data and test data, namely 90:10, 80:20, and 70:30. The results of this research show that the SVM algorithm with a composition of 80:20 is proven to be able to provide the best performance with an accuracy value of 94.10% and shows high consistency in precision, recall and f1-score values in all sentiment categories. Furthermore, the classification method with the SVM algorithm can be implemented as a solution to filling in inaccurate feedback categories by users of the Pusintek ICT service.

There are still shortcomings in the research that has been done, but there are also some recommendations that can be used to develop and improve the work. For example, it is preferable to add Deviation Detection before preprocessing to identify anomalies in documents that will be processed automatically with specific algorithms. This way, the documents can be retrieved and processed in preprocessing without generating noisy data because the representation of the data has strange values when compared with other data values. This needs to be done so that the processed data has good quality so that it can produce even higher accuracy. Apart from that, other methods need to be used and supplemented with other feature selections such as Pearson Correlation, Chi-Square, or Wrapper Feature Selection.

## 5. ACKNOWLEDGMENTS

Thank you to the Ministry of Finance of the Republic of Indonesia for providing data for this research and the Information Technology Department of Budi Luhur University for providing facilities for carrying out this research.

## REFERENCES

[1] D. Istanbulluoglu and E. Sakman, "Successful complaint handling on social media predicts increased repurchase intention: The roles of trust in company and propensity to trust," *Eur. Manag. J.*, vol. 42, no. 1, pp. 11–22, 2024.

[2] K. Lou Musu, C. Suryawati, and H. Warsono, "Analisis Sistem Penanganan Komplain di Rumah Sakit Permata Medika Semarang," 2020.

[3] F. V. Sari and A. Wibowo, "Analisis Sentimen Pelanggan Toko Online JD.ID Menggunakan Metode Naïve Bayes Classifier Berbasis Konversi Ikon Emosi," *J. Simetris*, vol. 10, no. 2, pp. 681–686, 2019.

[4] M. Birjali, M. Kasri, and A. Beni-Hssane, "A comprehensive survey on sentiment analysis: Approaches, challenges and trends," *Knowl.-Based Syst.*, vol. 226, p. 107134, 2021.

[5] C. Zong, R. Xia, and J. Zhang, *Text Data Mining*, vol. 711, Singapore: Springer, 2021.

[6] M. E. Basiri, M. Abdar, M. A. Cifci, S. Nemati, and U. R. Acharya, "A novel method for sentiment classification of drug reviews using fusion of deep and machine learning techniques," *Knowl.-Based Syst.*, vol. 198, p. 105949, 2020.

[7] M. Wongkar and A. Angdresey, "Sentiment analysis using Naive Bayes Algorithm of the data crawler: Twitter," in *2019 Fourth Int. Conf. Informatics Comput. (ICIC)*, 2019, pp. 1–6, doi: 10.1109/ICIC47613.2019.8985837.

[8] S. H. Kusumahadi, H. Junaedi, and J. Santoso, "Klasifikasi Helpdesk Menggunakan Metode Support Vector Machine," *J. Inform.*, vol. 4, no. 1, pp. 54–60, Jan. 2019, doi: 10.30591/jpit.v4i1.1125.

[9] E. Indrayuni, A. Nurhadi, and D. A. Kristiyanti, "Implementasi Algoritma Naive Bayes, Support Vector Machine, dan K-Nearest Neighbors untuk Analisa Sentimen Aplikasi Halodoc," *Faktor Exacta*, vol. 14, no. 2, p. 64, Aug. 2021, doi: 10.30998/faktorexacta.v14i2.9697.

[10] M. Azhar, N. Hafidz, B. Rudianto, and W. Gata, "Marketplace Sentiment Analysis Using Naive Bayes And Support Vector Machine," *PIKSEL*, vol. 8, no. 2, pp. 91–100, Sep. 2020, doi: 10.33558/piksel.v8i2.2272.

[11] Hermanto, A. Mustopa, and A. Y. Kuntoro, "Algoritma Klasifikasi Naïve Bayes dan Support Vector Machine Dalam Layanan Komplain Mahasiswa," 2020. [Online]. Available: www.bsi.ac.id

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

[13] T. Mardiana, H. Syahreva, and T. Tuslaela, "Komparasi Metode Klasifikasi Pada Analisis Sentimen Usaha Waralaba Berdasarkan Data Twitter," *J. Pilar Nusa Mandiri*, vol. 15, no. 2, pp. 267–274, Sep. 2019, doi: 10.33480/pilar.v15i2.752.

[14] Salma and W. Silfianti, "Sentiment Analysis of User Review on COVID-19 Information Applications Using Naïve Bayes Classifier, Support Vector Machine, and K-Nearest Neighbors," *Int. Res. J. Adv. Eng. Sci.*, vol. 6, no. 4, pp. 158–162, 2021.

[15] K. S. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transit. Proc.*, vol. 3, no. 1, pp. 91–99, 2022.

[16] F. Febriantho, S. Samidi, G. Mikael, and E. Saputra, "Sistem Penentuan Paket Penjualan dengan Algoritma FP-Growth Serta Metode Up dan Cross Selling," *J. Media Inform. Budidarma*, vol. 6, no. 4, p. 2269, Oct. 2022, doi: 10.30865/mib.v6i4.4800.

[17] C. Schröer, F. Kruse, and J. M. Gómez, "A Systematic Literature Review on Applying CRISP-DM Process Model," in *Procedia Comput. Sci.*, Elsevier B.V., 2021, pp. 526–534, doi: 10.1016/j.procs.2021.01.199.

[18] J. Zheng et al., "Research on complaint prediction based on feature weighted Naive Bayes," in *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 983, no. 1, p. 012117, 2022.

[19] G. Ou, Y. He, P. Fournier-Viger, and J. Z. Huang, "A Novel Mixed-Attribute Fusion-Based Naive Bayesian Classifier," *Appl. Sci.*, vol. 12, no. 20, p. 10443, 2022.

[20] I. Gifari, M. Adha, I. R. Hendrawan, F. Freddy, and S. Durrand, "Analisis Sentimen Review Film Menggunakan TF-IDF dan Support Vector Machine," *J. Inf. Technol.*, vol. 2, no. 1, 2022.

[21] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, 2020.

[22] D. M. Abdullah and A. M. Abdulazeez, "Machine learning applications based on SVM classification: A review," *Qubahan Acad. J.*, vol. 1, no. 2, pp. 81–90, 2021.

[23] E. Indrayuni, A. Nurhadi, and D. A. Kristiyanti, "Implementasi Algoritma Naive Bayes, Support Vector Machine, dan K-Nearest Neighbors untuk Analisa Sentimen Aplikasi Halodoc," *Faktor Exacta*, vol. 14, no. 2, p. 64, Aug. 2021, doi: 10.30998/faktorexacta.v14i2.9697.

[24] M. Suyal and P. Goyal, "A review on analysis of k-nearest neighbor classification machine learning algorithms based on supervised learning," *Int. J. Eng. Trends Technol.*, vol. 70, no. 7, pp. 43–48, 2022.

[25] D. Cheng, S. Zhang, Z. Deng, Y. Zhu, and M. Zong, "kNN Algorithm with Data-Driven k Value," in *Lecture Notes Comput. Sci.*, Springer, 2014, pp. 499–512, doi: 10.1007/978-3-319-14717-8_39.

[26] Y. Song, X. Kong, and C. Zhang, "A Large-Scale k-Nearest Neighbor Classification Algorithm Based on Neighbor Relationship Preservation," *Wireless Commun. Mobile Comput.*, vol. 2022, p. 7409171, 2022.

[27] S. Firman, W. Desena, and A. Wibowo, "Penerapan Algoritma Stemming Nazief & Adriani Pada Proses Klasterisasi Berita Berdasarkan Tematik Pada Laman (Web) Direktorat Jenderal HAM Menggunakan Rapidminer," 2022.