JURNAL INFORMATIKA, Vol. 11 No. 2 October 2025, Page 90-96
ISSN (Print) 2355-6579 | ISSN (Online) 2528-2247
DOI: https://doi.org/10.31294/inf.v12i2.24639

Evaluation of Machine Learning Algorithms for Classifying User Perceptions of a Child Health Monitoring Application

Eka Rahmawati[1*], Adi Wibowo[2], Budi Warsito[3]

[1,2,3]Doctoral Program of Information System, Universitas Diponegoro
Jl. Prof. Soedarto No.13, Tembalang, Kec. Tembalang, Kota Semarang, Indonesia
[1]Information Systems, Faculty of Engineering and Informatics, Universitas Bina Sarana Informatika
Jl. Kramat Raya No 98, Senen, Jakarta Pusat, Indonesia
[2]Department of Informatics, Faculty of Science and Mathematics, Universitas Diponegoro
Jl. Prof. Soedarto No.13, Tembalang, Kec. Tembalang, Kota Semarang, Indonesia
[3]Department of Statistics, Faculty of Science and Mathematics, Universitas Diponegoro
Jl. Prof. Soedarto No.13, Tembalang, Kec. Tembalang, Kota Semarang, Indonesia

Correspondence e-mail: eka.eat@bsi.ac.id

Submission:	Revision:	Acceptance:	Available Online:
10-12-2024	23-06-2025	07-07-2025	02-10-2025

Abstract

Supporting children’s early development requires consistent attention, ensuring their growth aligns with health standards. PrimaKu is one of the mobile applications developed by the Indonesian Pediatric Society. That application was created to assist parents in recording developmental milestones, monitoring immunization schedules, and accessing practical health information. This study investigates user perceptions of the application by analyzing publicly available reviews and ratings from the Google Play Store. Four supervised machine learning algorithms were applied to classify the sentiment expressed in the reviews: Support Vector Machine (SVM), Random Forest, Decision Tree, and Naive Bayes. Among the models tested, SVM achieved the highest classification accuracy (81%), followed by Random Forest (77%), Decision Tree (74%), and Naive Bayes (73%). Precision, recall, and F1-score were also used to evaluate the performance of each model. The results highlight the relevance of machine learning in capturing and interpreting user sentiment toward digital health tools. Further exploration of deep learning architectures is encouraged to enhance classification accuracy and understanding of features.

Keywords : machine learning; user perception; chield health monitoring

1. Introduction

Child growth and development are critical aspects that every parent needs to monitor attentively. Proper monitoring ensures that children achieve their developmental milestones, fostering a generation of healthy and high-quality individuals who will contribute positively to society (Schickedanz, 2020). In recent years, technological advancements have introduced innovative solutions, such as mobile applications, to assist parents in monitoring their children's growth. One such application, PrimaKu, developed by the Indonesian Pediatric Society provides parents with tools to monitor their children’s growth and development, schedule immunizations, and access health guidelines.

PrimaKu is designed to address the need for systematic and comprehensive child growth monitoring. The application allows parents to record key growth parameters such as weight, height, and head circumference while comparing them against standard growth charts provided by the World Health Organization (WHO). Additionally, it provides timely reminders for immunization schedules and delivers health tips to enhance parenting practices. Despite its practical benefits, the success of an application like PrimaKu relies heavily on its acceptance and usability among its target users. Understanding user feedback and perceptions is crucial for improving the application's functionality and ensuring widespread adoption.

Machine learning has emerged as a powerful tool for evaluating user perceptions. Machine learning algorithms can classify user feedback efficiently, providing insights into the strengths and weaknesses of the application(Kaluarachchi et al., 2021). Several machine learning algorithm has been used to evaluate user prediction. Random forest used to predict factors affecting the perceived usability of a covid-19 contact tracing mobile application with

92% accuracy (Ong et al., 2022). Support Vector Machine (SVM) and Naive Bayes were also used to evaluate the research about e-commerce online review for detecting influencing factors users' perception that has a high accuracy value (Arsad et al., 2021). Decision Tree is also considered an effective algorithm(Ren et al., 2020). The other research to explore consumers' purchasing decision pathway (Carrillo et al., 2023).

A method of ensemble learning that integrates numerous decision trees to enhance the overall efficacy of classification or regression tasks is referred to as the Random Forest algorithm (Anwar et al., 2024). Every decision tree inside the forest is constructed utilizing a random subset of the training data and a random selection of features, hence maintaining variation among the trees. During prediction, the random forest is performed because the output of all trees is averaged (for regression) or voted upon (for classification), allowing for very strong and reliable predictions to be made(Salman et al., 2024). This approach reduces the risk of overfitting, which is a common issue with individual decision trees, and enhances the model's ability to generalize to unseen data.

One of the key strengths of Random Forest is its capability to handle high-dimensional data, including datasets with both categorical and numerical features (Heritage Samuel, 2024). It is less sensitive to hyperparameter tuning and can provide insights into feature importance, which is useful for understanding the relative contributions of variables in the dataset.

Despite its robustness and effectiveness, Random Forest can become computationally expensive when dealing with very large datasets or a high number of trees. However, its performance and versatility make it a popular choice for a wide range of machine learning problems.

Naive Bayes is called "naive" because it assumes that all features are independent of one another, an assumption that often does not hold in real-world datasets. Despite this simplification, Naive Bayes performs remarkably well in many applications, especially for text classification tasks like spam detection, sentiment analysis, and document categorization. Its computational efficiency and simplicity make it a popular choice for large-scale datasets (Peretz et al., 2024).

A key advantage of Naive Bayes is its ability to handle imbalanced datasets and perform well even with limited training data. However, its performance may degrade if the independence assumption is violated significantly or if there are strong correlations between features. Additionally, it may struggle with numerical data unless properly preprocessed. Nevertheless, its speed, scalability, and ability to work well in a variety of domains make Naive Bayes a valuable tool for machine learning practitioners.

Support Vector Machine (SVM) is a supervised learning algorithm designed to find the optimal hyperplane that separates data into distinct classes with the maximum margin (Lu et al., 2024). This margin maximization ensures that the model achieves a robust decision boundary, minimizing the risk of misclassification. SVM is especially appropriate for linearly separable data but it has also the ability to classify non-linear cases with the help of some kernel functions (such as polynomial, radial basis function (RBF), or sigmoid kernels); this allows data to be transformed into higher dimensions, in which a separating hyperplane can be found within a more complex dataset.

Decision trees are very simple yet really strong machine learning algorithm used to represent their decision a feature value using a tree structure (Adegbehingbe et al., 2024). A decision rule is represented by each internal node in the tree, and its consequence is represented by each branch, culminating in a leaf node that offers the ultimate forecast. These algorithms are perfect for comprehending the connections between features and results because they are very interpretable and simple to display. Additionally, they are adaptable and proficient in managing both numerical and categorical data.

This research aims to classify user perceptions of the PrimaKu application based on their reviews and evaluate the performance of different machine learning algorithms in this context. The study not only identifies the most effective algorithm for this task but also provides insights for improving the PrimaKu application, ultimately enhancing user satisfaction and fostering better engagement in child health monitoring.

However, few studies have applied machine learning models specifically to classify user sentiment in the context of digital child health monitoring applications in Indonesia. Most prior evaluations of user perception rely on descriptive surveys, which may not capture nuanced textual feedback from users. Thus, this research addresses the gap by utilizing real-world user reviews and evaluating different machine-learning algorithms for sentiment classification.

2. Research Methods

This study aims to classify user perceptions of the PrimaKu application using various machine learning algorithms. The research methods are divided into several stages, including data collection, preprocessing, model training, evaluation, and performance optimization. Figure 1 show research methods.

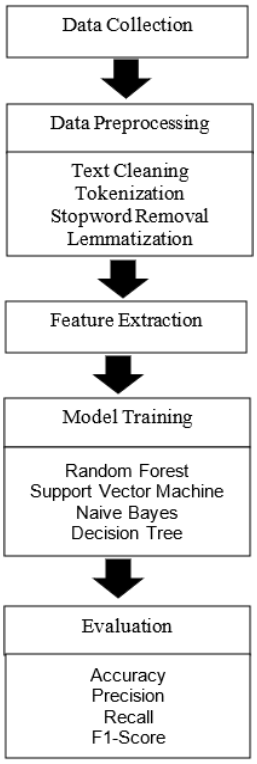

Figure 1 Research Methods

Figure 1 contains the research method that details as follows:

1. Data Collection

The dataset used in this study consists of 1000 user reviews of the PrimaKu application at Google Playstore. Each review includes textual feedback and a corresponding star rating provided by users.

2. Data Preprocessing

Several preprocessing steps were applied to prepare data. Text Cleaning removes noise such as punctuation, numbers, and URLs while converting text to lowercase to ensure uniformity(Daraghmi et al., 2024). Tokenization splits the cleaned text into smaller units (tokens), typically words, making the text easier to analyze(Siino et al., 2024). Stopword Removal eliminates common but less meaningful words, allowing the model to focus on the relevant terms(Yousafzai et al., 2024). Finally, Lemmatization standardizes words to their root forms, reducing redundancy and improving the consistency of the dataset(Abidin et al., 2024).

3. Feature Extraction

TF-IDF (Term Frequency-Inverse Document Frequency) vectorization was applied to convert the preprocessed textual data into numerical features. This method emphasizes unique and meaningful words by reducing the influence of commonly used terms. TF-IDF was chosen over alternatives like Word2Vec or BERT due to its simplicity, interpretability, and suitability for limited datasets in classical machine learning models.

4. Model Training

User perceptions classification was done using four techniques of machine learning: Random Forest, Support Vector Machine (SVM), Naive Bayes, and Decision Tree. The implementation was carried out using the Python programming language with the scikit-learn library. Model parameters were tuned with default settings, as the primary objective of this study was to compare baseline performance among the algorithms rather than to conduct hyperparameter optimization.

5. Evaluation

A separate test set was utilized to assess the performance of each model. The key metrics for evaluation include the following: Accuracy, Precision, Recall and F1-Score.

3. Results and Discussion

The research use the star to classify the class of the dataset. The star ratings used to label the data into three categories.

Table 1. Data Label

Rating	Label
1-2	Negative
3	Neutral
4-5	Positive

Table 1 categorizes user application ratings into three sentiment categories: negative, neutral, and positive, based on the given scores. Ratings of 1-2 are classified as negative, reflecting poor user experiences often accompanied by complaints or dissatisfaction with the application's features or performance. Ratings of 3 are considered neutral, indicating average user experiences without significant expressions of satisfaction or dissatisfaction. Reviews with this rating often provide constructive feedback. Meanwhile, ratings of 4-5 are classified as positive, representing good or excellent user experiences, where users typically praise the application's features, benefits, or design.

Table 2 Sentiment Labeling

No	Content	Rating	Label
1	Semua nya bagus tpi knpa ya aplikasi nya mau m...	2	negative
2	baru coba semoga bagus	3	neutral
3	bagus	5	positive
4	Habis download heran knp hp jd lemot ternyata ...	2	negative
5	masih,mau dicoba aplikasinya	5	positive

The next step of the research is to pre-process the data. Text cleaning aims to remove special characters, punctuation, numbers, and

URLs from the reviews.

Table 3 Cleaned Data

No	Content
1	semua nya bagus tpi knpa ya aplikasi nya mau m...
2	baru coba semoga bagus
3	bagus
4	habis download heran knp hp jd lemot ternyata ...
5	masihmau dicoba aplikasinya

Table 3 shows the cleaned data obtained after applying the preprocessing steps to the user reviews. Then, tokenization done by splitting text into individual words or tokens.

Table 4 Tokens

No	Content
1	['nya', 'bagus', 'tpi', 'knpa', 'ya', 'aplikas...
2	['coba', 'semoga', 'bagus']
3	['bagus']
4	['habis', 'download', 'heran', 'knp', 'hp', 'j...
5	['masihmau', 'dicoba', 'aplikasinya']

Table 4 illustrates the tokenized data generated from the cleaned user reviews. Tokenization is a preprocessing step in which the text is divided into individual words, or tokens, to facilitate further analysis.

Table 5 Stopwords Removal

No	Content
1	['nya', 'bagus', 'tpi', 'knpa', 'ya', 'aplikas...
2	['coba', 'semoga', 'bagus']
3	['bagus']
4	['habis', 'download', 'heran', 'knp', 'hp', 'j...
5	['masihmau', 'dicoba', 'aplikasinya']

Table 5 presents the data after the stopword removal process, where common but less meaningful words (e.g., "and", "the") have been eliminated from the tokenized text. Stopwords are typically words that appear frequently in text but do not contribute significantly to the overall meaning or context. The removal of these words helps to reduce noise and focus on the most relevant terms for analysis.

Table 6 Lemmatization

No	Content
1	['nya', 'bagus', 'tpi', 'knpa', 'ya', 'aplikas...
2	['coba', 'semoga', 'bagus']
3	['bagus']
4	['habis', 'download', 'heran', 'knp', 'hp', 'j...
5	['masihmau', 'dicoba', 'aplikasinya']

Table 6 presents the data after applying the Lemmatization process, a key step in text preprocessing where words are converted into their base or root forms. This process helps standardize the text by reducing word variations, such as "running," "runs," and "ran," into their root form, "run". The next step involves the implementation of Random Forest, Support Vector Machine, Naive Bayes, and Decision Tree algorithms. The results are presented in Table 7.

Table 7 Evaluation Results

Algorithm	Accuracy	Precision	Recall	F1-Score
Random Forest	0.770	0.620	0.540	0.520
SVM	0.810	0.690	0.570	0.550
Naïve Bayes	0.730	0.380	0.380	0.350
Decision Tree	0.740	0.510	0.490	0.490

Table 7 summarizes the performance of four machine learning models—Support Vector Machine (SVM), Naive Bayes, Decision Tree, and Random Forest—on the task of classifying user perceptions of the application. Among the models, SVM achieved the highest accuracy of 81%, demonstrating its effectiveness in correctly classifying the majority of the reviews. Random Forest followed with an accuracy of 77%, while Decision Tree and Naive Bayes scored 74% and 73%, respectively. Precision, which measures the correctness of positive predictions, was highest for SVM (0.69) and Random Forest (0.62), with Naive Bayes performing poorly at 0.38. In terms of recall, which evaluates the ability to capture all relevant instances, SVM also led with 0.57, while Random Forest achieved 0.54, outperforming Decision Tree and Naive Bayes. F1-score, which balances precision and recall, further highlights the superiority of SVM (0.55) and Random Forest (0.52) over Decision Tree (0.49) and Naive Bayes (0.35). These results indicate that SVM is the most reliable model for this task, providing a balance between precision, recall, and overall accuracy. Random Forest also performed well and serves as a competitive alternative, especially due to its robustness and generalization capabilities. Naive Bayes, on the other hand, struggled with both precision and recall, highlighting its limitations in handling complex textual data.

Confusion matrix is also important to evaluate models. Figure 2-6 show the confusion matrix results for each algorithm.

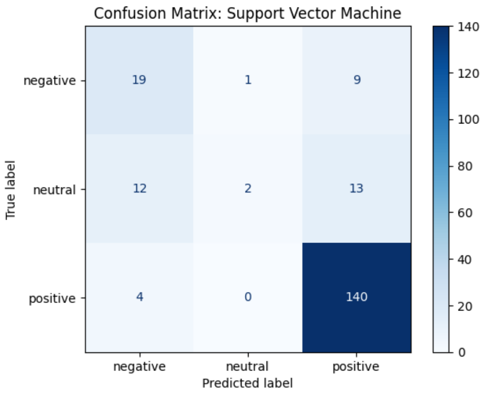

Figure 2 Confusion Matrix SVM

Figure 2 shows the confusion matrix for the Support Vector Machine (SVM) model, illustrating its performance in classifying user perceptions into three categories: negative, neutral, and positive. The model demonstrates strong performance in predicting positive reviews, with 140 instances correctly classified as positive and only 4 instances misclassified as negative. For the negative class, 19 instances were accurately classified, but the model struggled with misclassifications, where 1 instance was predicted as neutral and 9 as positive. The neutral class posed the greatest challenge for the model, with only 2 instances correctly classified, while 12 were misclassified as negative and 13 as positive.

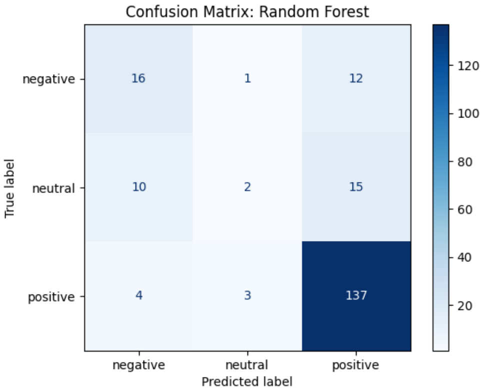

Figure 3 Confusion Matrix Random Forest

Figure 3 illustrates the confusion matrix for the Random Forest model, which evaluates its performance in classifying user perceptions into three categories: negative, neutral, and positive. The matrix provides a breakdown of true labels (actual categories) versus predicted labels (model classifications). The model performed well in identifying positive sentiments, with 137 instances correctly classified as positive. However, there were a few misclassifications, where 4 positive instances were misclassified as negative, and 3 as neutral. For the negative class, the model correctly identified 16 instances but misclassified 1 as neutral and 12 as positive. The neutral class posed a greater challenge for the model, with only 2 instances correctly classified, while 10 were misclassified as negative and 15 as positive

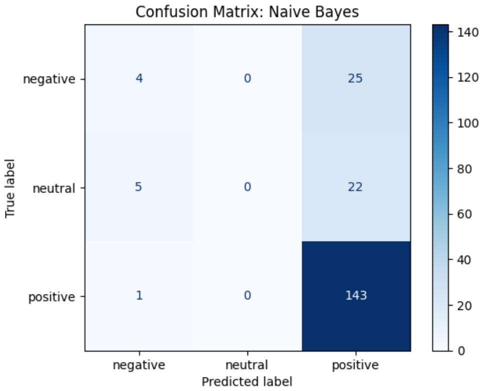

Figure 4 Confusion Matrix Naive Bayes

Figure 4 depicts the confusion matrix for the Naive Bayes model, illustrating its performance in classifying user perceptions into the categories of negative, neutral, and positive. The matrix highlights the distribution of correctly and incorrectly classified instances for each class. The model performed strongly in classifying positive sentiments, with 143 instances correctly identified. However, there were minor misclassifications, with 1 positive instance misclassified as negative. For the negative class, the model correctly identified only 4 instances, while the majority (25 instances) were misclassified as positive. The neutral class showed moderate performance, with 22 instances correctly classified, but 5 instances were misclassified as negative.

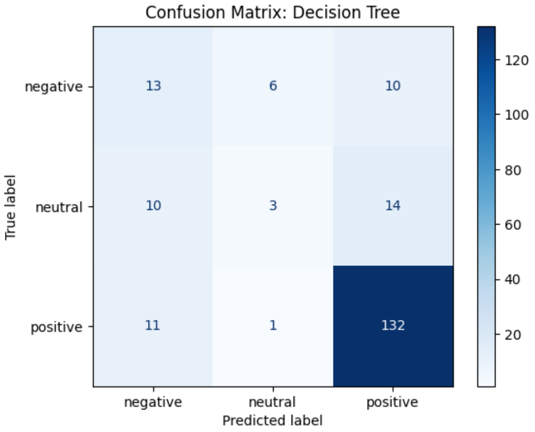

Figure 5 Confusion Matrix Decision Tree

Figure 5 shows the confusion matrix for the Decision Tree model, which illustrates how the algorithm classifies user sentiment into three categories: negative, neutral, and positive. The model correctly identified 132 positive reviews while misclassifying 11 as negative and one as neutral. 13 instances were classified correctly for the negative class. In contrast, 6 and 10 were misclassified as neutral and positive, respectively. The neutral class had the lowest performance, with only three correctly identified reviews and a larger number misclassified into other categories. These results highlight the decision tree model’s strength in identifying positive sentiment and its limitations in accurately capturing neutral and negative categories.

The performance of SVM is attributed to its ability to construct a robust hyperplane that maximizes the margin between classes, making it effective in handling high-dimensional and sparse data typically found in text classification. In contrast, Naive Bayes underperformed due to its strong independence assumptions, which do not hold well in real-world textual datasets. The neutral class posed classification challenges across all models, indicating the need for improved feature representation or class balancing methods.

4. Conclusion

This study evaluated the performance of four machine learning models: Support Vector Machine (SVM), Naive Bayes, Decision Tree, and Random Forest in classifying user perceptions of a child health monitoring application. Among these models, SVM proved the most effective, achieving the highest accuracy (81%) and outperforming the others in precision, recall, and F1-score. Its impressive performance underscores its ability to handle high-dimensional textual data while ensuring consistent predictive accuracy. Thanks to its ensemble learning architecture, the Random Forest model delivered reliable results, achieving an accuracy of 77% and an F1-score of 0.52. In contrast, the Decision Tree model exhibited moderate accuracy at 74% but struggled to balance the evaluation metrics effectively. Naive Bayes showed the weakest performance due to its limitations in handling complex textual input. These results suggest that SVM and Random Forest are well-suited for sentiment classification tasks involving user-generated reviews in health applications. Future research is encouraged to explore deep learning approaches such as LSTM and BERT and incorporate class balancing techniques like SMOTE to enhance model accuracy and provide deeper insights into user sentiment.

Reference

Abidin, Z., Junaidi, A., & Wamiliana. (2024). Text Stemming and Lemmatization of Regional Languages in Indonesia: A Systematic Literature Review. *Journal of Information Systems Engineering and Business Intelligence*, *10*(2), 217–231. https://doi.org/10.20473/jisebi.10.2.217-231

Adegbehingbe, O. D., Akinsemoyin, Z., Okunade, J. O., Arbor, A., Thompson, K. E., Akoriola, T. T., Adebayo, B. R., Etim, M. R., & Chinonyerem, C. A. (2024). Flood Prognosis Using Aggregation Machine Learning. *International Journal of African Innovation and Multidiciplinary*, *6*(2). https://doi.org/10.70382/mejaimr.v6i2.00

Anwar, A. S., Sayem, A., Shabbir, A., Taslima, N., Sikder, A. R., & Sidhu, G. S. (2024). Analyzing Enterprise Data Protection and Safety Risks in Cloud Computing Using Ensemble Learning International Journal on Recent and Innovation Trends in Computing and Communication Analyzing Enterprise Data Protection and Safety Risks in Cloud Computing Using Ensemble Learning. *Article in International Journal on Recent and Innovation Trends in Computing and Communication*. http://www.ijritcc.org

Arsad, I. K., Setyohadi, D. B., & Mudjihartono, P. (2021). E-commerce online review for detecting influencing factors users perception. *Bulletin of Electrical Engineering and Informatics*, *10*(6), 3156–3166. https://doi.org/10.11591/eei.v10i6.3182

Carrillo, E., González, M., Parrilla, R., & Tarrega, A. (2023). Classification trees as machine learning tool to explore consumers' purchasing decision pathway. A case-study on parent's perception of baby food jars. *Food Quality and Preference*, *109*. https://doi.org/10.1016/j.foodqual.2023.104916

Daraghmi, E. Y., Qadan, S., Daraghmi, Y. A., Yousuf, R., Cheikhrouhou, O., & Baz, M. (2024). From Text to Insight: An Integrated CNN-BiLSTM-GRU Model for Arabic Cyberbullying Detection. *IEEE Access*, *12*, 103504–103519. https://doi.org/10.1109/ACCESS.2024.3431939

Heritage Samuel, A. (2024). *A Comparative Evaluation of Machine Learning Algorithms for Predictive Modeling*. https://www.researchgate.net/publication/385945679

Kaluarachchi, T., Reis, A., & Nanayakkara, S. (2021). A review of recent deep learning approaches in human-centered machine learning. In *Sensors* (Vol. 21, Issue 7). MDPI AG. https://doi.org/10.3390/s21072514

Lu, J., Xie, X., & Xiong, Y. (2024). Multi-view hypergraph regularized Lp norm least squares twin support vector machines for semi-supervised learning. *Pattern Recognition*, *156*. https://doi.org/10.1016/j.patcog.2024.110753

Ong, A. K. S., Chuenyindee, T., Prasetyo, Y. T., Nadlifatin, R., Persada, S. F., Gumasing, M. J. J., German, J. D., Robas, K. P. E., Young, M. N., & Sittiwatethanasiri, T. (2022). Utilization of Random Forest and Deep Learning Neural Network for Predicting Factors Affecting Perceived Usability of a COVID-19 Contact Tracing Mobile Application in Thailand "ThaiChana." *International Journal of Environmental Research and Public Health*, *19*(10). https://doi.org/10.3390/ijerph19106111

Peretz, O., Koren, M., & Koren, O. (2024). Naive Bayes classifier – An ensemble procedure for recall and precision enrichment. *Engineering Applications of Artificial Intelligence*, *136*. https://doi.org/10.1016/j.engappai.2024.108972

Ren, G., Wang, Y., Ning, J., & Zhang, Z. (2020). Using near-infrared hyperspectral imaging with multiple decision tree methods to delineate black tea quality. *Spectrochimica Acta - Part A: Molecular and Biomolecular Spectroscopy*, *237*. https://doi.org/10.1016/j.saa.2020.118407

Salman, H. A., Kalakech, A., & Steiti, A. (2024). Random Forest Algorithm Overview. *Babylonian Journal of Machine Learning*, *2024*, 69–79. https://doi.org/10.58496/bjml/2024/007

Schickedanz, A. (2020). Evolving Roles for Health Care in Supporting Healthy Child Development HHS Public Access. In *Future Child* (Vol. 30, Issue 2).

Siino, M., Tinnirello, I., & La Cascia, M. (2024). Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers. *Information Systems*, *121*. https://doi.org/10.1016/j.is.2023.102342

Yousafzai, S. N., Shahbaz, H., Ali, A., Qamar, A., Nasir, I. M., Tehsin, S., & Damaševičius, R. (2024). X-News dataset for online news categorization. *International Journal of Intelligent Computing and Cybernetics*. https://doi.org/10.1108/IJICC-04-2024-0184