Multitek Indonesia: Jurnal Ilmiah
Volume: 18 No. 2, Desember 2024, Hal. 105 - 113
ISSN: 1907-6223 (print)
ISSN: 2579-3497 (Online)
http://journal.umpo.ac.id/index.php/multitek

OPTIMIZING SENTIMENT ANALYSIS FOR USABILITY TESTING: ENHANCING SVM ACCURACY THROUGH KERNEL SELECTION AND TUNING METHODS

Hasan Basri[1], Irpan Kusyadi[2], Mayang Anglingsari Putri[3]
[1,2,3] Information System, Faculty of Science and Technology, Universitas Terbuka
Jl. Pd. Cabe Raya, Kota Tangerang Selatan, Indonesia
Email : hasan.basri@ecampus.ut.ac.id[1], irpan.kusyadi@ecampus.ut.ac.id[2], mayang.anglingsari@ecampus.ut.ac.id[3]

Dikimkan: 20 Juni 2024 **Direvisi: 01 September 2024** **Diterima: 04 Oktober 2024**

Abstract

With over 2.4 million apps on the Google Play Store by 2023, app developers face increasing demands to ensure high usability quality to remain competitive. Traditional usability testing methods, including heuristic evaluations and user questionnaires, are often limited by high costs, time constraints, and lack of real-world context. Sentiment analysis presents an alternative approach, leveraging user reviews as a resource for usability insights. This research applies Support Vector Machine (SVM) for sentiment analysis and usability testing on Google Play Store reviews, focusing on five usability criteria. Data collection yielded 2,000 reviews from a banking app, with two annotators conducting multi-label labeling for both sentiment and usability criteria. Through a series of experiments, the Linear Kernel in SVM demonstrated the highest performance, achieving 70.50% accuracy, an F1 Score of 0.8618, and a Hamming Loss of 0.0783. Grid Search was employed to optimize the C parameter for the linear kernel, revealing an optimal C value of 0.01, which resulted in an improved accuracy of 75.20%, F1 Score of 0.8775, and Hamming Loss of 0.0686. Experiments with values above or below 0.01 showed decreased accuracy, underscoring the importance of a balanced C value to enhance model generalization and avoid overfitting. These findings suggest that sentiment analysis via SVM can effectively capture usability feedback from user reviews, providing a scalable, data-driven solution for app usability assessment. This study is part of the Machine Learning for Software Engineering (ML4SE) domain, where machine learning techniques are applied to enhance software engineering practices, specifically in optimizing usability assessment through automated analysis of user feedback. Given the promising results of combining usability testing and sentiment analysis, future research should explore the use of deep learning approaches, such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), or Transformer-based models, to further enhance classification accuracy. Additionally, improvements in preprocessing techniques, particularly in handling slang words and informal language in user reviews, should be considered to improve data quality and ensure more accurate usability insights.

Keyword: Usability Testing, Sentiment Analysis, Support Vector Machine, ML4SE.

INTRODUCTION

With more than 2.4 million apps on the Google Play Store by 2023[1], app developers are required to ensure good usability quality in order to compete. Apps that are easy to use and efficient tend to have higher user retention rates[2]. Usability Testing is usually done through conventional methods such as usability testing (questionnaires), heuristic evaluation, and expert reviews [3] [4].Although effective, these methods have limitations, especially in terms of cost, time, and resources. Conventional testing methods require active participation from users in controlled scenarios, which sometimes do not reflect real-world application usage [5].

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

In this context, sentiment analysis offers a more efficient and adaptive solution. Sentiment analysis is a technique used to extract opinions and emotions from text, such as user reviews [6]. In the Google Play Store, users often leave reviews containing their opinions about the apps they use. These reviews, if properly analyzed, can provide valuable insights into the usability aspects of the app. Sentiment analysis is part of Natural Language Processing (NLP), a field that has been widely applied in various fields. One example is in Software Engineering, where NLP is used to analyze user stories. By using NLP, developers can automatically extract user needs and expectations from narrative text, which can help in designing software that is more in line with user needs [7] [8].

Applying sentiment analysis to user reviews offers several advantages. The technique enables automated data collection at scale, as well as providing real-world context since reviews are created by users without the influence of controlled scenarios. In addition, sentiment analysis can identify patterns and trends in user perceptions, helping developers to understand their app's strengths and weaknesses more quickly. This study is part of the broader field of Machine Learning for Software Engineering (ML4SE), where machine learning techniques, including Natural Language Processing, are applied to enhance and optimize various aspects of software development. By combining usability testing with sentiment analysis, this research leverages ML4SE to enable a data-driven, scalable, and adaptive approach to understanding user feedback. This integration not only accelerates the feedback loop for developers but also aligns usability insights more closely with real-world user expectations, making it a valuable contribution to the intersection of machine learning and software engineering practices.

In this study, Support Vector Machine (SVM) is employed as a classification algorithm to analyze sentiment and usability criteria from user reviews. SVM is widely recognized for its effectiveness in handling high-dimensional data and text classification tasks, making it a suitable choice for sentiment analysis [9]. By using SVM, this research ensures robust classification performance, allowing the model to differentiate between usability aspects such as Learnability, Efficiency, Memorability, Error Tolerance, and Satisfaction, as well as to identify the sentiment (Positive or Negative) expressed in user reviews. The incorporation of SVM further enhances the precision and reliability of sentiment analysis in usability testing.

This research aims to evaluate the extent to which sentiment analysis is reliable in measuring usability aspects of mobile applications, as well as identify its challenges and limitations. Through this approach, this research is expected to provide a more adaptive and data-driven solution in Usability Testing. By integrating sentiment analysis, developers can respond to user needs faster and more precisely. This research also contributes to expanding the understanding of the application of sentiment analysis beyond its traditional context, as well as assisting developers in optimizing applications based on user review data.

METHODS

This research consists of several key stages depicted in a flowchart, covering the entire process from literature study to reporting the final results. Each stage is designed to provide optimal results in sentiment analysis and usability testing based on user reviews on Google Playstore.

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

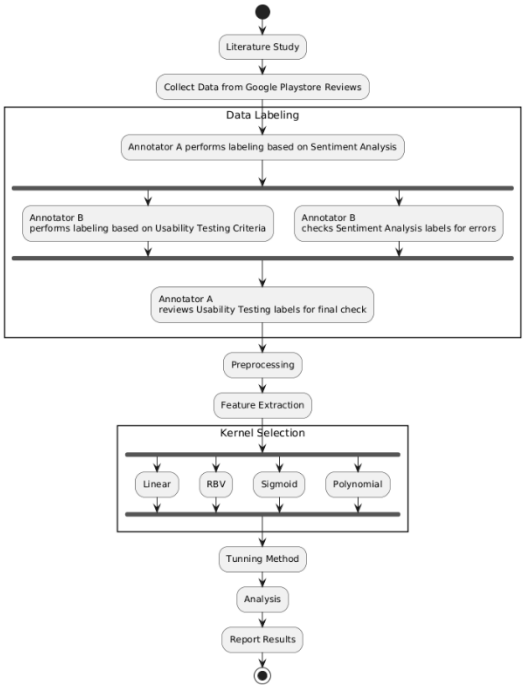

Figure 1. Research Procedure

The research began with a literature study to gain an in-depth understanding of existing methods in usability testing and sentiment analysis. The literature reviewed included text feature extraction techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) and the Support Vector Machine (SVM) classification model [10]. This literature study also aims to understand previous approaches in the application of SVM algorithms and data labeling strategies in the context of sentiment and usability analysis. After the literature study, data was collected from Google Play Store user reviews. The reviews collected are quantitative reviews that are relevant to the usability aspects of the application. This data collection forms the basis for further labeling and analysis.

The labeling process is done in two stages involving two annotators, Annotator 'A' and Annotator 'B', with a multilabel approach that allows each review to receive more than one label from different categories, namely Sentiment Analysis and Usability Testing Criteria. Annotator 'B' checks the results of Annotator 'A' and vice versa. This method was chosen because it is considered more efficient in terms of budget and time, allowing the labeling process to be done quickly without reducing the quality of the results. After the labeling is complete, the data then goes through a preprocessing process to clean and prepare the data to be ready for use in modeling. The preprocessing stages include converting numbers to words ("1" to "one"), removal of empty data, normalization, tokenization, and removal of stop words. These steps aim to ensure that the data used in the analysis process is of high quality and clean of irrelevant elements.

Feature extraction, a numerical representation of the text data is obtained through the TF-IDF (Term Frequency-Inverse Document Frequency) technique. TF-IDF is used to calculate the weight of each word in the text document, resulting in features that will be used as input for the SVM classification model. The selection of the kernel for SVM is an important step in this research. Several kernels were tested, namely Linear Kernel, Radial Basis Function (RBF) Kernel, Sigmoid Kernel and Polynomial Kernel. Each kernel is tested to evaluate its performance in a multi-label classification task on text data. This process aims to find the kernel that best matches the characteristics of the data.

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

After the kernel selection is done, the model is optimized using the Grid Search method to find the best combination of C and gamma parameters [11]. This tuning process aims to improve the performance of the SVM model, especially for non-linear kernels such as RBF and Polynomial, which require more careful parameter optimization. This tuning process is validated with k-fold cross-validation to ensure the resulting model is not overfitting.

After the SVM model is trained with the best parameters, performance evaluation is performed using various metrics, including accuracy, F1-score, precision, recall, and hamming loss [12]. The evaluation results of each kernel are compared to determine the most effective kernel in classifying the data. The confusion matrix is also analyzed to get an idea of the misclassification that occurs in each class.

RESULTS AND DISCUSSION

1. Data Collections

This study uses review data from one banking company's application available on Google Playstore, with a total of around 1.5 million reviews. This huge number of reviews provides significant potential to be used in research related to sentiment analysis and usability testing. The data reflects users' opinions about their experience in using banking apps, in terms of satisfaction, efficiency, and errors that may occur. On August 24, 2024, 2,000 reviews were taken from the total dataset for further analysis based on usability testing criteria. The labeling process was done by following 5 usability testing criteria [13] [14]. The 5 main aspects used in this research are Efficiency, User Satisfaction, Learnability, Memorability and Error Rate.

2. Data Labeling

This research uses a multi-label approach in labeling review data, with two annotators playing a role in this process. Annotator ‘A’ was responsible for labeling the reviews based on sentiment analysis with two categories, Positive and Negative. Once the sentiment labels were set, Annotator ‘B’ labeled the reviews based on usability testing criteria, which included five main aspects: Learnability, Efficiency, Memorability, Error Rate , and User Satisfaction. In addition, Annotator ‘B’ also checks the sentiment labels that have been assigned by Annotator ‘A’ to ensure there are no labeling errors. The final stage in this labeling process is that Annotator ‘A’ again checks the labels that have been given for Usability Testing Criteria by Annotator ‘B’ to ensure consistency and accuracy.

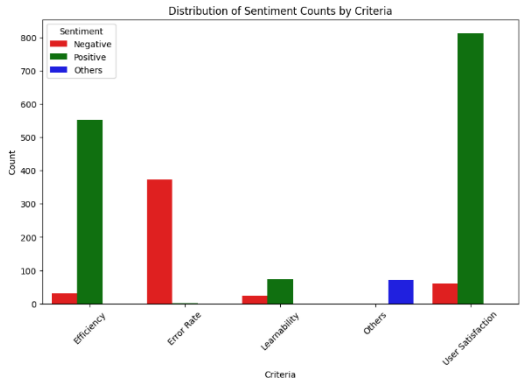

Figure 2. Sentiment distribution based on usability testing criteria

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

Based on the data obtained from the results of labeling reviews, it can be concluded that the distribution of reviews on usability testing criteria and sentiment analysis is that most reviews commenting the efficiency criteria are also dominated by positive reviews (552 reviews). While the Error Rate criterion related to errors in using the application has a negative sentiment (373 reviews), which indicates that users experience many technical problems or errors. The Learnability criterion shows that the app is quite easy to learn, with 74 positive reviews. Furthermore, User Satisfaction, this criterion dominates with 813 positive reviews, reflecting a high level of satisfaction with the app.
However, in the labeling results, the Memorability criterion was not found. This may indicate that users did not explicitly comment on the ease of remembering how to use the app, or that this criterion was not relevant in the context of the reviews. This could be an important finding in understanding users' primary focus on app usability. Another 72 reviews provided reviews outside the context of the app being used.

3. Data Preprocessing (Data Cleaning and Filtering)

After analyzing the data distribution, some combinations of criteria and sentiment with a small number of reviews will be eliminated to maintain the relevance of the data in the study. The eliminated combinations are as follows:

Table 1. Eliminated Data Combinations in Preprocessing Stage

Combination	
Error Rate + Positive	(1 review)
Efficiency + Negative	(30 reviews)
Learnability + Negative	(24 reviews)

These combinations were removed due to the small amount of data, which may affect the quality of analysis and generalization of the model. Thus, the remaining data includes a more significant distribution of reviews to support usability analysis and sentiment analysis more effectively.

In the preprocessing stage, a series of steps were taken to clean and prepare the user review data to make it consistent and ready for analysis. This process starts with converting numbers into words, for example the number '1' is converted into 'satu'. Next, irrelevant non-ASCII characters are removed to ensure the data is more organized and easy to process. In addition, lemmatization is performed, which is changing each word into its basic form, for example the word 'membeli' is changed to 'beli'. Lemmatization is important to reduce the variety of words that actually have the same meaning, so that the model can more easily recognize patterns in the text data. The last step is additional cleaning, which includes removing punctuation, redundant spaces, and other irrelevant characters. This process aims to ensure that the data used in the analysis is completely clean and ready for further processing.

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

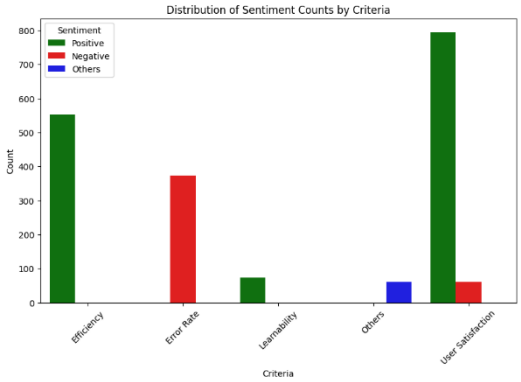

Figure 3. Sentiment distribution based on usability testing criteria after preprocessing

4. Kernel Selection

The next step in this research is kernel selection for the Support Vector Machine (SVM) model. Kernel selection is very important because the kernel determines how the data is mapped to a higher dimensional space, thus affecting the performance of the model in the classification task. The kernel selection is done based on initial experimental results with default parameters to evaluate the basic performance of each kernel.

Table 2. Performance Comparison of Different SVM Kernels

Kernel	Accuracy	Hamming Loss	F1 Score	Precision	Recall
Linear	70.50 %	0.0783	0.8618	0.8573	0.8664
RBF	64.23 %	0.1037	0.8075	0.8474	0.7712
Sigmoid	68.41 %	0.0783	0.8564	0.8867	0.8200
Poly	42.30 %	0.2167	0.6145	0.6165	0.6124

Linear kernel gives the best overall result, with the highest accuracy value (70.50%), the highest F1 Score (0.8618), and the lowest Hamming Loss (0.0783). This shows that the linear kernel is most effective in classifying review data, especially in maximizing the balance between precision and recall. This model successfully minimizes the error and provides the optimal separation margin. The RBF kernel also performed quite well, especially in terms of precision (0.8474). However, the lower recall (0.7712) shows that this kernel has difficulty in identifying reviews thoroughly, resulting in more errors than the linear kernel. The Sigmoid kernel has the highest precision (0.8867), which shows a good ability to correctly predict positive labels. However, the lower recall compared to the linear kernel (0.8200) indicates that this model is less comprehensive in capturing all reviews correctly, as seen from the lower accuracy (68.41%). Meanwhile, the Polynomial Kernel showed the lowest performance on all metrics tested. With the lowest accuracy (42.30%), low F1 Score (0.6145), and highest Hamming Loss (0.2167), this kernel is not suitable for the data used in this study. This model has difficulty in classifying reviews accurately and consistently.

Based on the comparison results, the Linear Kernel is the best for the data used in this study. It provides the highest accuracy, highest F1 Score, and lowest Hamming Loss, indicating that this model is able to optimally separate the data and produce the best performance compared to other kernels. The RBF and Sigmoid kernels also showed competitive performance, especially in terms of precision, but still lost out in terms of recall and overall accuracy.

5. Tuning Method

In this research, a tuning method using Grid Search is performed to find the optimal C value in the SVM model with a linear kernel. Based on the results of the kernel selection conducted

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

previously, the linear kernel was chosen because it provides the best performance in classifying text-based review data with TF-IDF representation. This kernel has been proven to be able to handle high-dimensional data effectively, so the tuning process is only focused on optimizing the hyperparameter C. The C value is an important parameter in SVM, which controls the trade-off between a large margin and proper classification of the training data. In Grid Search, several C values are tested to find the combination that gives the best accuracy. This process is done with cross-validation to ensure that the model is not overfitting and can provide good generalization to the test data.

The parameter tuning process is performed with Grid Search to find the most optimal C value in the SVM model with a linear kernel. In the initial stage, the C values tested were 0.1, 1, 10, 100, and 1000, using 3-fold cross-validation to ensure the results were not overfitting. From these tuning results, it was found that the best parameter was C = 0.1, which gave the best performance on the data used.

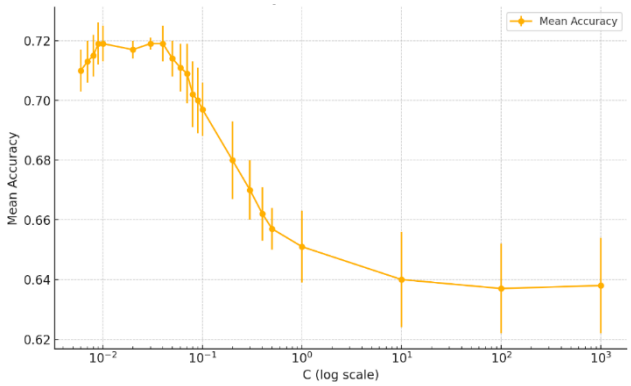

Figure 4. Mean Accuracy VS C with Standard Deviation

Furthermore, the tuning process was extended by testing C values between 0.1 to 0.5 ([0.1, 0.2, 0.3, 0.4, 0.5]), but still showed that C = 0.1 was the most optimal. To further check the accuracy of the optimal parameters, Grid Search was performed again with a smaller range of C ([0.05, 0.06, 0.07, 0.08, 0.09, 0.1]), and the optimal value changed to C = 0.05. In the final stage, the tuning was refined by testing lower values, namely between 0.01 to 0.05 ([0.01, 0.02, 0.03, 0.04, 0.05]). From these results, it was found that the most optimal C value was 0.01 because when another experiment was conducted with smaller C ([0.006, 0.007, 0.008, 0.009, 0.01]), the optimal C remained 0.01. Evaluation of the model with this C value resulted in the following performance:

Table 3. The resulting performance

No	Performance	Value
1	Accuracy SVM (multi-label):	75.20 %
2	Hamming Loss (which shows a minimal number of misclassifications)	0.0686
3	F1 Score (shows a good balance between precision and recall)	0.8775
4	Precision	0.8834
5	Recall	0.8717

These results show that C = 0.01 is the most optimal value for the multi-label classification task in this study. This small value of C allows for a larger margin on the model, which helps prevent overfitting and improves the generalization of the model to the test data. With these parameters, the SVM model managed to perform very well in sentiment classification and usability criteria testing.

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

CONCLUSION

This research uses Support Vector Machine (SVM) to perform sentiment analysis and usability testing on mobile application reviews from Google Play store. After going through the process of data collection, labeling, preprocessing, kernel selection, and parameter tuning. The data taken was 2,000 reviews from a total of 1.5 million reviews reflecting various user experiences, focusing on 6 aspects of usability testing criteria. However, the Memorability criterion was not found in the analyzed reviews, indicating that users did not provide explicit comments regarding the memorability of the application. The labeling process was multi-labeled by two annotators, with Annotator 'A' labeling based on sentiment analysis (Positive and Negative), and Annotator 'B' based on usability testing criteria. This allows reviews to be comprehensively analyzed from two points of view, namely sentiment and usability aspects. After conducting experiments with several kernels (Linear, RBF, Sigmoid, Polynomial), the results show that the Linear Kernel provides the best performance with 70.50% accuracy, F1 Score 0.8618, and Hamming Loss 0.0783. The linear kernel proved to be the most effective in classifying review data, especially in maximizing the balance between precision and recall. Through Grid Search to find the optimal C value, it is found that the best value is C = 0.01, resulting in 75.20% accuracy, F1 Score 0.8775, and Hamming Loss 0.0686 on the test data. Experiments with C values greater than 0.01 show that the average accuracy decreases as the C value increases, as seen in the mean accuracy result of 71.4% at C = 0.05 and continue to decrease to 69.7% at C = 0.1. This suggests that larger C values make the model too focused on the training data, potentially leading to overfitting which reduces performance on the test data. Conversely, experiments with C values smaller than 0.01 also resulted in a decrease in accuracy. A C value that is too small causes the model to lose the capacity to capture relevant patterns, thus reducing classification ability. Thus, an optimal C value of 0.01 provides the best balance between accuracy and model generalization, without unduly restricting or expanding the margin of separation in classification. Since the combination of usability testing and sentiment analysis in this study has yielded promising results, future research should explore deep learning approaches, such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), or Transformer-based models, to enhance classification accuracy. Deep learning models can better capture complex linguistic patterns and context, potentially leading to higher performance in sentiment-based usability analysis. Additionally, preprocessing improvements should be considered, particularly in handling slang words and informal language commonly found in user reviews. Developing a text normalization module to convert slang and non-standard words into standard forms would further enhance model accuracy. By integrating these improvements, future studies can refine usability testing methodologies, providing more robust and scalable solutions for mobile app evaluation.

REFERENCES

[1] L. Ceci, "Number of available applications in the Google Play Store from March 2017 to June 2024," *Statista*, 2024. .

[2] Y. Wang, J. Wang, H. Zhang, X. Ming, L. Shi, and Q. Wang, "Where is your app frustrating users?," in *Proceedings of the 44th International Conference on Software Engineering*, May 2022, pp. 2427–2439, doi: 10.1145/3510003.3510189.

[3] Z. Galavi, S. Norouzi, and R. Khajouei, "Heuristics used for evaluating the usability of mobile health applications: A systematic literature review," *Digit. Heal.*, vol. 10, Jan. 2024, doi: 10.1177/20552076241253539.

[4] H. Zulzalil, H. Rahmat, A. A. A. Ghani, and A. Kamaruddin, "Expert Review on Usefulness of an Integrated Checklist-based Mobile Usability Evaluation Framework," *J. Comput. Sci. Res.*,

ISSN: 1907- 6223 (Print)
ISSN: 579 – 3497 (Online)

vol. 5, no. 3, pp. 57–73, Aug. 2023, doi: 10.30564/jcsr.v5i3.5816.

[5] L. Panizo, A. Díaz, and B. García, "Model-based testing of apps in real network scenarios," *Int. J. Softw. Tools Technol. Transf.*, vol. 22, no. 2, pp. 105–114, Apr. 2020, doi: 10.1007/s10009-019-00518-2.

[6] H. Singh and D. Srivastava, "Sentiment Analysis: Quantitative Evaluation of Machine Learning Algorithms," in *2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Jan. 2023, pp. 946–951, doi: 10.1109/ICSSIT55814.2023.10061130.

[7] S. Chakraborty, "Sentiment Analysis in the Perspective of Natural Language Processing," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 11, no. 11, pp. 2235–2241, Nov. 2023, doi: 10.22214/ijraset.2023.56925.

[8] S.-C. Necula, F. Dumitriu, and V. Greavu-Șerban, "A Systematic Literature Review on Using Natural Language Processing in Software Requirements Engineering," *Electronics*, vol. 13, no. 11, p. 2055, May 2024, doi: 10.3390/electronics13112055.

[9] P. Vijayaragavan, R. Ponnusamy, and M. Aramudhan, "An optimal support vector machine based classification model for sentimental analysis of online product reviews," *Futur. Gener. Comput. Syst.*, vol. 111, pp. 234–240, Oct. 2020, doi: 10.1016/j.future.2020.04.046.

[10] Y. Xu, Z. Li, and X. Wang, "Specific Search Engine Identification Model Based on Improved TF-IDF and SVM," in *Proceedings of the 2022 10th International Conference on Information Technology: IoT and Smart City*, Dec. 2022, pp. 122–126, doi: 10.1145/3582197.3582217.

[11] C. Dewi, F. A. Indriawan, and H. J. Christanto, "Spam classification problems using support vector machine and grid search," *Int. J. Appl. Sci. Eng.*, vol. 20, no. 4, pp. 1–10, 2023, doi: 10.6703/IJASE.202312_20(4).006.

[12] V. N. Jenipher and S. Radhika, "SVM kernel Methods with Data Normalization for Lung Cancer Survivability Prediction Application," in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, Feb. 2021, pp. 1294–1299, doi: 10.1109/ICICV50876.2021.9388543.

[13] P. Weichbroth, "Usability of Mobile Applications: A Systematic Literature Study," *IEEE Access*, vol. 8, pp. 55563–55577, 2020, doi: 10.1109/ACCESS.2020.2981892.

[14] S. Shareef and M. N. A. Khan, "Evaluation of Usability Dimensions of Smartphone Applications," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 9, 2019, doi: 10.14569/IJACSA.2019.0100956.