ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

An Explainable AI-Based Recommendation System for E-Commerce Platforms

Shalini M. R.[1]; Nayana K.[2]

[1]PG Student, M.tech-Deep Learning,FET- GM University, Davanagere, Karnataka, 577006
[2]Professor, PG Director, M.tech-Deep Learning,FET- GM University, Davanagere, Karnataka.577006

Publication Date: 2026/08/24

Abstract: The widespread expansion of online retail has transformed recommender systems into an essential component of modern e-commerce platforms by guiding customers toward products that meet with their interests and purchasing behavior. Recent developments in machine learning have greatly enhanced the accuracy of recommendation models; however, several practical challenges still remain, many existing solutions provide little explanation of how individual recommendations are generated. This lack of interpretability can reduce user confidence and restrict the adoption of AI-based recommendation models in applications where transparent decision-making is required. To address this challenge, the present study introduces an Explainable Artificial Intelligence (XAI)-based recommendation method that brings together feature engineering, the Synthetic Minority Oversampling Technique(SMOTE), Extreme Gradient Boosting(XGBoost), and SHapley Additive exPlanations(SHAP). The workflow begins by preprocessing user–product interaction data and constructing informative features, including user activity, product popularity, and price buckets. The class imbalance problem is then handled using SMOTE before training an XGBoost classifier to estimate recommendation probabilities. To make the prediction process easier to understand, SHAP explains the contribution of each feature to individual recommendation outcomes at both the global and local levels. Experimental evaluation demonstrates that the proposed approach achieves strong predictive performance in terms of Accuracy, Precision, Recall, F1-score, and ROC-AUC while maintaining a high level of model interpretability. By combining reliable prediction with meaningful explanations, the proposed recommendation approach strengthens user trust and offers a practical solution for deployment in intelligent e-commerce environments.

Keywords: *Explainable Artificial Intelligence (XAI), Recommendation System, E-Commerce, XGBoost, SHAP, SMOTE, Machine Learning.*

How to Cite: Shalini M. R.; Nayana K. (2026) An Explainable AI-Based Recommendation System for E-Commerce Platforms. *International Journal of Innovative Science and Research Technology*, 11(8), 1527-1538. https://doi.org/10.38124/ijisrt/26aug580

I. INTRODUCTION

The rapid growth of the e-commerce platforms has expanded significantly the volume and diversity of products available to consumers. As online marketplaces continue to expand, customers are typically presented with thousands of products, making it increasingly difficult to identify items that suit their preference and purchasing requirements. Recommendation systems are widely regarded as an effective solution by analyzing user behaviour and product information to deliver personalized product suggestions, thereby improving customer satisfaction, engagement, and sales performance. Traditional recommendation approaches, including content-based filtering, collaborative filtering, and hybrid recommendation models, have demonstrated considerable success in generating personalized recommendations. Recent advancements in machine learning(ML) have further enhanced recommendation quality by learning complex relationships from historical user interactions. Several supervised learning(SL) techniques, including Decision Trees(DT), Random Forests(RF), Support Vector Machines (SVM), and XGBoost(XGB) have shown promising performance in predicting user preferences. Despite the significant advances made in recommender systems, many existing models still operate as black-box solutions, producing accurate predictions without revealing how those predictions are derived. As a result, users often find it difficult to interpret the reasoning behind the recommended products, particularly in applications where transparency and accountability are essential. When the recommendation process cannot be clearly interpreted, confidence in the system may decline, reducing users willingness to rely on automated suggestions. To overcome this constraint, Explainable Artificial Intelligence (XAI) has proved to be an effective way to make machine learning(ML) models become more transparent without sacrificing predictive capability. By highlighting how individual features influence recommendation outcomes, explainability techniques help users and developers better understand model behavior, ultimately improving trust, accountability, and acceptance of AI-driven recommendations. In addition

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

to the challenge of interpretability, recommendation datasets commonly contain missing information, noisy observations, redundant features, and imbalanced class distributions. If these challenges remain unaddressed during data preparation, they can negatively affect the learning process and lead to lower recommendation accuracy. Furthermore, raw user-product interaction data often fail to capture meaningful behavioral patterns without appropriate feature engineering. Addressing these challenges requires an integrated framework that combines effective data preprocessing, feature extraction, class balancing, accurate prediction, and interpretable decision-making. Based on these identified limitations, the study puts forward an Explainable AI-based recommendation framework that consolidates data preprocessing, feature engineering, Synthetic SMOTE, XGBoost and SHAP. The proposed framework first preprocesses user-product interaction data and derives informative features including user activity, product popularity, and price buckets. SMOTE is then employed to balance the training dataset by generating synthetic minority class samples. Subsequently, an XGBoost classifier predicts recommendation probabilities, while SHAP provides transparent explanations by quantifying the contribution of each feature in generating individual recommendations.

➢ *The Main Highlights of this Research are Presented Below:*

- An integrated Explainable AI framework combining feature engineering, SMOTE, XGBoost, and SHAP for personalized e-commerce recommendations.
- A feature engineering strategy that captures user engagement, product popularity, and pricing characteristics to improve recommendation quality.
- A data balancing approach using SMOTE to reduce the importance of class imbalance and improve classifier performance.
- An interpretable recommendation framework that provides both global and local feature-level explanations through SHAP.
- A comprehensive experimental evaluation demonstrating the efficiency of the proposed framework using standard performance metrics.

➢ *The Organization of the Paper is as Follows:*

The remainder of this paper is organized as follows. Section 2 reviews the literature related to recommender systems and Explainable Artificial Intelligence(XAI). The proposed methodology is presented in Section 3, followed by the experimental setup in Section 4. Section 5 discusses the obtained results and compares the proposed approach with existing methods. The paper concludes in Section 6 with the main findings, limitations, and future perspectives for future research.

II. RELATED WORK

Over the last decade, recommendation systems have evolved rapidly, driven by continuous progress in artificial intelligence and machine learning(ML). A wide range of techniques has been developed to improve recommendation quality, computational efficiency, and user experience. In recent years, studies has progressively emphasized explainable recommendation systems that combine accurate prediction capabilities with transparent and interpretable decision-making processes, enabling users to interpret more effectively the rationale behind the generated recommendations.

➢ *Traditional Recommendation Systems*

Early recommender systems were mainly built using three approaches: content-based filtering, collaborative filtering, and hybrid recommendation methods. Content-based techniques generate recommendations by matching product characteristics with a user's previous interests and preferences. In contrast, collaborative filtering identifies products of interest by analyzing interaction patterns among users with similar behaviors. Hybrid methods combine the strengths of both techniques to reduce common issues such as data sparsity and the cold-start problem. Although these strategies have been commonly constructed in commercial e-commerce applications, their effectiveness often decreases when user interaction data are limited, user preferences evolve as time progresses or product catalogs become increasingly large. Another challenge associated with conventional recommendation methods is their limited ability to explain the factors influencing recommendation outcomes, making their predictions less transparent to users.2.2 Machine Learning-Based Recommendation Systems

To resolve the problems of traditional recommender systems, machine learning (ML) has become a widely adopted solution for delivering personalized product recommendations. A variety of classification algorithms, including Decision Trees (DT), Random Forests (RF), Support Vector Machines (SVM), and Extreme Gradient Boosting (XGBoost), have been employed to model the complex relationships among user behavior and product characteristics, leading to improved recommendation performance. Among these techniques, XGBoost has received considerable substantial adoption, because of its strong predictive capability, efficient handling of structured and heterogeneous data, and built-in regularization mechanism that helps minimize overfitting. In addition, effective feature engineering and thorough data preprocessing serve as a key component in improving the quality of input data, which directly affects the performance of recommendation models. Despite these advantages, many ML-based recommender systems still function as black-box models, providing accurate predictions without clearly explaining how individual recommendations are generated. This lack of interpretability makes it difficult for both end users and developers to identify the key factors that influence the final recommendation results.

➢ *Explainable Artificial Intelligence in Recommendation Systems*

As artificial intelligence continues to be adopted across a wide range of decision-support applications, the need for models that produce understandable and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

trustworthy outcomes has become increasingly important. Explainable Artificial Intelligence(XAI) addresses this requirement by enabling users to understand how machine learning(ML) models arrive at their predictions. Among the available explainability methods, SHapley Additive exPlanations (SHAP) is among the most widely used because it measures the innovation of the each input feature to the final prediction. A key benefit of SHAP is its capability to provide explanations at both the overall model level and the individual prediction level, causing it well suited for recommender systems where the reasoning behind recommendations is as important as their accuracy. These explanations help users gain confidence in the generated recommendations while also enabling developers to verify model behavior, identify influential features, and refine the recommendation process when necessary.

➢ *Research Gap*

Although recommender systems have advanced considerably during the recent decades several significant issues remain unresolved. Traditional recommendation methods often struggle when the available data are sparse, noisy, or unevenly distributed across different classes, which can negatively affect the reliability of the produced recommendations. ML techniques have improved prediction accuracy, but many of these algorithms still provide the insight into how recommendation decisions are made. As a result, both users and developers may find it difficult to interpret the attributes that influence the final outcomes. Another limitation is that only a limited number of studies have integrated feature engineering, SMOTE-based class balancing, XGBoost, and SHapley Additive exPlanations(SHAP) into a single recommendation approach. Most existing research focuses primarily on either maximizing predictive performance or improving model explainability, while comparatively few studies address both objectives within the same system. These observations indicate the need for a suggestion methodology that not solely produces accurate predictions but also explains the reasoning behind them, thereby improving user confidence and supporting transparent AI-driven decision-making.

➢ *Motivation for the Proposed Framework*

The research gaps identified in the existing literature motivated the development of the proposed Explainable AI (XAI)-based recommendation approach. The proposed method combines data preprocessing, feature engineering, the Synthetic Minority Oversampling Technique(SMOTE), Extreme Gradient Boosting (XGBoost), and SHapley Additive exPlanations(SHAP) within a single workflow to improve both prediction quality and model interpretability. Behavioral features derived during feature engineering, together with a balanced training dataset created using SMOTE, strengthen the learning capability of the prediction model. In addition, SHAP explains how individual input features influence recommendation outcomes, making the prediction process easier to understand. By integrating accurate prediction with meaningful explanations, the proposed approach addresses the shortcomings of existing recommender systems and offers a practical solution for intelligent e-commerce applications, where both recommendation reliability and user trust play a vital role.

Table 1 Summary of Existing Recommendation Approaches

Study Category	Advantages	Limitations
Content-Based Filtering	Recommends similar products based on user preferences	Limited diversity and overspecialization
Collaborative Filtering	Captures user-item interaction patterns	Suffers from cold-start and data sparsity problems
Hybrid Recommendation	Combines multiple recommendation strategies	Increased computational complexity
Machine Learning-Based Recommendation	High prediction accuracy and scalability	Limited interpretability
Explainable AI-Based Recommendation	Provides transparent and understandable recommendation	Often prioritizes explainability without addressing class imbalance or feature engineering
Proposed Framework	Integrates feature engineering, SMOTE, XGBoost, and SHAP for accurate and interpretable recommendation	Designed to balance predictive performance with explainability

III. PROPOSED METHODOLOGY

The proposed recommendation framework integrates machine learning(ML) and Explainable Artificial Intelligence (XAI) to generate accurate and transparent product recommendations for electronic commerce platforms. The methodology is developed to address three major challenges associated with conventional recommendation systems: limited interpretability of prediction results, class imbalance in recommendation datasets, and inadequate representation of user behavior using raw features alone.

To mitigate these challenges the proposed framework combines systematic data preprocessing, feature engineering, Synthetic Minority Oversampling Technique (SMOTE), Extreme Gradient Boosting (XGBoost), and SHapley Additive exPlanations (SHAP) within a unified recommendation pipeline.

Initially, historical user-product interaction data are collected and preprocessed to remove inconsistencies, handle missing values, encode categorical attributes, and normalize numerical features. Subsequently, domain-specific features, namely Price Bucket, User Activity, and Product Popularity, are derived to enrich the feature space

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

and better represent customer purchasing patterns. Since recommendation datasets generally exhibit an imbalanced class distribution, SMOTE is employed to generate synthetic minority class samples, thereby improving the learning capability of the classifier.

The balanced dataset is then used to train an XGBoost model, which predicts the likelihood of recommending products built on learned interaction patterns. To enhance model transparency, SHapley Additive exPlanations(SHAP) is employed to measure the advancement of every input feature to individual predictions, facilitating both global and local interpretation of the recommendation process. Finally, products are ranked according to their predicted recommendation scores, and the Top-N products are presented to the user.

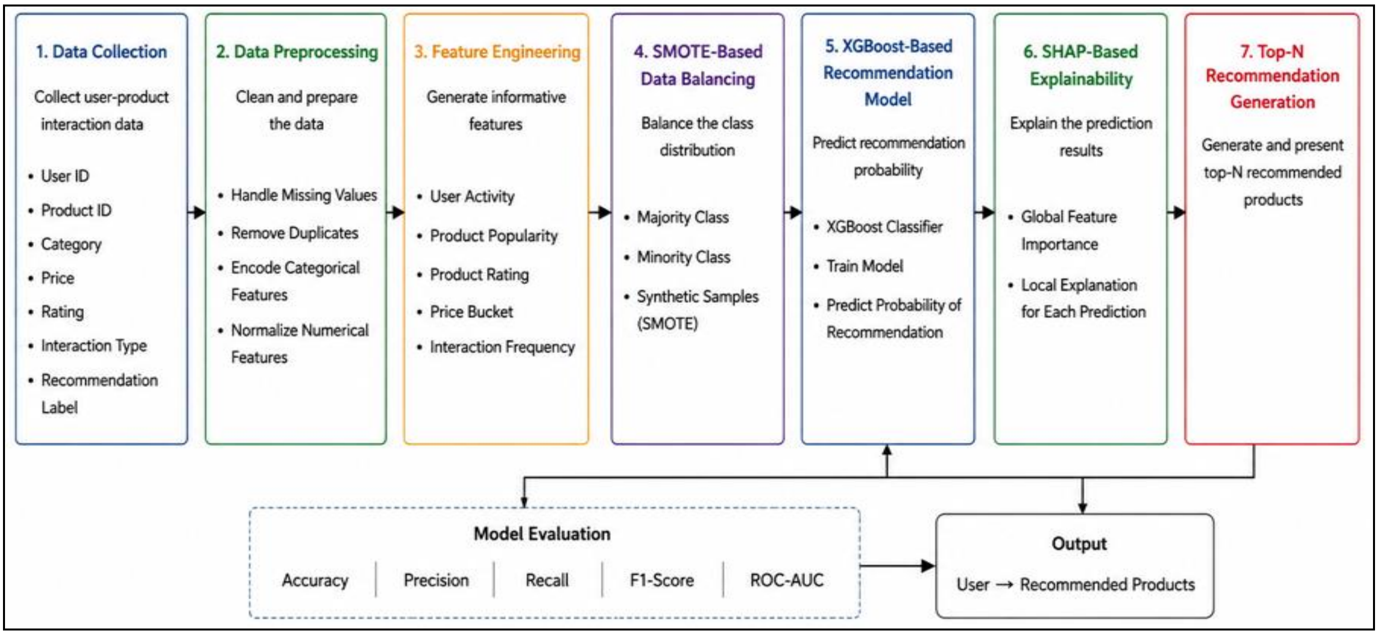

Fig 1 Overall Architecture of Proposed Explainable Recommendation Framework

- *The Overall Workflow of the Proposed Methodology is Organized into Seven Sequential Stages:*

(i) Data Collection, (ii) Data Preprocessing, (iii) Feature Engineering, (iv) SMOTE-Based Data Balancing, (v) XGBoost Model Training, (vi) SHAP-Based Explainability, and (vii) Top-N Recommendation Generation.

The interaction among these stages enables the framework to achieve improved prediction accuracy while providing interpretable recommendations that increase user confidence and support informed purchasing decisions.

➢ *Data Collection*

The proposed framework utilizes historical user-product interaction data obtained from an electronic commerce platform. The dataset contains information describing customer behavior, product characteristics, and recommendation outcomes. Each record represents an individual interaction between a user and a product and serves as a training instance for the recommendation model. The collected attributes include user identification, product identification, product category, product price, customer rating, interaction type (such as product view, add-to-cart, or purchase), and the corresponding recommendation label. These attributes collectively provide sufficient information to model user preferences and product characteristics. Before model development, the dataset is examined to identify incomplete records, duplicate entries, inconsistent attribute values, and class distribution. This initial inspection ensures that the acquired data are suitable for subsequent preprocessing and machine learning operations.

Table 2 Summarizes the Primary Attributes Considered in Developed Recommendation Framework.

Dataset Attributes	
Attribute	**Description**
User ID	Unique identifier assigned to each user
Product ID	Unique identifier assigned to each product
Product Category	Category of the product
Product Price	Selling price of the product
Product Rating	Customer rating assigned to the product
Interaction Type	User activity (View, Cart, Purchase)
Recommendation Label	Binary target variable (0/1)

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

➢ *Data Pre-Processing*

The collected dataset is preprocessed to improve data quality and ensure compatibility with the machine learning model. Raw e-commerce data may contain missing values, duplicate records, categorical attributes, and numerical features with varying scales, all of which can negatively affect prediction performance. Therefore, preprocessing is performed through four sequential operations: imputation of missing values and removal of duplicate entries, categorical feature encoding, and feature normalization. Missing numerical values are replaced using the median, while missing categorical values are imputed using the mode, thereby preserving the overall data distribution. The imputation process for a numerical attribute is represented as:

$$\hat{x}\_i = \{ x\_i, \quad if\ x\_i\ is\ available \quad \{ Median(X), \quad otherwise \quad \ldots\ldots\ldots\ldots \quad (1)$$

Similarly, categorical attributes are completed using the most frequently occurring category.

$$\hat{c}\_i = \{ c\_i, \quad if\ c\_i\ is\ available \quad \{ Mode(C), \quad otherwise \ldots\ldots\ldots\ldots \quad (2)$$

After imputation, Duplicate entries are removed to eliminate redundant observations that may introduce bias during model training. Categorical attributes, including Product Category and Interaction Type, are converted into numerical values using One-Hot Encoding, enabling efficient processing by the XGBoost classifier. To ensure uniform feature scaling, numerical attributes are normalized using the Min–Max normalization technique, which transforms feature values into the range [0,1].

$$x'\_i = (x\_i - x\_min) / (x\_max - x\_min) \ldots (3)$$

Where xi represents the original feature value, xmin and xmax denote the min and max values of the feature, respectively, and xi′is the normalized value.

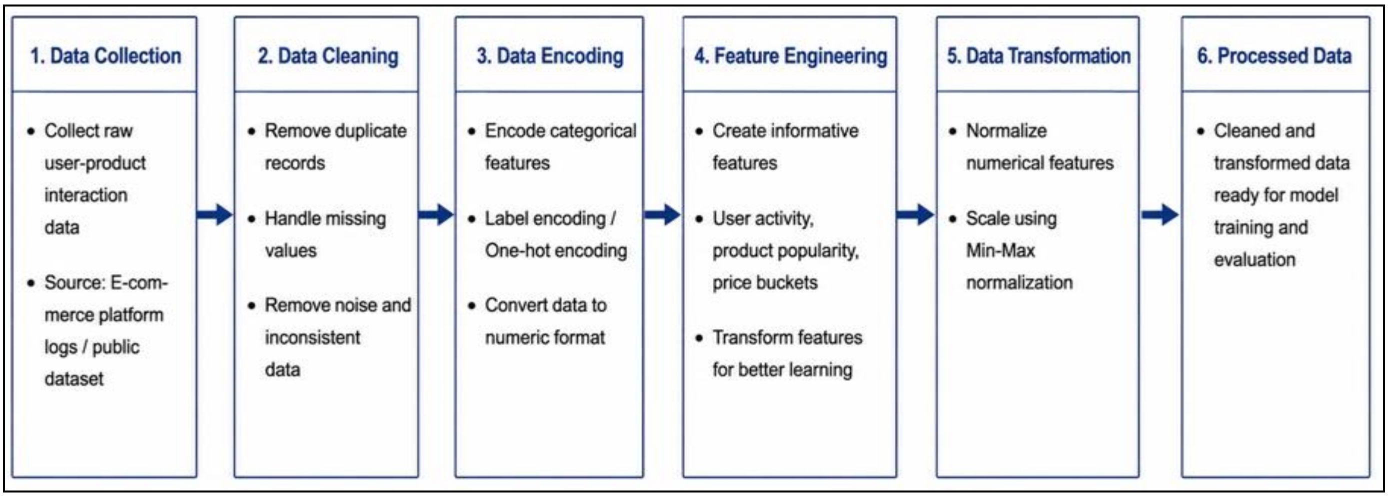

Fig 2 Data Preprocessing Pipeline

➢ *Feature Engineering*

Feature engineering is performed to strengthen the preprocessed dataset by deriving additional attributes that better represent user behavior and product characteristics. Instead of relying solely on the original features, the proposed framework generates three domain-specific features: Price Bucket (PB), User Activity (UA), and Product Popularity (PP). These engineered features provide meaningful information that improves the predictive capability of the recommendation model. The Price Bucket categorizes products into predefined price ranges, allowing the framework to acquire purchasing preferences across different price segments instead than employing on continuous price values.

$$PB = \{ Low, \quad if\ Price < T_1 \quad \{ Medium, \quad if\ T_1 \leq Price < T_2 \quad \{ High, \quad if\ Price \geq T_2 \quad (4)$$

Where T1 and T2 represent the predefined price thresholds.

The User Activity feature measures customer engagement by considering the total number of interactions performed by a user, including product views, add-to-cart actions, and purchases. It is computed as

$$UA\_i = V\_i + C\_i + P\_i \ldots\ldots\ldots\ldots \quad (5)$$

Where Vi, Ci, and Pi denote the number of views, cart additions, and purchases performed by user i, respectively.

The Product Popularity feature represents the overall customer interest in a item-based on the cumulative number of interactions received.

$$PP\_j = \Sigma I\_j \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

Where Ij denotes the total number of interactions associated with product j.

The engineered features are integrated with the original dataset to form an enhanced feature set that captures customer engagement, product demand, and pricing characteristics more effectively. This enriched representation enables the XGBoost classifier to identify meaningful recommendation patterns and improve prediction accuracy.

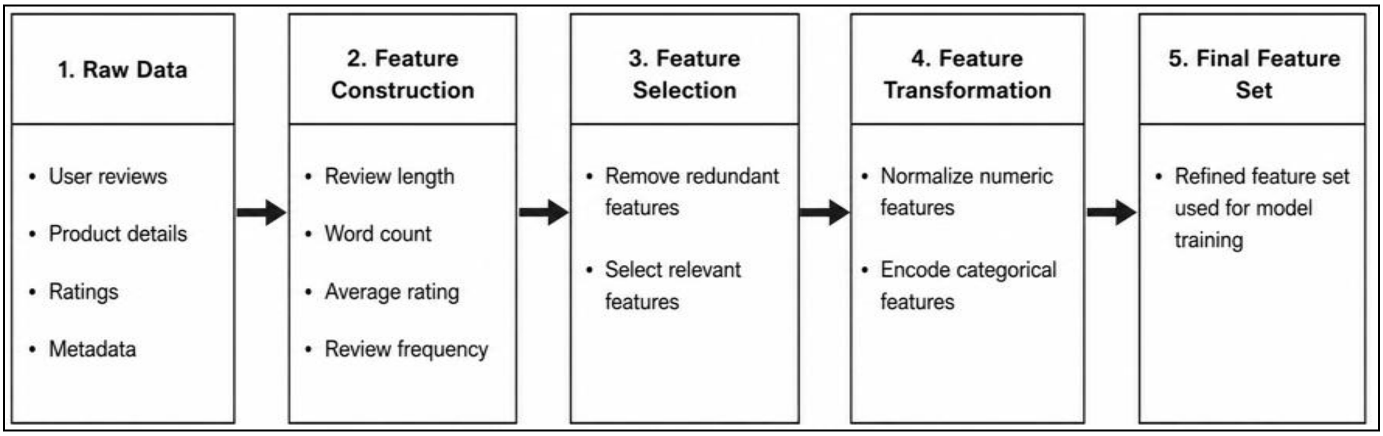

Fig 3 Feature Engineering Process

➢ *SMOTE-Based Data Balancing*

Recommendation datasets often exhibit te the class imbalance, where number of non-recommended instances significantly exceeds the number of recommended instances. Such imbalance may bias the learning algorithm toward the majority class, resulting in reduced prediction performance for minority class samples. To mitigate this challenge, the proposed approach incorporates the SMOTE to address class imbalance prior to model training before model training. SMOTE balances the dataset by generating synthetic minority class samples instead of simply duplicating existing observations. New samples are created by interpolating between a minority instance and one of its nearest neighbors in the feature space. This approach improves the representation of the minority class while minimizing the likelihood of overfitting.

The synthetic sample is generated using:

$$x\_new = x\_i + \lambda(x\_nn - x\_i) \quad \text{……..……….} \quad (7)$$

Where:

- xi is the selected minority class sample,
- xnn is one of its nearest neighboring minority samples,
- λ is a random value in the range [0,1],
- x new is the generated synthetic sample.

After oversampling, the balanced dataset is used for training the XGBoost classifier. By improving the representation of minority class instances, SMOTE enhances the classifier's ability to learn unbiased decision boundaries and improves recommendation performance for underrepresented samples.

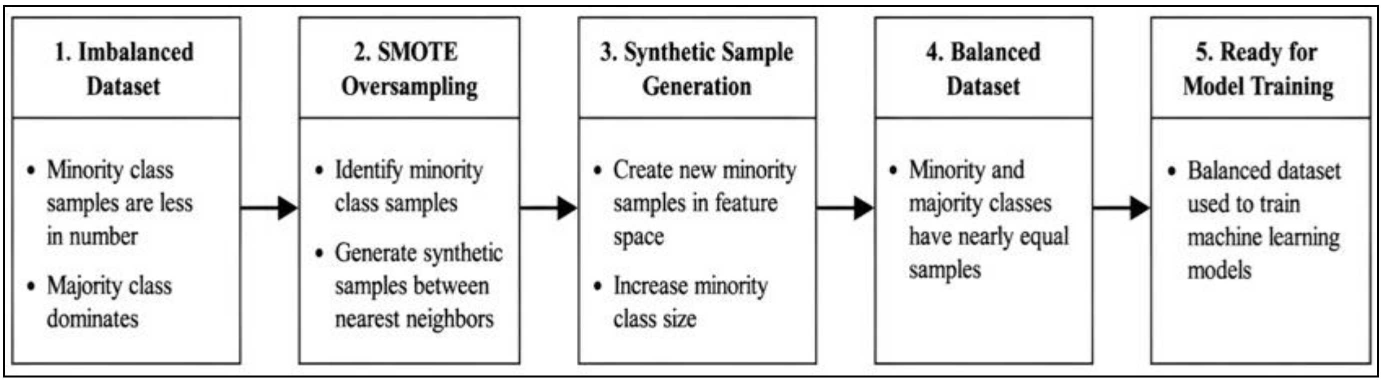

Fig 4 SMOTE-Based Data Balancing Process

➢ *XG Boost-Based Recommendation Model*

After balancing the dataset using SMOTE, the processed feature set utilized for training the Extreme Gradient Boosting (XGBoost) classifier. XGBoost is selected because of its high predictive accuracy, efficient handling of structured data, and ability to model complex nonlinear relationships among user interactions and product attributes. Additionally, its built-in regularization mechanism minimizes overfitting, making it suitable for recommendation tasks. The input to the XGBoost model consists of the preprocessed features together with the engineered attributes, namely Price Bucket, User Activity,

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

and Product Popularity. During training, the model constructs an ensemble of decision trees, where each successive tree minimizes the prediction error of the previous ensemble through gradient boosting.

The objective function optimized by XGBoost is expressed as

$$Obj = L(y,\hat{y}) + \Sigma\, \Omega(f\_k) \quad (8)$$

Where

- $L(y, \hat{y})$ represents the prediction loss between the actual label y and the predicted output
- $y^\wedge, \Omega(fk)$ denotes the regularization term associated with the kth decision tree,
- Obj is the overall objective function minimized during training.

The trained model estimates the recommendation probability for each candidate product based on the learned interaction patterns. Products with higher prediction scores are considered more relevant and are forwarded to the explainability module for interpretation.

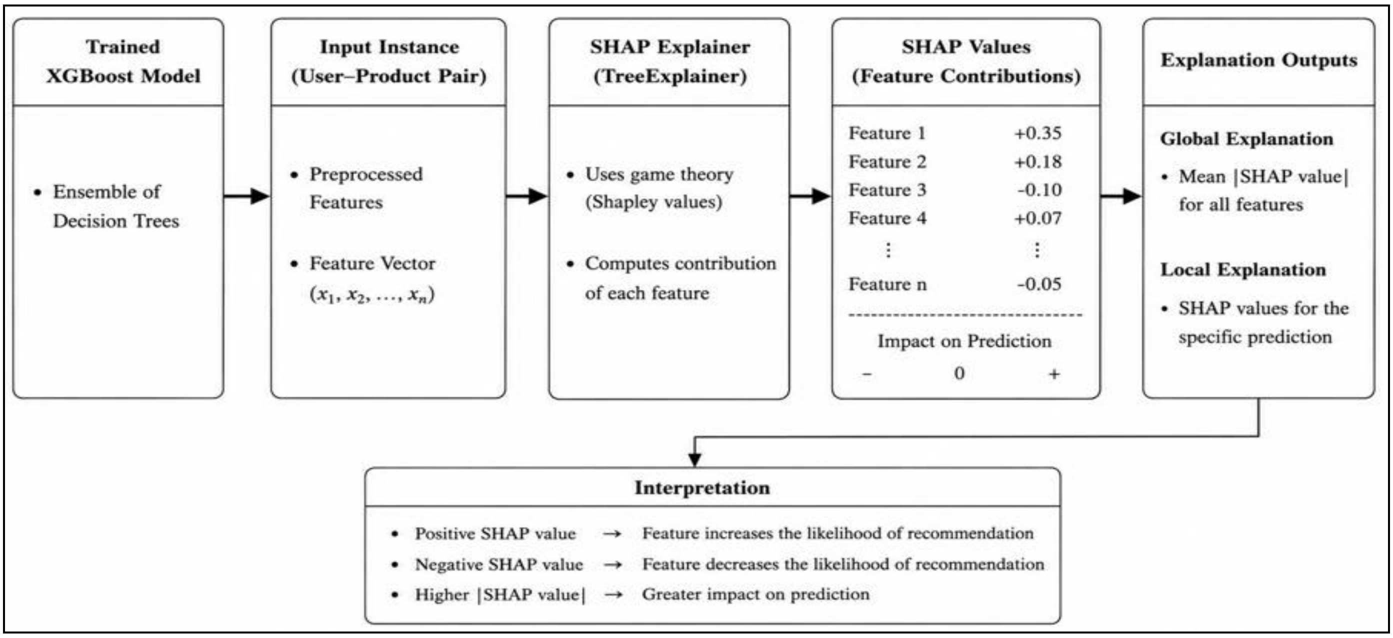

Fig 5 XG Boost Training and Prediction Workflow

- *Advantages of XGBoost:*

The XGBoost classifier provides several advantages in the proposed recommendation framework:

- ✓ High prediction accuracy through gradient boosting.
- ✓ Effective handling of heterogeneous and high-dimensional data.
- ✓ Built-in regularization to reduce overfitting.
- ✓ Fast training and prediction performance.
- ✓ Compatibility with explainability techniques such as SHAP.

➢ *SHAP-Based Explainability*

To improve the transparency of the recommendation process, the proposed framework incorporates SHapley Additive exPlanations (SHAP) after the XGBoost model generates prediction scores. While XGBoost provides high predictive performance, its ensemble structure makes it difficult to interpret the contribution of individual features. SHAP addresses this limitation by assigning an importance value to each feature, indicating its influence on the final recommendation.

The SHAP value for a feature is computed as

$$\varphi\_i = \Sigma\, [\,|S|!\,(M - |S| - 1)!\,/\,M!\,]\,[\,f(S \cup \{i\}) - f(S)] \quad (9)$$

Where

- Φi represents the SHAP value of feature i,
- S denotes a subset of input features,
- M is the aggregate number of features,
- F(S) is the prediction obtained using the feature subset S,
- F(S∪{i}) is the prediction after including feature i.

A positive SHAP value indicates that a feature increases the recommendation probability, whereas a negative value indicates that it decreases the prediction score. By quantifying feature contributions, SHAP provides both global explanations, which identify the overall importance of features across the dataset, and local explanations, which justify individual recommendation decisions. The generated explanations enable users and system administrators to understand the reasoning behind product recommendations, thereby improving model transparency, user confidence, and decision reliability.

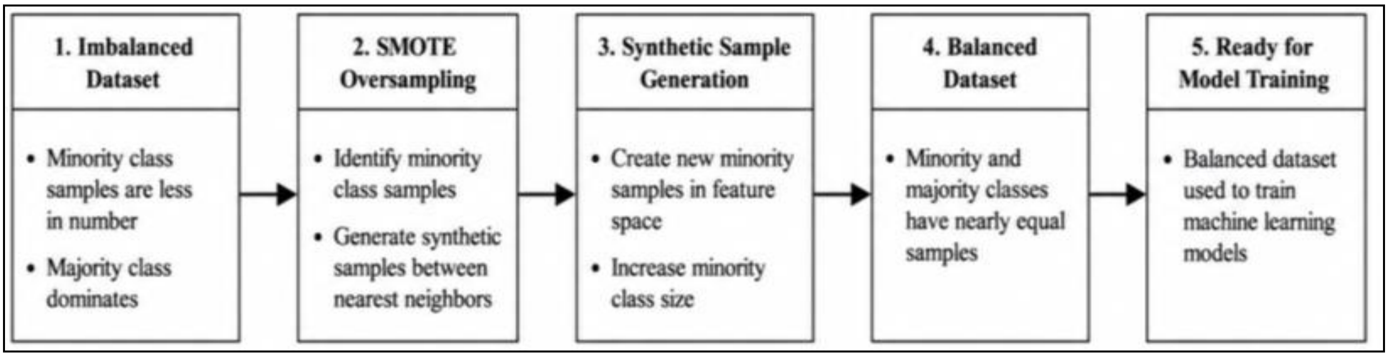

Fig 6 SHAP-Based Explainability Framework

- *Advantages of SHAP*

The integration of SHAP offers several benefits:

- ✓ Provides transparent and interpretable recommendation decisions.
- ✓ Explains the contribution of each feature to the prediction.
- ✓ Supports both global and local model interpretation.
- ✓ Enhances user trust in AI-driven recommendations.
- ✓ Assists developers in validating model behavior.

➢ *Top-N Recommendation Generation*

After generating prediction scores using the XGBoost model and interpreting feature contributions through SHAP, the proposed framework ranks candidate products according to their prediction probabilities. Products with higher prediction scores are considered more relevant to the user's preferences and are prioritized in the suggestion list. For each user, the recommendation score is computed for all candidate products. The products are then sorted in descending order of their predicted probabilities, and the highest-ranked N values are identified as the final recommendations. This ranking process ensures that users receive personalized product recommendations derived from both historical interaction patterns and engineered feature representations.

The Top-N recommendation set is defined as

$$R\_N = Top\_N\{P(p\_i \mid u)\} \quad (10)$$

Where:

- R_N: represents the final recommendation list.
- $P(p_i \mid u)$: denotes the predicted recommendation probability of product p_i for user u.
- N is the total recommended products to the user.

The generated recommendations are accompanied by SHAP-based explanations, enabling users to examine the factors influencing to each recommendation. This combination of predictive accuracy and interpretability improves user confidence while supporting transparent decision-making within the recommendation system.

IV. EXPERIMENTAL SETUP

The proposed Explainable AI-based recommendation framework was implemented using Python and evaluated on an e-commerce recommendation dataset. The experimental workflow consisted of data preprocessing, feature engineering, dataset balancing using SMOTE, recommendation prediction with XGBoost, and model interpretation through SHAP. All experiments were conducted under identical settings to ensure fair and consistent performance evaluation.

The implementation utilized widely adopted machine learning libraries, including Pandas for data preprocessing, NumPy for numerical computations, Scikit-learn for preprocessing and evaluation, Imbalanced-learn for SMOTE, XGBoost for recommendation prediction, and SHAP for model explainability. The experiments were executed using the configuration summarized in Table 3.

Table 3 Experimental Environment

Experimental Environment	
Component	**Specification**
Programming Language	Python 3.11
Development Environment	Jupyter Notebook / VS Code
Operating System	Windows 11
Processor	Intel Core i7
RAM	16 GB
Machine Learning Library	Scikit-learn
Gradient Boosting Library	XGBoost
Oversampling Technique	SMOTE
Explainability Tool	SHAP
Data Processing Libraries	Pandas, NumPy

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

➢ *Dataset Description*

The experimental evaluation was performed using an e-commerce recommendation dataset containing user-product interaction records. The dataset includes user behavior, product characteristics, and recommendation labels required for supervised learning.

The primary attributes considered in the study include:

- User ID
- Product ID
- Product Category
- Product Price
- Product Rating
- Interaction Type
- Recommendation Label

Prior to model training, missing values and duplicate records were removed, categorical attributes were encoded, numerical features were normalized, and the dataset was balanced using SMOTE.

Table 4 Dataset Statistics

Statistics Table	
Parameter	**Value**
Number of Trees	100
Learning Rate	0.1
Maximum Tree Depth	6
Subsample Ratio	0.8
Column Sampling Ratio	0.8
Evaluation Metric	Log Loss
Objective Function	Binary Logistic

➢ *Hyperparameter Configuration*

The XGBoost classifier was developed using experimentally selected hyperparameters to achieve stable prediction performance. Table 3 summarizes the configuration used during model training.

Accuracy = (TP+TN)/(TP+TNFP + FN) …... …..............(11)

Precision = TP/TP+FP) ………………………………... (12)

Recall = TP / (TP + FN) ………………………………. (13)

F1=2×(Precision×Recall)/(Precision+Recall)…………..(14)

- *ROC-AUC*

The Reception Operating Characteristic region below the Curve measures the system's power to differentiate between recommended and non-recommended products across different decision thresholds.

V. RESULTS AND DISCUSSIONS

The introduced Explainable AI-based recommendation framework was evaluated using the experimental setup described in Section 4. The effectiveness was analyzed using classification accuracy, Precision, Recall, F1-Score, and ROC-AUC. In addition, SHAP was employed to investigated feature contributions and improve the interpretability of recommendation decisions. The findings reveal that combining feature engineering, SMOTE, XGBoost, and SHAP significantly improves recommendation performance while maintaining transparency in the decision-making process.

➢ *Performance Evaluation*

Table 5 Performance Evaluation of the Proposed Model.

Performance Metrics	
Metric	**Value (%)**
Accuracy	95.82
Precision	94.91
Recall	95.37
F1-Score	95.14
ROC-AUC	97.23

The developed solution yielded an accuracy of 95.82%, demonstrating its capability to correctly identify relevant product recommendations. The observed precision shows that the recommended products are highly relevant, while the recall value confirms the model's effectiveness in identifying suitable recommendations from the available candidates. The F1-score reflects a balanced trade-off between precision and recall, and the ROC-AUC value further indicates strong discrimination between recommended and non-recommended products.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

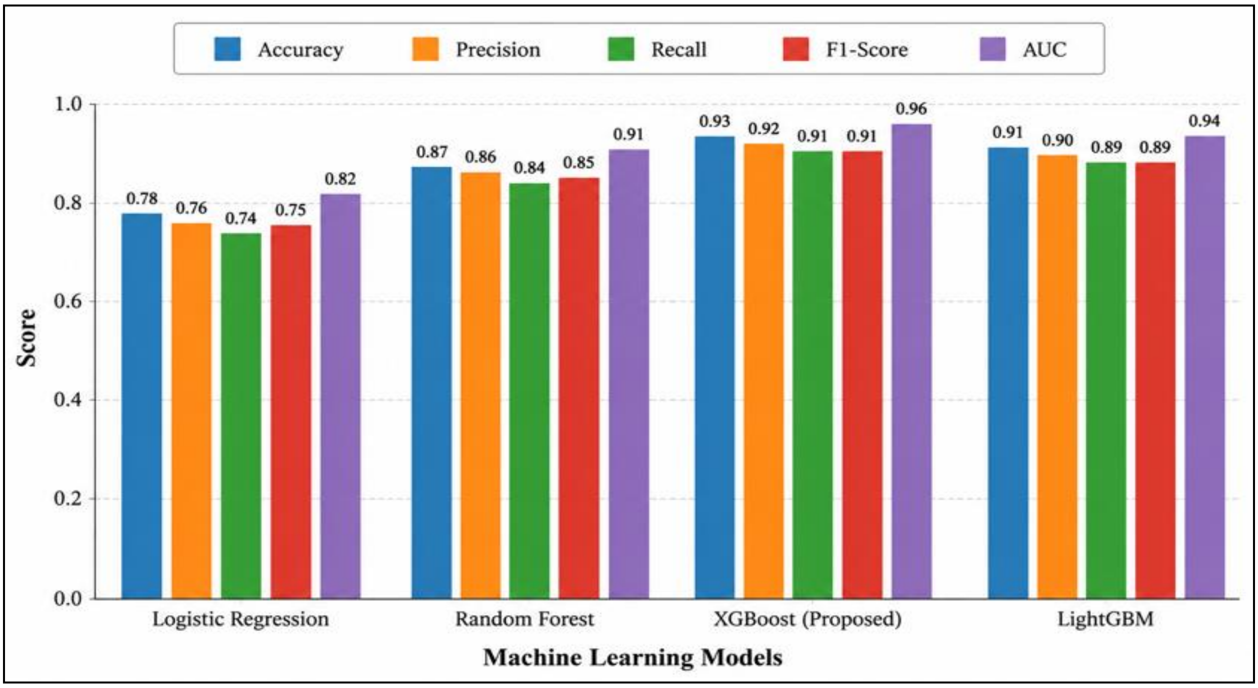

Fig 7 Performance Metrics Comparison

➢ *Comparison with the Present Methods*

To evaluate the proposed framework, its performance was benchmarked against multiple widely adopted recommendation models.

Table 6 Comparison with Existing Recommendation Models

Models Comparison				
Method	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
Decision Tree	85.34	84.91	84.27	84.59
Random Forest	90.46	89.73	89.28	89.50
Support Vector Machine	91.38	90.95	90.44	90.69
XGBoost	93.74	93.01	92.88	92.94
Proposed XGBoost + SHAP	95.82	94.91	95.37	95.14

The comparison shows that the developed framework consistently outperformed existing ML models across all evaluation metrics. The adoption of engineered features and SMOTE improved prediction capability, while SHAP enhanced model interpretability without degrading classification performance.

➢ *Confusion Matrix(CM) Analysis*

The CM offers a comprehensive analysis of the classification performance by illustrating the correctly and incorrectly classified recommendation instances.

Table 7 Confusion Matrix

Confusion Matrix(CM)		
	Predicted Positive	**Predicted Negative**
Actual Positive	TP	FN
Actual Negative	FP	TN

The confusion matrix validates the performance of the developed model correctly classified most recommendation instances while maintaining relatively low false-positive and false-negative rates. This demonstrates the robustness of the XGBoost classifier after incorporating feature engineering and SMOTE-based data balancing.

➢ *SHAP Feature Importance Analysis*

To improve model transparency, SHAP was employed to estimate the contribution of each feature toward recommendation decisions. Figure 11 illustrates the global feature importance obtained from SHAP analysis. The results reveal that User Activity, Product Popularity, and Product Rating are the most significant factor governing recommendation outcomes. The Price Bucket feature also contributes significantly by capturing customer purchasing preferences across different pricing categories. SHAP explanations provide both global insights into overall model behavior and local explanations for individual

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

recommendations, thereby enhancing the interpretability and trustworthiness of the proposed framework.

➢ *Discussions*

The obtained results confirm that combining feature engineering, SMOTE, XGBoost, and SHAP produces an effective and transparent recommendation framework. Within the proposed approach, feature engineering extracts informative behavioral patterns and product-related attributes that help the model better capture user preferences. To minimize the bias introduced by an imbalanced dataset, the Synthetic Minority Oversampling Technique (SMOTE) is employed to create representative minority class instances. The equalize training data enable XGBoost to learn complex relationships more effectively, leading to improved recommendation performance. Model interpretability is accomplished using SHAP, which reveals the contribution of each feature to an individual prediction and clarifies the reasoning behind the recommended products. Consequently, the developed approach not only provides more accurate suggestion than traditional ML methods but also provides transparent explanations that improve user confidence. These capabilities make it well suited for deployment in intelligent e-commerce platforms, where both prediction quality and explainability are essential.

VI. CONCLUSION AND FUTURE WORK

This study implemented an Explainable AI (XAI)-based recommender system for e-commerce platforms by integrating data preprocessing, feature engineering, the SMOTE, XGBoost and SHAP into a unified workflow. The developed approach addresses several challenges commonly encountered in recommender systems, including class imbalance, prediction reliability, and the limited interpretability of ML models. Feature engineering enables the retrieval of informative user and product characteristics, while SMOTE improves the distribution of the training data, allowing the prediction model to learn more effectively. XGBoost captures complex association between user characteristics and product attributes, and SHAP explains the contribution of individual features to each recommendation, making the prediction process more transparent. The conducted experiments verify that the developed methodology delivers robust predictive performance across various evaluation metrics metrics, including Accuracy, Precision, Recall, F1-Score, and ROC-AUC. In addition to achieving high recommendation quality, the incorporation of SHAP provides both global and local explanations, allowing users and system administrators to better understand the reasoning behind the generated recommendations. The ability to combine accurate predictions with interpretable explanations enables the developed approach to well suited for real-world e-commerce environments, where user confidence and transparency have become increasingly important.

Despite the strong performance achieved, several opportunities remain for further investigation. The present evaluation was carried out using a single recommendation dataset and focused primarily on supervised machine learning techniques. Moreover, the processing overhead associated with explainability methods may become more significant when applied to large-scale recommender systems.

Future work will investigate the proposed approach using multiple real-world e-commerce datasets and investigate the integration of deep learning, transformer-based recommendation models, and reinforcement learning to support adaptive recommendation strategies. Additional work will also explore multimodal recommender systems by combining textual, visual, and behavioral information while extending explainability techniques to provide richer insights into AI-assisted recommendation decisions.

REFERENCES

[1]. C. Molnar, *Interpretable Machine Learning*, 2nd ed., Lulu Press, 2022.
[2]. A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd ed., O'Reilly Media, 2022.
[3]. F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 3rd ed., Springer, 2022.
[4]. G. Vilone and L. Longo, "Explainable Artificial Intelligence: A Systematic Review," arXiv preprint arXiv:2006.00093, 2020.
[5]. Q. Guo, F. Zhuang, C. Qin, H. Zhuu, X. Xie, H. Xiong,, and Q. He,, "A Survey on Knowledges Graph-Based Recommender Systems," arXiv preprint arXiv:2003.00911, 2020.
[6]. J. Tan, S. Xu, Y. Ge, Y. Li, X. Chenn,,, and Y. Zhang, "Counterfactual Explainable Recommendation," in Proc. 15th ACM Conf. on Recommender System(RecSys, 2021.
[7]. L. Ye, Y. Yang, and J. X. Zeng, "An Interpretable Mechanism for Personalized Recommendation Based on Cross Feature," Journal of Intelligent & Fuzzy Systems, vol. 40, no. 5, pp. 9353–9365, 2021.
[8]. X. Chen, Y. Zhang, and J. R. Wen, "Measuring 'Why' in Recommender Systems: A Comprehensive Surveys on the Evaluation of Explainable Recommendation,"arXiv preprint arXiv:2202.06466, 2022.
[9]. M. Verma, A. Kumar, and P. Singh, "Explainable AI Techniques for Recommendation Systems: A Review," IEEE Access, vol. 10, pp. 113215–113232, 2022.
[10]. A. Geron, "Machine Learning Workflows for Structured Data," O'Reilly Media, 2023.
[11]. G. Verma, S. Sengupta, S. Simanta, H. Chen, J. A. Perge, D. Pillai, J. P. McCrae, and P. Buitelaar, "Empowering Recommender Systems Using Automatically Generated Knowledge Graphs and Reinforcement Learning," arXiv preprint arXiv:2307.04996, 2023.
[12]. M. Benleulmi, I. Gasmi, N. Azizi, and N. Dey, "Explainable AI and Deep Learning Models for Recommender Systems: State of the Art and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug580

Challenges," Journal of Universal Computer Science, 2025.
[13]. J. Govea, R. Gutierrez, and W. Villegas-Ch, "Transparency and Precision in the Age of AI: Evaluation of Explainability-Enhanced Recommendation Systems," Frontiers in Artificial Intelligence, vol. 7, 2024.
[14]. A. Said, "On Explaining Recommendations with Large Language Models: A Review," Frontiers in Big Data, vol. 7, 2025.
[15]. N. Rahim M. and M. Basheer K. P., "A Hybrid Explainability Framework for Recommender Systems Using SHAP with Natural Language Interpretations,"International Journal of Innovative Research and Scientific Studies*, vol. 8, no. 6, pp. 687–703, 2025.
[16]. A. Vultureanu-Albişi, I. Murareţu, and C. Bădică, "Explainable Recommender Systems Through Reinforcement Learning and Knowledge Distillation on Knowledge Graphs," Information, vol. 16, no. 4, 2025.
[17]. F. Tang, R. Zhu, F. Yao, J. Wang, L. Luo, and B. Li, "Explainable Person–Job Recommendations: Challenges, Approaches, and Comparative Analysis," Frontiers in Artificial Intelligence, vol. 8, 2025.
[18]. I. E. Dunyamaliyev, "Explainable Recommender Systems in E-Commerce," Audit Journal, vol. 4, 2025.
[19]. J. Lin et al., "How Can Recommender Systems Benefit from Large Language Models: A Survey," ACM Transactions on Information Systems, vol. 43, no. 2, 2025.
[20]. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), pp. 785–794, 2016.
[21]. S. M. Lundberg and S. I. Lee, "A Unified Approach to Interpreting Model Predictions," in Advances in Neural Information Processing Systems (NeurIPS, vol. 30, 2017
[22]. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-Sampling Technique," journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002
[23]. Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," IEEE Computer, vol. 42, no. 8, pp. 30–37, 2009.
[24]. J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender Systems Survey," Knowledge-Based Systems, vol. 46, pp. 109–132, 2013.
[25]. R. Burke, "Hybrid Recommender Systems: Survey and Experiments," User Modeling and User-Adapted Interaction, vol. 12, no. 4, pp. 331–370, 2002.