ICIC Express Letters
Part B: Applications
Volume **15**, Number **8**, August **2024**

ICIC International ©2024 ISSN 2185-2766
pp. **787–794**

USER PREFERENCE MINING USING SENTIMENT ANALYSIS ON E-WALLETS REVIEWS

Amil Ahmad Ilham*, Anugrayani Bustamin and Ainun Annisa Kahar

Department of Informatics
Universitas Hasanuddin
Jalan Poros Malino Km. 6, Gowa 92171, Indonesia
anugrayani@unhas.ac.id; kaharaa18d@student.unhas.ac.id
*Corresponding author: amil@unhas.ac.id

Received November 2023; accepted January 2024

Abstract. *During the pandemic era, there has been an increase in the use of E-Wallet as the payment alternative that supports physical distancing activities. However, with so many E-Wallet offers, it is difficult for users to choose the suitable E-Wallet service. In this study, we proposed an E-Wallet recommendation based on user preference mining using sentiment analysis. We developed three machine learning classification techniques: Multinomial Naïve Bayes, Jurafsky Naïve Bayes, and Support Vector Machines, for sentiment classification. The Ensemble Hard Voting model with modifications was also built to classify sentiment based on the results of those three classification algorithms. The experiment results indicate that Multinomial Naïve Bayes algorithm obtained the highest accuracy rate at 80.3%, while Ensemble Hard Voting, Jurafsky Naïve Bayes, and Support Vector Machine achieved accuracy rates of 80.1%, 79%, and 79%, respectively. Based on these results, we implemented the Multinomial Naïve Bayes model for E-Wallet recommendations.*

Keywords: Sentiment analysis, Naïve Bayes, Support Vector Machine, Ensemble learning

1. **Introduction.** The increasing trend and use of E-Wallets since the pandemic have increased rapidly as a payment method due to the limitations of community activities. This increase was accompanied by various online user reviews, especially in reviews of the E-Wallet application on the Google Play Store. Research conducted that ShopeePay, Gopay, Dana and LinkAja are some E-Wallets that are most widely used [1]. Bank Indonesia states that in 2020, 47 E-Wallet companies already have official licenses [2]. The study stated that 91% users examined customer reviews and 46% users claimed that reviews impacting their decisions before purchasing a product [3]. Analysis of reviews can be done by looking at product ratings, but ratings alone are not enough to represent the intent of the overall review by customers [4]. In recommendations, user reviews are essential tools [5]. Users are usually given recommendations based on user history and profiles, but now recommendations are starting to be explored more so that they make recommendations based on sentiment analysis. A survey on sentiment analysis covered many approaches, and different researchers discussed that machine learning techniques are deemed efficient for analyzing the sentiment-laden terms in a document [6]. Sentiment analysis employs various categories of machine learning algorithms, including the Multinomial Naïve Bayes technique, Support Vector Machine (SVM), Logistic Regression, Random Forest, and Decision Tree [7].

A comparative review of sentiment analysis was done using various machine learning techniques such as Naïve Bayes, Logistic Regression, SVM, and Random Forest. The results show SVM tends to excel in the majority of cases, but the outcomes may vary

DOI: 10.24507/icicelb.15.08.787

depending on the specific application [8]. Ahmad et al. [9] show that sentiment analysis of Indonesian tweets using the Naïve Bayes method obtained higher accuracy than the lexicon-based method. A proposed method by Deviani et al. [10] determines and analyzes sentiment towards the quality of digital wallets based on the e-servqual dimension using the Naïve Bayes algorithm. Customized user recommendations also offer a superior option for assessing a product by conducting sentiment analysis on reviews to ascertain the sentiment polarity of its features [11]. In [12], the authors develop an e-commerce product recommendation system by proposing a new model that improves the performance of the conventional Naïve Bayes algorithm. The proposed model only excels on negative rather than neutral and positive sentiments. In this study, we developed a similar system but used different classification algorithms, namely Multinomial Naïve Bayes, Jurafsky Naïve Bayes, and Support Vector Machines to classify sentiments. We also proposed an Ensemble Hard Voting model with modifications to combine the prediction results of those three classification techniques.

The remainder of this paper follows this structure. In Section 2, we present an outline of the research methodology, encompassing aspects like data collection, preprocessing, elucidation of particular algorithms, and evaluation methods. In Section 3, the results are discussed. Finally, Section 4 provides some conclusions and future work.

2. **Research Method.** This study built a sentiment analysis system and compared three classification techniques: Multinomial Naïve Bayes algorithm, Jurafsky Naïve Bayes algorithm, and Support Vector Machine algorithm. An Ensemble Hard Voting model with modifications was also built to combine the prediction results of those three classification techniques. The dataset was obtained from an Indonesian review of five E-Wallets on the Google Play Store. The built system will produce a polarity prediction output that will be used as an E-Wallet recommendation. Figure 1 illustrates the diagram depicting the proposed approach.

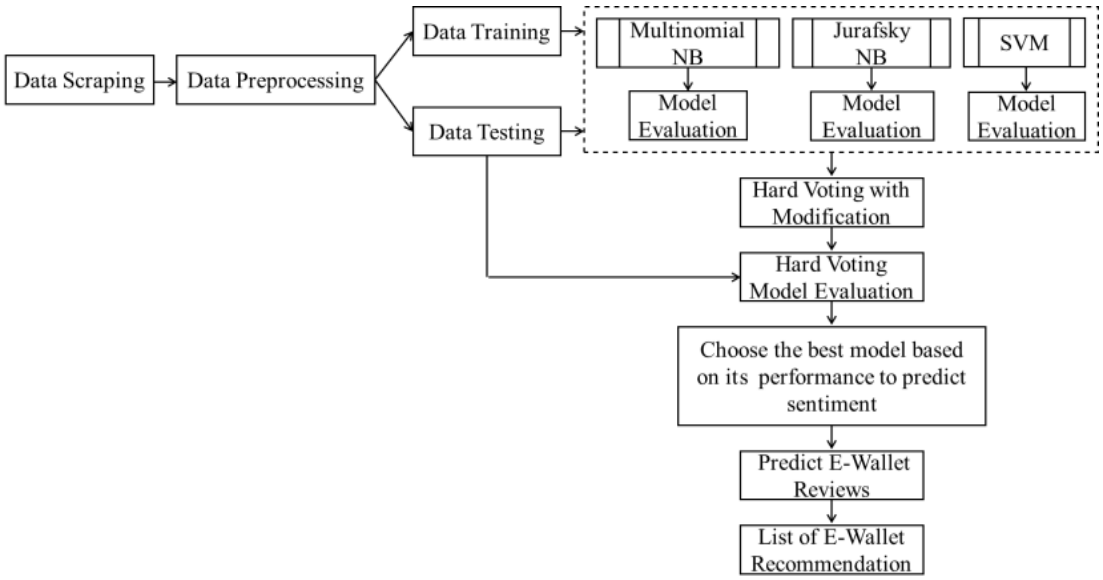


FIGURE 1. Proposed method diagram

2.1. **Data collection.** Data collection was done by scraping user reviews of the E-Wallets application on the Google Play Store. Dana, iSaku, Jenius, LinkAja, and OVO are E-Wallets that have the highest number of downloads in Google Play Store in Indonesia. The dataset that was successfully collected for each review was manually labeled according to the sentiment: negative was labeled as 0, neutral was labeled as 1, and positive was labeled as 2.

2.2. **Data preprocessing.** After sufficient datasets had been collected and labeled according to sentiment, the data was preprocessed first to obtain clean dataset. The data preprocessing included text cleansing, lowercasing, stopwords removal, and stemming.

2.3. **Multinomial Naïve Bayes.** Naïve Bayes is a commonly used method for categorization and training. Naïve Bayes is a classification method based on probability based on the Bayes theorem to predict a particular set of features as part of a certain label [13]. It is called a Naïve Bayesian classification because it makes simplistic (naive) assumptions about how features interact [14]. To classify text documents, the Multinomial Naïve Bayes algorithm will choose the class with the highest probability [15]. Figure 2 shows the implementation of the Multinomial Naïve Bayes algorithm. Two essential variables are needed to determine the highest-class probability. The multinomial parameter required to classify test data is the prior probability. Then, the results of the prior probability will be added to the conditional probability.

TrainMultinomialNB($\mathbb{C}, \mathbb{D}$)
1 $V \leftarrow$ ExtractVocabulary($\mathbb{D}$)
2 $N \leftarrow$ CountDocs($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow$ CountDocsInClass($\mathbb{D}, c$)
5 $prior[c] \leftarrow N_c / N$
6 $text_c \leftarrow$ ConcatenateTextOfAllDocsInClass($\mathbb{D}, c$)
7 **for each** $t \in V$
8 **do** $T_{ct} \leftarrow$ CountTokensOfTerm($text_c, t$)
9 **for each** $t \in V$
10 **do** $condprob[t][c] \leftarrow \frac{T_{ct}+1}{\sum_{t'}(T_{ct'}+1)}$
11 **return** $V, prior, condprob$

ApplyMultinomialNB($\mathbb{C}, V, prior, condprob, d$)
1 $W \leftarrow$ ExtractTokensFromDoc(V, d)
2 **for each** $c \in \mathbb{C}$
3 **do** $score[c] \leftarrow \log prior[c]$
4 **for each** $t \in W$
5 **do** $score[c] \mathrel{+}= \log condprob[t][c]$
6 **return** $\arg\max_{c \in \mathbb{C}} score[c]$

Figure 2. Multinomial Naïve Bayes algorithm

2.4. **Naïve Bayes optimization by Jurafsky and Martin.** Sentiment optimization is done by adding the prefix "NOT_" after the negation word until the next punctuation mark. In this study, the prefix "NOT_" was added only one word after the negation word. This was done to avoid adding the prefix to the following sentence [14]. So, the way this algorithm works is to change the log prior equation and its log likelihood, as shown in Figure 3. The Prior or $P(c)$ represents the proportion of each sentiment class c within the training dataset. N_c refers to the count of training data instances belonging to sentiment class c and N_{c_denom} is the number of training data in a class other than c summed. Likelihood or $P(w|c)$ is assumed to be the ratio of the number of times a word occurs in a document to the total number of words in class c.

2.5. **Support Vector Machine.** The way SVM works is a binary classification technique, where this technique can interpret the definition to describe a concrete search. Therefore, the One-Versus-All (OVA) technique is used to build a multi-class classification. With this method, the classification problem of N classes is broken down into class N binary problems. In the process of training, the N-th class sample was labeled positive $(+)$ and the remaining $N-1$ samples were labeled negative $(-)$ [16].

function TRAIN NAIVE BAYES(D, C) returns log $P(c)$ and log $P(w|c)$

for each class $c \in C$ # Calculate $P(c)$ terms

N_c = number of documents from D in class c

N_{c_denom} = number of documents from D in class other than c summed

$logprior[c] \leftarrow log \frac{N_c}{N_{c_denom}}$

$bigdoc[c] \leftarrow$ append(d) for $d \in D$ with class c

$V \leftarrow$ vocabulary of D

for each word $w \in V$ # Calculate $P(w|c)$ terms

for each class $c \in C$

$count(w, c) \leftarrow$ # of occurences of w in $bigdoc[c]$

$prob(w, c) \leftarrow \frac{count(w,c)+1}{(\Sigma_{w' \in V} count(w',c))+|V|}$

for each class $c \in C$

$prob_w_denom$ = ratio from $prob(w, c)$ other than c summed

$loglikelihood[w, c] \leftarrow log \frac{prob(w,c)}{prob_w_denom}$

return $logprior, loglikelihood, V$

FIGURE 3. Optimization of Jurafsky Naïve Bayes with negation algorithm

Model training with SVM implementation solves primal optimization problems with a sub-gradient solver using stochastic gradient descent. In Figure 4, the algorithm is called the Pegasos algorithm, namely the Primal Estimated sub-Gradient Solver [17]. First, the input data will be taken randomly, and each class label will be changed to $+1$ and -1. The gradient descent algorithm may have problems determining the minimum value if the variable or η step length is not set correctly. To avoid this difficulty, Pegasos uses a step length variable: $\eta = \frac{1}{\lambda \cdot t}$ [18].

The Pegasos algorithm.

Inputs: a list of example feature vectors $\boldsymbol{X}$
a list of outputs Y
regularization parameter λ
the number of steps T

$\boldsymbol{w} = (0, \ldots, 0)$
for t in $[1, \ldots, T]$
select randomly a training instance $\boldsymbol{x}_i$, with a corresponding output label y_i
$\eta = \frac{1}{\lambda \cdot t}$
score = $\boldsymbol{w} \cdot \boldsymbol{x}_i$
if $y \cdot$score < 1
$\boldsymbol{w} = (1 - \eta \cdot \lambda) \cdot \boldsymbol{w} + (\eta \cdot y_i) \cdot \boldsymbol{x}_i$
else
$\boldsymbol{w} = (1 - \eta \cdot \lambda) \cdot \boldsymbol{w}$
the end result is $\boldsymbol{w}$

FIGURE 4. Pegasos for SVM algorithm

2.6. **Ensemble Hard Voting with modification based on comparison of classification model performances.** The Hard Voting performs roughly the same as the voting method, where the input takes the predictive label results from each model, and then counts the votes for each predictive label. As the output, calculation is performed

where the label with the largest number of votes (majority) is the result of the prediction from the Ensemble Hard Voting model. However, the drawback of Ensemble Voting is that it is only suitable for an odd number of classification models, and for solving binary classification problems where the number of classes classified is even. This is because the model will probably predict each different class label so that no majority vote can be collected [19].

In this study, we use the Ensemble Hard Voting model with modifications, where the majority voting is done if it meets the conditions required in the conventional Hard Voting model. For other conditions, calculations will be performed based on a comparison of the performance of each classification technique using Multinomial Naïve Bayes algorithm, Jurafsky Naïve Bayes algorithm, and Support Vector Machine algorithm.

2.7. **Model evaluation.** Confusion Matrix is used to summarize the performance of the classification model on the test data. The Confusion Matrix is a table for visualizing how an algorithm performs against the actual class, using two dimensions, namely the predicted class and the actual class, and each cell gives a possible set of results [14]. The Confusion Matrix not only highlights classifier errors, but also the types of errors generated [20]. Confusion Matrix can be used to measure the Accuracy, Recall or True Positive Rate, and Precision or Positive Predictive Value [20]. Here are the equations for Accuracy (1), Recall (2), and Precision (3), where TP is True Positive, TN is True Negative, FP is False Positive and FN is False Negative.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

The F-measure is one of the metrics used to combine aspects of the two using parameters β to distinguish Recall (R) and Precision (P) weights [16], which is defined as

$$F_\beta = \frac{(\beta^2 + 1)\, PR}{\beta^2 P + R} \tag{4}$$

If value $\beta > 1$, then it supports Recall, whereas if $\beta < 1$, then it supports Precision. When $\beta = 1$, where Precision and Recall are balanced, the following equation $F1$ is usually used:

$$F1 = \frac{2PR}{P + R} \tag{5}$$

3. **Result and Discussion.** The testing process is carried out using a Confusion Matrix to see the performance of each class, namely Accuracy, Precision, Recall and F1-score.

3.1. **Result of sentiment analysis with Multinomial Naïve Bayes model.** Table 1 shows the Multinomial Naïve Bayes performance. The F1-score for negative and positive classes reached the same value, namely 86%, while for the neutral class reached 35%. This is because the amount of data in each class was not evenly distributed, especially in the neutral class which was relatively small in number. The model's accuracy rate achieved 80.3%.

3.2. **Result of sentiment analysis with Jurafsky Naïve Bayes model.** The optimization of the Naïve Bayes model by Jurafsky and Martin suggests adding the prefix "NOT_" to each word after the negation sign as used in this study, namely 'tidak', 'nggak', and 'g' (in Bahasa). The addition of prefixes is done during preprocessing and tokenization. Table 2 shows Jurafsky Naïve Bayes performance. The F1-score achieved

Table 1. Multinomial Naïve Bayes performance

	Multinomial Naïve Bayes			
	Negative	Neutral	Positive	Macro Avg
Precision	0.79	0.68	0.83	0.77
Recall	0.93	0.24	0.88	0.68
F1-score	0.86	0.35	0.86	0.69
Accuracy	0.88	0.24	0.88	0.803

Table 2. Jurafsky Naïve Bayes performance

	Jurafsky Naïve Bayes			
	Negative	Neutral	Positive	Macro Avg
Precision	0.92	0.21	0.87	0.67
Recall	0.79	0.73	0.81	0.78
F1-score	0.85	0.33	0.84	0.67
Accuracy	0.92	0.21	0.87	0.79

by the negative class is 85%, the neutral class is 33%, and the positive class is 84%. The accuracy of the model reached 79%.

3.3. **Result of sentiment analysis with Support Vector Machine model.** Table 3 shows SVM performance. The F1-score level reached 84% for the negative and positive classes while 35% for the neutral class. The accuracy achieved by the model is the same as the Jurafsky Naïve Bayes model, reaching 79%.

Table 3. SVM performance

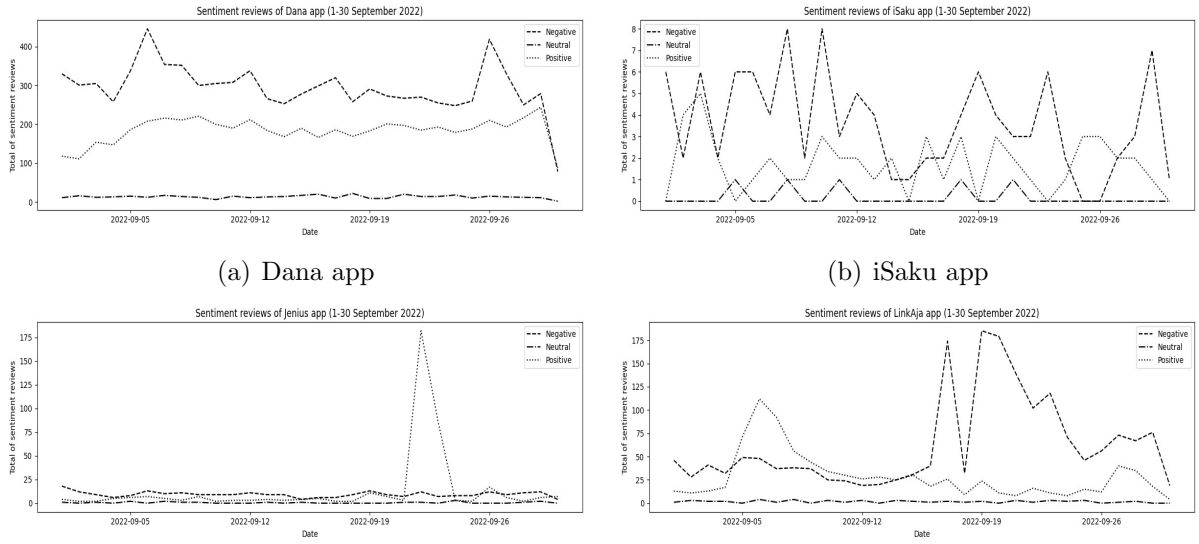
	SVM			
	Negative	Neutral	Positive	Macro Avg
Precision	0.75	0.82	0.86	0.81
Recall	0.96	0.22	0.82	0.67
F1-score	0.84	0.35	0.84	0.68
Accuracy	0.94	0.22	0.94	0.79

3.4. **Result of Ensemble Hard Voting with modification.** Table 4 shows Ensemble Hard Voting with modification performance. The F1-score for the negative class is 85%, the neutral class is 25%, and the positive class is 85%. The accuracy achieved by this model is 80.1%.

Table 4. Ensemble Hard Voting with modification performance

	Ensemble Hard Voting			
	Negative	Neutral	Positive	Macro Avg
Precision	0.78	0.74	0.84	0.79
Recall	0.93	0.23	0.87	0.68
F1-score	0.85	0.25	0.85	0.69
Accuracy	0.93	0.23	0.87	0.801

3.5. **Sentiment predictions for E-Wallet recommendations.** For E-Wallet recommendation, it is done by scraping user reviews in a span of one month from the Google Play Store on each E-Wallet. It aims to see the trend of users and sentiment reviews in range of one month. The Multinomial Naïve Bayes model is used to classify sentiments. Figure 5 shows the results of sentiment analyisis on four E-Wallets based on user reviews.



(a) Dana app (b) iSaku app

(c) Jenius app (d) LinkAja app

FIGURE 5. Sentiment reviews

4. **Conclusions.** In this study, three machine learning classification techniques were developed using Multinomial Naïve Bayes algorithm, Jurafsky Naïve Bayes algorithm, and Support Vector Machines algorithm. These techniques were utilized to classify the sentiment of user reviews. The Ensemble Hard Voting model with modifications was also built based on the predicted results of the three classification techniques. The results indicate that Multinomial Naïve Bayes algorithm obtained the highest accuracy rate at 80.3%, followed by Ensemble Hard Voting with an accuracy of 80.1%, Jurafsky Naïve Bayes with 79%, and Support Vector Machine with 79%. In the future, the performance of the text classification model can be improved in the data preprocessing section by emoticon conversion, augmentation, N-gram generation, and other preprocessing methods according to research needs. Additionally, ensemble soft voting models involving a number of machine learning algorithms can also be considered.

Acknowledgment. This work is supported by Engineering Faculty, Universitas Hasanuddin, via Labo-based Education Research Grant 2022, and Software Engineering and Information System Laboratory.

REFERENCES

[1] E. Wijaya and Y. Octafilia, Importance of consumer satisfaction to improve consumer trust LinkAja digital wallet, *Journal of Research in Business, Economics, and Education*, vol.3, no.6, pp.66-77, 2021.

[2] D. A. Kristiyanti, D. A. Putri, E. Indrayuni, A. Nurhadi and A. H. Umam, E-wallet sentiment analysis using Naïve Bayes and support vector machine algorithm, *Journal of Physics: Conference Series*, vol.1641, no.1, pp.1-7, DOI: 10.1088/1742-6596/1641/1/012079, 2020.

[3] D. Dwidienawati, D. Tjahjana, S. B. Abdinagoro, D. Gandasari and Munawaroh, Customer review or influencer endorsement: Which one influences purchase intention more?, *Heliyon*, vol.6, no.11, DOI: 10.1016/j.heliyon.2020.e05543, 2020.

[4] E. Ismagilova, E. L. Slade, N. P. Rana et al., The effect of electronic word of mouth communications on intention to buy: A meta-analysis, *Information Systems Frontiers*, vol.22, pp.1203-1226, DOI: 10.1007/s10796-019-09924-y, 2020.

[5] G. Khanvilkar and D. Vora, Sentiment analysis for product recommendation using random forest, *International Journal of Engineering and Technology*, vol.7, no.3, pp.87-89, DOI: 10.14419/ijet.v7i3.3.14492, 2018.

[6] D. M. E. D. M. Hussein, A survey on sentiment analysis challenges, *Journal of King Saud University – Engineering Sciences*, vol.30, no.4, pp.330-338, DOI: 10.1016/j.jksues.2016.04.002, 2018.

[7] G. Khanvilkar and D. Vora, Product recommendation using sentiment analysis of reviews: A random forest approach, *International Journal of Engineering and Advanced Technology*, vol.8, no.2, pp.146-152, 2019.

[8] A. Nair, C. Paralkar, J. Pandya, Y. Chopra and D. Krishnan, Comparative review on sentiment analysis-based recommendation system, *2021 6th International Conference for Convergence in Technology (I2CT)*, Maharashtra, India, pp.2-7, DOI: 10.1109/I2CT51068.2021.9418222, 2021.

[9] M. Ahmad, M. F. Octaviansyah, A. Kardiana and K. F. Prasetyo, Sentiment analysis system of Indonesian tweets using lexicon and Naïve Bayes approach, *2019 4th International Conference on Informatics and Computing (ICIC)*, Semarang, Indonesia, pp.1-5, 2019.

[10] W. A. Deviani, K. Kusumahadi and E. Nurhazizah, Service quality for digital wallet in Indonesia using sentiment analysis and topic modelling, *International Journal of Technology Management*, vol.4, no.1, pp.46-58, DOI: 10.55057/ijbtm.2022.4.1.6, 2022.

[11] M. V. K. Kiran, R. E. Vinodhini, R. Archanaa and K. Vimalkumar, User specific product recommendation and rating system by performing sentiment analysis on product reviews, *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pp.1-5, DOI: 10.1109/ICACCS.2017.8014640, 2017.

[12] A. A. Ilham, A. Bustamin and E. Wahyudiarto, Customer satisfaction assessment system on transactions e-commerce product purchases using sentiment analysis, *International Journal on Advanced Science, Engineering and Information Technology*, vol.13, no.3, pp.1041-1051, DOI: 10.18517/ijaseit.13.3.18273, 2023.

[13] M. Wankhade, A. C. S. Rao and C. Kulkarni, A survey on sentiment analysis methods, applications, and challenges, *Artificial Intelligence Review*, vol.55, pp.5731-5780, 2022.

[14] D. Jurafsky and J. Martin, Naïve Bayes and sentiment classification, *Speech and Language Processing*, 1024, https://web.stanford.edu/∼jurafsky/slp3/6.pdf, 2019.

[15] C. D. Manning, P. Raghavan and H. Schütze, Text classification and Naïve Bayes, *Introduction to Information Retrieval*, no.c, pp.234-265, DOI: 10.1017/cbo9780511809071.014, 2012.

[16] D. Żurek and M. Pietroń, Training with reduced precision of a support vector machine model for text classification, *arXiv.org*, arXiv: 2007.08657, 2020.

[17] S. Shalev-Shwartz, Y. Singer, N. Srebro and A. Cotter, Pegasos: Primal estimated sub-GrAdient SOlver for SVM, *Mathematical Programming*, vol.127, pp.3-30, 2011.

[18] R. Johansson, *Clarification of the Pseudocode in the Pegasos Paper*, https://svn.spraakdata.gu.se/repos/richard/pub/ml2015_web/a2_clarification.pdf, 2016.

[19] N. Peppes, E. Daskalakis, T. Alexakis, E. Adamopoulou and K. Demestichas, Performance of machine learning-based multi-model voting ensemble methods for network threat detection in Agriculture 4.0, *Sensors*, vol.21, no.22, DOI: 10.3390/s21227475, 2021.

[20] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis and N. Doulamis, Multiclass confusion matrix reduction method and its application on net promoter score classification problem, *Technologies*, vol.9, no.4, 81, DOI: 10.3390/technologies9040081, 2021.