International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I1P121

Volume 73 Issue 1, 250-254, January 2025
© 2025 Seventh Sense Research Group®

Original Article

Comparing Multiple Ensemble Classifiers for E-commerce Recommendation System

Ignatius Michael Dinata[1], Vincentius Loanka Sinaga[2], Antoni Wibowo[3]

[1,2,3]BINUS Graduate Program - Master of Computer Science, Bina Nusantara University Jakarta, Indonesia.

[1]Corresponding Author: ignatius.dinata@binus.ac.id

Received: 21 June 2024 Revised: 10 December 2024 Accepted: 07 January 2025 Published: 31 January 2025

Abstract *- E-commerce recommendation systems face challenges with data sparsity, which impacts the accuracy of user engagement and product recommendations. This research evaluates the performance of multiple Machine Learning classifiers, including Extreme Gradient Boosting (XGBoost), K-Nearest Neighbor (KNN), Random Forest, and Support Vector Machine (SVM), with and without the use of the Synthetic Minority Over-sampling Technique (SMOTE). The results indicate that XGBoost with SMOTE achieves the highest performance across all evaluation metrics (accuracy, precision, recall, and F1-score), scoring 0.97 in each metric. Random Forest also performs well, achieving 0.95 for all metrics, while KNN scores moderately at 0.83. SVM shows the lowest performance, with an accuracy of 0.59 and an F1-score of 0.51. These findings highlight the robustness of XGBoost combined with SMOTE in handling imbalanced data and improving prediction accuracy in e-commerce recommendation systems, offering valuable insights for researchers and practitioners in this domain.*

Keywords *- XGBoost, KNN, Random Forest, SVM, E-Commerce.*

1. Introduction

In the rapidly evolving E-Commerce landscape, recommendation systems have become crucial in improving user experience and driving business growth. As consumers are inundated with an ever-growing variety of products and services, personalized recommendations have become a powerful tool to guide users through the digital marketplace. The data obtained will also increase with the increasing number of E-Commerce users. However, not all this data can be used to guide the purchasing habits of e-commerce users.

Therefore, a Data Mining process is needed so that the data can provide value and meaning effectively and increase the intentions of E-Commerce users. Based on predictive analysis of E-Commerce user habits, this paper will provide recommendations for more effective methods in predicting E-Commerce user behaviour. Throughout the year, recommendation systems attempt to solve a cold start problem and data sparsity problem while increasing the accuracy of a recommendation. Multiple algorithms were tested and evolved to the current recommendation system algorithm that is available right now. Commonly used methods to tackle these problems are Content-Based Filtering (CBF) and Collaborative Filtering (CF).

However, these methods have their own respective downsides, with that in mind, researchers tend to combine multiple methods, hence creating hybrid methods which are expected to perform better than the traditional methods. However, regardless of its downside, the CF approach is still the most implemented technique. Many approaches are used in different research, which will be discussed in the next section. The authors compare how different ensemble approaches perform on a given dataset.

The authors hope that the availability of this research can help more researchers develop better recommendation systems by using the information related to the comparison results of each machine learning prediction method in this paper. This research paper contains the evaluation and results of several ensemble classifiers with and without the SMOTE algorithm. The author applies Extreme Gradient Boosting (XGBoost), K-nearest neighbor (KNN), Random Forest, and Support Vector Machine (SVM) in this study.

However, not all classifiers are included; the algorithms mentioned are the most widely applied in recommendation systems and focus on machine learning algorithms only. The subsequent sections of this paper are structured as follows: Section 2 delves into related research in this field, followed by a presentation of the methodology to be employed in Section 3. The methodology's implementation outcomes will be elucidated in Section 4 and further examined. Finally, Section 5 will encapsulate the concluding findings from this paper.

2. Related Works

In previous research, Anitha and Kalaiarasu proposed an Optimized Machine Learning-based Collaborative Filtering (OMLCF) algorithm [1]. They implemented SVM for items' classification and filtered out items users disliked, reducing the recommended commodities. They concluded that the proposed approach of SVM-IACO-CF (Support Vector Machine-Improved Ant Colony Optimization based Collaborative Filtering) classifier shows a better predictive accuracy of 20% than K-RecSys-CF and SVM-CF. Another research done to increase the accuracy of a recommendation system involves the usage of one of the algorithms discussed, the XGBoost algorithm [2]. Research conducted by Yutong shows that when XGBoost is used to predict the purchasing behaviours of e-commerce platform consumers, it can improve the method's performance and obtain a better prediction effect than the Random Forest Algorithm. [3]. The result here shows that compared to Random Forest (RF), the result is 0.00%-0.06% better.

In research conducted by Widayanti, the authors proposed a hybrid Collaborative Filtering-Content-Based Filtering (CF-CBF) approach to enhance the efficacy of recommendation systems where historical user interaction data are collected to build the model [4]. The experiment aims to improve recommendation accuracy and personalization. The results show that the hybrid approach significantly outperformed CF and CBF with a relevance accuracy rate of 90% to 80% and 75%, respectively, and a performance level of 95% to 85% and 80%, respectively. In another research study, predictions were made using machine learning methods, specifically Extreme Gradient Boosting (XGBoost). XGBoost is an algorithm dominating the applied Machine Learning fields. It is also a scalable, distributed Gradient-Boosted Decision Tree (GBDT) Machine Learning Library.

XGBoost uses more accurate approximations to find the best tree model. [2]. In research conducted by Yutong, an experiment was carried out to compare XGBoost and Random Forest. XGBoost exhibited a higher accuracy level than Random Forest, with an improvement of 0.06% [5]. In another research by Nuanmeesri and Sriurai, the authors developed a second-hand car recommender system model that uses SMOTE with the Random Forest to address data imbalance [6]. SMOTE oversamples the minority classes, increasing the dataset size by 400%. Random Forest was applied to make tree decisions for car recommendations based on car specifications and consumer profiles.

The model was evaluated using a 10-fold cross-validation and shows an accuracy of 98.84%, precision of 98.89%, recall of 98.80%, and an F1 score of 98.80%. These results outperform the model, which only uses Random Forest. The research that was conducted by Lubis, the integration of KNN with SMOTE, was explored and compared with other methods such as AdaBoost and XGBoost [7]. The findings revealed that the accuracy of KNN without SMOTE was only 64%, whereas that of KNN with SMOTE was 77%. Additionally, combining SMOTE with XGBoost and KNN produced the best model, achieving an accuracy of 88%. These results underscore the importance of data balancing techniques prior to implementing boosting algorithms. Based on the research conducted before and shown above, the authors evaluated whether SMOTE could provide even better results if implemented on several ensemble classifications (XGBoost, KNN, Random Forest, and SVM).

3. Materials and Methods

Figure 1 provides an overview of the experiment's process, incorporating SMOTE and supervised learning models for predicting recommendation items. The initial step in the data processing involves extracting the dataset's features using the “Pandas” library. Subsequently, feature selection is performed due to the inclusion of string features. Consequently, fitting and transformation are executed for these features. After the fitting and transformation process, the following process will be divided into two processes. The first process uses SMOTE, and the other data will be directly fed to the models. The prediction output will be compared and evaluated with the actual labels of the test dataset.

3.1. Dataset

The dataset used in this paper is obtained from Kaggle [15], where the dataset is the Customer Shopping Trends Dataset. This dataset offers valuable insights into consumer behavior and purchasing patterns. This dataset consists of 21 columns with 52954 data inside.

The columns include no, CustomerID, Gender, Location, Tenure_months, and Transaction_ID. Transaction_Date, Product_SKU, Product_Description, Product_Category, Avg_Price, GST, Offline_Spend, Online_Spend, Month, Coupon_Code, Coupon_Status, Quantity, Date, and Discount_pct.

Table 1. Imbalance count in product category

Column	Data Counted	Column	Data Counted
Nest-USA	14013	Nest-Canada	317
Office	6513	Bottles	270
Apparel	18126	Gift Cards	160
Bags	1882	More Bags	46
Drinkware	3483	Backpacks	89
Lifestyle	3092	Housewares	125
Waze	554	Android	47
Headgear	771	Nest	2205
Fun	160	Accessories	235
Notebooks & Journals	750	Notebooks	12
Google	105		

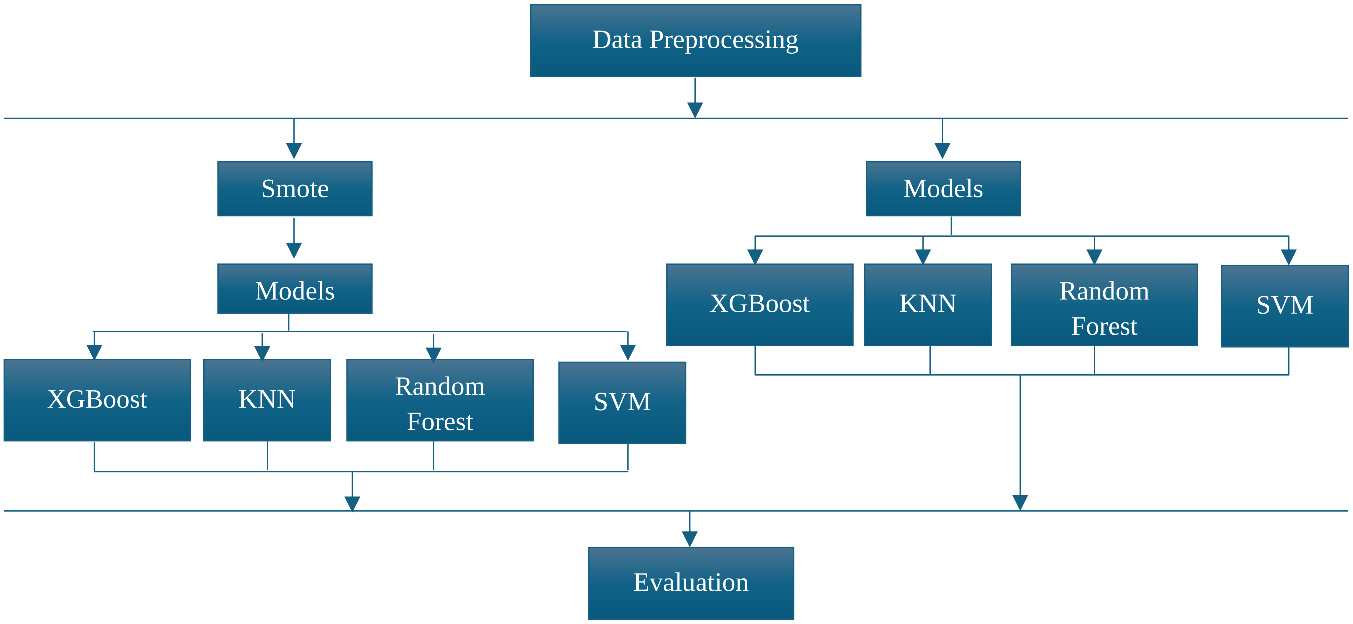

Fig. 1 Research workflow

However, some features are not used because the prediction has no relevance. In this research, the "product category" is used as the predicted item.As shown in the Table. 1, there is a data imbalance in the dataset provided; only 12 data are categorized as Notebooks. Meanwhile, Nest_USA and Apparel have data exceeding 14000. Processing this data will cause the models not to perform well. The precision and accuracy will be low. Since imbalanced datasets might cause such problems, resampling methods, such as SMOTE for over-sampling, are used for this dataset with the expectation that the model will perform better. The dataset will be divided into 80% data for the training process, and 20% will be divided for the testing process using a random state value of 42.

3.2. Device Specification

For conducting this research, the authors use a local device with the following specifications:

- CPU: Intel Core i7-9750 HF @260GHz
- RAM: 32GB

3.3. Pre-Processing

The pre-processing carried out in this study involves handling missing values, feature selection, and categorical data encoder. Firstly, the authors select columns related to consumer data which are CustomerID, Gender, Location, Tenure_Month, Product_Category, GST, Offline_Spend, Online_Spend, Month, Coupon_Code, and Discount_pct.

Next, the author's approach to handling missing values is to discard rows containing empty values. This is done due to the consideration that the rows with empty values, which are 400 rows, far outnumbered by the total number of rows in the dataset, which is 52,999. Lastly, the next process is to encode categorical data after all missing values are handled. The columns encoded are Gender, Location, Product_Category, Coupon_Code, and Discount_pct.

3.4. Synthetic Minority Over-sampling Technique

The Synthetic Minority Over-Sampling Technique (SMOTE) Algorithm employs an oversampling approach to rebalance the original training set by generating synthetic data using a k-nearest neighbor algorithm. SMOTE is initiated by randomly selecting data from the minority class and implementing k-nearest neighbors. As indicated by prior research [8], the technique focuses on the values and relationships of the features rather than analyzing the data points. Applying SMOTE to the dataset significantly augmented the number of synthetic cases, escalating from 324 to 1737. The results create an increased ratio of outerwear classes, achieving a 1:1 balance.

3.5. Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is a dominant algorithm in applied Machine Learning. Additionally, it serves as a scalable, distributed library for Gradient-Boosted Decision Trees (GBDT) in Machine Learning. XGBoost employs more precise approximations to identify the optimal tree model [2].

3.6. K-Nearest Neighbor

The K-Nearest Neighbor (KNN) algorithm, a straightforward supervised Machine Learning approach, is widely applied to address classification and regression problems. The algorithm operates by determining the distances between a query and every instance in the data, selecting the K instances closest to the query, and subsequently casting votes for the label with the highest frequency in the context of classification or computing the average labels in the case of regression [9].

3.7. Random Forest

The Random Forest is a supervised machine learning algorithm extensively applied in Classification and Regression problems. It is regarded as an accessible algorithm that often yields excellent results even without hyperparameter tuning, as asserted by Niklas Donges, an AI expert and founder of AM Software [10]. In the context of Random Forest, the concept of Ensemble arises, involving the merge of multiple models. Ensemble methods include two types: Bagging and Boosting. Random Forest exemplifies Bagging, whereas XGBoost serves as an illustration of Boosting [11].

3.8. Support Vector Machine

The Support Vector Machine (SVM) is a supervised machine learning method commonly used for classification, regression, and outlier detection [12]. Several kernels, namely the Radial Basis Function (RBF), Linear (LIN), Sigmoid (SIG), and Polynomial (POL), are commonly employed in SVM to address the given problem. In machine learning, "kernel" typically denotes the kernel trick—a method to adapt a linear classifier for solving non-linear problems. This approach becomes particularly relevant when aiming for a linear separation of data in regression tasks [13].This paper will use RBF instead of LIN, SIG, and POL since the RBF adapts well to non-linear data. RBF itself is the most popular kernel among all kernels in SVM. Executing RBF SVM involves mapping the input data into a higher-dimensional feature space, enabling a hyperplane for class division. [14].

3.9. Evaluation Method

The evaluation will encompass existing models' accuracy, precision, recall, and f1-score. It is calculated by measuring the ratio of correctly detected and predicted instances for accuracy. Precision, determined by the division of projected positive labelled data comparisons by the total number of correctly labelled data, signifies clearer model results as it approaches 1. Recall, analogous to precision, computes the ratio between anticipated positively labelled data and the total available data, with increased proximity to 1 indicating enhanced model clarity. The f1-score, representing the sum of precision and recall, assesses the impact of False Positives and False Negatives. The formulas for accuracy, precision, recall, and f1-score [16] are detailed below:

$$Accuracy = \frac{\alpha+ \beta}{\alpha+ \gamma+ \omega+ \beta} \tag{1}$$

$$Precision = \frac{\alpha}{\alpha+ \gamma} \tag{2}$$

$$Recall = \frac{\alpha}{\alpha+ \omega} \tag{3}$$

$$F1- \ Score = \frac{2\ X\ Precision\ X\ Recall}{Precision\ X\ Recall} \tag{4}$$

4. Results and Discussion

Before discussing further, note that during the experimentation of this research, no parameters of the models were tuned, and only default or standard values were used. This is to make sure that the results obtained are the standard or general results. With that in mind, the table below shows the accuracy, precision, recall, and f1-score results for the earlier methods.

From Table 2, it is observed that XGBoost combined with SMOTE achieved the highest performance across all evaluation metrics-accuracy, precision, recall, and F1-score-reaching a value of 0.97. This approach showed an improvement of 0.02 compared to using XGBoost alone. Random Forest demonstrated consistent results, maintaining a score of 0.95 across all metrics regardless of the SMOTE application. KNN showed significant improvement with SMOTE, increasing its F1-score from 0.71 to 0.83. In contrast, SVM performed the worst among the methods, with an F1-score of 0.51 and an accuracy of 0.59, highlighting its challenges in managing imbalanced datasets. XGBoost and Random Forest exhibit minimal performance improvement after applying SMOTE, largely due to the inherent characteristics of these algorithms. XGBoost, being a gradient boosting variant, employs a customizable loss function that directly addresses data imbalance, making it naturally robust in such scenarios. Similarly, Random Forest mitigates the impact of imbalance by constructing multiple decision trees and aggregating their predictions through averaging or voting, which reduces the effect of outliers and underrepresented minority classes. In contrast, algorithms like KNN and SVM, which rely on the local data distribution, benefit significantly from SMOTE, as it enhances the representation of minority classes in the dataset.

Table 2. Comparison between machine learning methods

Comparison	Accuracy	Precision	Recall	F1-Score
XGBoost	0.95	0.95	0.95	0.95
XGBoost + SMOTE	0.97	0.97	0.97	0.97
Random Forest	0.95	0.95	0.95	0.95
RandomForest + SMOTE	0.95	0.95	0.95	0.95
KNN	0.73	0.71	0.73	0.71
KNN + SMOTE	0.83	0.83	0.83	0.83
SVM	0.59	0.68	0.59	0.51
SVM + SMOTE	0.59	0.68	0.59	0.51

5. Conclusion

In conclusion of the research, XGBoost combined with SMOTE achieved the highest performance, attaining accuracy, precision, recall, and F1-score values of 0.97. This marks a modest improvement of 0.02 points compared to XGBoost without SMOTE, highlighting the ability of SMOTE to enhance predictions.

Random Forest maintained stable performance, achieving a consistent score of 0.95 across all metrics, regardless of whether SMOTE was applied, demonstrating its robustness in handling data imbalance independently. KNN, however, saw a significant boost with SMOTE, as its F1-score increased from 0.71 to 0.83, reflecting the effectiveness of SMOTE in enhancing the representation of minority classes, which is crucial for local distribution-based algorithms like KNN. In contrast, SVM performed the worst, with an F1-score of 0.51 and an accuracy of 0.59, both with and without SMOTE, indicating its limited capacity to manage imbalanced datasets, even with improved minority class representation.

5.1. Future Works

In the future, more ensemble classifiers and algorithms could be used compared to the algorithms used in this paper and the usage of SMOTE in their implementations. The methods examined in this work might be applied to a different dataset from fields outside the recommendation system and analysed further. The author also recommends implementing XGBoost with Content-Based Filtering as an algorithm that can increase the relevance of the system's recommendation list. In addition, since the current era has started using Deep Learning to make predictions. The author recommends using Deep Learning to make predictions while still paying attention to the model's performance.

References

[1] J. Anitha, and M. Kalaiarasu, "RETRACTED ARTICLE: Optimized Machine Learning Based Collaborative Filtering (OMLCF) Recommendation System in E-Commerce," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 6, pp. 6387-6398, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Tianqi Chen, and Carlos Guestrin, "XGBoost: A Scalable Tree Boosting System," *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, pp. 785-794, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[3] Sunny Sharma, Vijay Rana, and Manisha Malhotra, "Automatic Recommendation System Based on Hybrid Filtering Algorithm," *Education and Information Technologies*, vol. 27, no. 2, pp. 1523-1538, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Riya Widayanti et al., "Improving Recommender Systems Using Hybrid Techniques of Collaborative Filtering and Content-Based Filtering," *Journal of Applied Data Sciences*, vol. 4, no. 3, pp. 289-302, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Peiyi Song, and Yutong Liu, "An XGBoost Algorithm for Predicting Purchasing Behaviour on E-Commerce Platforms," *Technical Bulletin*, vol. 27, no. 5, pp. 1467-1471, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Sumitra Nuanmeesri, and Wongkot Sriurai, "Second-Hand Cars Recommender System Model Using the Smote and the Random Forest Technique," *Journal of Xi'an University of Architecture and Technology*, vol. 12, no. 4, pp. 3687-3695, 2020. [Google Scholar]

[7] Adyanata Lubis et al., "Leveraging K-Nearest Neighbors with Smote and Boosting Techniques for Data Imbalance and Accuracy Improvement," *Journal of Applied Data Sciences*, vol. 5, no. 4, pp. 1625-1638, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Alberto Fernandez et al., "Smote for Learning from Imbalanced Data: Progress and Challenges, Marking The 15-Year Anniversary," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863-905, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[9] Amit Pandey, and Achin Jain, "Comparative Analysis of KNN Algorithm using Various Normalization Techniques," *International Journal of Computer Network and Information Security*, vol. 9, no. 11, pp. 36-42, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[10] Niklas Donges, Random Forest Algorithm: A Complete Guide, Built In, 2022. [Online]. Available: https://builtin.com/data-science/random-forest-algorithm

[11] Sajib Kabiraj et al., "Breast Cancer Risk Prediction using XGBoost and Random Forest Algorithm," *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, pp. 1-4, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Haifeng Wang, and Dejin Hu, "Comparison of SVM and LS-SVM for Regression," *2005 International Conference on Neural Networks and Brain*, Beijing, pp. 279-283, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[13] Danial Jahed Armaghani et al., "Examining Hybrid and Single SVM Models with Different Kernels to Predict Rock Brittleness," *Sustainability*, vol. 12, no. 6, pp. 1-17, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Arepalli Peda Gopi et al., "Classification of Tweets Data Based on Polarity Using Improved RBF Kernel of SVM," *International Journal of Information Technology*, vol. 15, no. 2, pp. 965-980, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[15] R. Jackson Divakar, Online Shopping Dataset, Kaggle, 2024. [Online]. Available: https://www.kaggle.com/datasets/jacksondivakarr/online-shopping-dataset

[16] Bay Vo et al., "Efficient Methods for Clickstream Pattern Mining on Incremental Databases," *IEEE Access*, vol. 9, pp. 161305-161317, 2021. [CrossRef] [Google Scholar] [Publisher Link]