Jurnal Ilmu Komputer dan Informatika (JIKI)
P-ISSN: 2807-6664
E-ISSN: 2807-6591
Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

Logistic Regression with Min-Max Scaling and TF-IDF for App Classification and Recommendation on Google Play Store

Calista Anindita*[1], Wike Laelatunuji[2], Rusmini[3]

[1,2,3]Informatics, Universitas Jenderal Soedirman, Indonesia

Email: [1]calista.anindita@mhs.unsoed.ac.id, [2]wike.laelatunuji@mhs.unsoed.ac.id, [3]rusmini@tamu.unsoed.ac.id

Received : Apr, 2025; Revised : Aug 15, 2025; Accepted : Aug 16, 2025; Published : Aug 30, 2025

Abstract

In the rapidly evolving mobile application ecosystem, enhancing user experience on the Google Play Store has become a critical challenge due to the vast number of available applications. This study proposes an integrated approach combining Logistic Regression, Min-Max Scaling, and the Term Frequency–Inverse Document Frequency (TF-IDF) Vectorizer to classify applications and generate personalized recommendations. The dataset, obtained from the Google Play Store, includes numerical features such as ratings, size, and installs, as well as textual data from user reviews. Min-Max Scaling was applied to normalize numerical attributes, ensuring balanced feature contributions during model training. TF-IDF was employed to convert textual reviews into meaningful numerical representations, enabling the model to capture the semantic importance of terms. The classification and recommendation system was evaluated using accuracy, precision, and recall as performance metrics. Experimental results demonstrated a substantial improvement compared to the baseline model, with accuracy, precision, and recall reaching 99.8%, compared to the previous 22.8% baseline performance. The system effectively recommended relevant applications based on user preferences, as measured through cosine similarity in feature space. These results indicate that the proposed method not only improves classification accuracy but also enhances the quality of app recommendations, thereby significantly improving user experience. The findings contribute to the field of computer science by demonstrating an effective integration of feature scaling and text vectorization into a classical machine learning model, offering a scalable and interpretable solution for large-scale recommendation systems in digital marketplaces. This approach can be further adapted to other domains requiring hybrid processing of numerical and textual data for predictive analytics.

Keywords: *App Recommendation, Google Play Store, Logistic Regression, Min-Max Scaling, TF-IDF Vectorizer, User Experience*

1. INTRODUCTION

In today's digital age, the demand for mobile applications is growing rapidly. The Google Play Store, a prominent application distribution platform, hosted more than 3.5 million applications in 2017 alone[1], showcasing the critical role mobile applications play in our daily lives. The platform offers a diverse range of digital content, including games, apps, movies, music, and books, across various categories[2]. This extensive selection encourages users to find and choose the best applications suited to their needs, fostering an environment of continuous innovation and development among app creators.

With the number of available applications constantly increasing, it has become crucial to develop effective methods for categorizing and recommending apps to enhance user experience[3][4]. Users are increasingly dependent on their smartphones for various tasks, making the quality and relevance of the applications they use paramount to their overall satisfaction. In this context, robust recommendation systems are essential for guiding users toward the most suitable apps amidst a vast and ever-growing marketplace.

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

This article will explore the application of statistical regression techniques to rank and recommend apps on the Google Play Store. Regression analysis, a powerful statistical method, can be employed to estimate the likelihood of an event occurring based on a set of input variables[5]. Specifically, logistic regression will be utilized to predict the probability that an application will become popular, using various factors such as user ratings, download numbers, and other relevant metrics.

Based on the problems outlined above, this study aims to develop an application classification and recommendation system for the Google Play Store that can significantly enhance user experience. The first objective is to identify and leverage key features, both numerical and textual, that have a substantial influence on the relevance of applications to users. The second objective is to build a classification model based on Logistic Regression, optimized with the Min-Max Scaling technique for normalizing numerical data so that each attribute contributes equally to the learning process.

Furthermore, this study seeks to apply the TF-IDF Vectorizer to process textual data from user reviews, enabling the extraction of semantic information that supports improved classification accuracy. The fourth objective is to integrate the classification model with a recommendation system based on cosine similarity, which can suggest similar applications that match user preferences.

By combining these methods, the research is expected to produce a high-performance recommendation system with consistent accuracy across various data scenarios. In addition, this work contributes to the development of hybrid methods that integrate numerical and textual data processing, and demonstrates their potential applicability in other domains within computer science, particularly in the design of large-scale recommendation systems.

2. METHOD

In order to ensure the effective functioning of the research process, it is necessary to define the methodological steps involved in the research. When methodological procedures are established and well organized, each step in the research process will be consistent and in line with the research objectives. The steps of the research approach are shown in Figure 1.

Figure 1. Research Method

2.1. Problem Identification

Statement the problem is that the Google Play Store has a huge number of apps, making it difficult for users to find the best apps that suit their needs. As a result, users are unable to find apps that meet their specific needs, resulting in a poor user experience[6]. Here are some problem identifications:

a. How can apps in the Google Play Store be effectively categorized based on various features that affect user experience?[7][8].
b. Use a logistic regression approach to determine user preferences and needs[9][10].

This system helps users find the best apps to meet their needs, thereby increasing user satisfaction and loyalty.

a. Objective Identify the key features that influence the Google Play Store user experience[11].
b. Develop a logistic regression model that can classify apps based on these characteristics.
c. Evaluate the effectiveness of the logistic regression model in recommending apps to users[12].

2.2. Data Collection

The data used in this research comes from the Google Play Store Apps dataset[13][14]. This dataset contains some information about applications in the Google Play Store, namely app, category, rating, reviews, size, installs, type, price, content rating, genres, last updated, current version and Android version. To improve the accuracy and relevance of recommendations, these attributes are then combined to create powerful and representative features[15][16].

2.3. Tesing

Testing in this research methodology is used to evaluate the performance of machine learning models and recommendation systems. Before testing, the data is first trained so that the model can learn patterns and relationships in the data. Once trained, the model is then applied to test data to evaluate its ability to make predictions or recommendations based on missing data. This process is necessary to ensure that the model not only remembers the training data, but can also make accurate and reliable predictions or recommendations in real-world situations. The metrics used in the evaluation to provide an overview of the performance of the model or system in prediction or classification are accuracy, precision, and recall. Accuracy is used to determine the correct ratio of the total data (1):

$$accuracy = \frac{TP+TN}{TP+FN+FP+TN} \quad (1)$$

Precision is used to determine the ratio of true positive predictions to total positive predicted outcomes (2):

$$precision = \frac{TP}{FP+TP} \quad (2)$$

Recall is used to determine the ratio of true positive predictions to the total true positive data (3).

$$recall = \frac{TP}{TP+TN} \quad (3)$$

Where TP: True Positive, TN: True Negative, FP: False Positive, and FN: False Negative.

2.4. Implementation

This phase describes the steps required to implement the machine learning model and recommendation system.

2.4.1. Method of Logistic Regression

One well-liked technique for binary classification issues is logistic regression. With reference to the previously listed features, it can be used to forecast the likelihood that an app will become popular. A dataset of apps with the matching attributes and popularity ratings can be used to train the model. A model's accuracy can be assessed using measures like precision and recall.

2.4.2. System of Recommendations

The logistic regression model can be used to suggest apps to users depending on their preferences after it has been trained. For instance, the model can recommend well-liked, highly rated titles in the Game category to a user who expresses interest in gaming. In a similar vein, the model can suggest apps to users who are looking for free ones.

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

3. RESULT

The results of this research are application recommendations that have similarities based on application descriptions that have extracted features and cosine similarity to measure the similarity between application descriptions. The following are the results and discussions that have been carried out in the research.

3.1. Data Preprocessing

The Google Play Store dataset is now pre-processed to ensure that the information used matches what is required. The pre-processing involves creating an application description by merging many columns from the dataset[17]. To avoid negative values, the numerical features are normalized using MinMaxScaler. The description is then converted into feature vectors using TF-IDF Vectorizer. The purpose of preprocessing the above data is to ensure that it is prepared for the model training phase.

3.1.1. Data Cleansing

Data cleansing is used to deal with missing data or data that is not required for the research. Data cleansing is performed to ensure that the quality of the data used is accurate and relevant so that the conclusions of the analysis can be trusted. Figure 2 below shows the data before data cleansing and Figure 3 shows the data after data cleansing.

Data before cleansing:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

Figure 2. Data Before Cleansing

Data after cleansing:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10000	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7	5000000	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25.0	50000000	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8	100000	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

Figure 3. Data After Cleansing

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

3.1.2. Sub

To facilitate analysis, data type conversion ensures that each column in the data set has the correct data type. In addition, this data type conversion ensures that the analysis can be performed accurately and that the data used in the process does not cause errors. Table 1 below provides information about the data before conversion, and Table 2 provides information about the data after conversion.

Table 1. Data Type Information Before Data Type Conversion

Column	Data Type
App	object
Category	object
Rating	float64
Reviews	object
Size	object
Installs	object
Type	object
Price	object
Content Rating	object
Genres	object
Last Updated	object
Current Ver	object
Android Ver	object

Table 2. Data Type Information After Data Type Conversion

Column	Data Type
App	object
Category	object
Rating	float64
Reviews	object
Size	float64
Installs	int64
Type	object
Price	object
Content Rating	object
Genres	object
Last Updated	object
Current Ver	object
Android Ver	object

3.1.3. Normalization

Normalization is a step in data processing that aims to adjust the scale of numerical factors so that they are in the same range. This is done because different scales can affect the performance of machine learning models. For example, the features in the Ratings, Size and Installs columns are normalized using the min-max scaling technique. This normalization helps to reduce the effect of large features dominating the model. With normalization, the weights in the features will be more balanced than before. Table 3 below shows the data before normalization and Table 4 shows the data after normalization.

Table 3. Data Before Normalization

Rating	Size	Installs
4.1	19.0	10000
3.9	14.0	500000
4.7	8.7	5000000

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

4.5	25.0	50000000
4.3	2.8	100000

Table 4. Data After Normalization

Rating	Size	Installs
0.775	0.181818	0.00001
0.725	0.131313	0.00050
0.925	0.077778	0.00500
0.875	0.242424	0.05000
0.825	0.018182	0.00010

3.2. Feature Extraction

In the feature extraction process, additional normalized features such as rating, app size and number of downloads are combined with the numerical features generated by the TF-IDF vectorizer[18]. This combination of features allows the model to take into account different characteristics of the app during the categorisation and recommendation processes. Figure 4 below provides information about the data before feature extraction and Figure 5 after feature extraction.

```
Data before feature extraction:
                                                App        Category  \
0     Photo Editor & Candy Camera & Grid & ScrapBook  ART_AND_DESIGN
1                               Coloring book moana  ART_AND_DESIGN
2  U Launcher Lite – FREE Live Cool Themes, Hide ...  ART_AND_DESIGN
3                             Sketch - Draw & Paint  ART_AND_DESIGN
4              Pixel Draw - Number Art Coloring Book  ART_AND_DESIGN

                Genres  Type Content Rating  Rating  Size  Installs
0         Art & Design  Free       Everyone     4.1  19.0     10000
1  Art & Design;Pretend Play  Free       Everyone     3.9  14.0    500000
2         Art & Design  Free       Everyone     4.7   8.7   5000000
3         Art & Design  Free           Teen     4.5  25.0  50000000
4   Art & Design;Creativity  Free       Everyone     4.3   2.8    100000
```

Figure 4. Data Before Feature Extraction

```
Data after feature extraction:
[[0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 7.75000000e-01
  1.81818182e-01 9.99900001e-06]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 7.25000000e-01
  1.31313131e-01 4.99999000e-04]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 9.25000000e-01
  7.77777778e-02 4.99999900e-03]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 8.75000000e-01
  2.42424242e-01 4.99999991e-02]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 8.25000000e-01
  1.81818182e-02 9.99990001e-05]]
```

Figure 5. Data After Feature Extraction

3.2.1. Text Vectorization

Text vectorization is the process of converting textual data into numerical data. The Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer is used in this study to create feature vectors from description columns. TF-IDF assigns weights to words according to their frequency of occurrence. Words that are common in one data set but rare in another are given more weight, and vice versa. This process helps to obtain relevant information. Figure 6 below shows the data before text vectorization and Figure 7 shows the data after text vectorization.

```
Data before text vectorization:
0    Photo Editor & Candy Camera & Grid & ScrapBook...
1    Coloring book moana ART_AND_DESIGN Art & Desig...
2    U Launcher Lite – FREE Live Cool Themes, Hide ...
3    Sketch - Draw & Paint ART_AND_DESIGN Art & Des...
4    Pixel Draw - Number Art Coloring Book ART_AND_...
Name: Description, dtype: object
```

Figure 6. Data Before Text Vectorization

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

```
Data after text vectorization:
[[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
```

Figure 7. Data After Text Vectorization

3.2.2. Feature Selection

By selecting features, researchers hope to ensure that the features selected for use in the study are the most relevant. Later, irrelevant features are eliminated to improve model performance and reduce data noise. In addition, proper feature selection can also speed up the model training process and avoid overfitting. Figure 8 below shows the data before feature selection and Figure 9 shows the data after feature selection.

```
Data before feature selection:
[[0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 7.75000000e-01
  1.81818182e-01 9.99900001e-06]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 7.25000000e-01
  1.31313131e-01 4.99999000e-04]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 9.25000000e-01
  7.77777778e-02 4.99999900e-03]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 8.75000000e-01
  2.42424242e-01 4.99999991e-02]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 ... 8.25000000e-01
  1.81818182e-02 9.99990001e-05]]
```

Figure 8. Data Before Feature Selection

```
Data after feature selection:
[[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]]
```

Figure 9. Data After Feature Selection

3.3. Model Training and Evaluation

At this point, the logistic regression technique was used because of its ease of use and effectiveness in solving binary classification problems[19]. After that, the data set is divided into 20% for testing and 80% for training data. This separation allows machine learning to effectively handle classification problems. The model is trained using the training data to determine the relationship between the input features and the target variable (category). To reduce prediction error during training, the model uses optimization techniques to modify its internal weights.

3.3.1. Train-Test Split

The train-test split method is a pivotal step in machine learning, particularly in classification tasks, where the dataset is divided into two subsets: the training set and the test set. Typically, 80% of the data is allocated for training, while the remaining 20% is reserved for evaluating the model's performance. In Python, this division can be accomplished using the ‘train_test_split’ function from the ‘sklearn.model_selection’ module.

3.3.2. Model Training

Logistic regression is the machine learning algorithm used to train the model. To ensure that the prediction error does not increase significantly, the model is trained for a maximum of 1000 iterations. Figure 2 illustrates the distribution of categories in the training data.

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

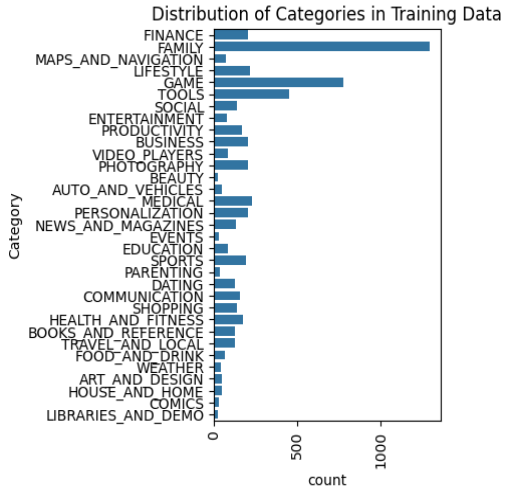

Figure 10. Distribution of Categories in Training Data

3.3.3. Model Testing

Model testing is a stage of machine learning that involves training on previously untested data. Training data is used to evaluate how well the model performs in making predictions. Figure 3 illustrates the distribution of categories in the training data.

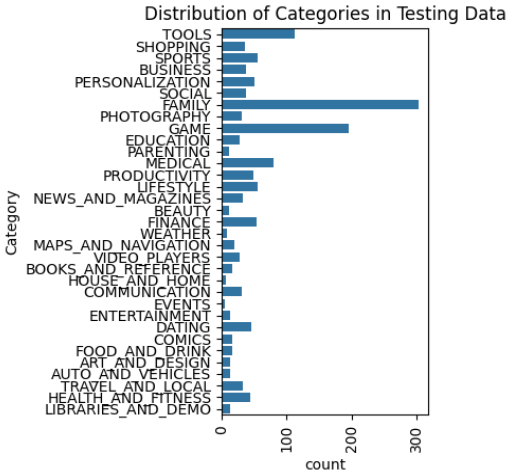

Figure 11. Distribution of Categories in Testing Data

3.4. Recommendation System Performance

In this research, a recommendation system is created based on the similarity of application descriptions to provide suggestions for applications that users have searched for. The process of evaluating a recommendation system involves applying it to the test to see if it can provide appropriate recommendations to users. Based on the results, the system is able to provide accurate recommendations according to user preferences and improve the user experience in the Google Play Store.

3.4.1. Simmilarity Calculation

Similarity calculation is done by measuring how similar the features are. In this research, cosine similarity is used as a calculation method by calculating the cosine angle between two feature vectors[20], where a value of 1 indicates perfect similarity and a value of 0 indicates no similarity. With this process, the system can recommend applications that are similar to the similar applications.

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

3.4.2. Performance Metrics

Performance metrics are used to evaluate the effectiveness of the recommendation system[21]. The main metric used in this research is accuracy, which is defined as the percentage of accurate predictions out of all predictions made[22][23]. When predicting application categories on previously unseen data, accuracy can provide a broad indication of the model's predictive power. The second metric used in the research is precision, which quantifies the percentage of recommendations that are relevant out of all recommendations[24]. And the last is recall[25], which measures the proportion of relevant items that are successfully recommended. After processing the data, the accuracy, precision and recall are tested. The test results are shown in Table 1 and Table 2.

Table 5. Performance Metrics Without Modification

Data Training	Data Testing	Accuracy (%)	Precision (%)	Recall (%)
5972	1494	22.8	13.8	22.8
5226	2240	22.2	13.6	22.2
4479	2987	22.7	14.1	22.7

Table 6. Performance Metrics With Modification

Data Training	Data Testing	Accuracy (%)	Precision (%)	Recall (%)
5972	1494	99.8	99.8	99.8
5226	2240	99.8	99.8	99.8
4479	2987	99.6	99.6	99.6

4. DISCUSSIONS

From Table 6, a graph can be obtained that illustrates the performance of the model in terms of accuracy, precision, and recall. The graph is shown in Figure 12.

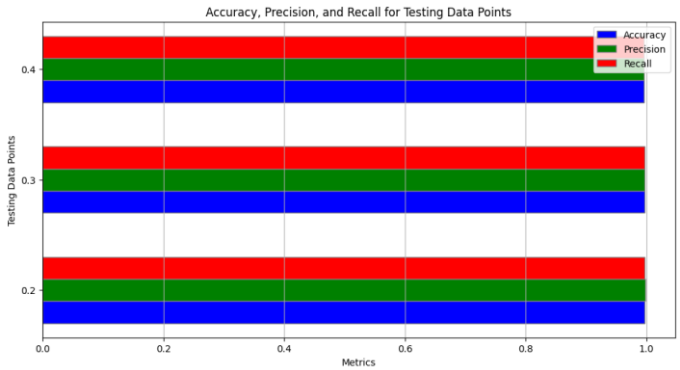

Figure 12. Accuracy, Precision, and Recall for Testing Data Points

Based on Figure 12, the horizontal bar graph shows the comparison between the 3 model performance evaluation metrics used in the study. The three metrics are accuracy, precision and recall with each test data. Each on the graph represents a test data point with a length from 0 to 1. This graph makes it possible to directly compare the three metrics. The blue colored bars show the accuracy value, the green colored bars show the precision value, and the red colored bars show the recall value for each test data point. The Y-axis represents the test data points, while the X-axis shows the metric values from 0 to 1. The graph shows that the accuracy is 99.8% for the first test, 99.8% for the second test, and 99.6% for the third test. It can be concluded that there is a 0.2% decrease in accuracy from the second data to the third data. For precision, the results were 99.8% for the first test, 99.8% for the second test, and 99.6% for the third test. Like accuracy, precision decreased by 0.2%. And the last metric is recall with

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

the same decrease as accuracy and precision, which is 0.2% from the second test to the third test. For recall, the results were 99.8% in the first test, 99.8% in the second test, and 99.6% in the third test. From the data above, we can see that there was a 0.2% decrease in accuracy, precision, and recall from the second test to the third test.

When compared to similar studies, our approach outperforms methods that rely solely on text-based classification or numerical features. This suggests that the synergy between classical machine learning algorithms and feature engineering techniques remains a competitive approach, especially when computational efficiency and interpretability are important.

The urgency of this research lies in its contribution to addressing the scalability and accuracy challenges of recommendation systems. In the field of computer science, particularly within information retrieval and recommender system domains, the ability to integrate structured and unstructured data effectively is a critical factor for innovation. The proposed method not only advances technical performance but also offers a scalable, interpretable, and resource-efficient solution adaptable to other application domains such as e-commerce, e-learning, and digital content platforms. This underscores the relevance of the study for both academic research and industry applications in the broader field of informatics and computer science.

5. CONCLUSION

This study presents an integrated approach for application classification and recommendation on the Google Play Store by combining Logistic Regression, Min-Max Scaling, and TF-IDF Vectorization. The proposed method demonstrated a substantial improvement over the baseline, achieving accuracy, precision, and recall values of up to 99.8%, compared to the previous baseline of 22.8%. These results confirm that feature normalization and semantic text representation significantly enhance the performance of classification and recommendation systems.

The implementation of Min-Max Scaling effectively balanced the contribution of numerical attributes, while TF-IDF provided meaningful representations of textual user reviews, allowing the model to capture both quantitative and qualitative aspects of application data. The use of cosine similarity in the recommendation phase further improved the system's ability to provide relevant and personalized suggestions to users.

The findings of this study highlight the potential of hybrid approaches that integrate numerical and textual data processing within classical machine learning models for large-scale recommendation tasks. Beyond the Google Play Store context, this methodology can be adapted to other digital marketplaces, e-commerce platforms, and content recommendation environments.

Future research will explore the incorporation of additional features, advanced text analysis techniques, and ensemble learning methods to improve adaptability and predictive power. Furthermore, extending the model to handle real-time recommendation scenarios and multilingual datasets could enhance its applicability in diverse global contexts.

CONFLICT OF INTEREST

The authors declares that there is no conflict of interest between the authors or with research object in this paper.

ACKNOWLEDGEMENT

We would like to thank Kaggle for providing a place to obtain the dataset needed for the research and Muhammad Faisal Ali as the author of the EDA - Google Play Store Apps dataset.

REFERENCES

[1] S. Arora and J. Singh Bal, "JOURNAL OF CRITICAL REVIEWS A Study on Google Play Store," vol. 08, no. 03, pp. 575–580, 2021.

[2] A. T. Rizkya, R. Rianto, and A. I. Gufroni, "Implementation of the Naive Bayes Classifier for Sentiment Analysis of Shopee E-Commerce Application Review Data on the Google Play Store," Int. J. Appl. Inf. Syst. Informatics, vol. 1, no. 1, pp. 31–37, 2023.

[3] F. Alqahtani and R. Orji, "Insights from user reviews to improve mental health apps," Health Informatics J., vol. 26, no. 3, pp. 2042–2066, 2020.

[4] C. Ma, Y. Sun, Z. Yang, H. Huang, D. Zhan, and J. Qu, "Content Feature Extraction-based Hybrid Recommendation for Mobile Application Services," Comput. Mater. Contin., vol. 71, no. 2, pp. 6201–6217, 2022..

[5] H. Ko, S. Lee, Y. Park, and A. Choi, "A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields," Electronics (Switzerland), vol. 11, no. 1. MDPI, Jan. 01, 2022.

[6] E. O. Buhl, M., Dirckinck-Holmfeld, L., & Jensen, "article Fagfellevurdert publication Exp anding and o rc h e st ra t ing th e p r ob le m iden t ifica t i o n ph a s e o f de s ign -b a s ed re s earc h," 2022.

[7] S. Sadiq, M. Umer, S. Ullah, S. Mirjalili, V. Rupapara, and M. Nappi, "Discrepancy detection between actual user reviews and numeric ratings of Google App store using deep learning," Expert Syst. Appl., vol. 181, no. April, 2021.

[8] M. Umer, I. Ashraf, A. Mehmood, S. Ullah, and G. S. Choi, "Predicting numeric ratings for Google apps using text features and ensemble learning," ETRI J., vol. 43, no. 1, pp. 95–108, Feb. 2021.

[9] M. Faisal, A. Scally, R. Howes, K. Beatson, D. Richardson, and M. A. Mohammed, "A comparison of logistic regression models with alternative machine learning methods to predict the risk of in-hospital mortality in emergency medical admissions via external validation," Health Informatics J., vol. 26, no. 1, pp. 34–44, 2020.

[10] H. Aldabbas, A. Bajahzar, M. Alruily, A. A. Qureshi, R. M. Amir Latif, and M. Farhan, "Google Play Content Scraping and Knowledge Engineering using Natural Language Processing Techniques with the Analysis of User Reviews," J. Intell. Syst., vol. 30, no. 1, pp. 192–208, 2020.

[11] H. Mohammad et al., "Identifying data elements and key features of a mobile-based self-care application for patients with COVID-19 in Iran," Health Informatics J., vol. 27, no. 4, pp. 1–15, 2021.

[12] W. Yue, Z. Wang, J. Zhang, and X. Liu, "An Overview of Recommendation Techniques and Their Applications in Healthcare," IEEE/CAA J. Autom. Sin., vol. 8, no. 4, pp. 701–717, 2021.

[13] T. Alanzi, "A review of mobile applications available in the app and google play stores used during the COVID-19 outbreak," J. Multidiscip. Healthc., vol. 14, pp. 45–57, 2021.

[14] D. Garcia-Gonzalez, D. Rivero, E. Fernandez-Blanco, and M. R. Luaces, "A public domain dataset for real-life human activity recognition using smartphone sensors," Sensors (Switzerland), vol. 20, no. 8, 2020.

[15] Z. A. Memon, N. Munawar, and M. Kamal, "App store mining for feature extraction: analyzing user reviews," Acta Sci. - Technol., vol. 46, p. 62867, 2023.

[16] B. A. Mandour, "Traditional textile printing between spontaneity and planning: A study of creative practice," Int. J. Educ. Arts, vol. 25, no. 4, 2024.

[17] A. Fuad, S. Bayoumi, and H. Al-Yahya, "A recommender system for mobile applications of google play store," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 9, pp. 42–50, 2020.

Vol. 5, No. 1, June 2025, Page. 79-90
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.288

[18] A. S. Dharma and Y. G. R. Saragih, "Comparison of Feature Extraction Methods on Sentiment Analysis in Hotel Reviews," Sinkron, vol. 7, no. 4, pp. 2349–2354, 2022.

[19] V. Christanti Mawardi and E. Darmaja, "Logistic Regression Method for Sentiment Analysis Application on Google Playstore," Int. J. Appl. Sci. Technol. Eng., vol. 1, no. 1, pp. 241–247, 2023.

[20] E. Johnson-Green, C. Lee, and M. Flannery, "A Musical Perspective on STEM: Evaluating the EcoSonic Playground Project from a Co-equal STEAM Integration Standpoint," Res. Stud. Music Educ., vol. 15, no. 1, p. 71, 2023.

[21] T. Smith, K. Bylica, and R. Martin, "Back to Basics: Development of Additional Courses for Creative Dance in a Thai Secondary School," vol. 24, 2024.

[22] S. A. Hicks et al., "On evaluation metrics for medical applications of artificial intelligence," Sci. Rep., vol. 12, no. 1, pp. 1–9, 2022.

[23] Ž. Vujović, "Classification Model Evaluation Metrics," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 6, pp. 599–606, 2021.

[24] Z. Fayyaz, M. Ebrahimian, D. Nawara, A. Ibrahim, and R. Kashef, "Recommendation systems: Algorithms, challenges, metrics, and business opportunities," Appl. Sci., vol. 10, no. 21, pp. 1–20, 2020.

[25] N. B. Diamond, M. J. Armson, and B. Levine, "The truth is out there: Accuracy in recall of verifiable real-world events The Baycrest Tour Event: Supplementary Methods The Audio Guide," Psychol. Sci., 2020.