Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2026. Vol. 5. No. 2; e-ISSN: 2808-4519

Smartphone Rating Classification Based on Technical Specifications Using Naive Bayes and Feature Importance

Eka Yunizar Zalimatus Sa'diyah[1*], Mellanda Kurniawati[2], Harun Al Rosyid[3]

[1,2,3]*Universitas Negeri Surabaya*
Eka.22089@mhs.unesa.ac.id[1], mellanda.22116@mhs.unesa.ac.id[2], harunrosyid@unesa.ac.id[3]*

Abstract

Classifying smartphone ratings based on technical specifications is crucial for market analysis and consumer electronics marketing strategies. This study applies the Naive Bayes algorithm to categorize smartphone ratings into low, medium, and high levels, while identifying influential features via Mutual Information scores. The dataset includes 1,159 smartphones with eight categorical predictors: screen size, RAM capacity, pixel density (PPI), battery capacity, water resistance, display type, IP rating, and Android OS version.
Data preprocessing involved handling missing values with mode imputation, Label Encoding for categoricals, and stratified 80:20 train-test split to preserve class balance. The model achieved 77.16% accuracy and a weighted F1-score of 0.7724 on test data, with 5-fold cross-validation yielding a mean accuracy of 76.53%.Mutual Information analysis ranked RAM capacity (32.51%), PPI (25.78%), and display type (21.09%) as top features. These findings highlight key specs for differentiating rating groups, aiding product analysis, market positioning, and smartphone marketing in Indonesia.

***Keywords**: Naive Bayes, Feature Importance, Rating Classification, Mutual Information, Smartphone, Machine Learning*

1. Introduction

The smartphone industry is growing rapidly and is characterized by intense competition, forcing manufacturers to compete to deliver increasingly high-tech specifications to attract and maintain consumer loyalty [1]. In the context of digital marketing, ratings listed on e-commerce platforms and review sites serve as important indicators of perceived product quality, while also influencing smartphone interest and purchasing decisions [2]. Various studies have concluded that product attributes such as technical features, quality, and price are key factors influencing consumer purchase intention and behavior in selecting a particular smartphone brand [3].

Empirically, smartphone technical characteristics—such as performance, battery capacity, screen quality and size, camera, and operating system—have been shown to contribute to user satisfaction and product evaluation [4]. Product features have even been identified as one of the most dominant factors in explaining purchase intention and preference for smartphone brands [2]. In this context, technical specifications such as screen size, RAM capacity, pixel density (PPI), battery capacity, water resistance, screen type, IP protection level, and Android version have the potential to be important determinants of variations in smartphone ratings given by users [5]. However, the relationship between these specification combinations and rating categories (e.g., low, medium, high) is complex and not always linear, making it difficult to explain using conventional descriptive approaches alone [6].

The development of machine learning methods allows modeling the complex relationship between specification features and ratings through quantifiable and replicable approaches [7]. The Naive Bayes algorithm is widely used in text and review classification tasks due to its simple structure, high computational efficiency, and competitive performance on high-dimensional data, including sentiment analysis and smartphone reviews [8]. Various studies have shown that Naive Bayes is capable of achieving good accuracy on app and digital product review data, making it relevant for classifying user ratings or opinions [8].

Based on this foundation, research on classifying smartphone ratings based on technical specifications using the Naive Bayes algorithm combined with feature importance analysis using Mutual Information is relevant and important [4]. The use of variables such as screen size, RAM, PPI, battery capacity, air resistance, screen type, IP protection level, and Android version as predictors allows the identification of specifications that contribute most to smartphone rating formation (Nugroho, 2024; Niel van den Berg, 2021). The research results are expected to provide an empirical basis for producers and digital business actors in designing product specification compositions, determining strategic positioning, and developing smartphone marketing strategies that are oriented towards increasing ratings and consumer satisfaction in the Indonesian market [9].

2. Materials and Methods

This study uses a quantitative approach with a machine learning-based classification method to analyze the relationship between smartphone technical specifications and product rating levels. The research methods include data collection from the Versus.com platform, data preprocessing to prepare categorical features, implementation of the Naive Bayes algorithm as a classification model [10], feature importance analysis using Mutual Information Score as a feature selection/assessment method [11], and model evaluation using standard metrics and k-fold cross-validation techniques to assess model performance stability [12]. The entire research process was carried out using the Python programming language with the Pandas library for data manipulation, Scikit-learn for algorithm implementation and evaluation, and Matplotlib and Seaborn for result visualization.

2.1. Dataset and Data Collection

The dataset used in this study consisted of 1,159 smartphones collected from the technology product comparison platform Versus.com (https://versus.com/en/smartphone). Versus.com is a platform that provides comprehensive technical specifications for various consumer electronics products, including smartphones from various brands and price ranges. Each smartphone in the dataset was equipped with eight categorical predictor variables and one target variable. The data was collected through a web scraping process conducted over a period of 2025, ensuring the data's relevance to current smartphone market conditions. Web scraping is a technique for automatically collecting data from websites using efficient programming scripts to access large amounts of data without having to use public APIs, which often have limited access [13].

These predictor variables include: (1) screen size, categorized as Small Screen (<5.5 inches), Medium Screen (5.5-6.5 inches), and Large Screen (>6.5 inches); (2) RAM capacity which is categorized into Low RAM (< 4 GB), Mid RAM (4-8 GB), and High RAM (> 8 GB); (3) pixel density (PPI) which is categorized into Low PPI (< 250 ppi), Mid PPI (250-400 ppi), and High PPI (> 400 ppi); (4) battery capacity which is categorized into Low Battery (< 3500 mAh), Mid Battery (3500-5000 mAh), and High Battery (> 5000 mAh); (5) water resistance level which is categorized into Low Protect (non-waterproof), Mid Protect (IPx3-IPx4), and High Protect (IPx5 or higher); (6) screen type which is categorized into LCD and OLED; (7) IP protection level (Ingress Protection) which is categorized into Low IP (< IPx3), Mid IP (IPx3-IPx5), and High IP (> IPx5); and (8) Android operating system versions categorized into Old Android (Android < 10), Mid Android (Android 10-12), and New Android (Android > 12). This smartphone specification categorization follows the standards used in consumer preference analysis research on mobile phone products using machine learning methods such as Naive Bayes and Neural Networks [14].

The target variable is the smartphone rating, categorized into three classes based on the aggregated assessment scores on the Versus.com platform: Low Rating (rating < 3.5 stars), Mid Rating (rating 3.5-4.2 stars), and High Rating (rating > 4.2 stars). The data distribution shows that the dataset is relatively balanced, with 511 units (44.1%) in the Low Rating category, 545 units (47.0%) in the Mid Rating category, and 103 units (8.9%) in the High Rating category.

2.2. Data Preprocessing

The data preprocessing stage includes several important steps to prepare the data before modeling. First, each variable was checked for missing values. The results revealed only three missing values in the battery capacity variable (0.26% of the total data), which were then handled using imputation using the mode value of that variable. Handling missing values using the mode was chosen because all variables are categorical, making the mode the most appropriate method for replacing missing values while maintaining the original distribution of the data, as described in research on handling missing values using various imputation techniques, including the mean, interpolation, and KNN imputation [15].

Second, categorical variables were encoded using a Label Encoder to convert each category into an integer numeric value. This encoding process is necessary because machine learning algorithms like Naive Bayes require input in the form of numeric values that can be mathematically processed. Each category in each variable is converted to an integer ranging from 0 to (number of categories - 1). For example, for a RAM variable with three categories (LowRAM, MidRAM, HighRAM), each is encoded as 0, 1, and 2. Label encoding is one of the most fundamental methods in data preprocessing to ensure compatibility with machine learning algorithms [16], and research shows that choosing the right encoding technique can impact model performance.

Third, the data is divided into two sets: training data and testing data in an 80:20 ratio. This ratio is chosen based on best practices in machine learning to ensure the model has sufficient data for training while maintaining representative testing data for evaluation. The data is split using a stratified split, meaning the distribution of the target classes (Low Rating, Mid Rating, High Rating) in both sets is kept balanced and representative of the overall data distribution. With the stratified split, the training data consists of 927 units (80%) and the testing data consists of 232 units (20%), with the class distribution in both sets remaining balanced (Low Rating ~44%, Mid Rating ~47%, High Rating ~9%). Stratified splits are crucial in datasets that are not fully balanced to avoid bias in the testing data and ensure accurate model performance estimates, especially when the dataset has an uneven class distribution, as demonstrated by research on the effect of data split composition on classification accuracy using various machine learning algorithms [16].

After preprocessing is complete, all categorical variables are converted to numeric format and ready for use in training a Naive Bayes model. No additional normalization or standardization is required because Naive Bayes is insensitive to the scale of the numeric features resulting from categorical encoding.

2.3. Naïve Bayes Algorithm

Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem with the assumption that all features are conditionally independent (conditional independence assumption), which makes it one of the most fundamental and efficient algorithms in machine learning for classification tasks [10]. Mathematically, the probability of class C given features $F_1, F_2, ..., F_n$ can be expressed as:

$$P(C \mid F_1, F_2, \dots, F_n) = \frac{P(F_1, F_2, \dots, F_n \mid C) \times P(C)}{P(F_1, F_2, \dots, F_n)} \text{...(1)}$$

Assuming conditional independence, equation (1) can be simplified to:

$$P(C \mid F_1, F_2, \dots, F_n) \propto P(C) \times \prod_{i=1}^{n} P(F_i \mid C) \text{...(2)}$$

P(C) is the prior probability of class C which represents the initial probability of the class before considering any features, $P(F_i|C)$ is the likelihood of feature F_i given class C which indicates how likely a particular feature is to occur in that class, and $P(F_1, F_2, ..., F_n)$ is the evidence which is constant for all classes and therefore can be ignored in the comparison. For classification, the class C that maximizes the posterior probability is chosen:

$$\hat{C} = arg\ max_C\ P(C) \times \prod_{i=1}^{n} P(F_i \mid C) \text{...(3)}$$

In this study, a CategoricalNB variant is used which is specifically designed to handle categorical data, where each feature is assumed to follow a categorical distribution. The CategoricalNB variant is very suitable for datasets with categorical variables such as in this study, because it is able to handle nominal data without requiring transformation to continuous values [14]. The model is trained on the training data to calculate the prior probability P(C) of each class based on the sample proportion of each class in the training data, and the likelihood probability $P(F_i|C)$ for each combination of features and classes based on the empirical frequency in the training data. The main advantages of Naive Bayes are its simplicity of implementation, high computational speed, and its ability to work well on datasets with high dimensions and limited training data.

2.4. Feature Importance Analysis using Mutual Information

Feature importance is analyzed using the Mutual Information (MI) Score, which measures the statistical dependence between each predictor feature and the target variable, which is a very effective method for identifying the most informative features in a dataset [11]. The Mutual Information between feature F_i and target C is defined as:

$$MI(F_i; C) = \sum f \in F_i \sum c \in C\ P(f,c) log\big[P(f,c)/\big(P(f) \times P(c)\big)\big] (4)$$

Di mana P(f, c) adalah *joint probability* dari fitur F_i dan kelas C yang merepresentasikan peluang kemunculan bersama fitur dan kelas, P(f) adalah marginal *probability* dari fitur F_i yang menunjukkan distribusi probabilitas fitur tanpa mempertimbangkan kelas, dan P(c) adalah marginal *probability* dari kelas C yang merepresentasikan distribusi probabilitas kelas dalam *dataset*. Nilai MI yang lebih tinggi mengindikasikan bahwa fitur tersebut lebih informatif dan memiliki hubungan yang lebih kuat dengan target klasifikasi. MI *Score* dinormalisasi menjadi persentase kontribusi untuk setiap fitur agar mudah diinterpretasi:

$$Kontribusi_i = \left(MI(F_i; C) / \sum_{j=1}^{n} MI(F_j; C) \right) \times 100\% (5)$$

This method was chosen because the MI Score does not assume a linear relationship between features and targets, making it more suitable for categorical data and can identify non-linear interactions between variables. Research shows that mutual information-based feature importance analysis can improve classification accuracy by 3-5% by reducing the number of features used in the model, resulting in a simpler yet still accurate model [11]. In this study, the MI Score was used to identify the most important features that contribute to smartphone rating prediction, helping to understand which technical specifications most influence user satisfaction levels with smartphone products.

2.5. Model Evaluation

The performance of the Naive Bayes model is evaluated using several standard metrics on testing data that is not used during model training to ensure an objective and unbiased evaluation [12]. These metrics include accuracy, precision, recall, and F1-score, which are formulated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$F1\text{-}Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (9)$$

True Positive (TP) is the number of correctly predicted positive samples, True Negative (TN) is the number of correctly predicted negative samples, False Positive (FP) is the number of negative samples predicted as positive, and False Negative (FN) is the number of positive samples predicted as negative. Accuracy measures the overall proportion of correct predictions, precision measures how accurately the model predicts a particular class, recall measures how completely the model identifies samples from a particular class, and the F1-score provides a balance between precision and recall, which is useful when there is a class imbalance in the dataset. Furthermore, a confusion matrix is used to display the number of correct and incorrect predictions for each class, helping to identify specific misclassification patterns and indicating which classes are most frequently misclassified. The confusion matrix provides in-depth insight into the model's performance on each individual class, which cannot be obtained solely from overall accuracy.

5-fold cross-validation is applied by splitting the training data into five parts in turn to train and test the model, resulting in more stable and reliable estimates of model performance and reducing bias from specific data splits [12]. In this technique, the model is trained on 4 folds (80% of the training data) and tested on 1 fold (20% of the training data), this process is repeated 5 times until all folds have been used as testing data. The estimated cross-validation performance is calculated as the average accuracy of all 5 folds, providing a more generalizable performance assessment to new data. The standard deviation of the accuracy values in each fold is also reported to measure the stability of the model performance. All preprocessing, model training, and evaluation processes are performed using Python libraries: Pandas for data manipulation and transformation, Scikit-learn for the implementation of the Naive Bayes algorithm and the calculation of evaluation metrics, and Matplotlib/Seaborn for visualizing the analysis results in the form of gr

3. Result and Discussion

3.1. Dataset Description and Preprocessing

Secondary data sources included technical specifications, price, and user ratings. The data was divided into two subsets: 927 training units (80 percent) and 232 testing units (20 percent), using stratification to ensure a balanced class distribution across both subsets. The dataset includes eight predictor features relevant to smartphone popularity: RAM capacity (ram_cat), screen pixel density (PPI) (ppi_cat), screen type (screen_type_cat), Android operating system level (android_level), water-resistance level (water_level), screen size category (screen_size_cat), IP rating (dust and water protection) (ip_level), and battery capacity category (battery_cat).

The target variable consists of three smartphone popularity categories classified based on user ratings: LowRating, MidRating, and HighRating. The class distribution in the dataset shows a reasonable imbalance, with the majority of data falling into the MidRating category, followed by HighRating and LowRating with smaller percentages. Data preprocessing involved encoding categorical features using LabelEncoder to convert the category values into a numeric format that the algorithm could process. There were no missing values or significant outliers in the dataset, so all data was usable for analysis without the need for additional filtering.

The class distribution across the dataset showed a clear pattern with informative groupings for training the classification model. The training data maintained the same proportions as the overall data, ensuring a fair representation of each rating level category during the model training process. Similarly, for the testing data, stratification ensured that the model evaluation was conducted on a dataset that reflected the true distribution of the smartphone population in the market.

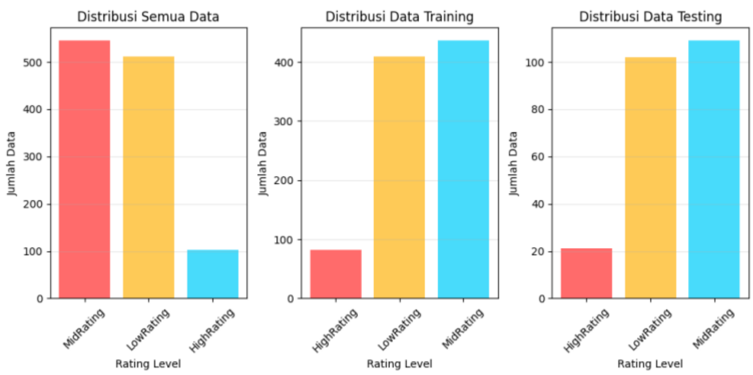

Figure 1: Class distribution in overall data, training data, and testing data

3.2. Naive Bayes Classification Results

The Naive Bayes Categorical algorithm was implemented using the scikit-learn library on a preprocessed dataset. The model was trained on the training data and then evaluated for its predictive performance on testing data, which was not used during the training phase. The model evaluation results on the testing data showed an accuracy of 77.16 percent, meaning the model correctly predicted the smartphone popularity category for 77.16 percent of the 232 test samples. The precision metric reached 0.7737, indicating that of all positive predictions made by the model, approximately 77.37 percent were correct. The recall metric reached 0.7716, indicating that of all samples that were actually

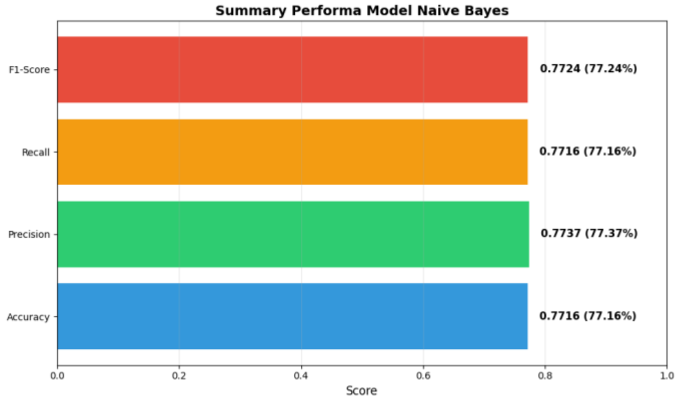

Figure 2: Summary of Naive Bayes model performance with accuracy, precision, recall, and F1-score metrics

Five-fold cross-validation was performed to obtain a more robust estimate of model performance and reduce bias from specific data splits. The cross-validation results showed an average accuracy of 76.53 percent with a standard deviation of 0.0224. This low standard deviation indicates that the model has stable and consistent performance across folds, with no extreme performance variations between folds. The five-fold accuracy values, in sequence, are as follows: fold 1 reached 76.72 percent, fold 2 reached 77.16 percent, fold 3 reached 72.41 percent, fold 4 reached 77.16 percent, and fold 5 reached 79.22 percent. This consistent performance confirms that the Naive Bayes model has good generalization to previously unseen data.

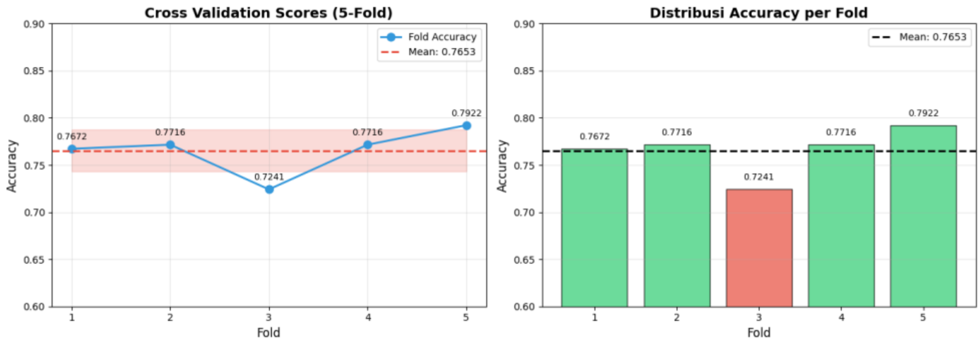

Figure 3: Cross-validation scores show stable performance with line plot and bar chart per fold

Table 1: Summary of Evaluation Metrics of Naive Bayes Model for Smartphone Popularity Classification

Metrik Evaluasi	Nilai	Interpretasi
Akurasi (*Testing*)	77,16%	Model memprediksi dengan benar 77,16% sampel *testing*
Precision	0,7737	77,37% dari prediksi positif adalah benar
Recall	0,7716	Model mengidentifikasi 77,16% dari kelas positif aktual
F1-Score	0,7724	Keseimbangan baik antara *precision* dan *recall*
Akurasi CV (*Mean*)	76,53%	Performa stabil dengan variasi lintas *fold*
Std Dev CV	0,0224	Variasi performa sangat kecil, model konsisten

3.3. Feature Importance Analysis

Feature importance analysis was performed using the Mutual Information Score to identify the features that contribute most to the classification of smartphone popularity categories. The Mutual Information Score measures the statistical dependence between each predictor feature and the target variable of the popularity class. The higher the MI Score, the more important the feature is in distinguishing and predicting popularity categories. The results of the feature importance analysis indicate that not all eight features contribute equally to the model's prediction accuracy.

Table 2 : Feature Importance using Mutual Information Score

Fitur	MI Score	Kontribusi (%)	Rank
ram_cat	0,2844	32,51%	1
ppi_cat	0,2255	25,78%	2
screen_type_cat	0,1844	21,09%	3

android_level	0,0656	7,50%	4
water_level	0,0532	6,08%	5
screen_size_cat	0,0303	3,47%	6
ip_level	0,0207	2,37%	7
battery_cat	0,0105	1,20%	8

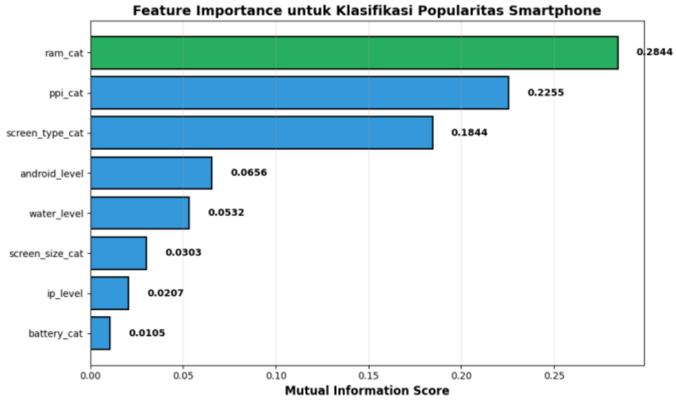

Figure 4: Feature importance bar chart shows the MI score for each feature.

RAM capacity (ram_cat) demonstrated the highest dominance, with an MI score of 0.2844 and a 32.51 percent contribution to predicting smartphone popularity. This finding indicates that Indonesian consumers place significant importance on RAM capacity when determining smartphone popularity, as RAM significantly impacts multitasking performance and application execution speed. Screen pixel density (ppi_cat) was the second most important feature, with an MI score of 0.2255 and a 25.78 percent contribution, indicating that screen visual quality is a crucial consideration for consumers in assessing smartphone popularity. Screen type (screen_type_cat) was the third most important feature, with an MI score of 0.1844 and a 21.09 percent contribution. These three features combined contributed 79.38 percent of the total feature importance, indicating that they are key drivers in classifying smartphone popularity.

Android operating system level (android_level), with a 7.50 percent contribution, demonstrated a more moderate influence than the top three features. This feature is still relevant but less dominant in determining the popularity category. Water resistance level (water_level), with a contribution of 6.08 percent, had a smaller impact, likely because not all consumers prioritize water resistance in their purchasing decisions. Features with very low contributions, such as category screen size (3.47 percent), IP rating (2.37 percent), and category battery capacity (1.20 percent), indicate that these features have minimal influence on popularity classification. This is likely because screen size has become standard in most modern smartphones, resulting in less differentiation between categories, while IP rating and category battery capacity may be less important considerations for most consumers.

3.4. Performance Analysis per Rating Category

A more detailed analysis of model performance across each rating level category revealed significant differences in model predictive ability. The LowRating category performed the best with a precision of 0.837, a recall of 0.804, and an F1-score of 0.820, indicating that the model was highly accurate in identifying low-rated smartphones. The MidRating category, which is the majority category, showed a precision of 0.750, a recall of 0.771, and an F1-score of 0.760, indicating solid performance with a recall capability slightly higher than precision. The HighRating category showed the lowest performance with a precision of 0.591, a recall of 0.619, and an F1-score of 0.605, indicating that

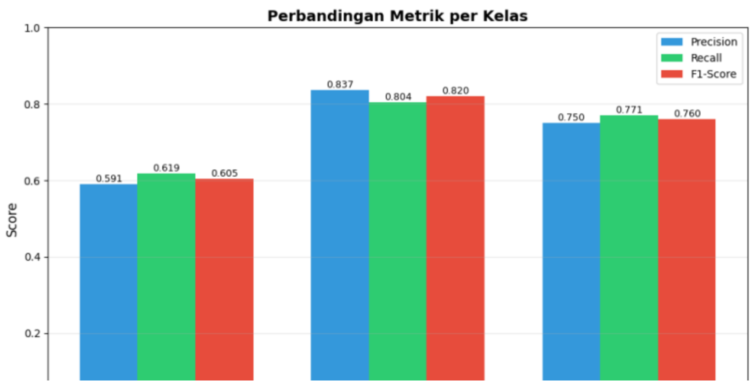

Figure 5: Comparison of precision, recall, and F1-score metrics for each rating level category

The confusion matrix provides in-depth insights into the model's misclassification patterns. Of the 21 HighRating samples in the testing data, the model correctly predicted 13 samples (correct predictions), but 8 samples were predicted as MidRating. Of the 102 LowRating samples, the model predicted 82 samples correctly, although 20 samples were still predicted as MidRating. Of the 109 MidRating samples,

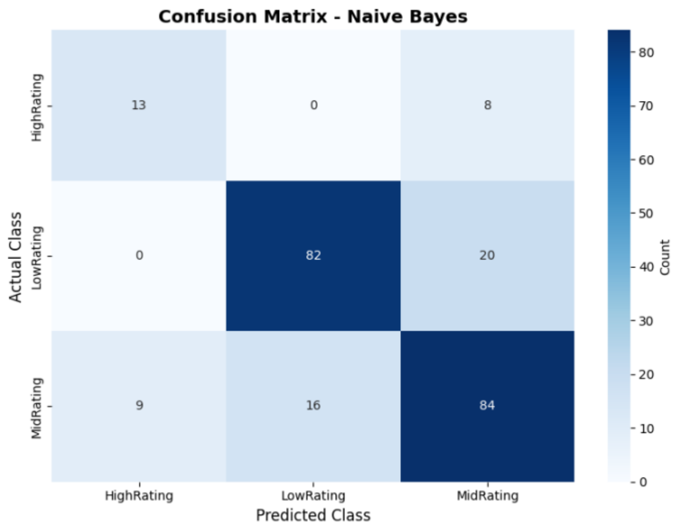

Figure 6: Confusion matrix menunjukkan prediksi vs actual class distribution

the model correctly predicted 84 samples, but 9 samples were predicted as HighRating and 16 samples were predicted as LowRating. This pattern suggests that the main misclassifications occurred between adjacent categories (HighRating ↔ MidRating and MidRating ↔ LowRating), which makes sense considering that all three categories are ordinalizations of the user rating continuum.

3.5. Discussion

The results show that the Naive Bayes algorithm successfully classified smartphone popularity levels with 77.16 percent accuracy on the test data and 76.53 percent accuracy on the five-fold cross-validation. Performance stability, with a standard deviation of only 0.0224, indicates that the model has good generalization and is not overfitting. The finding that RAM capacity, screen pixel density, and screen type are the three most important features (79.38 percent contribution) provides valuable insights for smartphone manufacturers in their product development strategies, while for consumers, it suggests that focusing on these specifications is a rational decision when choosing a popular smartphone.

The model's prediction patterns, which show different performance in each rating category, provide opportunities for further development. The model's preference for the MidRating category as a result of prediction conservation, can be addressed through class balancing techniques, weighted classification, or the use of more sophisticated algorithms such as ensemble methods or neural networks capable of modeling non-linear relationships between features. The concentrated contribution of feature importance to the first three features opens up opportunities for feature selection and model simplification without significantly sacrificing accuracy. Further research could test these techniques to improve model performance, especially in the HighRating category, as well as integrate additional features such as price, brand reputation, and design preferences for a more comprehensive understanding of smartphone popularity drivers in the Indonesian market.

4. Conclusion

This study successfully implemented the Naive Bayes algorithm to classify smartphone popularity levels based on eight technical specification features. The model, trained on 1,159 smartphone data sets, achieved 77.16 percent accuracy on the test data with a precision of 0.7737, a recall of 0.7716, and an F1-score of 0.7724. Five-fold cross-validation yielded an average accuracy of 76.53 percent with a standard deviation of 0.0224, indicating stable performance and good generalization. Thus, the Naive Bayes algorithm proved effective in predicting smartphone popularity categories based on available technical specifications.

Feature importance analysis revealed that RAM capacity (32.51 percent), display pixel density (PPI) (25.78 percent), and display type (21.09 percent) were the three most important features in the classification, and together contributed 79.38 percent of the total contribution. These findings indicate that Indonesian consumers prioritize performance, visual quality, and display technology when assessing smartphone popularity, providing practical implications for manufacturers in product development strategies and consumers in purchasing decisions. Further research could explore the use of alternative algorithms such as Random Forest or Neural Networks and the integration of additional features such as price and brand to improve accuracy and provide a more comprehensive understanding of the drivers of smartphone popularity in the Indonesian market.

References

[1] S. Rai, "Factors influencing online purchase intention of smartphones," *Cogent Business & Management*, vol. 10, no. 1, hlm. 1–15, 2023, doi: 10.1080/23311975.2018.1496612.

[2] J. R. Vergara dan P. A. Estévez, "A review of feature selection methods based on mutual information," *Neural Comput Appl*, vol. 24, no. 1, hlm. 175–186, 2014, doi: 10.1007/s00521-013-1368-0.

[3] S. Rai, "Factors influencing online purchase intention of smartphones," *Cogent Business & Management*, vol. 10, no. 1, hlm. 1–15, 2023, doi: 10.1080/23311975.2018.1496612.

[4] G. Brown, A. Pocock, M. J. Zhao, dan M. Luján, “Conditional likelihood maximisation: A unifying framework for information theoretic feature selection,” *Journal of Machine Learning Research*, vol. 13, hlm. 27–66, 2012, [Daring]. Tersedia pada: https://www.sciencedirect.com/science/article/pii/S0031320313002227

[5] N. van den Berg, “Driving factors behind product ratings in the smartphone market,” Rotterdam, 2021. [Daring]. Tersedia pada: https://thesis.eur.nl/pub/60845/

[6] Y. Kim dan J. Lee, “Factors affecting young customers’ smartphone purchase intention,” *Front Psychol*, vol. 13, hlm. 1–12, 2022, doi: 10.3389/fpsyg.2022.946780.

[7] W. Zhang, M. Xu, Y. Feng, Z. Mao, dan Z. Yan, “The Effect of Procrastination on Physical Exercise among College Students—The Chain Effect of Exercise Commitment and Action Control,” *International Journal of Mental Health Promotion*, vol. 26, no. 8, hlm. 611–622, 2024, doi: 10.32604/ijmhp.2024.052730.

[8] T. Shintarani, “Sentiment Analysis of Mobile Provider Application Reviews Using Naive Bayes,” *Sinkron*, vol. 9, no. 2, hlm. 100–110, 2024, [Daring]. Tersedia pada: https://jurnal.polgan.ac.id/index.php/sinkron/article/view/13469

[9] S. Goswami dan V. Chouhan, “A Study on Factors Affecting Smartphone Purchase Decisions in India,” *International Journal of Social Science and Humanities Research*, vol. 7, no. 8, hlm. 315–324, 2020, [Daring]. Tersedia pada: https://ijsshr.in/v7i8/43.php

[10] L. J. Anreaja, N. N. Harefa, J. G. P. Negara, V. N. H. Pribyantara, dan A. B. Prasetyo, “Naive Bayes and Support Vector Machine Algorithm for Sentiment Analysis Opensea Mobile Application Users in Indonesia,” *JISA(Jurnal Informatika dan Sains)*, vol. 5, no. 1, hlm. 62–68, Jun 2022, doi: 10.31326/jisa.v5i1.1267.

[11] S. Situju, N. Nur, dan N. Halal, “Analisis Penerapan Mutual Information pada Klasifikasi Status Studi Mahasiswa Menggunakan Naïve Bayes,” *Journal of Applied Computer Science and Technology*, vol. 6, no. 1, hlm. 23–28, Jun 2025, doi: 10.52158/jacost.v6i1.1106.

[12] W. Wijiyanto, A. I. Pradana, S. Sopingi, dan V. Atina, “Teknik K-Fold Cross Validation untuk Mengevaluasi Kinerja Mahasiswa,” *Jurnal Algoritma*, vol. 21, no. 1, Mei 2024, doi: 10.33364/algoritma/v.21-1.1618.

[13] A. Z. Rizquina dan C. I. Ratnasari, “Implementasi Web Scraping untuk Pengambilan Data Pada Website E-Commerce,” *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 4, hlm. 377–383, Okt 2023, doi: 10.47233/jteksis.v5i4.913.

[14] N. N. M, S. Z. Harahap, dan I. Irmayanti, “Implementasi Metode Naive Bayes dan Neural Network Untuk Menentukan Minat Masyarakat Pada Handphone Samsung,” *INFORMATIKA*, vol. 12, no. 3, hlm. 602–608, Agu 2024, doi: 10.36987/informatika.v12i3.6163.

[15] A. Widianti dan I. Pratama, “PENANGANAN MISSING VALUES DAN PREDIKSI DATA TIMBUNAN SAMPAH BERBASIS MACHINE LEARNING,” *Rabit : Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 9, no. 2, hlm. 242–251, Jul 2024, doi: 10.36341/rabit.v9i2.4789.

[16] F. R. Aftha Harianto, Z. Alawi, dan I. A. Sa’ida, “PENGARUH KOMPOSISI SPLIT DATA PADA AKURASI KLASIFIKASI PENDERITA DIABETES MENGGUNAKAN ALGORITMA MACHINE LEARNING,” *Jurnal Sistem Informasi dan Informatika (Simika)*, vol. 8, no. 1, hlm. 36–44, Jan 2025, doi: 10.47080/simika.v8i1.3663.