LONTAR KOMPUTER VOL. 15, NO. 2 AUGUST 2024 **p-ISSN 2088-1541**
DOI : 10.24843/LKJITI.2024.v15.i02.p03 **e-ISSN 2541-5832**
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

Aspect Based Sentiment Analysis on Shopee Application Reviews Using Support Vector Machine

Dyah Ayu Wulandari[a1], Fitra Abdurrachman Bachtiar[a2], Indriati[a3]

[a]Informatics Engineering Department, Faculty of Computer Science
Universitas Brawijaya
Malang, Indonesia
[1]dyahaw_@student.ub.ac.id
[2]fitra.bachtiar@ub.ac.id (Corresponding author)
[3]indriati.tif@ub.ac.id (Corresponding author)

Abstract

One of the e-commerce in Indonesia is Shopee. Feedback from users is needed to improve the quality of e-commerce services and user satisfaction. This research process includes data scraping, labeling, text pre-processing, TF-IDF, aspect, and sentiment classification. The novelty of this research is using the SVM method with SGD to classify Indonesian language application reviews based on aspect categories consisting of 7 dimensions of service quality and sentiment so that the website created in this research can display the aspects and sentiments of the input reviews. This research also builds an Indonesian normalization dictionary to optimize the terms used to increase model accuracy. The test in aspect classification resulted in a precision value of 90%, recall of 88.73%, accuracy of 88.57%, and f1-score of 89%. Meanwhile, the sentiment classification resulted in a precision value of 96.15%, recall of 91.91%, accuracy of 94.28%, and f1-score of 93.98%. In addition, the test results (accuracy, f1-score, precision, recall) show that the lemmatization process is better than stemming and term weighting using the TF-IDF method is better than other methods (raw-term frequency, log-frequency weighting, binary-term weighting).

Keywords: *Aspect Based Sentiment Analysis, Text Pre-processing, Support Vector Machine, Stochastic Gradient Descent, TF-IDF*

1. Introduction

E-commerce comes from the words electronic and commerce, which means electronic commerce. One example of a well-known e-commerce in Indonesia is Shopee. This research uses Shopee application review data because it is based on iPrice data; it shows that Tokopedia leads the e-commerce market with monthly visitors to the Tokopedia page, reaching 157.2 million in the first quarter of 2022, while Shopee is in second place with an average monthly visitor of 132.77 million in the first quarter of 2022 [1]. There is a difference of around 24 million visitors between Tokopedia and Shopee, which causes Shopee not to create customer satisfaction and loyalty compared to its competitors. Research from [2] shows that the good and bad aspects of e-commerce can be analyzed based on the quality of service. Therefore, Shopee application review data can be analyzed based on service quality, which in this research is used as an aspect in the form of 7 dimensions of service quality and application reviews are classified based on aspects and sentiments to find out which categories of aspects and sentiments the reviews fall into. E-service quality or electronic service quality is the process of creating more value in a product so that a product gets added value from consumers and maintains the image of a company. There are seven dimensions of the approach used in measuring service quality with the E-S-QUAL and E-RecS-QUAL methods developed by Parasurman, including efficiency, fulfillment, system availability, privacy, responsiveness, compensation, and contact [3]. Feedback from users is also needed to improve the quality of service and increase user satisfaction. Shopee application feedback can be seen on the Google Play Store in the review feature that assesses the service of an application. User reviews are textual comments.

With the review feature, textual comment data is available in very large quantities. The sheer number of reviews makes it challenging to analyze sentiment. Sentiment analysis is the automated process of understanding an opinion about a given subject from spoken or written language; sentiment analysis can identify opinions into polarity, positive or negative [4]. Sentiment analysis consists of three scopes: document, sentence, and aspect levels [5]. Conducting aspect-based sentiment analysis (ABSA) is essential to get a more complete analysis. One of the processes in aspect-based sentiment analysis is word weighting. In the previous study [6] a word weighting process was carried out by applying four methods, namely term occurrence / raw term frequency, binary term frequency, term frequency, and term frequency inverse document frequency. The research shows that classification with the KNN method using the word weighting method with term occurrence / raw term frequency produces the highest accuracy compared to other word weighting methods, which is 96.53%. Meanwhile, classification using the Naïve Bayes method using word weighting with binary term frequency resulted in the highest accuracy compared to other word weighting methods of 91.68%. Until now, the most frequently used method of word weighting is the TF-IDF method. Hence, this research will compare the outcomes related to accuracy, precision, recall, and f1-score across different term weighting techniques, including term occurrence / raw term frequency, binary term frequency, term frequency, and term frequency-inverse document frequency.

Research related to ABSA (Aspect Sentiment Analysis) was conducted [7] on customer reviews from case studies in the hotel industry. In the study, a comparison of methods between Naïve Bayes and SVM was conducted. The study shows that evaluation of the classification results proves the effectiveness of the SVM method from Naïve Bayes. Research on sentiment analysis was conducted [8], in that study compared three methods, namely Multinomial Naïve Bayes, Logistic Regression, and Stochastic Gradient Descent against the Arabic corpus to analyze sentiment into two classes of figurative features, namely hyperboles and similes. This study involves several phases, including data collection and manual labeling, followed by a quality assessment conducted by experts proficient in Arabic. Then, text cleaning and preprocessing are done to clean the Arabic text. The next stage is text feature extraction and TF-IDF. Then, a machine learning model can be created to classify text in binary form to determine which falls into a hyperbole or simile class. From the study, it is known that Stochastic Gradient Descent and Logistic Regression have better performance than Multinomial Naïve Bayes. The F1-score obtained by Multinomial Naïve Bayes is 87%, while Stochastic Gradient Descent and Logistic Regression is 89%. Furthermore, research conducted [9] categorized Bengali documents based on the word2vec word embedding model and Stochastic Gradient Descent (SGD) learning algorithm with multi-class SVM. The semantic features of documents extracted by word2vec and SGD increase the classification complexity with multi-class SVM that performs data classification into 9 class categories. The results of the study resulted in an accuracy value of 93.33%.

The disadvantage of the research conducted [8] is that the corpus used in classifying sentiments based on hyperbole or simile classes is less in data size, thus affecting the accuracy of results. The shortcomings of the research conducted [9] are that the f1-score of the environment class reaches a minimum value of 78%, and the accident class overlaps with the crime class. This is because the training data for each aspect of the class is not balanced. In addition, in the research conducted [10] comparison at the pre-processing stage between the lemmatization and stemming processes, the lemmatization process had better results than stemming, but these results were not significant. Therefore, in this study, a comparison of the values of accuracy, precision, recall, and f1-score between the lemmatization and stemming processes was carried out.

Building upon the earlier research, this study delves into "Aspect Based Sentiment Analysis on Shopee Application Reviews Using Support Vector Machine". In this study, the main emphasis lies in testing the Support Vector Machine method for classifying categories, aspects, and sentiments. The aspects used in this study are seven dimensions of the approach used in measuring service quality with the E-RecS-QUAL and E-S-QUAL methods, namely system availability, efficiency, fulfillment, privacy, responsiveness, contact, and compensation so that classification is carried out into seven classes of aspect categories. Following the aspect class categorization, sentiment classification is then performed, categorizing it into two classes: positive sentiment and negative sentiment. The data used in this study has a balanced label. Before the classification process, text pre-processing is carried out first, namely case folding, punctuation removal, repetition of any character, normalization, whitespace and number, tokenizing, stopword

removal, lemmatization, and empty tokens. In the normalization process, a slang dictionary is created to convert slang words into words according to KBBI to optimize the terms used in word weighting and classification processes. The dictionary used already has a large number and is adjusted to the Shopee application review data for maximum accuracy. After that, the term weighting process was carried out.

2. Research Methods

Figure 1 shows the research flow in this study. The process encompasses a series of steps, which include a literature study, data collection, data labeling, text pre-processing, word weighting using TF-IDF, implementation of methods with a Support Vector Machine using Stochastic Gradient Descent, and testing and analysis. After testing and analysis, the study can be concluded and provide suggestions for future research.

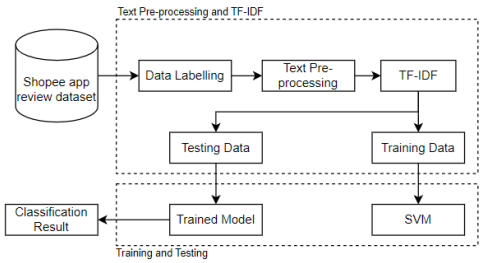

Figure 1. Research Flow

2.1. Literature Study

Various theories and references are needed to support research at the literature study stage: theoretical frameworks and reference materials acquired from journals, books, and earlier studies concerning aspect-based sentiment analysis. Commonly used theories are aspect-based sentiment analysis, text pre-processing, term weighting TF-IDF, and Support Vector Machine using Stochastic Gradient Descent.

2.2. Data Scraping

The research commences with the primary step of data scraping, where secondary data is employed. Specifically, this data pertains to Shopee application reviews sourced from the Google Play Store. The data scraping process is carried out using the Google Play Scraper library. The feature needed in this study is the text containing customer reviews. Data scraping is done by sorting based on the most relevant variables to get data that includes the entire review. The gathered data is stored in a comma-separated value (CSV) file, primarily comprising review text data written in Indonesian. For this research, 1,400 data points were utilized, consisting of 1,120 training and 280 test data. It's worth noting that the data used in this study is well-balanced in terms of both aspects and sentiments.

2.3. Data Labelling

After the data is collected, the next stage is data labeling, where labeling is carried out based on aspect class categories using GPT-3, and sentiment classes are carried out manually based on the review text. Selection of aspect categories in the labeling process based on the dimensions of the approach in measuring service quality with the E-S-QUAL and E-RecS-QUAL methods developed by Parasurman. These categories include efficiency, fulfillment, system availability, privacy, responsiveness, compensation, and contact. Therefore, the aspect column has seven label targets, and the sentiment column has two label targets: positive and negative.

The aspect class labeling data was carried out using GPT-3. In the process, it was known that the accuracy value obtained was 0.9333, with a training loss value of 0.0105 and a validation loss value of 0.01462. The pseudocode of the aspect class labeling using GPT-3 is shown in Table 1.

Table 1. GPT-3 algorithm for aspect class labeling

GPT-3 algorithm for aspect class labeling
1. Load the example aspect data that have been created
2. For each row of example aspect dataset in the ‘reviews’ column:
a. Add the word ‘\n\nLabel:\n\n’’at the end of the sentence
3. For each row of example aspect dataset in the 'label' column:
a. Add the word ‘ END’ at the end of the sentence
4. Rename the columns to ‘prompt’ and ‘completion’ in the example aspect data
5. Save the example aspect data in JSONL format to “labeling_sample.jsonl”
6. Install a specific version of the openai library using pip
7. Prepare fine-tuning data with the command “openai tools fine_tunes.prepare_data”
8. Set the environment variable OPENAI_API_KEY with the API key
9. Create a fine-tuned model with the command “openai api fine_tunes.create”
10. Monitor the fine-tuning process with the command “openai api fine_tunes.follow”
11. Upgrade openai and wandb libraries using pip
12. Synchronize the project with wandb using the command “openai wandb sync”
13. Specify the fine-tuned model to be used
14. Load the reviews data file of the Shopee app
15. Rename the ‘content’ column to ‘prompt’ column in the Shopee reviews data
16. Create a new ‘completion’ column in the Shopee reviews data
17. For each row of the Shopee reviews dataset in the ‘prompt’ column:
a. Add the word ‘\n\nLabel:\n\n’ at the end of the sentence
18. For each row of the Shopee reviews dataset in the ‘completion’ column:
a. Add the word ‘ END’ at the end of the sentence
19. Save the Shopee reviews data in JSONL format to “siaplabeling.jsonl”
20. Load data file “siaplabeling.jsonl”
21. Initialize project and job_type using wandb
22. Specify the number of samples to be tested (n_samples)
23. Loop through the rows of the Shopee reviews data:
a. Retrieve the ‘prompt’ from the current row
b. Generate labelling results using the fine-tuned model from openai
c. Preprocess the labeling results
d. Save the prompt and label results into ‘data’
24. Create a ‘prediction_table’ table using wandb. Table from the labelling results data
25. Log the ‘prediction_table’ table using wandb.log
26. Finish and end the wandb session

Tables 2 and 3 show the amount of data based on aspect and sentiment categories after labeling. As can be seen in Table 2 and Table 3, the data distribution for both aspect and sentiment is relatively balanced. In the next step, these datasets will be used to be preprocessed prior to aspect and sentiment modeling.

Table 2. Aspect Class Statistics

Aspect Class	Amount of Train Data	Amount of Test Data
Compensation	158	39
Contact	151	39
Efficiency	180	37
Fulfillment	168	46
Privacy	149	43
Responsiveness	163	33
System Availability	151	43

Table 3. Sentiment Class Statistics

Sentiment Class	Amount of Train Data	Amount of Test Data
Positive	562	136
Negative	558	144

2.4. Text Pre-processing

After data labeling is done, text pre-processing is carried out. Text pre-processing is the initial stage in making document representations neater because algorithms on search engines can only translate documents in the form of numerical data, so documents that were initially in the form of text must be converted into documents in the form of numerical data. Text pre-processing carried out in this study is first-case folding to convert uppercase characters to lowercase [11]. As the second step, eliminate punctuation marks and symbols, which includes characters like periods (.), commas (,), mentions (@), and hashtags (#), to exclude elements that don't impact the classification process and don't alter the text's meaning after performing case folding [12]. Third, remove repetition of any character to convert more than two repetitive characters in a word string into two characters. Fourth, normalization converts non-standard or abbreviations into proper words according to KBBI (Big Dictionary Indonesian) for each candidate corpus of slang words. A corpus with a list of appropriate (KBBI) and inappropriate (slang words) terms is used in this instance to normalize the use of words, and the terms in the dataset are generally matched by the corpus, which was acquired from Git Hub [13]. Fifth, remove whitespace and number to remove more than one space and number character. Sixth, tokenizing is dividing a text into manageable units known as tokens, including words, phrases, symbols, or other elements, and even complete sentences. This segmentation aids in more efficient text processing [14]. Seventh, stopword removal removes unnecessary words from the dataset based on a list stopword list, thereby improving accuracy [15]. Eighth, lemmatization. Ninth, remove empty tokens to remove empty tokens. This study only used lemmatization and did not use stemming because, based on the function, lemmatization and stemming have the same function, namely removing the prefix or suffix from a word, thereby reducing the length of the token. The difference is that lemmatization functions to transform a word into a basic form by removing the prefix or suffix from a word and knowing the context of the word, for example in Indonesian text, the word "pengiriman" has the prefix "peng-", the suffix "-an", and the basic word "kirim". So, the lemmatization process changes the word "pengiriman" to the word "kirim". Meanwhile, stemming removes prefix and suffix without knowing the word's context. An example of stemming in an Indonesian text changes the word "pengiriman" to the word "irim". In this research, lemmatization has better accuracy than stemming, so it uses lemmatization.

2.5. Term Weighting

Data that has been pre-processed is converted from text data to numeric data, and then the words contained in the sentence are given values according to the frequency of occurrence in the text using TF. Then, the IDF calculates by dividing the total number of documents by the number of documents containing a word. Then, the TF-IDF value is calculated by multiplying the result of the TF and IDF values. This stage aims to classify a value.
To calculate the term frequency value according to equation (1):

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (1)$$

To calculate the value of inverse document frequency according to equation (2):

$$idf(t,D) = log\frac{N}{df(t)} + 1 \quad (2)$$

To calculate the value of term frequency-inverse document frequency according to equation (3):

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) \quad (3)$$

where $f_{t,d}$ is the value of the number of occurrences of a word in the Shopee review document (t and d show the word/*term* and the Shopee review documents), D is entire Shopee review document, N is number of all Shopee review documents, $df(t)$ is number of Shopee review documents containing a word or *term* t, $tfidf(t,d,D)$ is weight of a *term* t on the Shopee review

document d, $tf(t,d)$ is frequency of occurrence of *terms* t on the Shopee review document d, and $idf(t,D)$ is inverse value *of* the number of Shopee review documents containing *term* t.

2.6. Method Implementation

The next step is aspect classification, which aims to classify aspects based on the class of aspects in a dataset. This study has seven classes of aspects, including efficiency, fulfillment, system availability, privacy, responsiveness, compensation, and contact. This study has seven classes, including the multi-class classification with the Support Vector Machine method using Stochastic Gradient Descent. This study used a one-vs-rest strategy for multi-class classification. In this strategy, if you have a k-class classification problem, k is found to be a separator function where k is the number of classes. This strategy needs to generate K binary SVM classifiers in total, and in this approach, we choose the ith class as a positive class and the remaining classes as negative [16]. So, if you have a classification problem using seven classes, there will be seven types of Support Vector Machines with binary models in the training process. SVM operates based on Structural Risk Minimization (SRM), striving to identify the optimal hyperplane capable of dividing two classes within the input space to establish distinct decision boundaries among a collection of data points categorized with varying labels. This is a supervised classification algorithm, and the best hyperplane has the most significant margin [17]. In this study, the hyperplane separates two binary classes in aspect and sentiment classification because the aspect classification uses the One-vs-Rest strategy to deal with multi-class classification problems using binary models in the training process. In the One-vs-Rest approach, each class in a multi-class classification problem is considered a positive class in one binary model and a negative class in another. In other words, it trains a binary model to predict whether a sample belongs to a particular class (positive class) or not (negative class). SVM looks for the best hyperplane that has minimal cost functions. The cost function aims to measure how good a hyperplane is. Several processes are carried out at the aspect classification stage, including forming one vs rest training data. Then, calculate the cost function value and change the weight value. Then, test the test data. The Support Vector Machine (SVM) technique with Stochastic Gradient Descent (SGD) is employed in the Sentiment classification technique. Each review is categorized into one of two classes: positive or negative sentiment. In the sentiment classification phase, two distinct processes are undertaken, which involve computing the cost function value and adjusting the weight values. SGD uses only one sample of training data at each iteration step to perform a weight value update. Subsequently, the test data is evaluated. The data split for training and testing follows an 80:20 ratio, with 80% used for training and 20% for testing.

To calculate the cost function value, one of them is to use hinge loss according to equation (4):

$$l_{hinge} = \max(0,1 - yx.w) \begin{cases} w, & if\ max\ (0, 1 - yx.w) = 0 \\ w - Cy_i x_i, & others \end{cases} \quad (4)$$

where l_{hinge} is *hinge loss*, w is weight, x is training data from the Shopee app review document, y is train label from the Shopee app review document, and C is value of *regularization*.

Then change the weight in the opposite direction to the value of the cost function according to equation (5):

$$w_{i+1} = w_i - \alpha \Delta_{wi} l(w_i) \quad (5)$$

where $\Delta_{wi} l(w_i)$ is calculation of the *hinge loss* value, w_{i+1} is new weights, w_i is starting weight, and α is value of *learning rate*.

After that, do the testing according to equation (6):

$$Testing = \sum w.x \begin{cases} > 0, & +class \\ < 0, & -class \end{cases} \quad (6)$$

where w is weight and x is train label from the Shopee app review document.

2.7. Testing and Analysis

After the model has been implemented, hyperparameter tuning is carried out to get the best model. The parameters used in the test include the regularization value, the maximum number of epochs, and the learning rate value. Various word weighting methods were also tested at this stage, including raw-term frequency, log-frequency weighting, binary-term weighting, and term frequency-inverse document frequency (TF-IDF). In addition, testing is carried out between lemmatization and stemming at the text pre-processing stage. It aims to get the best evaluation results. The model's performance is assessed by applying the confusion matrix method, enabling the determination of precision, recall, f1-score, and accuracy values. This evaluation is conducted to gauge the model's performance in aspect classification and sentiment analysis. Following this evaluation, a thorough analysis of the test results is performed.

3. Result and Discussion

3.1. Regularization Value Testing

Regularization is a method employed to prevent machine learning models from overfitting. Overfitting occurs when a model is too complex and closely attuned to training data, so it cannot generalize well to new, never-before-seen data. Through regularization, machine learning models can be rendered more general and better equipped to adapt to novel data.

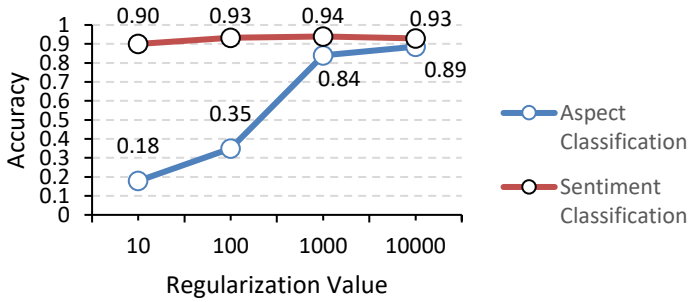

Figure 2. Graph of Regularization Value Testing in Aspect Classification and Sentiment Classification

Based on Figure 2, it can be concluded that the best regularization value in aspect classification is 10000, which produces an accuracy value of 88.57%, while in sentiment classification, it is 1000, which produces an accuracy value of 93.92%. In aspect classification, the optimal regularization value ranges from 1000 – 10000, resulting in an accuracy value of 83% – 88%. Meanwhile, regularization values of 10-100 produce an accuracy value of 17.85% - 35%. It can be concluded that the search for regularization values is necessary to avoid overfitting.

3.2. Testing the Maximum Number of Epochs

In machine learning, epoch refers to one complete iteration or process through the entire training dataset during the training phase of the model. The model receives the entire training dataset at each epoch, processes it, and updates weights based on a defined optimization algorithm. Training a model with a larger number of epochs can help improve its performance up to a point, as it allows the model to learn more from the data. However, training a model with too many epochs can lead to overfitting, where the model becomes too specialized to train data and performs poorly on never-before-seen data.

Based on Figure 3, it can be concluded that the best maximum number of epochs in aspect classification is 1000 and 1500, which produces an accuracy value of 86.42%, while in sentiment classification, it is 100 – 1500, which produces an accuracy value of 93.92%. Based on Figure 3, it can be seen that the more epochs there are, the more accuracy the model learns from the data. According to research conducted [18], the accuracy increases as the epochs increase. However,

using epochs in large quantities can have stable (not increased) accuracy results. Therefore, it is necessary to use the correct epoch value to get maximum results.

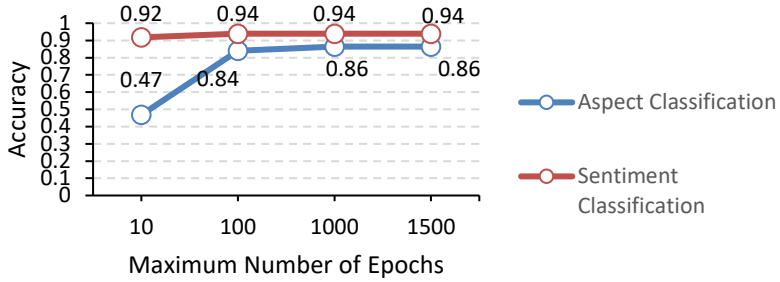

Figure 3. Graph of Testing the Maximum Number of Epochs in Aspect Classification and Sentiment Classification

3.3. Learning Rate Value Testing

Learning rate in machine learning is one of the parameters to calculate weight corrections or changes in weight values so that it can regulate how many steps are taken in the model training process. The smaller learning rate increases the number of epochs because adjustments to the weights become smaller. Conversely, higher learning rates require fewer epochs due to their swifter adjustments. When the learning rate is excessively high, the model convergence rapidly. In contrast, if the learning rate is too small, the process may become stuck to progress effectively at some point [19]. The learning rate determines how quickly or slowly a model changes its parameters when learning patterns in training data. Therefore, choosing the correct learning rate is very important. Figure 4 indicates a rise in accuracy if the learning rate value is getting smaller in aspect and sentiment classification because it is stable in convergence or reaches the desired local minimum.

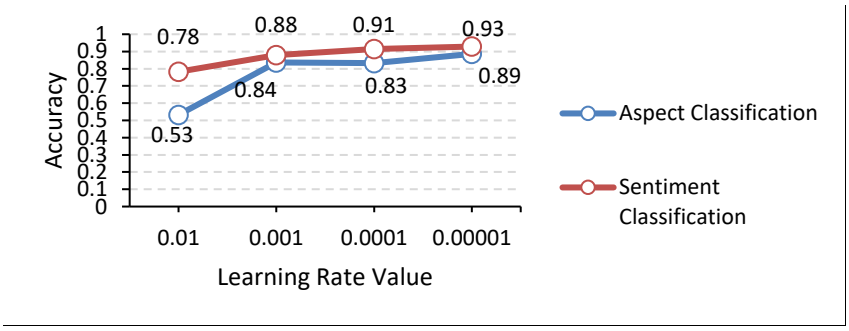

Figure 4. Graph of Learning Rate Value Testing in Aspect Classification and Sentiment Classification

3.4. Comparison of Evaluation Results between Lemmatization and Stemming in Aspect Classification

Based on the graph of Figure 5, it can be concluded that doing aspect classification using the lemmatization process has better accuracy, precision, recall, and f1-score results than stemming. Still, the difference in results is not too significant. Lemmatization has an accuracy result of 88.57% while stemming has an accuracy result of 87.85%. The result of the f1-score lemmatization is 89.09%, while the stemming value is 88.41%. The result of precision lemmatization value is 90.34% while stemming is 89.93%. The result of the lemmatization recall value was 88.73% while stemming was 87.91%. This is because lemmatization can maintain the meaning of the word very well. Words transformed into their root form are known as "lemmas", which retain the semantics and meaning of the word [20]. As in research by Rio Pramana [10], the lemmatization process has a better accuracy value than stemming, although the difference in

results is insignificant. However, stemming is better in terms of computational speed than lemmatization. Therefore, stemming is the best option if computational speed optimization is a top priority. If accurate results are a top priority, lemmatization is the best option.

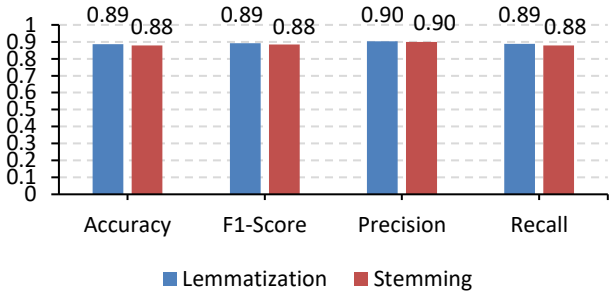

Figure 5. Comparison Graph of Evaluation Results between Lemmatization and Stemming

3.5. Comparison of Evaluation Results between Lemmatization and Stemming in Sentiment Classification

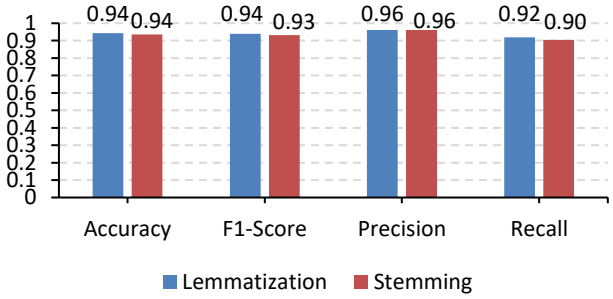

Figure 6. Comparison Graph of Evaluation Results between Lemmatization and Stemming

Based on the graph of Figure 6, it can be concluded that sentiment classification using the lemmatization process yields better recall, accuracy, f1-score, and precision results than the stemming process. However, the difference in results is not too significant. Lemmatization has an accuracy result of 94.28%, while stemming has an accuracy result of 93.57%. The result of the f1-score lemmatization is 93.98%, while the stemming value is 93.18%. The result of precision lemmatization value is 96.15%, while stemming is 96.09%. The result of the lemmatization recall value was 91.91%, while stemming was 90.44%.

3.6. Comparison of Word Weighting Evaluation Results in Aspect Classification

Based on the graph of Figure 7, it can be concluded that doing aspect classification using TF-IDF word weighting has better recall, accuracy, f1-score, and precision results than using other word weighting (raw-term frequency, log-frequency weighting, binary-term weighting). This occurs because TF-IDF takes into account the word frequency within a document (Term Frequency) and the importance of words in the corpus as a whole (Inverse Document Frequency) by giving higher weight to words that appear frequently in a document but rarely appear in the corpus as a whole, thus helping to identify unique or distinctive words in a document.

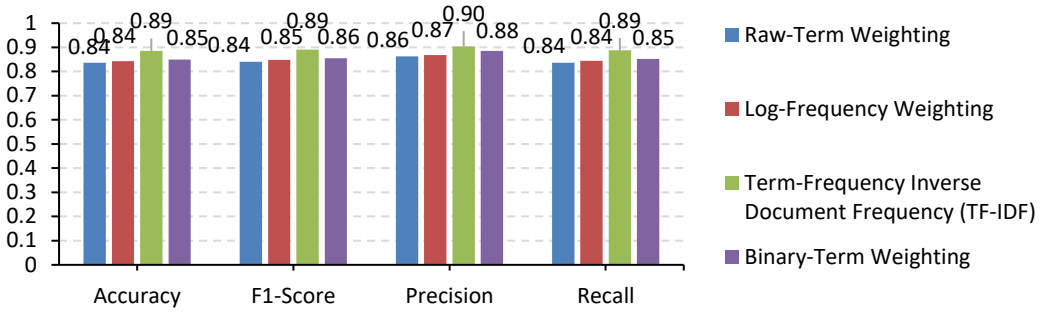

Figure 7. Comparison Graph of Word Weighting Evaluation Results in Aspect Classification

Based on the graph of Figure 7, it can be concluded that doing aspect classification using TF-IDF word weighting has better recall, accuracy, f1-score, and precision results than using other word weighting (raw-term frequency, log-frequency weighting, binary-term weighting). This occurs because TF-IDF takes into account the word frequency within a document (Term Frequency) and the importance of words in the corpus as a whole (Inverse Document Frequency) by giving higher weight to words that appear frequently in a document but rarely appear in the corpus as a whole, thus helping to identify unique or distinctive words in a document.

3.7. Comparison of Word Weighting Evaluation Results in Sentiment Classification

Based on the graph of Figure 8, it can be concluded that doing sentiment classification using TF-IDF word weighting has better recall, accuracy, f1-score, and precision results than using other word weighting (raw-term frequency, log-frequency weighting, binary-term weighting).

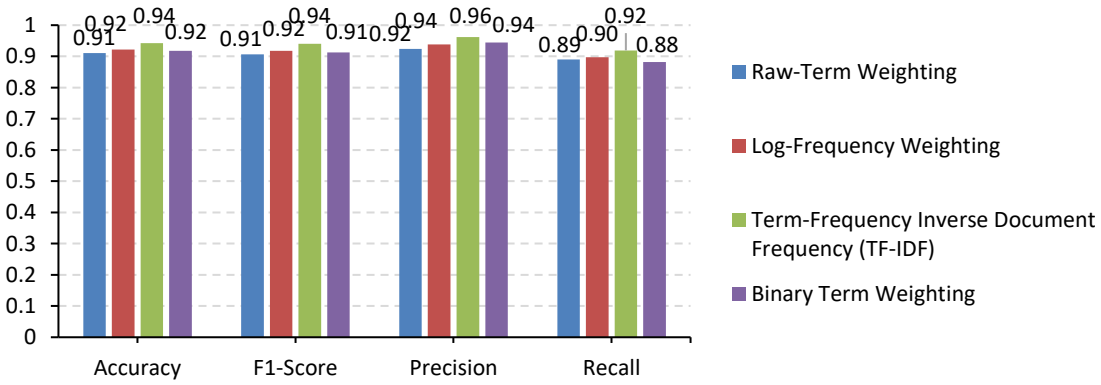

Figure 8. Comparison Graph of Word Weighting Evaluation Results in Sentiment Classification

3.8. Aspect Classification Testing

Aspect classification testing is done on data features using the best parameters, the Support Vector Machine method, and Stochastic Gradient Descent (SGD). Testing is carried out with a confusion matrix from each aspect class to determine performance evaluation results from the Support Vector Machine method using Stochastic Gradient Descent (SGD) in classifying an aspect. The macro average approach can calculate the average evaluation value from all aspect classes. The evaluation results of each aspect class are shown in Table 4, while the average evaluation results of all aspects are shown in Table 5.

Table 4. Classification Evaluation Results of Each Aspect Class

Class	Precision	Recall	Accuracy	F1-Score
Compensation	1	0,846	0,885	0,916
Contact	0,972	0,897		0,933
Efficiency	0,885	0,837		0,861

Fulfillment	0,851	0,869	0,860
Privacy	0,724	0,976	0,831
Responsiveness	0,969	0,969	0,969
System Availability	0,921	0,813	0,864

Table 5. Results of Evaluation of Average Aspect Classification

	Precision	Recall	Accuracy	F1-Score
Macro Average	0,903	0,887	0,885	0,890

3.9. Sentiment Classification Testing

In sentiment classification testing, data features use the best parameters with the Support Vector Machine method using Stochastic Gradient Descent (SGD). The test was carried out with a confusion matrix to determine the performance evaluation results from the Support Vector Machine method using Stochastic Gradient Descent (SGD) in classifying a sentiment. Performance evaluation obtained from the confusion matrix consists of precision, recall, accuracy, and f1-score.

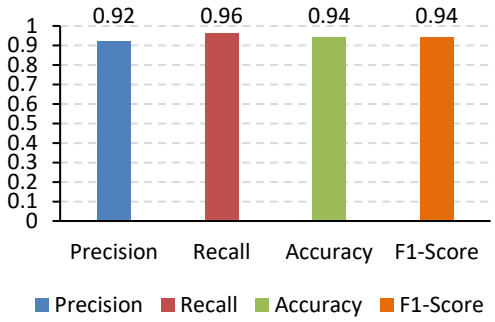

Figure 9. Sentiment Classification Evaluation Graph

4. Conclusion

This research is about aspect-based sentiment analysis that can be used to determine the influence of hyperparameters on the model, assess the impact of *lemmatization* and *stemming processes*, and find out the evaluation results of the model used. The Support Vector Machine algorithm classifies reviews based on aspects and sentiment. From research experiments, it was found that hyperparameters affect the results of the performance evaluation of a model. Hyperparameter values should be searched carefully so as not to cause underfitting or overfitting. In aspect classification, the lemmatization process + TF-IDF has better accuracy, precision, recall, and f1-score results than the stemming process + TF-IDF, but the difference in results is not too significant. The evaluation results of the Support Vector Machine (SVM) method using Stochastic Gradient Descent (SGD) in conducting aspect classification on a Shopee review using the macro average approach resulted in precision of 90%, recall of 88.73%, accuracy of 88.57%, and f1-score by 89%. Meanwhile, the evaluation results of the Support Vector Machine (SVM) method using Stochastic Gradient Descent (SGD) in conducting sentiment classification on a Shopee review resulted in a precision of 96.15%, recall of 91.91%, accuracy of 94.28%, and f1-score of 93.98%.

This study could be expanded for further research and development. First, text preprocessing plays an essential step in the modeling. Thus, text pre-processing detects non-standard words, such as typos and slang, and converts them into standard words by KBBI (Big Dictionary Indonesian) standards. In the dataset, some typos or slang words have not been entered into the dictionary, so the data still has noise. In addition, translating foreign language words into Indonesian at the text pre-processing stage is essential because some words in foreign languages have not entered the dictionary, so the data still has noise. It aims to optimize the terms used in word weighting and classification processes. Perform data labeling on aspects manually so that

the system can be more accurate because, in the data labeling process, this aspect of research uses GPT-3 and adds training data and test data on each element so that the more training data is carried out, the more accurate the system will be in classifying aspects.

References

[1] V. A. Dihni, "10 E-Commerce dengan Pengunjung Terbanyak Kuartal I 2022," 2022. [Online]. Available: https://databoks.katadata.co.id/datapublish/2022/07/19/10-e-commerce-dengan-pengunjung-terbanyak-kuartal-i-2022#:~:text=Tokopedia%20dan%20Shopee%20masih%20memimpin,juta%20pada%20kuartal%20I%202022.. [Acesso em 15 Agustus 2022].

[2] R. J. S. I. Abdillah Taufikqurrochman, "The Impact of E-Service Quality and Price on Customer Satisfaction of Tokopedia," *Jurnal Manajemen Bisnis dan Kewirausahaan,* vol. 1, nº 2, pp. 88-96, 2021.

[3] K. Çelik, "The effect of e-service quality and after-sales e-service quality on e-satisfaction," *Business & Management Studies: An International Journal,* vol. 9, nº 3, pp. 1137-1155, 2021.

[4] M. T. Ibrahim Moge Noor, "Sentiment Analysis using Twitter Dataset," *IJID (International Journal on Informatics for Development),* vol. 9, nº 2, pp. 84-94, 2019.

[5] P. Ray e A. Chakrabarti, "A Mixed approach of Deep Learning method and Rule-Based method to improve Aspect Level Sentiment Analysis," *Applied Computing and Informatics,* vol. 18, nº 1/2, pp. 163-178, 22 2 2019.

[6] S. C. J. S. V. Shitanshu Jain, "Analysis of Text Classification with various Term Weighting Schemes in Vector Space Model," *International Journal of Innovative Technology and Exploring Engineering (IJITEE),* vol. 9, nº 10, pp. 390-393, 2020.

[7] W. P. A. N. R. Fitra A. Bachtiar, "Text Mining for Aspect Based Sentiment Analysis on Customer Review : A Case Study in the Hotel Industry," em *IICST2020: 5th International Workshop on Innovations in Information and Communication Science and Technology*, Malang, 2020.

[8] S. A. Nouh Sabri Elmitwally, "Arabic Corpus for Figurative Sentiment Analysis," *International Journal of Advanced Science and Technology,* vol. 29, nº 3, pp. 3391- 3404, 2020.

[9] M. M. H. Md. Rajib Hossain, "Automatic Bengali Document Categorization Based on Word Embedding and Statistical Learning Approaches," Rajshahi, Bangladesh, 2018.

[10] D. J. J. S. A. Rio Pramana, "Systematic Literature Review of Stemming and Lemmatization Performance for Sentence Similarity," Yogyakarta, Indonesia, 2022.

[11] S. P. Nur Fadilah, "Automatic Essay Scoring Using Data Augmentation in Bahasa Indonesia," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems),* vol. 16, nº 4, pp. 401-410, 2022.

[12] M. D. S. M. E. E. M. M.-S. Gustavo Candela, "Reusing digital collections from GLAM institutions," *Journal of Information Science,* vol. 48, nº 2, pp. 251-267, 2022.

[13] M. Z. A. Y. R. A. R. M. P. H. P. A. K. A. S. B. Tiara Lailatul Nikmah, "Comparison of LSTM, SVM, and naive Bayes for classifying sexual harassment tweets," *Journal of Soft Computing Exploration,* vol. 3, nº 2, pp. 131-137, 2022.

[14] H. D. Muhittin IŞIK, "The impact of text preprocessing on the prediction of review ratings," *Turkish Journal of Electrical Engineering & Computer Sciences,* vol. 28, nº 3, pp. 1405-1421, 2020.

[15] K. S. Neha Garg, "Text pre-processing of multilingual for sentiment analysis based on social network data," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 12, nº 1, pp. 776-784, 2022.

[16] S. L. Z. Z. Huiru Wang, "Ramp loss for twin multi-class support vector classification," *INTERNATIONAL JOURNAL OF SYSTEMS SCIENCE,* vol. 51, nº 8, pp. 1448-1463, 2020.

[17] B. D. G. R. P. M. Hajah T. Sueno, "Multi-class Document Classification using Support Vector Machine (SVM) Based on Improved Naïve Bayes Vectorization Technique," *International Journal of Advanced Trends in Computer Science and Engineering,* vol. 9, nº 3, pp. 3937-3944, 2020.

[18] S. D. M. A. L. Owais Mujtaba Khanday, "Effect of filter sizes on image classification in CNN: a case study on CFIR10 and Fashion-MNIST datasets," *IAES International Journal of Artificial Intelligence (IJ-AI),* vol. 10, nº 4, pp. 872-878, 2021.

[19] D. M. M. B. K. K. E. C. T. Jennifer Jepkoech, "The Effect of Adaptive Learning Rate on the Accuracy of Neural Networks," *International Journal of Advanced Computer Science and Applications,* vol. 12, nº 8, pp. 736-751, 2021.

[20] H. J. B. J. M. C. C. Aachal Jakhotiya, "Text Pre-Processing Techniques in Natural Language Processing: A Review," *International Research Journal of Engineering and Technology (IRJET),* vol. 09, nº 02, pp. 878-880, 2022.