ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 14 (2024) No. 2, pp. 214-228
DOI:10.33168/JSMS.2024.0213

Sentiment Analysis using Machine Learning Models on Shopee Reviews

Ahmad Hariz Imran bin Ahmad Azrir, Palanichamy Naveen, Su-Cheng Haw

Faculty of Computing and Informatics, Multimedia University,63100, Cyberjaya, Malaysia
p.naveen@mmu.edu.my

Abstract. Sentiment analysis is essential for understanding customer opinions and feedback in the e-commerce industry. It is a valuable tool for businesses, providing insights into customer preferences and opinions. By understanding customer sentiment, companies can tailor their messaging to better resonate with their target audience and identify areas of improvement to increase customer satisfaction. However, sentiment analysis precision requires enhancement when informal language is present in the reviews. Therefore, we aim to enhance and boost sentiment analysis's accuracy when informal language is present in the reviews. In this study, the shopee reviews are extracted; preprocessed and sentiments are extracted and labelled as positive or negative. We employ feature extraction approaches such as N-grams, Bag of Words (BOW), and Term Frequency-Inverse Document Frequency (TF-IDF). Next, we apply machine learning methods such as Naive Bayes (NB), Support Vector Machine (SVM), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM). The performance of the models are evaluated using Area Under the Curve (AUC) and Receiver Operating Characteristic (ROC). Our findings show that the SVM classifier using 2-gram TF-IDF features achieves the best performance with an accuracy score of 86% when compared to NB, LSTM and GRU. Overall, our results suggest that using machine learning algorithms can effectively analyze user sentiment in e-commerce platforms like Shopee when informal language is present.

Keywords: Sentiment Analysis, Machine Learning, Reviews, Naïve Bayes, Support Vector Machine, Long Short-Term Memory, Gated Recurrent Unit

1. Introduction

Shopee is a popular e-commerce platform in Southeast Asia that has gained immense popularity in recent years. The platform allows sellers to create online stores and sell their products to millions of buyers across the region. Shopee has transformed the e-commerce industry in Southeast Asia by providing a seamless shopping experience, offering a wide range of products at affordable prices, and offering fast and reliable delivery services. Shopee's popularity can be attributed to its user-friendly interface, convenient payment options, and a robust customer support system that ensures buyers have a hassle-free shopping experience. As a result, Shopee has emerged as a top player in the e-commerce industry and continues to grow its user base across the region.

Sentiment analysis has become an important tool in understanding customer opinions and feedback in the e-commerce industry. With the rise of online shopping platforms such as Shopee, it has become increasingly important for businesses to understand the sentiment behind customer reviews and feedback. This is especially relevant in today's digital age, where customers have the ability to voice their opinions online and share their experiences with a wide audience. It is feasible to automatically categorise the sentiment of a review based on the words and phrases it contains by utilising machine learning models.

The development of efficient sentiment analysis techniques is gaining more and more attention in the research community. The majority of these methods were developed predominantly for English content. Additionally, most currently available research only focuses on formal language, while customer reviews are typically informal and include slang words, which could result in incorrect sentiment classification. For these reasons, the paper aims to explore the problem of handling informal language in sentiment analysis through an experiment conducted on Shopee customer reviews. Preprocessing methods like abbreviation expansion, misspelling removal, and language standardisation can be utilised to solve this problem. Additionally, to ensure proper sentiment categorization in reviews that contain these phrases, machine learning models can be trained on datasets that include a variety of slang words and informal language. By addressing the challenges posed by colloquial words in sentiment analysis, we hope to provide more accurate insights into customer sentiment on online platforms such as Shopee.

The structure of this paper is as follows: Section 2 lists some relevant studies in the area of sentiment analysis. Then, Section 3 describes the methods used in this experiment. Next, section 4 contains the details of this experiment and our findings. Finally, Section 5 presents the conclusion.

2. Literature Review

In this literature review, we will analyze some of the existing methods and techniques used in sentiment analysis for Indonesian and Malay languages.

2.1. Pre-Processing Methods

Preprocessing techniques, such as stemming and stopword removal, are commonly used to clean and prepare text data before feeding them into a sentiment analysis model. However, the impact of these techniques on the accuracy of sentiment analysis is still a topic of debate. In paper (Pradana and Hayaty, 2019), the authors focused on the impact of preprocessing techniques on the accuracy of sentiment analysis in Indonesian language texts. The authors applied four different preprocessing conditions with and without stemming and stopword removal and used a Support Vector Machine classifier with TF-IDF weighting. The results indicated that the application of stemming and stopword removal has a small effect on the accuracy of sentiment analysis in Indonesian text documents.

In the paper (Iswanto and Poerwoto, 2018), the authors investigated the impact of stemming and stopword removal on the accuracy of sentiment analysis in Indonesian text documents. The accuracy and recall of the automatic sentiment analysis of twitter documents for the Indonesian language were

up to 85.50%. The study found that these techniques had a small effect on the accuracy of sentiment analysis. In paper (Tyagi and Tripathi, 2019) the authors used Tweepy to extract Twitter data and implemented a K-Nearest Neighbor algorithm with N-gram modeling to categorize sentiments as positive, negative, or neutral.

Web scraping is a technique used to extract data from websites automatically. In paper (Tedjojuwono and Neonardi, 2021), web scraping using the BeautifulSoup4 python library was employed to extract restaurant reviews from Tripadvisor. The resulting data scraped included the restaurant name, reviewer's name, comment reviews, and ratings result. However, only the restaurant name and customer reviews were used in this project. The extracted data was then subjected to data pre-processing, which involved using the Sastrawi python library, an Indonesian functional library. The Sastrawi python library was used for stemming, a process of reducing words to their root form, to remove any inflections and variations in the Indonesian language. NLTK functions were also utilised to tokenize and remove stopwords, which are commonly used words in a language that do not carry much meaning in the context of the analysis. The resulting pre-processed data was then used for sentiment analysis to determine the sentiment of the customer reviews.

2.2. Supervised Machine Learning

The authors (Fitri et al., 2019) used the Naive Bayes algorithm to classify the sentiment polarity of Twitter users in Indonesia. The authors found that the majority of comments on the Anti-LGBT campaign were neutral, with an accuracy of 86.43% obtained using the Naive Bayes algorithm. Similarly, in the paper (Yin et al., 2021), the authors proposed and tested two algorithms, Naive Bayes and Random Forest, on two datasets of Malay Twitter comments that use internet slang and short forms. The authors found that Random Forest was a better classifier compared to Naive Bayes, with an accuracy of 81.25%.

The results of the studies varied, with some reporting high accuracy rates in sentiment analysis while others reporting lower rates. In the study (Pradana and Hayaty, 2019), the results showed that the application of stemming and stopword removal had a small effect on the accuracy of sentiment analysis in Indonesian text documents. Paper (Fitri et al., 2019) found that the NB algorithm had higher accuracy in sentiment analysis compared to DT and RF models, and that the majority of comments on the Anti-LGBT campaign were neutral. The authors (Saad and Yang, 2019) in their study found that the DT algorithm had the best accuracy in detecting ordinal regression, while paper (Wibowo and Musdholifah, 2021) reported that the use of Fasttext Embedding improved the accuracy of sentiment analysis compared to traditional methods. In the paper (Yin et al., 2021), the authors found that Random Forest was a better classifier compared to Naive Bayes, and that the use of supervised machine learning algorithms improved the accuracy of sentiment analysis compared to traditional methods. The authors of the paper (Adam et al., 2021) reported that their proposed approach using automatically annotated category labels based on emojis outperformed traditional word feature-based methods. Paper (Iswanto and Poerwoto, 2018) found that Naive Bayes classifiers using unibigram feature models produced the best performance without eliminating stop words or stemming the pre-processed tweets, with an accuracy and recall of up to 85.50%. Based on paper (Mussalimun et al.,2021), K-NN and Naïve Bayes classification have similar accuracy, precision, and recall tests. Naïve Bayes classification gets better results in terms of accuracy and precision edging out with a 2% higher result.

2.3. Hybrid Models

Another method that some authors proposed is a hybrid machine learning model, which combines different feature extraction and selection techniques to improve the accuracy of sentiment analysis. In paper (Hassonah et al.,2020), the authors proposed a hybrid machine learning approach that combines Support Vector Machines with two feature selection techniques, ReliefF and MVO. The method was tested using over 6900 tweets from Twitter and was compared with other methods. The results

showed that the author's method performs better than existing techniques, by up to 96.85%. Similarly, in paper (Kumar et al., 2019), the authors focused on the use of a hybrid feature extraction method for sentiment analysis of IMDb movie reviews. The method involves combining feature extraction methods such as TF and TF-IDF with a lexicon corpus to improve the accuracy and complexity of sentiment analysis. The authors found that their method gave better results compared to other classifiers such as SVM, Naive Bayes, KNN, and Maximum Entropy.

2.4. Deep Learning Models

In paper (Matsumoto et al., 2018) focuses on classifying emojis used in tweets using deep learning methods. The authors compared the performance of various neural network models, including Feed Forward Neural Network, CNN, BiLSTM RNN, and BiGRU, and found that the models using BiLSTM as the foundation achieved the highest accuracy. They also evaluate the models using a confusion matrix and precision, recall, and F1 score as evaluation metrics. This study shows the potential for using deep learning to classify non-textual elements in social media data, such as emojis.

The following paper (Aljedaani et al., 2022) presents a hybrid sentiment analysis method that combines deep learning models and lexicon-based techniques to increase sentiment accuracy. The authors evaluate the classification accuracy of various machine learning models, including LR, RF, SVC, DT, GBC, CNN, LSTM, GRU, and LSTM-GRU, and compare the effects of TextBlob, Afinn, and VADER. They use accuracy and F1 score as evaluation measures, and they also employ TF-IDF and BoW as feature extraction methods. The results demonstrate that the LSTM-GRU model performs better than all other models, with an accuracy of 0.97 and an F1 score of 0.96. This study highlights the importance of combining different techniques in order to improve sentiment analysis accuracy.

The authors of paper (Sunitha et al.,2022) proposed a sentiment analysis model to examine tweets about the coronavirus. The authors collect 3100 tweets from European and Indian individuals between March 23, 2020, and November 1, 2021, and use TF-IDF, GloVe, pre-trained Word2Vec, and quick text embedding to preprocess the data and extract features. They then use an ensemble classifier made up of a GRU and a CapsNet to categorize the users' emotions as fear, joy, sadness, and rage. The experimental findings demonstrate that the proposed model classifies the emotions of Indian and European individuals with 97.28% and 95.20% prediction accuracy, respectively. This study shows the potential for using deep learning to classify emotions in social media data related to current events, such as the COVID-19 pandemic.

The conclusion drawn from the paper (Vijayaraghavan and Basu, 2020) results is that deep learning algorithms perform better in predicting sentiment within reviews compared to other machine learning algorithms. The accuracy results for the best models using TFIDF feature and Condition: Depression where deep learning algorithms, namely ANN, LSTM, and GRU, achieved the highest accuracy scores, with ANN performing the best at 90.1993%, followed closely by LSTM at 90.6085%, and GRU at 90.0162%. On the other hand, SVM, Logistic Regression, and RF algorithms achieved lower accuracy scores, with SVM being the highest at 77.6737%, followed by LogReg at 74.2811%, and RF at 61.1632%. This is because deep learning algorithms, particularly neural networks, are capable of capturing significant features that are crucial for classifying the sentiment within the reviews. Additionally, it was found that using Countvectorizer encoding in conjunction with deep learning algorithms results in better performance compared to other models. The study also concludes that both LSTM and GRU models perform similarly for every condition, indicating that recurrent neural networks have a similar performance among each other.

3. Methodology

The study aims to analyze customer reviews of Shopee, a popular online shopping platform. Fig. 1. shows the overall flow of the proposed approach. The Data is extracted from Shopee's app on Google

Play using the Google-Play-Scraper API, and pre-processing techniques are applied to ensure that the dataset is clean and ready for analysis. A lexicon-based method is used to label each review with a

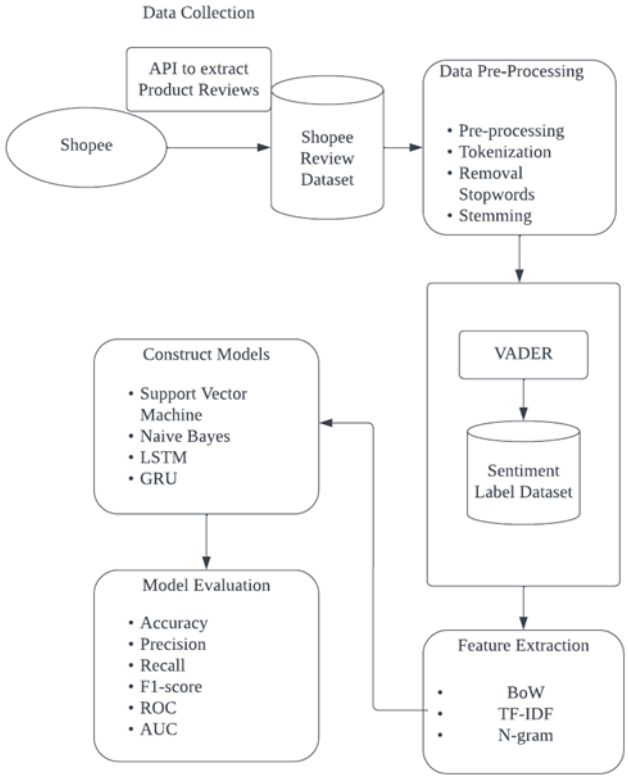

Fig. 1: Proposed architecture

polarizing value, indicating the sentiment of the review. The dataset is then divided into training and testing sets, and feature extraction methods such as BoW, N-gram, and TF-IDF are applied to process the data. Machine learning models are used to predict the sentiment of new customer reviews, including both deep learning and traditional machine learning algorithms. The performance of these models is evaluated using metrics such as accuracy, precision, recall, and F1-score. By analyzing the sentiment of Shopee's customer reviews, valuable insights can be gained into areas for improvement in the platform's user experience.

3.1. Data Collection

Web scraping is a commonly used method for collecting data from the internet, and in this report, we will be using the Google Play Scraper API to extract reviews for a specific app from the Google Play Store. To do this, we first need to install the "google-play-scraper" package in the Python environment using the command "pip install google-play-scraper". Next, we specify the package name of the app we want to scrape and the date range for the reviews. For example, the package name for the "Shopee" app is "com.shopee". Using the Google Play Scraper API, we can extract reviews in the form of dictionaries that contain various details such as 'reviewId', 'userName', 'userImage', 'content', 'score', and 'date'. We store this data in a CSV file to use for our machine learning models.

3.2. Data Preprocessing

The first step involves collecting data using an API and saving it in a CSV file. The next step is data pre-processing, which includes exploratory data analysis, handling null values, and transforming the dataset by dropping unneeded columns and changing data types.

Standard preprocessing techniques such as removing hashtags, mentions, numerical values, non-alphabetic symbols, case folding, URLs, emojis, and hyperlinks are applied. Tokenization is then used to break down the dataset into smaller, more manageable pieces or tokens, which can be individual

words or phrases that are extracted from the dataset. Stop words removal is applied to remove commonly used words from the dataset that do not carry any meaningful information and are not useful in tasks such as sentiment analysis. Stemming is used to simplify words to their most fundamental form by removing suffixes, prefixes, and other grammatical identifiers that are not necessary for understanding the word's meaning.

The next step is text preprocessing, which entails a series of adjustments to the text data to make it more manageable. Standard preprocessing, tokenization, stop word removal, and stemming are the four phases of the approach. The 'BeautifulSoup' library is used to first remove HTML tags from the raw text, followed by regular expressions to eliminate any numbers, non-letter characters, punctuation, and multiple spaces from the text. The final output of the function is a cleaned and prepared text ready for analysis. A slang dictionary is then established to replace all the slangs in the dataset with the corresponding formal translation. Finally, emoticons and emojis are replaced with the words they stand for to convey feelings and sentiments.

3.3. VADER

VADER is an essential tool for sentiment analysis in this work because it uses a lexicon-based approach to determine the sentiment of text data. Based on the paper (Elbagir and Yang, 2019), with the help of VADER, the code can calculate polarity scores for each review in the dataset, which is represented by a score between -1 and 1, indicating the overall sentiment of the text. The sentiment of the reviews plays a significant role in understanding the opinions and feelings of the customers, which helps businesses make informed decisions. Without VADER, it would be challenging to analyze and comprehend the sentiments expressed in the text, and the accuracy of the analysis would be compromised. Furthermore, paper (Bonta and Janardhan, 2019) suggests based on the experimental results of this work that VADER outperforms the Text blob. Therefore, it is an indispensable tool for sentiment analysis in this work.

3.4. Feature Extraction

Bag of Words (BoW): The technique is a simple and flexible method for extracting features from text data. It works by creating a histogram of all the words in the text, treating each word as a feature, and using the frequency of each word as a function in the training set. The CountVectorizer function from Scikit-learn is used to implement the BoW method, which counts the tokens and generates a matrix of limited tokens.

Term frequency-inverse document frequency (TF-IDF): It is another commonly used feature extraction technique that assigns a weight to each word in the corpus, converting the text data into numerical representation. The weight of each word is computed by multiplying its Term Frequency (TF) and Inverse Document Frequency (IDF). The TF is computed as the frequency of a term in a document, and IDF is calculated based on the number of documents containing the term. The most frequent word combinations of size n are sought out in the text using the N-gram parameter shown in the figures. 1-gram and 2-gram ranges were discovered to be the most efficient N-gram range combinations for this dataset. These parameters are excellent for ensuring that a review contains all of the features needed. As a result, the N-gram range for this project using the TF-IDF approach will be between 1 and 2 grams.

3.5. Supervised Machine Learning Models

Each model is created using a combination of different feature extraction techniques and parameters. The parameter settings are shown in Table 1 below.

Support Vector Machines (SVMs) are a machine learning algorithm used for classification and regression tasks. SVMs aim to find the hyperplane that best separates data into distinct classes with the largest margin between the closest data points of the two classes. SVMs are robust to noise and generalizable to new data, making them well-suited for imbalanced, high-dimensional, and non-linear

datasets. However, they can be computationally expensive and sensitive to hyperparameter selections. Overall, SVMs are an effective and versatile algorithm for classification and regression tasks.

Naive Bayes is a popular supervised learning algorithm used for classifying data into different groups. However, its assumption that a data point's attributes are independent of each other is often unrealistic for real-world data. To overcome this limitation, variations of the algorithm have been developed, including Gaussian, Bernoulli, and Multinomial Naive Bayes. The algorithm calculates the probability of a new data point belonging to each category based on its attribute values and selects the class with the highest probability. This involves calculating the posterior probabilities of each class using Bayes' theorem, which considers the likelihood and prior probability of the data point and each class. The algorithm can be used to classify a large dataset by repeating this process for multiple data points.

Table 1. Hyperparameters used

Model	Hyperparameter	Feature extraction
SVM	cvec__max_features: 300 cvec__min_df: 2, 3 cvec__max_df: 0.9, 0.95 cvec__ngram_range: (1, 1), (1, 2) svc__kernel: 'linear', 'poly', 'rbf' svc__degree: 3 svc__C: 0.1	BoW + TF-IDF + N-Gram
NB	cvec__max_features: 500 cvec__min_df: 2, 3 cvec__max_df: 0.9, 0.95 cvec__ngram_range: (1, 1), (1, 2)	BoW + TF-IDF + N-Gram

3.6. Deep Learning models

LSTM: RNN architectures with LSTM are commonly used due to their ability to process sequential data such as text data. LSTMs are well-suited for handling long-term dependencies and can remember past events to predict future ones. During training, the LSTM gates adjust their weights to focus on the most important information in the sequence, making them effective for sentiment analysis where context and sequence of words are critical in determining the sentiment of a text. Overall, the use of LSTM-based RNN architectures in Shopee sentiment analysis allows for more accurate and efficient analysis of customer feedback and reviews.

GRU: It is a type of RNN used for analyzing sequential data, such as text or time series data. They have a simpler structure and fewer parameters than LSTMs, which makes them faster to train and easier to optimize. The update and reset gates in GRUs allow the network to selectively choose which data to store in its memory and make accurate predictions by capturing long-term dependencies in the data. GRUs are used for sentiment analysis in text data by processing it as a sequence of words or tokens and using the gates to learn patterns indicative of different sentiments. They are trained on a labeled dataset of text data and can be used to classify new, unseen data into the appropriate sentiment category.

4. Results and Discussion

The data pre-processing involves various techniques such as exploratory data analysis, handling null values, and transforming the dataset. Standard pre-processing techniques such as removing hashtags, mentions, numerical values, non-alphabetic symbols, case folding, URLs, emojis, and hyperlinks are applied, followed by tokenization to break the dataset into smaller pieces. Stop words removal is used to remove commonly used words, and stemming simplifies words to their fundamental form by

removing suffixes and prefixes the results of can be found in Table 2

Table 2. Sample reviews after data preprocessing

Preprocessing step	Review
Original	"The search function used to be very good and would give relevant items for your search. Now, it is cluttered up with unrelated items that are barely relevant and not what I'm looking for."
General Preprocessing	'The search function used to be very good and would give relevant items for your search Now it is cluttered up with unrelated items that are barely relevant and not what I m looking for"
Remove stopwords	"search function used good would give relevant search. now, cluttered unrelated barely relevant i'm looking for. "
Stem text	"search function used good would give relevant search. now, cluttered unrelated barely relevant i'm looking for."

Fig 2 shows the Number of Meaning words in Negative and Positive Reviews. This refers to the number of words in a review that convey meaning or content, rather than being filler words or stop words that convey the person's sentiment, opinion, or description of the subject of the review. The mean of negative reviews is 21 and the mean of positive reviews is 13, this suggests that the negative reviews tend to have a higher average score than the positive reviews. In other words, customers who leave negative reviews tend to rate the product or service lower on average than those who leave positive reviews. This may indicate that there are some significant issues or drawbacks with the product or service that are affecting the overall satisfaction of customers.

In Fig 3, the accuracy is a measure of how well a model is able to correctly predict the target variable. In this case, having a 78% accuracy on a 0.11 threshold suggests that the model is able to correctly predict the target variable 78% of the time when Vader's compound score is above 0.11.

Fig 4 shows the distribution of the compound scores calculated by Vader for positive and negative reviews. The x-axis represents the range of compound scores (-1 to 1), and the y-axis shows the density of reviews falling within that range. The graph illustrates that the majority of positive reviews have a higher compound score, indicating a stronger positive sentiment, while the majority of negative reviews have a lower compound score, indicating a stronger negative sentiment. Additionally, the graph also shows that there is some overlap in the compound scores of positive and negative reviews, suggesting that some reviews may contain mixed sentiments.

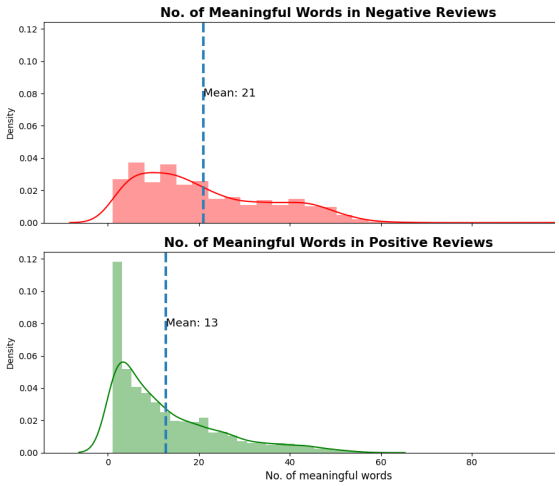

Fig. 2. Number of Meaning words in Negative and Positive Reviews

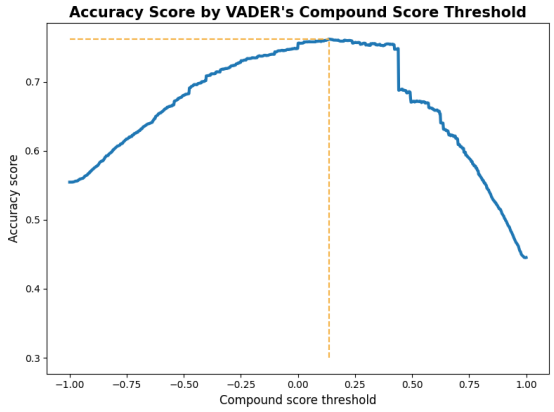

Fig. 3. Average Score by Vader's Compound Score Threshold

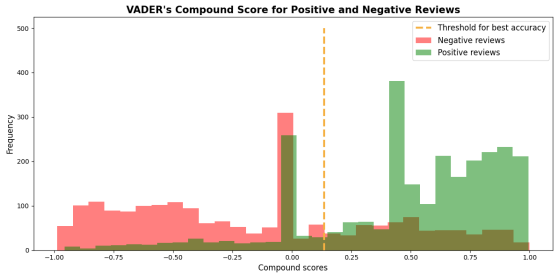

Fig. 4. Vader's Compound Score for Positive and Negative Reviews

In Fig 5,6,7 and 8 It shows the lists of unigrams and bigrams and reveals the most common words and phrases used in positive and negative reviews. For example, in negative reviews, words such as "use," "time," and "app" are frequently used. In positive reviews, words such as "good," "shopping," and "great" are frequently used. By analysing these words, we can roughly figure out what consumers feel towards the service. However, it is difficult to get the full context of the sentiment. Thus Bigrams should be able to capture the full sentiment clearer. Bigrams such as "customer service," "server error," and "went wrong" are commonly used in negative reviews. These phrases highlight the proper context and we can deduce that business needs to improve on customer service in order to satisfy the consumers. While bigrams such as "easy use," "online shopping," and "customer service" are commonly used in positive reviews. These insights can be useful for businesses to understand their customers' sentiment and improve their products or services accordingly.

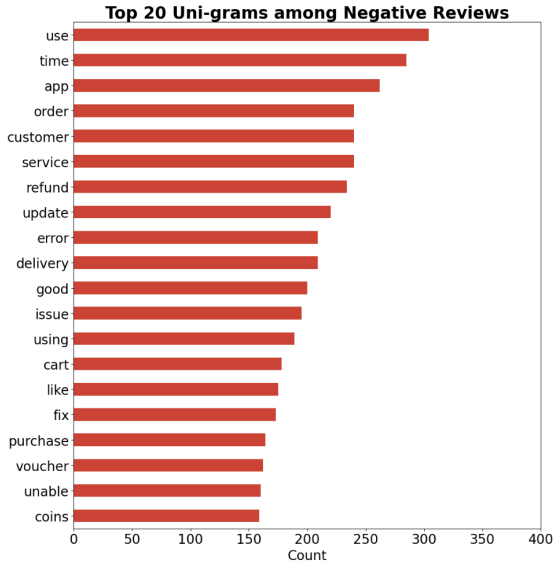

Fig. 5. Top 20 Uni-grams for Negative Reviews

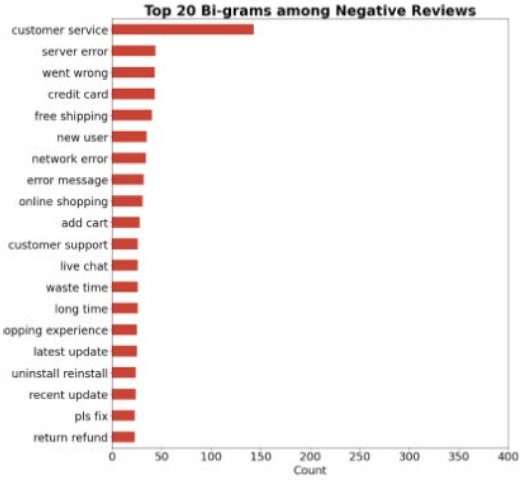

Fig. 6. Top 20 Bi-grams for Negative Reviews

The results are shown in tables 3 and 4. The Bag of Words (BoW) technique was used in the first table, while the second table used the Term Frequency-Inverse Document Frequency (TF-IDF) feature representation. In the first table, both SVM and NB models achieved high accuracy scores of 0.83 and 0.85, respectively. The models had similar precision, recall, and F1-score for class 0, but NB performed slightly better than SVM for class 1. These results suggest that BoW may be a suitable technique for text classification tasks, but may not be optimal for datasets with imbalanced classes, as seen in class 1. In the second table, both SVM and NB models achieved higher accuracy scores than in the first table, with SVM achieving an accuracy of 0.85 and NB achieving an accuracy of 0.84. The models had higher precision and F1-scores for class 0 than in the first table, but had slightly lower scores for class 1. The precision and recall of the models were also more balanced across both classes, indicating that TF-IDF may be more suitable for datasets with imbalanced classes.

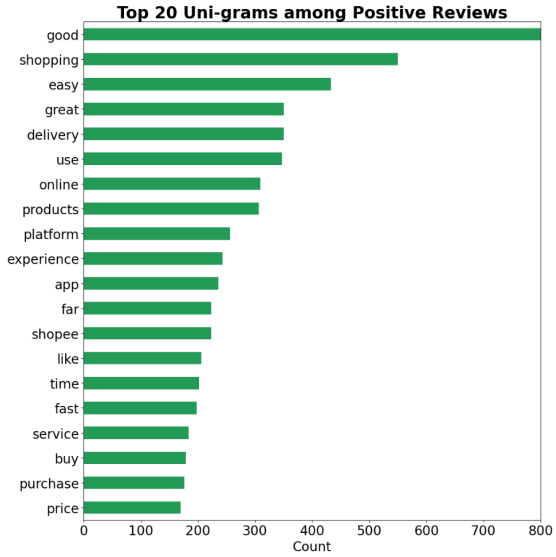

Fig. 7. Top 20 Uni-grams for Positive Reviews

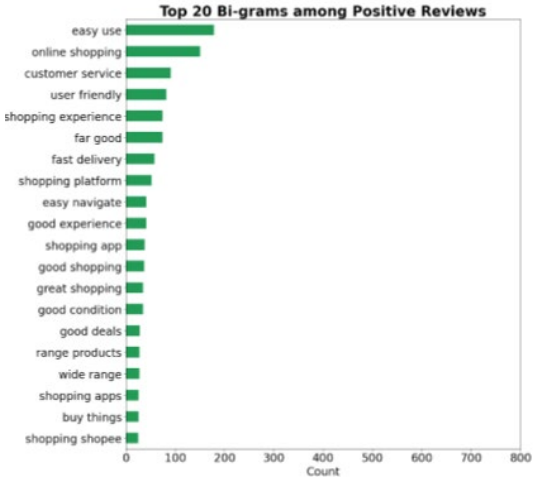

Fig. 8. Top 20 Bi-grams for Positive Reviews

The ROC curve and AUC values were calculated for both classifiers. The AUC value of 0.890 for SVM + BoW indicates that the classifier has good performance, while an AUC value of 0.91 for SVM + TF-IDF indicates slightly better performance. Comparing the results between BoW and TF-IDF, the TF-IDF technique outperformed BoW in terms of accuracy, precision, recall, and F1-score, suggesting that it may be a more robust technique for text classification.

Table 3. Performance of Machine learning with BoW

Model	Accuracy	Class	Precision	Recall	F1 score
SVM	0.83	0 1	0.81 0.85	0.9 0.74	0.85 0.79
NB	0.85	0 1	0.84 0.85	0.87 0.82	0.86 0.83

Table 4. Performance of Machine learning with TF-IDF

Model	Accuracy	Class	Precision	Recall	F1-score
SVM	0.85	0 1	0.88 0.82	0.85 0.85	0.86 0.83
NB	0.84	0 1	0.84 0.84	0.88 0.80	0.86 0.82

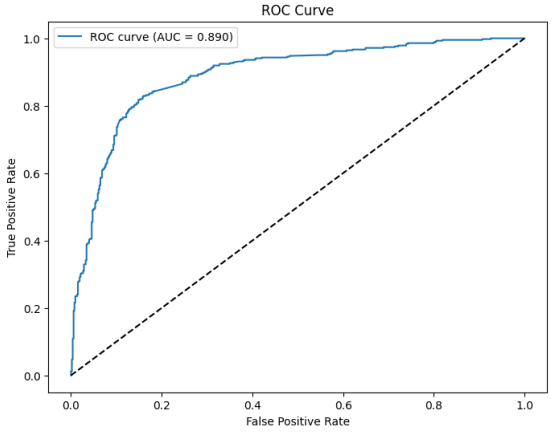

Fig. 9. ROC Curve for SVM + BoW

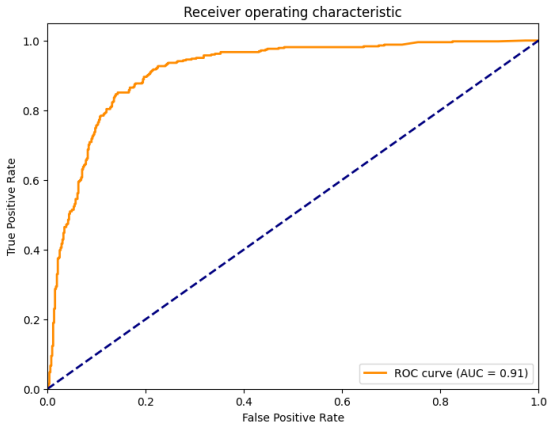

Fig. 10. ROC Curve for SVM + TF-IDF

4.1. Results of Deep Learning Models

The two models shown are deep learning models designed for text classification tasks. Both models use an embedding layer to represent each word in the input text as a vector, which is then processed by a recurrent layer. The LSTM model uses a Long Short-Term Memory layer, while the GRU model uses a Gated Recurrent Unit layer. Both of these layers allow for the network to keep track of information from previous time steps, which can be important for understanding the context of text data. Both models also use dropout regularization to prevent overfitting, and an output layer with sigmoid activation for binary classification. The main difference between the two models is the number of units used in the recurrent layer (32 for LSTM and 64 for GRU), as well as the size of the embedding layer (8 for LSTM and 64 for GRU). Overall, both models have a similar architecture and are suitable for text classification tasks.

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 60, 8)	50640
spatial_dropout1d (SpatialDropout1D)	(None, 60, 8)	0
bidirectional (Bidirectional)	(None, 60, 64)	10496
dense (Dense)	(None, 60, 8)	520
dense_1 (Dense)	(None, 60, 1)	9

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 60, 64)	405120
spatial_dropout1d_1 (SpatialDropout1D)	(None, 60, 64)	0
bidirectional_1 (Bidirectional)	(None, 60, 128)	49920
dense_2 (Dense)	(None, 60, 32)	4128
dropout (Dropout)	(None, 60, 32)	0
dense_3 (Dense)	(None, 60, 1)	33

Fig. 11. model architecture

Based on the table provided, both LSTM and GRU models have achieved an accuracy of 0.84, which indicates that both models have performed equally well in predicting the classes. The precision values for both models are also similar for class 0 and 1, indicating that both models can predict the positive and negative classes accurately. The recall values for class 0 are also quite similar for both models, while the recall value for class 1 is slightly higher for the GRU model. This suggests that the GRU model is better at identifying the positive class, while the LSTM model is better at identifying the negative class.

Table 5. Performance of LSTM and GRU

Model	Accuracy	Class	Precision	Recall	F1- score
LSTM	0.84	0 1	0.86 0.80	0.84 0.84	0.85 0.82
GRU	0.84	0 1	0.86 0.83	0.86 0.82	0.86 0.83

Looking at the F1-scores, we can see that both models have similar scores for class 0, but the GRU model has a higher score for class 1. This indicates that the GRU model is better at balancing precision and recall for the positive class. Overall, both models have performed well, but if identifying the positive class is more important, then the GRU model may be preferred. However, if identifying the negative class is more important, then the LSTM model may be more suitable. Possible limitations can include contextual ambiguity where models may struggle with contextual ambiguity, which occurs when the sentiment of a word or phrase depends on the surrounding words or the broader context of the sentence or document. For example, the word "sick" could be used positively ("That concert was sick!") or negatively ("I'm feeling sick after that meal."), and a model may struggle to correctly identify the sentiment without considering the broader context.

Next, irony may be a possible factor as these words can invert the sentiment of a word or phrase. For example, the phrase "not bad" could be interpreted as positive or negative depending on the context and the speaker's intent. A model that does not take into account negation and irony may

perform poorly in these cases.

5. Conclusion

The paper aimed to handle the informal language present in customer reviews, by providing more accurate insights into customer sentiment on online platforms such as Shopee. To explore the sentiment of users, BOW, TF-IDF, and N-grams feature extraction approaches were applied. Next, sentiments were extracted, machine learning models were employed and performance of the models were evaluated. Comparing the SVM and NB results with the deep learning models, we can see that the SVM model has the highest accuracy at 0.85, followed by the LSTM and GRU models at 0.84. The NB model has an accuracy of 0.84, which is the same as the deep learning models.

One of the major limitations is contextual ambiguity, where the sentiment of a word or phrase depends on the surrounding words or the broader context of the sentence or document. Additionally, irony can be a possible factor that inverts the sentiment of a word or phrase. Therefore, it is important to take into account these limitations and use more advanced techniques to improve the accuracy and effectiveness of sentiment analysis models. By doing so, we can better understand and analyze textual data for various applications such as marketing, customer feedback analysis, and political analysis.

References

Adam, N. L., Rosli, N. H., & Soh, S. C. (2021). Sentiment Analysis on Movie Review using Naïve Bayes. *2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, IPOH, Malaysia.

Aljedaani, W., Rustam, F., Mkaouer, M. W., Ghallab, A., Rupapara, V., Washington, P. B., Ashraf, I. (2022). Sentiment analysis on Twitter data integrating TextBlob and deep learning models: The case of US airline industry. *Knowledge-Based Systems*, 255, 109780. doi:10.1016/j.knosys.2022.109780.

Bonta, V., & Janardhan, N. K. N. (2019). A comprehensive study on lexicon based approaches for sentiment analysis. *Asian Journal of Computer Science and Technology*, 8(S2), 1-6.

Elbagir, S., & Yang, J. (2019). Twitter sentiment analysis using natural language toolkit and VADER sentiment. *International multiconference of engineers and computer scientists*, Vol. 122, p. 16.

Fitri, V. A., Andreswari, R., & Hasibuan, M. A. (2019). Sentiment Analysis of Social Media Twitter with Case of Anti-LGBT Campaign in Indonesia using Naïve Bayes, Decision Tree, and Random Forest Algorithm. *Procedia Computer Science*, 161, 765–772. doi:10.1016/j.procs.2019.11.181.

Hassonah, M. A., Al-Sayyed, R., Rodan, A., Al-Zoubi, A. M., Aljarah, I., & Faris, H. (2020). An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter. *Knowledge-Based Systems*, 192, 105353. doi:10.1016/j.knosys.2019.105353.

Iswanto, B., & Poerwoto, V. (2018). Sentiment analysis on Bahasa Indonesia tweets using Unibigram models and machine learning techniques. IOP Conference Series: *Materials Science and Engineering*, 434, 012255. doi:10.1088/1757-899X/434/1/012255.

Kumar, K., Harish, B. S., & Darshan, H. K. (2019). Sentiment analysis on IMDb movie reviews using hybrid feature extraction method. *International Journal of Interactive Multimedia and Artificial Intelligence*, 5(5), 109. doi:10.9781/ijimai.2018.12.005.

Matsumoto, K., Yoshida, M., & Kita, K. (2018). Classification of Emoji Categories from Tweet Based on Deep Neural Networks. *Proceedings of the 2nd International Conference on Natural Language Processing and Information Retrieval*, 17–25. Presented at the Bangkok, Thailand.

Mussalimun, E. H. Khasby, G. I. Dzikrillah and Muljono (2021), Comparison of K- Nearest Neighbor (K -NN) and Naïve Bayes Algorithm for Sentiment Analysis on Google Play Store Textual Reviews,

2021 8th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), Semarang, Indonesia, 2021, pp. 180-184, doi: 10.1109/ICITACEE53184.2021.9617217.

Pradana, A., & Hayaty, M. (2019). The Effect of Stemming and Removal of Stopwords on the Accuracy of Sentiment Analysis on Indonesian-language Texts. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 4(4).

Saad, S. E., & Yang, J. (2019). Twitter sentiment analysis based on ordinal regression. *IEEE Access: Practical Innovations, Open Solutions*, 7, 163677–163685. doi:10.1109/access.2019.2952127

Sunitha, D., Patra, R. K., Babu, N. V., Suresh, A., & Gupta, S. C. (2022). Twitter sentiment analysis using ensemble based deep learning model towards COVID-19 in India and European countries. *Pattern Recognition Letters*, 158, 164–170. doi:10.1016/j.patrec.2022.04.027

Tedjojuwono, S. M. and Neonardi, C. Aspect Based Sentiment Analysis: Restaurant Online Review Platform in Indonesia with Unsupervised Scraped Corpus in Indonesian Language, *2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI)*, Jakarta, Indonesia, 2021, 213-218.

Tyagi, P., & Tripathi, R. C. (2019). A review towards the sentiment analysis techniques for the analysis of twitter data. *In Proceedings of 2nd international conference on advanced computing and software engineering (ICACSE)*.

Vijayaraghavan, S., & Basu, D. (2020). Sentiment analysis in drug reviews using supervised machine learning algorithms. *arXiv preprint arXiv:2003.11643*.

Wibowo, D. A., & Musdholifah, A. (2021). Sentiments analysis of Indonesian tweet about covid-19 vaccine using support vector machine and fasttext embedding. *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*. Presented at the 2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia. doi:10.1109/isriti54043.2021.9702871.

Widjaja, J. A., & Wibowo, A. (2022). Sentiment Analysis with Slang Dictionary in Indonesian Social Media using Machine Learning Approach. ICIC Express Letters, 16(11), 1169-1177.

Yin, C. J., Ayop, Z., Anawar, S., Othman, N. F., & Zainudin, N. M. (2021). Slangs and Short forms of Malay Twitter Sentiment Analysis using Supervised Machine Learning. *International Journal of Computer Science and Network Security*, 21(11), 294–300.