JCRINN
Journal of Computing Research and Innovation
e-ISSN : 2600-8793

Available online at
https://jcrinn.com/
https://crinn.conferencehunter.com/

Journal of Computing Research and Innovation 9(2) 2024

Journal of Computing Research and Innovation

Performance of TF-IDF for Text Classification Reviews on Google Play Store: Shopee

Najwa Umaira Che Mohd Safawi[1], Nur Amalina Shafie[2*]

[1,2] *College of Computing, Informatics and Mathematics, Universiti Teknologi MARA Negeri Sembilan, Seremban Campus, Negeri Sembilan, Malaysia*

ARTICLE INFO

Article history:
Received 27 January 2024
Revised 18 February 2024
Accepted 29 February 2024
Online first
Published 1 September 2024

Keywords:
TF-IDF
Text Classification
Shopee
Feature Extaction

DOI:
10.24191/jcrinn.v9i2.410

ABSTRACT

TF-IDF is a technique used to extract features in the field of text classification. The TF-IDF approach extracts feature by considering the frequencies of terms and their inverse document frequencies. The performance of various feature extraction methods varies, and it is necessary to determine the most appropriate approach for accurately classifying Shopee's application user reviews to enhance the user experience in Malaysia. This study aims to assess the efficacy of TF-IDF in text classification tasks, analyze their advantages and disadvantages, and identify the specific conditions in TF-IDF. The study employs a dataset of Shopee customer reviews acquired from the Google Play Store as the main data source. The methodology entails pre-processing the text data by applying a text normalization procedure that includes several processes, such as eliminating stop words, Unicode characters, and lemmatizing. The Logistic Regression, Support Vector Machine, and Decision Tree classifiers are trained and graded using both feature extraction approaches. The research notes that the efficacy of feature extraction approaches may differ based on the specific data set and task being considered. Subsequent studies might examine alternative methods of extracting features and assess their efficacy across various domains and datasets.

1. INTRODUCTION

The early development of text classification dates back to the 1960s when machine learning algorithms were first used to categorize documents. In the 2000s, the explosion of digital text data and the rise of the internet led to a significant increase in the interest and popularity of text classification. One of the examples of text classification is a study by Turney (2002), where he proposed a method for using machine learning algorithms to classify movie reviews as either positive or negative. A study by Mikolov et al. (2015) proposed a novel method for automatically categorizing news articles into predefined topics using word embedding and clustering techniques.

An integral part of text classification is feature extraction, which aims to represent textual data in a meaningful way. One widely used technique is term frequency-inverse document frequency (TF-IDF), introduced by Karen Sparck Jones in the year 1972 (Sparck Jones, 1972). Today, TF-IDF is one of the most widely feature extraction techniques used for text classification. It evaluates how important a word is to a document in a collection or corpus. This technique calculates a score for each word in a document, which

[2*] Corresponding author. *E-mail address*: amalina@uitm.edu.my

https://doi.org/10.24191/jcrinn.v9i2.410

is then used to determine the relevance of that word to the document's content. TF-IDF has been used successfully in various applications such as text classification, text clustering, and information retrieval (Kanaris et al., 2020).

Once the features are extracted, classification algorithms are applied to learn patterns and make predictions. Classifiers employ a learning process to create a model that establishes the relationship between these features and the corresponding class labels. During the training phase, the classifier endeavours to recognize patterns and associations within the features that signify particular classes or categories. This involves fine-tuning internal parameters or weights based on the training data, with the ultimate objective of minimizing errors or maximizing the model's predictive accuracy (Bishop, 2006).

There are several well-known classifiers commonly employed in text classification tasks. Logistic regression, for instance, models the connection between the extracted features and the probabilities associated with each class by employing a logistic function. Through estimating the parameters of this function, logistic regression can effectively classify new texts by calculating the likelihood of each class based on the acquired model (Raj, 2020). Decision trees, on the other hand, provide another avenue for classification. They systematically split the feature space by utilizing specific criteria, such as entropy or information gain, to construct a hierarchical structure of decision rules. These rules enable the partitioning of data into subsets, and at each internal node, a decision is made based on the feature values to determine the appropriate path to follow. Ultimately, the leaves of the decision tree represent the predicted class labels for unseen texts (javaTpoint, 2021). Support vector machines (SVMs) offer yet another approach to classification. Their primary objective is to identify an optimal hyperplane within the feature space that effectively separates different classes. SVMs achieve this by maximizing the margin between the hyperplane and the nearest instances of each class, facilitating a clear distinction between the classes. By leveraging kernel functions to map the textual data into a higher-dimensional space, SVMs can adeptly handle complex relationships and capture intricate decision boundaries (Sunil, 2019).

The choice of classifier depends on the specific task and dataset characteristics. Each algorithm has its strengths and limitations, and researchers often experiment with different classifiers to find the most suitable one for a given text classification problem. One specific area where text classification is crucial is in analyzing user reviews in the e-commerce industry, especially on platforms like the Google Play Store. Understanding these reviews is vital for developers and businesses to improve their applications and services. Apptentive (2019) emphasizes the importance of higher ratings, which have a positive impact on downloads and revenue. Therefore, it becomes imperative to analyze and categorize user reviews of Shopee, the leading e-commerce platform in Malaysia (Similarweb, 2023), which are posted on the Google Play Store. The significance of this analysis is particularly noteworthy in Malaysia, where Shopee holds the distinction of being the top shopping app on both the Google Play Store and the Apple App Store, as reported by iPrice Group (2021) and Annie (2021).

Hu et al. (2019) conducted a study in which the researcher categorized customer reviews into three main aspects: e-commerce service, customer service, and application performance. Their research highlighted the importance of analyzing customer feedback in these specific areas. Furthermore, another relevant paper by Rajendran (2021) focused on the courier and delivery service industry, specifically addressing the challenges faced by courier companies in terms of customer ratings and the significance of employee commitment for customer loyalty. Building upon the insights from these two papers, this research aimed to conduct text classification based on three classes of customer reviews: application performance, customer service and delivery service. Analyzing customer reviews in relation to these three classes provides insights into the functionality and usability of e-commerce application, while evaluating reviews concerning delivery and customer service. By analyzing and categorizing user reviews on the Google Play Store, businesses and developers can gain valuable insights into the user experience of the Shopee application.

https://doi.org/10.24191/jcrinn.v9i2.410

This enables them to identify common issues and areas for improvement, taking actionable steps to enhance the application further. The benefits of text classification in the big data world are vast. By categorizing large amounts of textual data, businesses can identify patterns, trends, and insights that would otherwise be impossible to detect. This can help businesses make informed decisions and enhance their customer experience, according to a study by Duan et al. (2008).

However, it is important to note that there are also potential negative consequences of text classification. A study by Kim and Yoon (2020) argues that text classification algorithms may perpetuate existing biases and stereotypes present in the data, leading to discrimination against certain groups. Additionally, the authors argue that text classification may oversimplify complex issues, leading to inaccurate or incomplete conclusions. By categorizing customer feedback into specific categories like delivery service, customer service, and application performance, businesses can identify areas for improvement and enhance their offerings. TF-IDF is a technique that can help businesses analyze large amounts of textual data and make informed decisions. By using this technique, businesses can gain insights into customer sentiments and preferences, improve their offerings, and stay ahead of the competition.

The problem of text classification in the context of e-commerce is a complex one that has important implications for businesses, consumers, and society. In the digital age, the growth of e-commerce has led to an increasing volume of customer reviews and social media mentions, providing valuable insights into public sentiment towards products and services.

One of the best ways ensure the success of e-commerce business is to better serve customers and hearing them out when they have concerns or suggestions and improve their services accordingly. Due to this, Shopee must analyze thousands of complaints every day. Despite the dire need for an automated text classification, the key problem in natural language processing (NLP) is choosing the best feature extraction method that can be used to allow machines to automatically classify textual material into preset classes or categories. Feature extraction methods include the term frequency-inverse document frequency (TF-IDF) approach. Hence, this study evaluated the performance of TF-IDF as a feature extraction technique in text classification of Shopee user reviews on Google Play Store.

2. METHODOLOGY

The workflow begins with data gathering, data preparation and data analysis. First, the data gathering process starts with Python scraping of Shopee application reviews from the Google Play Store. The obtained data includes textual reviews, the number of ’thumbs up’ votes, and other relevant metadata. Then, a manual labelling procedure is initiated in which 3000 reviews are assigned to three distinct categories: “delivery service,” “customer service,” and “application performance.”

The reviews are then subjected to normalization procedures to facilitate reliable and consistent analysis. This includes spelling correction, conversion to lowercase, and removal of white space. In addition, standard text pre-processing procedures such as stop word removal, stemming, and lemmatization are implemented. After the data have been pre-processed, analysis begins. To accurately represent textual information, the TF-IDF feature extraction method is utilized. The extracted and pre-processed features are then used to train logistic regression, linear SVM, and decision tree classifier algorithms. The Fig. 1 shows the workflow of the research consisting of the first step which is data gathering, data preparation and data analysis.

https://doi.org/10.24191/jcrinn.v9i2.410

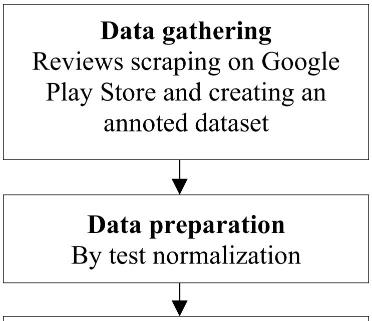

Data gathering
Reviews scraping on Google Play Store and creating an annoted dataset
Data preparation
By test normalization

Fig. 1. Research flowchart

2.1 Data Scraping

2.1.1 Extraction of Shopee Reviews on Google Play Store

User reviews of Shopee mobile application from the year 2021 to 2023 is scraped from Google Play Store using the google play scraper library in Python. The scraping criteria is defined by a target app package name of Shoopee (com.shopee.my), the language of English (en), the country of Malaysia (my), the sorting order of newest (Sort.NEWEST), and the desired number of reviews. The scraped reviews are then stored in a CSV file in write mode and uses the csv.writer to write the review data into the file. The CSV file is structured with headers: ’Text’, ’Date’ and ’Thumbs Up Count’. Each review is iterated through, and the relevant information (user name, review text, date, score, and thumbs-up count) is extracted and written as a row in the CSV file.

2.1.2 Creating an annotated data set

The first step in training a machine to carry the task of text classification is creating an annotated dataset. The process involves manually labelling 2309 text reviews into the class of customer service, delivery service and application performance. There are three steps of labelling textual reviews which is:

(i) **Define label categories**: Determining the labels that is assigned to textual reviews. In this research, the defined labels are delivery service, customer service and application performance.

(ii) **Create a Labelling System**: Establish guidelines or criteria for assigning labels to the text samples. Ensuring consistency and clarity in the labelling process to maintain the quality and reliability of the labelled data set. The guidelines used are:

- Customer Service: This label applies when the review specifically mentions the quality of customer service provided by the company or its representatives. Examples: “Excellent customer support,” “Unhelpful customer service,” “Quick response from support team.”

- Delivery Service: This label applies when the review discusses the delivery process, shipping speed, or overall satisfaction with the delivery service. Examples: “Fast delivery,” “Delayed shipment,” “Damaged package upon arrival.”

https://doi.org/10.24191/jcrinn.v9i2.410

• Application Performance: This label applies when the review focuses on the performance or functionality of the company's application, website, or digital platform. Examples: "App crashes frequently," "Smooth user experience," "Slow loading times."

(iii) **Label the Text Samples**: Go through each text sample and assign the appropriate label based on its content and context.

Table 1. Example of Shopee user reviews and its respective class label

Reviews	Label
Often times the app becomes unresponsive and laggy.	Application performance
Shopee express does not care to deliver the item on time.	Delivery service
I rarely buy anything here since the customer service is not very helpful.	Customer service

2.2 Data Preparation

2.2.1 Data Split

The original dataset that has been annotated is divided into input features (X) and corresponding labels (y). The dataset is then split into two parts - a training set and a test set. The purpose of this split is to train the model on the training data and evaluate its performance on the test data. 20% of the data is used for testing. The test set obtained from the previous step is further divided into two parts which are validation set and a final test set. 50% of the remaining test data is used for validation, and the remaining 50% is be used for the final test. The validation set is used for tuning the model's hyperparameters and selecting the best configuration while the final test set is kept separate and is used to evaluate the model's performance after all the hyperparameter tuning. Finally, separate datasets are created for the training, validation, and test sets, which include both the input features and their corresponding labels. By performing these splits, the model is trained on one set of data, tuned on another set, and evaluated on a completely independent set to get an unbiased assessment of its performance.

2.2.2 Text Normalisation

Text normalization in text classification refers to the process of transforming raw text data into a standardized format to improve the accuracy and effectiveness of classification models. It is needed because text data often contains inconsistencies, variations, and noise, which can negatively impact the performance of classification algorithms. Text normalization involves techniques such as converting text to lowercase, removing punctuation, stemming or lemmatization, removing stop words, and handling numerical and special characters. By normalizing the text, we reduce the dimensionality of the data, eliminate irrelevant details, and create a more consistent representation, which helps in improving the efficiency and accuracy of text classification models. In this subsection all of the steps in the text normalisation process are discussed.

2.3 Data Analysis

2.3.1 Descriptive Analysis

The purpose of descriptive analysis is to summarise or characterise a collection of data by the use of statistical methods. The ability of descriptive analysis to provide understandable insights from data that would otherwise be uninterpreted is one of the primary reasons for its extensive use (Bush, 2020). The comparison of class frequencies, the distribution number of characters and the average word length of each class reviews are analysed.

https://doi.org/10.24191/jcrinn.v9i2.410

2.3.2 Feature extraction

Feature extraction is a crucial step in text classification that entails transforming unstructured text data into a numerical representation. It seeks to identify the key characteristics of the text that can be used to differentiate between distinct classes or categories. These extracted features serve as input for the classifier, allowing it to discover patterns and make predictions based on the extracted features. Text classification relies on feature extraction because it enables the model to comprehend and interpret textual data effectively. TF-IDF is a feature extraction method used in this text classification tasks of three classes which is application performance, delivery service, and customer service. It aims to capture the importance of words within each class by considering both the term frequency (TF) and the inverse document frequency (IDF).

TF measures how frequently a word appears in a specific review. It is calculated by dividing the frequency of a term (word) in a review by the sum of frequencies of all terms in that class. The equation for TF is given by Eq. (1).

$$TF(t,d) = \frac{f_{t,d}}{\sum_{w \epsilon d} f_{w,d}} \qquad (1)$$

where ft,d represents the frequency of term t in review d, and the denominator is the sum of frequencies of all terms in class d. IDF, on the other hand, measures the rarity of a term across the entire class. It is calculated as the logarithm of the total number of documents in the class divided by the number of reviews that contain the term. The IDF equation is in Eq. (2).

$$IDF(t,D) = log\frac{N}{DF_t} \qquad (2)$$

where N represents the total number of reviews in the class, and DFt is the number ofreviews in the class that contain the term t. By combining TF and IDF, we obtain the TF-IDF weight, which reflects the importance of a word in a specific class. The TF-IDF score is calculated by multiplying the TF and IDF values as in Eq. (3).

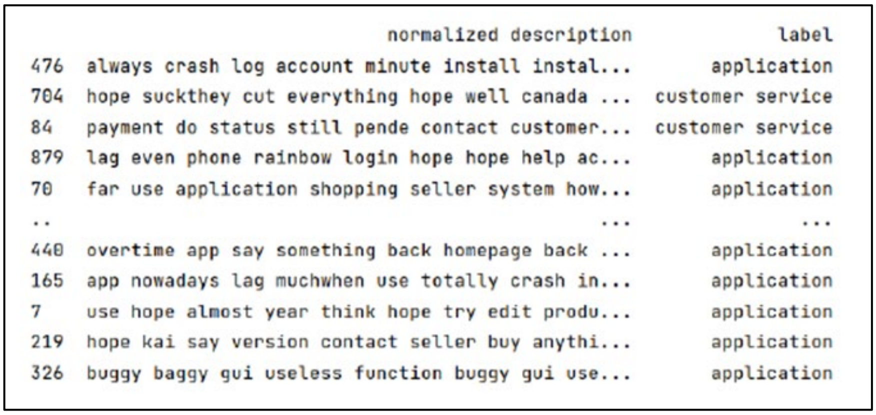
$$\text{TF} - \text{IDF}(t,d,D) = \text{TF}(t,d) \times \text{IDF}(t,D) \qquad (3)$$

TF-IDF assigns higher weights to words that are frequent within a specific review (high TF) and rare across the entire class (high IDF). These tend to be the most informative words for classification. Using TF-IDF as a feature extraction method for three classes like application performance, delivery service, and customer service allows us to capture the specific importance of words within each class. It enables the classification model to focus on the most relevant words for each class, improving the accuracy of the classification task.

3. RESULT AND DISCUSSION

Most of the reviews (51.5%) are related to application performance. Delivery service and customer service account for 24.5% (554 reviews) and 24.0% (501 reviews) of the reviews, respectively. In total, there are 2309 reviews in the dataset. Data is split into training data, validation data, and testing data. The training set 80% of the dataset with 1847 of the reviews, and the remaining 20% set is divided equally between the testing and validation set with 231 reviews respectively.

By employing text normalization techniques and preparing the user reviews of Shopee application for TF-IDF feature extraction, the resulting data in Fig. 2 is more consistent, less noisy, and optimised for capturing meaningful patterns and word relationships. This procedure improves the ability of machine learning models to extract insightful information from the reviews dataset.



```
                           normalized description              label
476  always crash log account minute install instal...       application
704  hope suckthey cut everything hope well canada ...  customer service
84   payment do status still pende contact customer...  customer service
879  lag even phone rainbow login hope hope help ac...       application
70   far use application shopping seller system how...       application
..                                                ...               ...
440  overtime app say something back homepage back ...       application
165  app nowadays lag muchwhen use totally crash in...       application
7    use hope almost year think hope try edit produ...       application
219  hope kai say version contact seller buy anythi...       application
326  buggy baggy gui useless function buggy gui use...       application
```

Fig. 2. TF-IDF normalization result on Shopee user reviews from Google Play Store

In order to assess the performance of TF-IDF as a feature extraction technique, the performance metrics of three classifiers on the task of text classification using TF-IDF feature extraction is presented as depicted in Table 2. The classifiers evaluated include Linear Support Vector Machine (SVM), Logistic Regression, and Decision Tree. The training accuracy and validation accuracy are reported for each classifier. The results provide insights into the effectiveness of these classifiers in accurately categorizing text documents based on TF-IDF features.

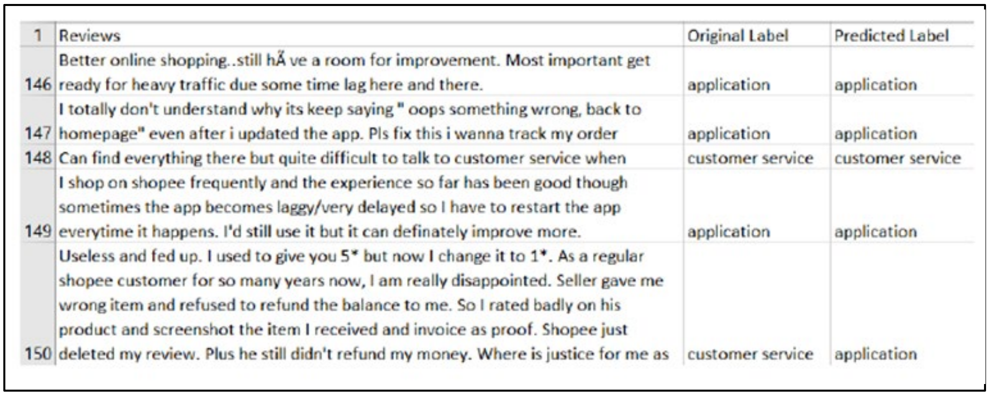
Table 2. Performance comparison of text classification using TF-IDF feature extraction techniques

No	Classifier	Training Accuracy	Validation Accuracy
1	Logistic Regression	0.885219	0.783550
2	Decision Tree	0.958311	0.696970
3	Linear SVM	0.908500	0.774892

Table 2 shows that Logistic regression classifier achieves a satisfactory degree of performance, with a training accuracy of 0.885219 and a validation accuracy of 0.783550. This demonstrates that the classifier performs well on both the training and validation sets, classifying nearly all instances accurately. Decision Tree, on the other hand, obtains a training accuracy of 0.958311 but a validation accuracy of 0.696970. The Linear SVM classifer exhibits strong performance, matching the training accuracy of the Linear SVM at 0.908500 and achieving a validation accuracy of 0.774. Overall, these classifiers exhibit varying levels of performance, with the Logistic Regression Classifier exhibiting the best generalisation capabilities to unseen data and demonstrating the highest level of performance.

Fig. 3 shows the result of text classification using TF-IDF technique as feature extraction and Logistic Regression as classifier in a csv file, the textual reviews from Google Play Store, original manually annotated labels and predicted labels is shown in this figure. It is shown that Logistic Regression classifier managed to label review number 146 until 149 correctly but fail to do so for review number 150 where it classifies a ’customer service’ review to an ’application’ review.

https://doi.org/10.24191/jcrinn.v9i2.410



1	Reviews	Original Label	Predicted Label
146	Better online shopping..still hĂ ve a room for improvement. Most important get ready for heavy traffic due some time lag here and there.	application	application
147	I totally don't understand why its keep saying " oops something wrong, back to homepage" even after i updated the app. Pls fix this i wanna track my order	application	application
148	Can find everything there but quite difficult to talk to customer service when	customer service	customer service
149	I shop on shopee frequently and the experience so far has been good though sometimes the app becomes laggy/very delayed so I have to restart the app everytime it happens. I'd still use it but it can definately improve more.	application	application
150	Useless and fed up. I used to give you 5* but now I change it to 1*. As a regular shopee customer for so many years now, I am really disappointed. Seller gave me wrong item and refused to refund the balance to me. So I rated badly on his product and screenshot the item I received and invoice as proof. Shopee just deleted my review. Plus he still didn't refund my money. Where is justice for me as	customer service	application

Fig. 3. Predicted labels using feature extraction TF-IDF and logistic regression classifier

4. CONCLUSION

TF-IDF is a highly used method for extracting features in text categorization. It assesses the significance of a word in a document inside a collection or corpus. This method computes a numerical value for every word in a document, which is subsequently employed to ascertain the significance of that word in relation to the content of the document. TF-IDF has shown effective in several applications, including text categorization, text clustering, and information retrieval. This research found that 51.5% which is majority of reviews are related to application performance. While delivery service is 24.5% and customer service account is 24%. In total, there are 2309 reviews in the dataset. Data is split into training data, validation data, and testing data. The training set 80% of the dataset with 1847 of the reviews, and the remaining 20% set is divided equally between the testing and validation set with 231 reviews respectively. This research also found that the Logistic Regression Classifier outperformed the other model and demonstrated the best generalization capabilities and proved to be the most effective in classifying text data accurately by using TF-IDF feature extraction.

5. ACKNOWLEDGEMENTS/FUNDING

This research was not funded by any grant.

6. CONFLICT OF INTEREST STATEMENT

The authors agree that this research was conducted in the absence of any self-benefits, commercial or financial conflicts and declare the absence of conflicting interests with the funders.

7. AUTHORS' CONTRIBUTIONS

Najwa Umaira binti Che Mohd Safawi carried out the research and wrote the article. **Nur Amalina Shafie** supervised the research progress, revised the article, and approved the article submission.

8. REFERENCES

Annie, A. (2021). *The state of mobile 2021*. https://www.appannie.com/en/go/state-of-mobile-2021/

Apptentive. (2019). Mobile app benchmarks: The average ratings, reviews, and retention rates. *Apptentive*

https://doi.org/10.24191/jcrinn.v9i2.410

Blog. https://www.apptentive.com/blog/2019/03/11/mobile-app-benchmarks-the-average-ratings-reviews-and-retention-rates/

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Bush, T. (2020, Jun). Descriptive analysis: How-to, types, examples. *Pestle Analysis*. https://pestleanalysis.com/descriptive-analysis/

Duan, W., Gu, B., & Whinston, A. B. (2008). Do online reviews matter? An empirical investigation of panel data. *Decision Support Systems*, *45*(4), 1007–1016.

Hu, P., Li, Q., & Ye, Y. (2019). Customer review analysis using natural language processing techniques: A case study of e-commerce platforms. *Sustainability*, *11*(8), 2234.

iPrice Group. (2021). *iPrice insights: State of ecommerce in Southeast Asia 2021*. https://iprice.my/insights/mapofecommerce/en/

javaTpoint. (2021). *Machine learning decision tree classification algorithm - javatpoint*. https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm

Kanaris, I., Stamatatos, E., & Fakotakis, N. (2020). Tf-idf vs word2vec vs glove: An overview. *arXiv preprint*. arXiv:2010.02545

Kim, M.-G., & Yoon, Y.-J. (2020). Negative consequences of text classification: A critical review and practical remedies. *Journal of the Association for Information Science and Technology*, *71*(8), 936–947.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2015). Efficient estimation of word representations in vector space. *arXiv preprint*. arXiv:1301.3781

Raj, A. (2020, Nov). Perfect recipe for classification using logistic regression. *Towards Data Science*. https://towardsdatascience.com/the-perfect-recipe-for-classification-using-logistic-regression-f8648e267592#:~:text=Logistic%20regression%20is%20a%20classification

Rajendran, S. (2021). Improving the performance of global courier & delivery services industry by analyzing the voice of customers and employees using text analytics. *International Journal of Logistics Research and Applications*, *24*(5), 473-493. https://doi.org/10.1080/13675567.2020.1769042

Similarweb. (2023, Jul). Top websites ranking most visited ecommerce shopping websites in Malaysia. *Similarweb LTD*. https://www.similarweb.com/top-websites/malaysia/e-commerce-andshopping/#:~:text=shopee.com.my%20ranked%20number,eCommerce%20%26%20Shopping%20websites%20in%20Malaysia.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, *28*(1), 11–21.

Sunil, R. (2019, Mar). *Understanding support vector machine algorithm from examples (along with code)*. https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

Turney, P. D. (2002). *Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews*. Association for Computational Linguistics.

Wei, H. (2019, Mar). Nlp pipeline 101 with basic code example—feature extraction. *Voice Tech Podcast*. https://medium.com/voice-tech-podcast/nlp-pipeline-101-with-basic-code-example-feature-

https://doi.org/10.24191/jcrinn.v9i2.410

extraction-ea9894ed8daf#:~:text=Feature%20extraction%20step%20means%20to

https://doi.org/10.24191/jcrinn.v9i2.410