Supervised Machine Learning Models for X Sentiment Analysis

Angeline Mary Marchella
Computer Science Department
School of Computer Science
Jakarta, 11480, Indonesia
angeline.marchella@binus.ac.id

Naufal Hardiansyah
Computer Science Department
School of Computer Science
Jakarta, 11480, Indonesia
naufal.hardiansyah@binus.ac.id

Esther Vemberly
Computer Science Department
School of Computer Science
Jakarta, 11480, Indonesia
esther.vemberly@binus.ac.id

Timothy Liundi
Computer Science Department
School of Computer Science
Jakarta, 11480, Indonesia
timothy.liundi@binus.ac.id

***Abstract*—Social media applications have been widely used to connect and express one's emotions and opinions on certain topics. Specifically, X is a platform where people can share their feelings in forms of text that they call tweets. To determine whether a tweet contains positive, negative, or neutral intention, sentiment analysis with Natural Language Processing is performed. Numerous techniques have been invented to tackle this task, either by utilizing deep learning or machine learning. This paper focused on discussing three different supervised machine learning models to perform tweets sentiment classification, namely Support Vector Machine (SVM), Logistic Regression (LR), and K-Nearest Neighbors (KNN). Each model is trained with a tweet dataset taken from Kaggle. Then, several pre-processing steps are done before passing the data to the model. TF-IDF vectorizer is also used to gain better results. Four common evaluation metrics are used to judge each model's performance: accuracy, precision, recall, and f1-score. The experiment proved that SVM had the highest accuracy (93%), followed by LR (89%) and KNN (81%). Other metrics followed the same order. This showed that SVM is better in handling text data without needing to put much attention on outliers that other two methods.**

Keywords—X, Sentiment Analysis, Natural Language Processing, Supervised Machine Learning, Support Vector Machine, Logistic Regression, K-Nearest Neighbors, Classification

I. Introduction

Nowadays, social media platforms are not only used to connect with each other, but also have been vastly used to express one's feeling towards one topic. Photo sharing, video sharing, blogs, forums, microblogs, and social networks are some examples of social media technologies that are widely used to express one's emotion on the internet. As the popularity of these technologies' increases, various online social networking applications, such as Instagram, Facebook, YouTube, and X (formerly Twitter), exist to accommodate people in sharing their opinions and thoughts.

There are around 4.9 billion social media users globally. This means that 60.49% of the global population are exposing their social life on the internet [1]. For this reason, the publications in social networks are useful sources for understanding and studying users' behavior, opinions, and intentions on certain topics, products, or organizations. As a result, thematic and emotive components of posts' content are significantly important for these kinds of tasks.

X is one of the most popular social media sites that provides writers freedom to share their opinions, thoughts, and emotions in the form of tweets. Based on data, X has 237.8 daily active users in 2023 [2]. Since the number of users is large, it is easy to gain text data through this platform. The variety of words is also more diverse than the other social media text platforms. Therefore, in this research, we focused on analyzing X users' feelings by looking at their tweets.

Feelings that people wrote in text can be extracted and analyzed by making use of natural language processing (NLP). This process is called sentiment analysis, in which the result can be grouped to be either positive, negative, or neutral. Specifically, sentiment analysis on X data can be done by extracting the tweets. Then, some sentiment algorithms are performed on these tweets data. In this study, we applied several machine learning (ML) algorithms, called multinomial logistic regression (LR), support vector machine (SVM), and K-Nearest Neighbors (KNN), to classify the text into the three basic categories.

II. Literature Review

For the past few years, many researchers have studied and developed different methods to tackle text sentiment analysis problems. In this section, we provide research summaries about related works.

A study suggested dependency parsing with sentiment relationship migration and adjusted distance for the sentiment analysis of short text in microblogs. The result of the study demonstrated the effectiveness of the strategy and the value of the sentiment structure in recognizing the sentiment of short texts. Additionally, it demonstrated how each sentence contributes differently to the short text sentiment computation and how varied relationships between the modifier and the emotion term might have an impact [3].

Another research provided a novel twitter sentiment analysis method that combines weighted text feature modeling (WTFM) and sentiment-specific word embeddings. In the study, they compared several models, such as Word2vec, SSWE, and NRC, with the proposed method. The research concluded that the WTFM model is quick to construct and efficient when compared to other methods for twitter sentiment analysis feature generation [4].

Three different sentiment analysis algorithms, namely logistic regression (LR), BERT, and VADER, have been compared in a published paper. The comparison result has shown that VADER and LR are both less accurate than BERT.

Since BERT looks for the aspect of the sentences, it is more accurate than other algorithms. In order to improve the performance of feature selection, VADER will search for valence and polarity scores. Logistic regression does not examine the tweets' polarity or other aspects [5].

Several researchers used four different Arabic datasets of various sizes to compare three text categorization classifiers. The classifiers that they have been used are decision tree (DT), K-Nearest Neighbors (KNN), and logistic regression (LR). They also implemented Bag of Word (BOW) to preprocess the data with tokenization and light stemming. In addition, TF-IDF was utilized to select features. Researchers concluded that, in most circumstances, especially when working with a large dataset, the LR classifier is more accurate than KNN and DT. In contrast, KNN performed better and was more accurate when dealing with small datasets [6].

A study conducted sentiment analysis experiments utilizing three different approaches: dictionary-based, machine learning, and deep learning. The results of an emotional analysis of the text are frequently subpar if the text has a significant number of new words and unregistered terms. Due of the richness of semantic expression, machine learning will not produce significant errors compared to dictionary matching. However, the selection of feature dimensions is crucial to the machine learning approach. In the study, the deep learning method outperformed the other two methods. But the deep learning approach is vulnerable to overfitting and necessitates the choice of the best parameters using cross-validation methods [7].

After analyzing previous works, we observed that there are many approaches that can be used to analyze the sentiments of texts. In this paper, we will focus on discussing and implementing three supervised learning models, namely multinomial logistic regression (LR), support vector machine (SVM), and K-Nearest Neighbors (KNN) to find the best model for sentiment analysis on tweets in X platform.

III. Methodolgy

For this study, the workflow consists of five main phases: dataset preparation, text preprocessing, data transformation with TF-IDF vectorizer, model implementation, and performance evaluation. The visualization of each step is visualized in a flowchart as shown in Fig. 1.

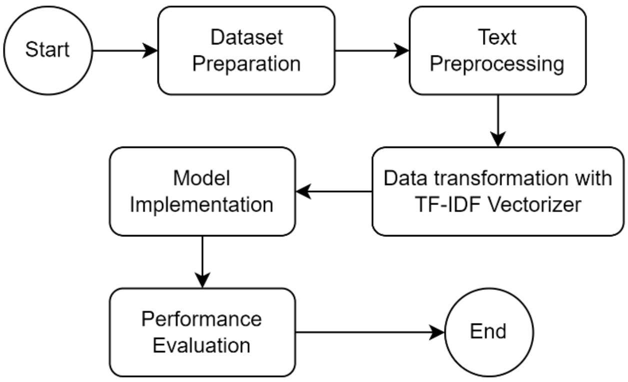


Fig. 1. Research Workflow

A. Dataset preparation

The dataset used in this study is taken from Kaggle, which consists of 298,728 rows of data [8]. There are four columns in the dataset, namely ID, entity, sentiment, and content. In this research, we only focused on analyzing the sentiment and content columns. The sentiment is divided into three classes: positive, negative, and irrelevant. The irrelevant class is renamed as neutral. Each sentiment class has a different proportion of data, which is shown in Fig. 2. The data consists of 27.9% positive data, 30.2% negative data, and 41.9% neutral data.

Some studies have determined the appropriate dataset splitting, which is 70–80% for training and 20-30% for testing, to produce the best results [9]. Hence, we later separate 59,196 rows for training and 14,800 rows for testing after the preprocessing stage.

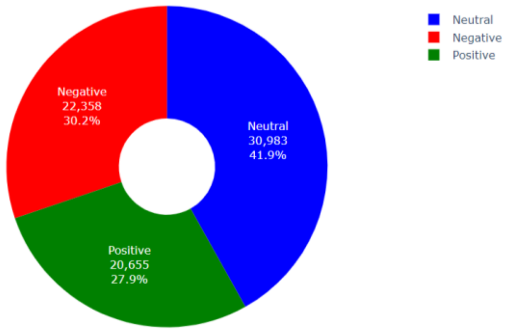


Fig. 2. Sentiment Distribution

B. Text preprocessing

Before the collected data are used for analysis, we need to perform several text preprocessing steps. For text sentiment analysis, text preprocessing is important to improve the performance and accuracy of the natural language processing (NLP) system [10]. These are some steps that we have implemented to turn the text into a more structured format:

1) Lower-casing

Converting case is one of the simplest text preprocessing stages, yet very important to make all strings follow a consistent format. In the NLP system, words like "Page" and "page" might be represented as two different words in the vector space model, thus resulting in more dimensions [11]. Therefore, we convert strings in the content column into lowercase by using Python built-in lower() function.

2) Tokenization

Tokenization is a technique used in NLP to break down sentences into smaller language-assignable units, called tokens [12]. For this research process, we utilized word tokenization to separate text into words. To achieve it, tokenize() function from NLTK library is imported.

3) Generalization

There are several tweet contents in the dataset that include links and username tags. Since different links and usernames have low occurrence probabilities, they are generalized into certain tokens by using re.sub() function. Links are replaced with "URL" token, while usernames are replaced with "USER" token.

4) Stopwords removal

Stopwords are frequently used words that are eliminated from the text since they add nothing to the analysis. These phrases have little or no meaning [13]. For every word in a tweet, the program checked whether it is belonged in the English stopword list. If it is not in the list, then the word is appended with other strings.

5) Lemmatization

Lemmatization is a process of transforming a word to its base or dictionary form. This stage is significant because it can increase the accuracy of NLP models. By reverting words to their root forms (or lemma), it can lower the percentage of unique words in a text [14]. WordNetLemmatizer() from NLTK.stem is used to perform this task.

C. Data transformation with TF-IDF Vectorizer

Machine learning models can only understand numerical data. Thus, tweet texts must be transformed into numerical. A popular transformation or vectorization technique in NLP is called TF-IDF. TF-IDF vectorizer is calculated by multiplying term frequency (TF) value and inverse document frequency (IDF) value. TF represents the probability of a term that appears in a document. On the other hand, IDF shows how the frequency of a term presents in a set of documents is inversely proportional, meaning the more often a term appears in a set of documents, the less important the term is [15]. This vectorizer can be easily implemented by calling TfidfVectorizer() function. Then, we also specified the ngram_range parameter into trigram. It is used to set the number of tokens in a term to be calculated by the vectorizer [16].

D. Model implementation

A type of machine learning (ML) called supervised machine learning discovers connections between input and output data. The inputs are typically known as features or "X variables" while the output is referred to as the target or "y variable". A dataset with features and target is called labeled data [17]. In addition, this kind of machine learning is often applied for tasks such as classification and regression. In this study, we implement ML models for classification or segmentation task. Classification involves the process of assigning data to different classes [18]. There are many types of algorithms for supervised learning to classify the feature with the target value. This research applied three types of algorithms, namely multinomial logistic regression (LR), support vector machine (SVM), and K-Nearest Neighbors (KNN).

1) Multinomial Logistic Regression

Logistic regression (LR) is a machine learning algorithm that classifies data based on logistic function. Basically, LR is applied when the dependent variable is binary or dichotomous [19]. Nevertheless, it is divided into three types, namely binary LR, multinomial LR, and ordinal LR. Specifically, since the tweets are segmented into three classes (positive, negative, and neutral), this paper implemented the multinomial LR, which is applied when there are multiple outcome categories and not ordered [20].

2) Support Vector Machine

Support vector machine (SVM) is a supervised machine learning approach for regression or classification. It seeks to find the best boundary between the possible outputs. In other words, the main objective of this technique is to find a hyperplane in an n-dimensional space that maximizes the separation of the data points to their prospective classes. The basic type of SVM does not support multiclass classification. However, we can still use SVM for multiclass classification. To achieve this, there are two approaches, which are one-to-toe approach and one-to-rest approach. Research has shown that one-to-one approach produced slightly better result than one-to-rest approach [21]. Hence, this study applied the one-to-one technique, meaning it breaks down the problem into multiple binary classification problems. More specifically, the classifier will use m SVMs, and each SVM predicts membership in one of the m classes [22].

3) K-Nearest Neighbors

The supervised learning technique K-Nearest Neighbors (KNN) is used for both regression and classification. By calculating the distance between the test data and all the training points, KNN can predict the proper class for the test data. The most popular way to find the distance between each point is by calculating the Euclidean distance [23]. Then, it chooses the K spots that are closest to the test data. The KNN method determines which of the classes of the K training data the test data will belong to by choosing the class with the highest probability [24].

E. Performance evaluation

In this study, we gave a thorough and objective evaluation of each model's performance through comparison of its accuracy, precision, recall, and f1-score. In terms of all predictions, accuracy refers to the percentage of predictions that are correct. Precision is a metric that counts the proportion of accurate positive predictions among all successful positive predictions. Recall gauges the proportion of true positive predictions to the total of true positives and false negatives. It provides a quantitative measurement based on the ratio of true positives that the model successfully detected. A statistic that combines recall and precision is called the f1-score. Since it combines precision and recall in a harmonic way, it equally weighs false positives and false negatives [25].

IV. Result & Discussion

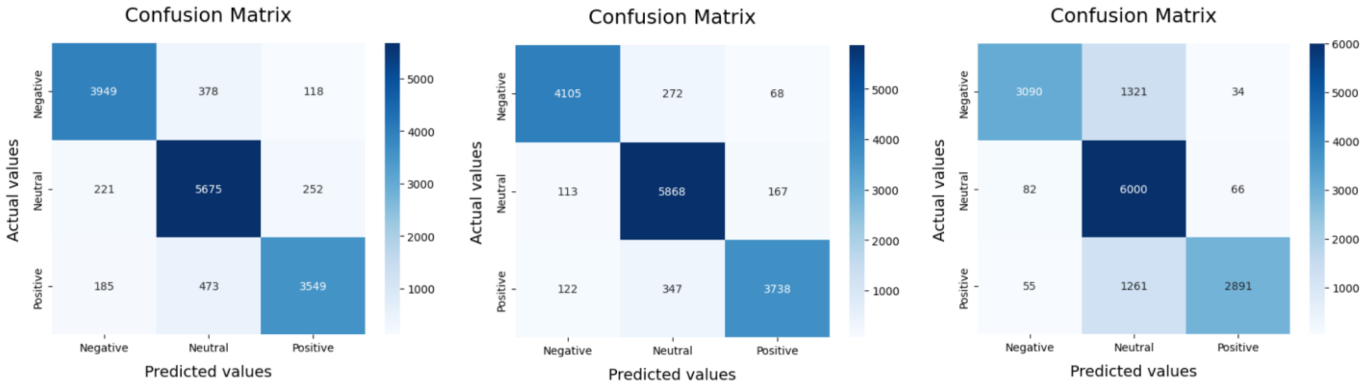
This section discusses the result of supervised machine learning algorithms in performing X sentiment analysis. The experiment is run by using the Google Colaboratory platform. Pandas, Numpy, WordCloud, Matplotlib, Plotly, Seaborn, NLTK, Sklearn, and Tensorfow are Python libraries that we used to perform data exploration, visualization, text preprocessing, and model training. More specifically, we called pretrained models, which are LogisticRegression, SVC, and KNeighborsClassifier from the Sklearn library.

Each model has its own parameters that need to be set. Since this study tried to classify the text into three classes, the multi_class parameter value is set to 'multinomial' and the max_iter is 1,000 for the LR model. According to our discussion in methodology section, we implemented a one-to-one SVC approach, thus the decision_function_shape parameter is set to 'ovo'. Ultimately, the n_neighbors should be set to three in KNN model according to the number of groups that we expect to have.

After models are trained by using the training set, the test set is tested and evaluated. Results are shown in the form of confusion matrix for each model as shown in Fig. 3. Previously, the sentiment distribution of our dataset shows that the neutral class has the largest proportion among the other classes. This can affect the performance of our model which the model might perform well on the majority class but poorly on the minority class. This statement aligns with the result which describes a slightly higher amount of the neutral class's True Positive (TP).

The confusion matrix then produced four main evaluation metrics, namely accuracy, precision, recall, and f1-score. The detailed scores are shown in Table 1. The SVM model occurred to have the highest accuracy score at 93%, followed by LR at 89% and KNN at 81%. On average, SVM's precision, recall, and f1-score results outperformed the other two models. SVM got 93% of precision, 92% of recall, and 93% of f1-score. Meanwhile, both precision, recall, and f1-score metrics of LR are finished at 89%. The least is KNN, which stopped at 87% precision, 79% of recall, and 81% of f1-score.

There may be several reasons why SVM turned out to be the best model for classifying the sentiment of this tweet dataset. First, SVM is well-known to be less sensitive to outliers, as it derives maximum margin solution. Hence, the possibility of the model to overfit is less than the others [26]. Also, it is proven to work well with unstructured and semi-structured data like text and images. However, the drawback of this model is the slow training process as the dataset becomes large [27]. LR is in the second position. It is a convenient, quick, and straightforward classification method. Nevertheless, the existence of outliers can plummet its performance. In addition, we can assume that the data points are likely followed almost a linear pattern since the LR result is better than KNN [28]. Despite its simplicity, the KNN model is sensitive to its hyperparameter value. Especially, if we do not tune the number of k properly, it will not give an optimal performance [29].



(a) Logistic Regression (b) Support Vector Machine (c) K-Nearest Neighbors

Fig. 3. Confusion Matrix of the Three Models

TABLE I. SUPERVISED MACHINE LEARNING MODELS PERFORMANCE COMPARISON

Model	Sentiment	Accuracy	Precision	Recall	F1-Score
Logistic Regression	Negative	0.89	0.91	0.89	0.90
	Neutral		0.87	0.92	0.90
	Positive		0.91	0.84	0.87
	Avg/Total		0.89	0.89	0.89
Support Vector Machine	Negative	**0.93**	0.95	0.92	0.93
	Neutral		0.90	0.95	0.93
	Positive		0.94	0.89	0.91
	Avg/Total		**0.93**	**0.92**	**0.93**
K-Nearest Neighbors	Negative	0.81	0.96	0.70	0.81
	Neutral		0.70	0.98	0.81
	Positive		0.97	0.69	0.80
	Avg/Total		0.87	0.79	0.81

V. CONCLUSION

In this study, three classic supervised machine learning models, namely LR, SVM, and KNN algorithms, were trained and tested for the X sentiment analysis task. The comparison result showed that SVM achieved the highest score in all evaluation metrics. It got 93% for the accuracy, precision, and f1-score, and 92% of recall. The LR method followed the second spot with a score of 89% for all metrics. The last one is KNN with 81% of accuracy and f1-score, 87% of precision, and 79% of recall. The result indicated that SVM architecture is more suitable for training text datasets than the others. Without taking any steps to remove outliers, SVM can still produce a promising result. Meanwhile, the other two models are sensitive to the existence of outliers. Several things can be improved in future research, including finding more appropriate hyperparameter values, thoroughly exploring and cleaning outliers in the dataset, and experimenting with more extensive and diverse datasets. These actions may help to improve models' performance and robustness in classifying the sentiment of tweets in X platform.

References

[1] R. Shewale, "Social Media Users — Global Demographics (2023)," DemandSage. Accessed: Sep. 27, 2023. [Online]. Available: https://www.demandsage.com/social-media-users/#:~:text=There%20are%204.9%20billion%20social,platform%2C%20with%203.03%20billion%20users.

[2] R. Shewale, "Twitter Statistics In 2023 — (Facts After 'X' Rebranding)," DemandSage. Accessed: Sep. 27, 2023. [Online]. Available: https://www.demandsage.com/twitter-statistics/#:~:text=Twitter%20Statistics%202023%20(Top%20Picks)&text=Twitter%20has%20around%20528.3%20million,daily%20active%20users%20(mDAU).

[3] J. Li and L. Qiu, "A Sentiment Analysis Method of Short Texts in Microblog," in *22017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, IEEE, Jul. 2017, pp. 776–779. doi: 10.1109/CSE-EUC.2017.153.

[4] Q. Li, S. Shah, R. Fang, A. Nourbakhsh, and X. Liu, "Tweet Sentiment Analysis by Incorporating Sentiment-Specific Word Embedding and Weighted Text Features," in *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, IEEE, Oct. 2016, pp. 568–571. doi: 10.1109/WI.2016.0097.

[5] A. J. Nair, V. G, and A. Vinayak, "Comparative study of Twitter Sentiment On COVID - 19 Tweets," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, Apr. 2021, pp. 1773–1778. doi: 10.1109/ICCMC51019.2021.9418320.

[6] N. K. Bolbol and A. Y. Maghari, "Sentiment Analysis of Arabic Tweets Using Supervised Machine Learning," in *2020 International Conference on Promising Electronic Technologies (ICPET)*, IEEE, Dec. 2020, pp. 89–93. doi: 10.1109/ICPET51420.2020.00025.

[7] W. Niu and L. Wu, "Sentiment Analysis and Contrastive Experiments of Long News Texts," in *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, IEEE, Dec. 2019, pp. 1331–1335. doi: 10.1109/IAEAC47372.2019.8997550.

[8] K. Morgado, K. Arbuzova, and N. G. Bảo, "Twitter Sentiment Analysis Dataset," Kaggle. Accessed: Oct. 01, 2023. [Online]. Available: https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis

[9] A. Gholamy, V. Kreinovich, and O. Kosheleva, "Why 70/30 or 80/20 Relation Between Training and Testing Sets: A Pedagogical Explanation," 2018. [Online]. Available: https://scholarworks.utep.edu/cs_techrephttps://scholarworks.utep.edu/cs_techrep/1209

[10] M. Saeed, "A Guide to Text Preprocessing Techniques for NLP," Scale Virtual Events. Accessed: Oct. 01, 2023. [Online]. Available: https://exchange.scale.com/public/blogs/preprocessing-techniques-in-nlp-a-guide

[11] Harshith, "Text Preprocessing in Natural Language Processing," Harshith.

[12] A. Burchfiel, "What is NLP (Natural Language Processing) Tokenization?," Tokenex. Accessed: Oct. 01, 2023. [Online]. Available: https://www.tokenex.com/blog/ab-what-is-nlp-natural-language-processing-tokenization/#:~:text=Tokenization%20is%20used%20in%20natural,into%20understandable%20parts%20(words).

[13] Deepanshi, "Text Preprocessing in NLP with Python Codes," Analytics Vidhya.

[14] E. Gomede, "The Importance of Lemmatization for Natural Language Processing," Medium. Accessed: Oct. 01, 2023. [Online]. Available: https://medium.com/@evertongomede/the-importance-of-lemmatization-for-natural-language-processing-24b6fe9f534b

[15] S. Somasunadaran, "TF-IDF in 2 minutes," Medium. Accessed: Oct. 01, 2023. [Online]. Available: https://medium.com/@soumyas567/tf-idf-in-2-minutes-3deb52f3a6fe

[16] Nithyashree, "What Are N-Grams and How to Implement Them in Python?," Analytics Vidhya. Accessed: Oct. 01, 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/09/what-are-n-grams-and-how-to-implement-them-in-python/#:~:text=N%2Dgrams%20are%20continuous%20sequences,(Natural%20Language%20Processing)%20tasks.

[17] M. All, "Supervised Machine Learning," Datacamp. Accessed: Oct. 01, 2023. [Online]. Available: https://www.datacamp.com/blog/supervised-machine-learning

[18] S. Shinde, "What is Supervised Learning in Machine Learning? A Comprehensive Guide," Emeritus. Accessed: Oct. 01, 2023. [Online]. Available: https://emeritus.org/blog/ai-and-ml-supervised-learning/#:~:text=Supervised%20learning%20is%20utilized%20in,learning%20algorithm%20in%20such%20scenarios.

[19] A. Saini, "Conceptual Understanding of Logistic Regression for Data Science Beginners," Analytics Vidhya. Accessed: Oct. 01, 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/08/conceptual-understanding-of-logistic-regression-for-data-science-beginners/

[20] Keerthana, "Multinomial Logistic Regression," Great Learning. Accessed: Oct. 01, 2023. [Online]. Available: https://www.mygreatlearning.com/blog/multinomial-logistic-regression/

[21] D. Alita, Y. Fernando, and H. Sulistiani, “Implementasi Algoritma Multiclass SVM Pada Opini Publik Berbahasa Indonesia di Twitter,” *Jurnal TEKNOKOMPAK*, vol. 14, no. 2, p. 86, 2020.

[22] Baeldung, “Multiclass Classification Using Support Vector Machines,” Baeldung. Accessed: Oct. 01, 2023. [Online]. Available: https://www.baeldung.com/cs/svm-multiclass-classification

[23] O. Harrison, “Machine Learning Basics with the K-Nearest Neighbors Algorithm,” Towards Data Science.

[24] A. Christopher, “K-Nearest Neighbor,” The Startup. Accessed: Oct. 01, 2023. [Online]. Available: https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4

[25] “Classification: Precision and Recall | Machine Learning | Google Developers,” Google. Accessed: Oct. 02, 2023. [Online]. Available: https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall

[26] D. Varghese, “Comparative Study on Classic Machine learning Algorithms,” Towards Data Science. Accessed: Oct. 15, 2023. [Online]. Available: https://towardsdatascience.com/comparative-study-on-classic-machine-learning-algorithms-24f9ff6ab222#:~:text=SVM%20can%20handle%20non%2Dlinear,it%20derives%20maximum%20margin%20solution.

[27] P. Bassey, “Logistic Regression Vs Support Vector Machines (SVM),” Axum Labs. Accessed: Oct. 15, 2023. [Online]. Available: https://medium.com/axum-labs/logistic-regression-vs-support-vector-machines-svm-c335610a3d16

[28] P. Shukla, “Logistic Regression vs. K Nearest Neighbors in Machine Learning,” PythonKitchen. Accessed: Oct. 15, 2023. [Online]. Available: https://www.pythonkitchen.com/logistic-regression-vs-k-nearest-neighbors-in-machine-learning/

[29] V. K. Das, “Logistic Regression vs K-Nearest Neighbors vs Support Vector Machine,” Global Tech Council. Accessed: Oct. 15, 2023. [Online]. Available: https://www.globaltechcouncil.org/machine-learning/logistic-regression-vs-k-nearest-neighbours-vs-support-vector-machine/