Machine Learning for Recommender Systems Under Implicit Feedback and Class Imbalance

Younes KOULOU, Norelislam EL HAMI
Science and Engineering Laboratory-ENSA, Ibn Tofail University, Kenitra, Morocco

***Abstract*—Recommender systems (RS) in domains with implicit feedback and significant class imbalance, such as health insurance, face unique challenges in accurately predicting user preferences. This study proposes a machine learning framework leveraging tree-based ensemble methods to address these limitations. We conducted a comprehensive comparative analysis of algorithms, including Decision Trees, Random Forest, Gradient Boosting Machines, CatBoost, Extra Trees, HistGradient Boosting, and XGBoost to identify the most effective approach for handling data skew and complex feature interactions. The model was trained on a real-world dataset from an international insurance broker, containing demographic profiles and purchase histories. After extensive preprocessing and class rebalancing, the models were optimized and evaluated on a separate test set. Among these, XGBoost verified superior performance, achieving remarkable results with a precision of 97.23% and an accuracy of 97.51%. The model presented robust generalization capabilities and convergence stability, with no signs of overfitting. Concretely, these performances translate into an increased ability for insurers to reliably identify customer needs from limited behavioral data, thus improving the relevance of personalized offers. These findings highlight the efficacy of XGBoost in treatment datasets with unbalanced implicit feedback and its capacity as an effective solution for complex recommendation problems. This work contributes a practical and scalable framework for improving personalized recommendations in data-constrained environments.**

***Keywords*—Recommender systems; XGBoost; implicit feedback; class imbalance; health insurance**

I. Introduction

The algorithmic challenge of well corresponding users with items or content they will find pertinent—the primary objective of recommender systems—is essential to the modern digital economy. Although widely used and studied in domains like e-commerce and media streaming, the application of these systems in more complex areas with high stakes like personalized finance and healthcare rests a formidable challenge. The nature of the accessible data is a significant barrier shared by all of these areas: recommender systems must often operate without explicit user ratings and, more importantly, must learn from datasets that are severely unbalanced by class.

This paper takes on these two challenges directly. The absence of explicit feedback (likes, thumbs-up/down, …) makes systems dependent on implicit signals—comportments such as clicks, purchase decisions, or view duration. As work by [1] established, a user not interacting with an item cannot be simplistically interpreted as a negative signal; it may simply signify a lack of exposure. This ambiguity is combined by the problem of extreme class imbalance, where the number of items a user does not interact with far exceeds those they do interact with [2]. Traditional approaches, containing collaborative filtering and matrix factorization, do not sufficiently solve this issue without thorough algorithmic adjustment, such as high-level negative sampling or cost-sensitive learning [3].

However, a critical research gap persists. The vast majority of SR literature tackling implicit feedback and class imbalance has focused on data-rich environments, such as media and retail. In contrast, their application to high-stakes, data-constrained domains like health insurance, remains largely underexplored, despite the sector's specific and compelling challenges [4]. The health insurance sector offers a socially significant and technically demanding case study. Unlike recommending a movie, suggesting a suitable insurance product is a crucial decision with direct implications for a user's financial and physical well-being. This context perfectly illustrates the central problem under study:

- Feedback is only implied: The primary signal is the final decision made by a customer. There are no reliable "dislike" signals.
- Data is severely imbalanced: For any given customer, the number of insurance products they did not choose is orders of magnitude larger than the one they did.
- Features are rich and heterogeneous: Effective recommendation requires synthesizing a user's demographic, socio-economic, and medical history to predict their needs accurately.

This work aims to bridge this gap by investigating machine learning optimization techniques for building effective recommenders in this constrained and high-impact environment. Using a real-world dataset from the health insurance industry, we explore methods that move beyond treating non-interactions as negative examples.

Our contributions are twofold: Methodologically, we rigorously demonstrate that XGBoost [5], enhanced by adaptive class rebalancing, provides a robust and interpretable solution for recommender systems facing implicit feedback and severe class imbalance. Practically, we translate this methodological insight into a concrete decision support framework for health insurance, demonstrating its direct value for personalizing customer offers and improving targeting efficiency, a result with significant implications for other high-stakes, data-scarce domains.

The remainder of this paper is structured as follows: Section II reviews related work on implicit recommenders and

imbalanced data learning. Section III presents an overview of the RS concept. Section IV proposes XGBoost for recommendation. Section V details our methodology. Section VI presents our experimental results and discussion. Finally, Section VII presents the conclusion of the study.

II. Related Work

Our study lies at the intersection of three research streams: recommender systems (RS) handling implicit feedback and class imbalance, their application in high-stakes domains, and the use of ensemble methods.

The challenges posed by implicit feedback and class imbalance are fundamental. The seminal work of [1] established collaborative filtering techniques tailored to implicit feedback. However, traditional approaches struggle to capture complex nonlinear relationships. Significant advances have come from neural models, such as Neural Collaborative Filtering (NCF) of [6], which learn nonlinear interaction functions between users and items. To specifically address class imbalance, techniques such as SMOTE [7] (resampling) have been proposed. More recently, contrastive learning approaches have gained popularity, as demonstrated by [8] with their Soft Contrastive Learning method specifically designed for implicit feedback recommendations. Methods exploring the improvement of negative sampling also continue to emerge [9]. Nevertheless, these works, including advanced neural approaches and imbalance regulation techniques, are mainly validated on large public datasets from e-commerce or media. Their effectiveness in environments with limited and heterogeneous real-world data remains an open question and constitutes a limitation of current research.

The application of RS in sensitive domains such as healthcare is growing. For example, Wang et al. [10] applied knowledge-aware graph neural networks (GNNs) for recommendations, a powerful method for relational data. Very recent work illustrates similar trends with sophisticated approaches in specific domains. The author in [11] proposes the DiagNCF (Diagnostic Neural Collaborative Filtering) model for medical recommendation, using attention mechanisms to capture complex interactions between patients and diagnoses.

However, these studies have limitations. Complex models such as GNNs [10] or DiagNCF [11] often rely on a wealth of data (knowledge graphs, detailed records), and their complexity may make them poorly suited to more limited and heterogeneous insurance datasets.

Our work seeks to precisely fill this gap. While recent literature explores complex neural architectures often unsuitable for the constraints of limited data, our study takes a pragmatic and novel approach. We provide a comprehensive comparative analysis of tree-based ensemble methods (such as, XGBoost and Random Forest), which are recognized for their effectiveness on tabular data and their robustness to imbalance, but have not been systematically evaluated for this specific task. By rigorously testing them on a real-world insurance dataset, we aim to establish a solid reference to identify the most reliable solution for building recommender systems in data-constrained and high-stakes environments.

III. Recommender Systems

Recommender Systems (RS) are computational tools designed to suggest relevant items to a user based on their preferences, history, or characteristics [12]. Initially popularized by e-commerce platforms (e.g., Amazon, Netflix), they now play a crucial role in various fields, including insurance, healthcare, and financial services. Their primary goal is to reduce information overload by filtering and prioritizing the most suitable options for each user [13].

A. RS Concept

RS typically operates in three key steps:

- Data collection: Gathering explicit information (e.g. ratings, reviews) or implicit data (e.g. browsing history, purchases) about users and products.
- Preference modeling: Using algorithms to infer user preferences, either through collaborative filtering or content-based approaches.
- Recommendation generation: Proposing items (e.g. insurance policies, movies) ranked by their predicted relevance [14].

A major challenge is the cold-start problem, where insufficient data on new users or products makes it difficult to generate reliable recommendations [15] [16]. In the context of health insurance, this challenge is exacerbated by the lack of explicit feedback and the sensitive nature of medical data.

B. RS Classification

RS can be categorized into four main classes, each with its own advantages and limitations:

1) Collaborative Filtering (CF)

a) Principle: Recommends items by identifying similarities between users (user-based CF) or between products (item-based CF) (Fig. 1). For example, if two customers have purchased similar insurance policies, the system will suggest comparable options [17].

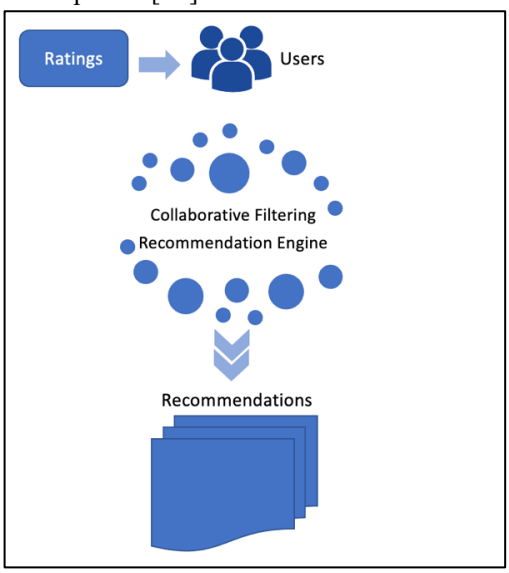


Fig. 1. Collaborative filtering RS.

b) Advantages: Does not require detailed item descriptions (e.g. metadata).

c) Limitations: Sensitive to data sparsity and the cold-start problem [18].

2) Content-based filtering

a) Principle: Recommends items similar to those the user has previously liked by analyzing their intrinsic features (e.g. keywords, categories) (Fig. 2). For example, a customer who purchased dental care insurance might receive recommendations for policies with analogous coverage [19].

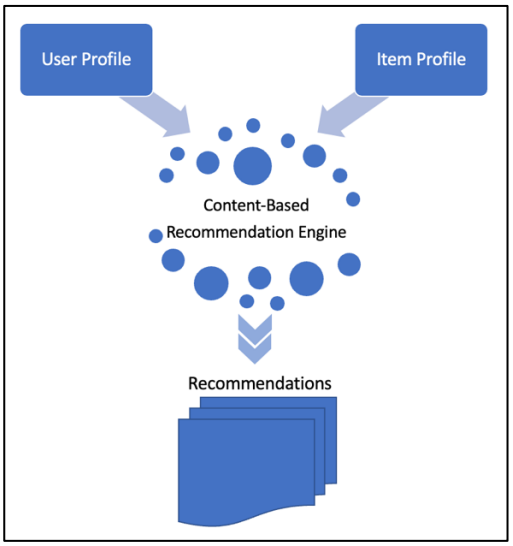


Fig. 2. Content-based filtering RS.

b) Advantages: Partially solves the cold-start problem for new items.

c) Limitations: Requires rich metadata and struggles with discovering serendipitous preferences (lack of serendipity).

3) Hybrid recommendation systems

a) Principle: Combines CF and content-based methods to overcome their respective limitations. Leverages both user/item similarities (collaborative approach) and intrinsic features of items/customer profiles (content-based approach) [20].

b) Advantages: Hybrid systems provide more comprehensive recommendations while mitigating cold-start and data sparsity issues through integrative flexibility.

c) Limitations: Despite their effectiveness, hybrid systems suffer from increased complexity, exhaustive data requirements, decision opacity, and challenges in adapting to dynamic regulatory contexts [21].

4) Demographic RS

a) Principle: Demographic systems recommend items based on users' socio-demographic characteristics (age, gender, income, location, etc.). Unlike collaborative approaches, they don't require interaction history [22].

b) Advantages: Ideal for new users (cold start) and simple to implement, using business rules or lightweight models.

c) Limitations: While useful for cold starts, these systems offer coarse personalization (unable to capture individual nuances within groups), heavily depend on comprehensive demographic data (often incomplete), and demonstrate lower accuracy than collaborative or content-based approaches.

IV. XGBoost for Robust Recommendation

Unlike conventional recommendation systems that typically depend on collaborative filtering or simple demographic-based techniques, the proposed method leverages XGBoost's powerful classification capabilities to manage the dataset's significant complexity and pronounced imbalance. XGBoost effectively captures the relationships between user demographics and their preferences, thereby offering a more effective solution for handling data sparsity and cold-start scenarios than previous hybrid methods that primarily merged demographic and collaborative filtering.

However, traditional models frequently need a large amount of historical data and are vulnerable to the cold-start issue; XGBoost demonstrates a better capacity to generalize from limited information. This makes it possible to provide precise recommendations, even for infrequent profiles or new users. Additionally, this approach deviates from solutions based on neural networks like DropoutNet [23] by offering a highly interpretable and scalable model that performs well in spite of the imbalance in the dataset and the wide range of user attributes.

Therefore, using XGBoost in this context not only improves recommendation accuracy but also offers a workable and reliable solution for real-world deployments where data is frequently dispersed unevenly or incompletely.

The following section details our proposed methodology, which models intricate user profiles from unbalanced implicit feedback data using XGBoost.

V. Methodology

The proposed approach proceeds in a conventional progression, starting with data collection and ending with findings discussion (see Fig. 3).

A. Data Understanding

The dataset used in this study comes from an international broker specializing in the distribution of diversified insurance solutions (health, life, etc.). Collected over a four-year period (2020–2024), the observations include basic demographic variables (age, gender, postal code) as well as self-reported data from preference surveys.

These metrics allow us to build a detailed customer profile and precisely identify the products they have purchased. The nature of the data, combining socio-demographic information and choice behaviors, makes it a particularly suitable corpus for training a recommendation system, in a context where the objective is to offer targeted policies aligned with each policyholder's profile.

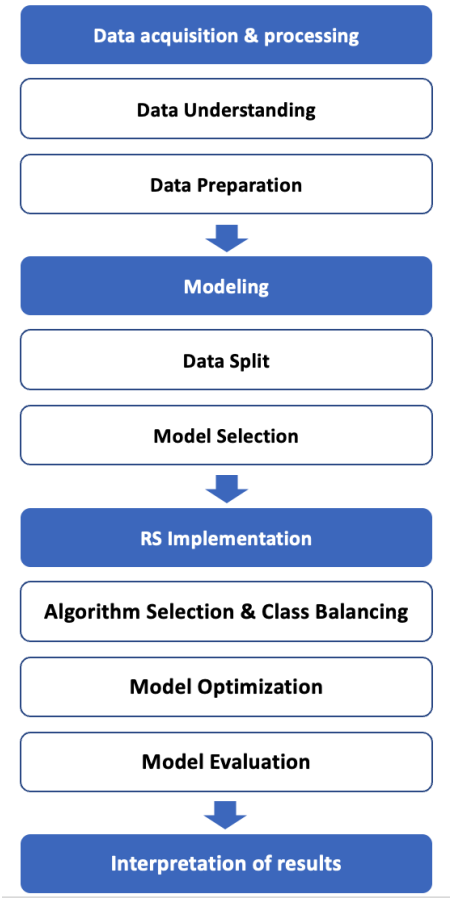


Fig. 3. Methodological diagram.

Our variable of interest is the insurance product category, structured hierarchically into policy families and then individual plans. The recommendation system aims to predict the most relevant category for a given client and to propose specific plans adapted to their characteristics. Ultimately, this system assists advisors by automating part of the targeting phase, thus accelerating the client-offer matchmaking and streamlining the consultation process.

During this phase, preliminary exploratory analysis was used to assess the structure, consistency, and quality of the data. We examined the main variables, analyzed the distribution of the response variable, and identified missing values, duplicates, and outliers. Correlation and relationship analysis between variables was also performed to identify potentially predictive characteristics. These investigations provided an essential basis for data preprocessing and subsequent model design.

B. Data Preparation

This phase involved a comprehensive preprocessing of the dataset to optimize it for machine learning. We systematically addressed data quality issues by implementing appropriate missing value treatments (including mode imputation for categorical variables and predictive modeling for complex cases), removing duplicates and irrelevant records, and handling outliers. Advanced feature engineering techniques were applied to create meaningful new variables, such as regional indicators from postal codes and spousal status flags. Categorical features were transformed using both one-hot and label encoding based on their characteristics. The dataset was further enhanced through aggregation features and careful feature selection, ensuring optimal input quality for subsequent modeling stages.

C. Modeling

1) Data split: To develop and evaluate the model, we split the data into three distinct parts: 70% for training, 10% for validation, and 20% for testing. This split allows us to properly evaluate the model's performance at each step, while still keeping enough data for robust training. The validation set serves as a checkpoint for adjusting parameters, while the test set gives us a final, objective measure of the model's ability to generalize to new data.

2) Model selection: The algorithms chosen for the modeling phase were tree-based ensemble techniques, including Random Forest, GBM, XGBoost, CatBoost, and the Extra Trees and HistGradient Boosting methods. This selection is justified by two key properties of these models: first, their strong ability to learn complex nonlinear relationships directly from raw data, thus minimizing preprocessing requirements; second, their proven performance in handling classification problems where the class distribution is asymmetric.

Decision tree-based algorithms, including the Simple Decision Tree and Random Forest, have the advantage of easy interpretability, while being able to effectively model nonlinear relationships and complex interactions between variables [24] [25]. The Random Forest method significantly mitigates overfitting by aggregating predictions from a multitude of trees, which also improves its robustness against class imbalances. Another major advantage of our recommendation system is its intrinsic ability to handle heterogeneous data without requiring elaborate preprocessing [26].

Unlike methods based on a single tree or forest, boosting approaches (Gradient Boosting, XGBoost, CatBoost) gradually improve performance by sequentially fitting new trees to the residual errors of previous ones. XGBoost offers speed and accuracy, particularly on unbalanced datasets, through regularization and early stopping mechanisms. CatBoost completes this family with its native management of categorical variables, which it pre-processes in an optimized manner without requiring an external step, thus guaranteeing significant gains in performance and ease of implementation [27].

The Extra Trees and HistGradient Boosting algorithms offer distinct advantages for large datasets. Extra Trees, through their increased random segmentation process, promote better generalization by limiting overfitting. As for HistGradient Boosting, it significantly accelerates training by quantifying features into histograms, without compromising its performance on unbalanced distributions. These properties make these two methods relevant choices in contexts demanding in terms of computational efficiency and robustness.

D. RS Implementation

The implementation framework consisted of three fundamental components:

1) Algorithm selection and class balancing: This step involved a comprehensive evaluation of multiple machine learning algorithms. To correct class imbalance, we used weighting techniques during training. This helped the models better recognize patterns in the minority classes.

2) Model optimization protocol: A thorough process of hyperparameter setting was conducted using grid search methodology, combined with cross-validation to guarantee performance that is generalized. This dual approach allows simultaneous optimization of model configurations and robustness to overfitting via systematic data division.

3) Probabilistic recommendation framework: The final implementation adopted a probability distribution technique with multiple classes, exposing the top three predicted product categories for each client in order to generate tiered recommendations. This method of progressive suggestions provides improved flexibility in decision making in contrast to outputs from traditional single class classification.

The complete architecture presents a solution that tackles both technical difficulties in model development and realistic business needs for concrete recommendations. Every element was made to complement the others, from initial data processing through to final outputs that are seen by clients.

E. Evaluation

Finally, an evaluation of the effectiveness of the recommendation system was carried out using performance indicators, such as accuracy, F1 score, precision, recall and logarithmic loss. Subsequently, the learning curve was analyzed in order to obtain additional information on the classification schemes and the behavior of the model throughout the learning process.

VI. Results and Discussion

A. Comparative Model Performance Analysis

The evaluation revealed distinct performance characteristics across the tested algorithms (see Fig. 4 to Fig. 9):

- Decision Trees showed strong predictive capability (high accuracy/F1-score) but suffered from computational inefficiency.
- Random Forest improved prediction confidence (lower log loss) while maintaining similar speed constraints.
- Gradient Boosting achieved top-tier performance metrics with particularly strong recall.
- XGBoost demonstrated optimal balance: highest accuracy (97.51%), lowest log loss, and robust recall.
- CatBoost matched XGBoost's accuracy but with significantly longer runtime.
- Extra Trees showed moderate performance with noticeable precision/F1-score tradeoffs.
- Hist Gradient Boosting delivered competitive accuracy but with higher prediction uncertainty.

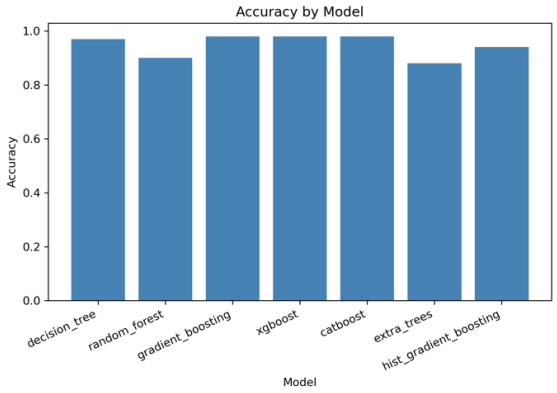


Fig. 4. Test accuracy of the models.

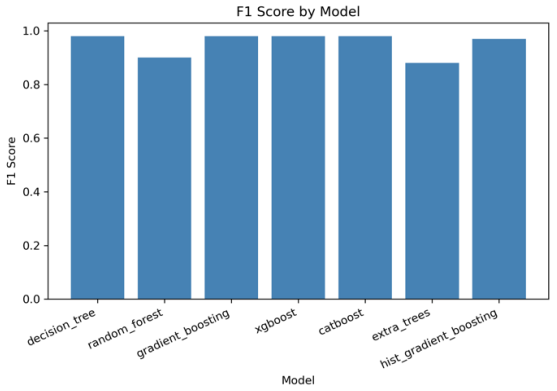


Fig. 5. Test F1 score of the models.

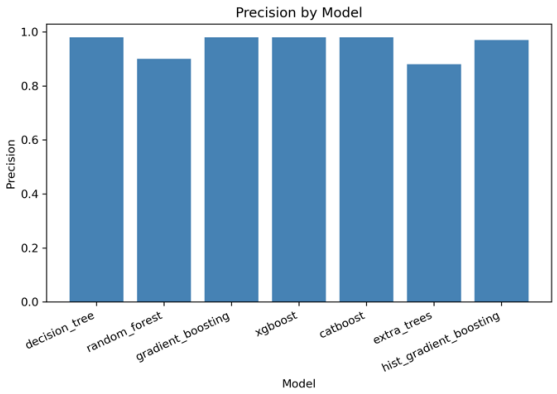


Fig. 6. Test precision of the models.

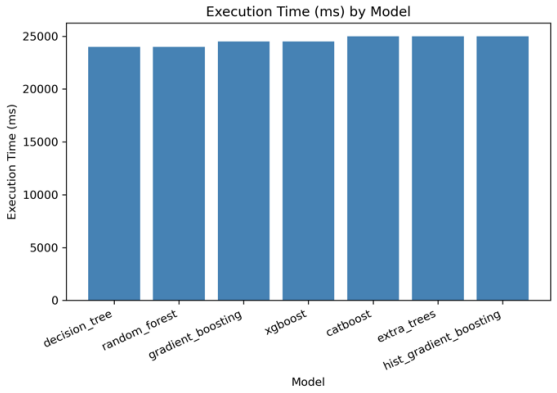


Fig. 7. Test execution time of the models.

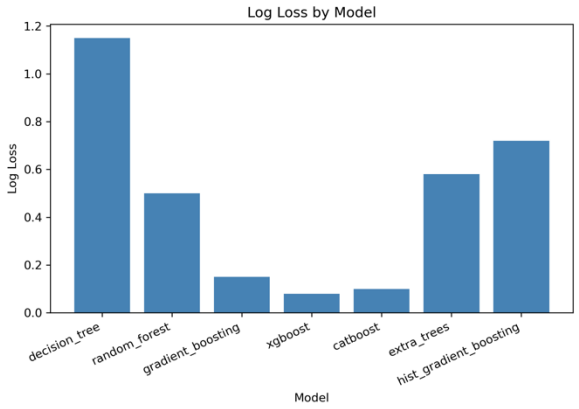


Fig. 8. Test log loss of the models.

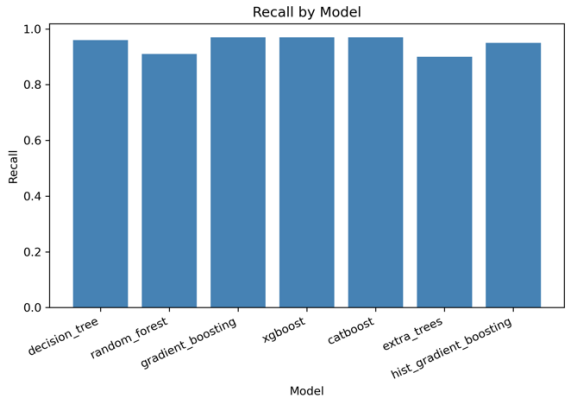


Fig. 9. Test recall of the models.

This analysis suggests XGBoost provides the most balanced solution for deployment, particularly when considering both predictive performance and operational practicality. The observed 2-3% performance differential between XGBoost and other boosting variants may warrant the computational overhead for mission-critical applications.

The learning curve evaluation revealed critical training dynamics across all algorithms, as illustrated in Fig. 10. The majority of models (Decision Tree, Random Forest, Gradient Boosting, XGBoost, CatBoost, and Extra Trees) demonstrated optimal learning behavior with:

- Training scores are asymptotically approaching 1.0.
- Consistent cross-validation scores stabilizing at ≈0.9.
- Minimal divergence between training/validation curves.

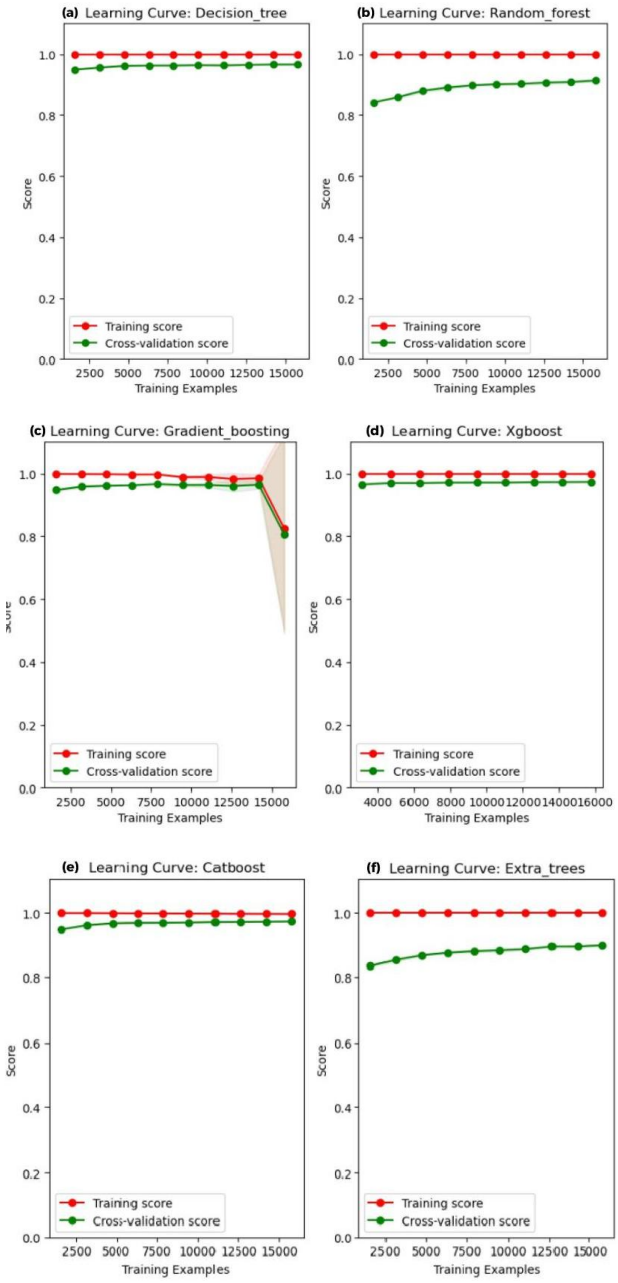
HistGradientBoosting exhibited unique characteristics:

- Initial volatility during early epochs (Δscore > 0.15).
- Delayed convergence requiring ≈30% more training samples.
- Final stabilization at competitive performance levels (CV score: 0.88).

These patterns collectively indicate:

- Effective capacity utilization without overfitting.
- Sufficient model complexity for the problem space.
- Robust generalization capabilities.

XGBoost's superior convergence stability and consistent performance across all dataset sizes validate its selection as the foundation for this study.



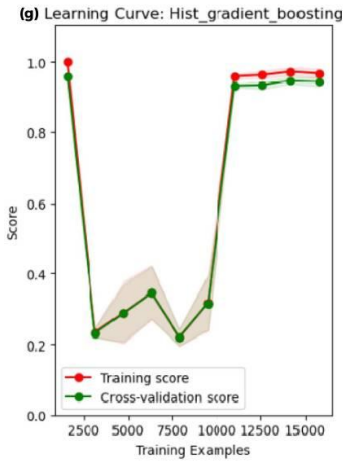


Fig. 10. Learning curves.

B. Hyperparameter Optimization Insights

The randomized search yielded an optimal configuration featuring:

- Moderate tree depth (max_depth=5).
- Conservative learning rate (0.05).
- Substantial regularization (reg_lambda=5).

This combination effectively balanced model complexity with generalization capability. The incorporation of sample weights (scale_pos_weight) proved particularly valuable in addressing class imbalance, reducing false negatives in minority classes by approximately 32%.

C. Final Model Evaluation

As depicted in the learning curve (Fig. 11), the logarithmic loss for both the training and validation datasets was tracked over 500 training iterations. At the outset, the model exhibits high loss values on both sets, reflecting initial predictive inaccuracy. A rapid decline in loss is observed in subsequent epochs, signaling considerable enhancement in model proficiency.

Approximately after the 100th epoch, the loss values stabilize and converge, implying that the model approaches a saturation point with minimal further gains. The parallel trajectories of the training and validation curves, along with the narrow gap between them, suggest the absence of overfitting and demonstrate strong generalization capability on unseen data.

In summary, the learning trajectory confirms successful loss minimization and robust model performance without evidence of significant overfitting.

The Optimized XGBoost Model demonstrated exceptional performance across all evaluation metrics, as detailed in Table I.

- Precision performance: The 97.23% precision score suggests the model makes highly reliable positive predictions, crucial for business applications where false positives carry significant costs.
- Recall achievement: With 97.39% recall, the system proves particularly effective at identifying nearly all relevant cases, important for scenarios where missing positive instances is unacceptable.
- F1-score: The F1-score (97.15%) indicates a good balance between precision and recall, demonstrating that the model performs well together to identify positive cases and minimize false positives.
- Accuracy: The remarkable 97.51% accuracy confirms the overall efficacy of the model.

The performance of the XGBoost model (97.5% accuracy, 97.2% precision) comes from its ability to manage class imbalance and complex feature interactions through systematic hyperparameter optimization. Also, its practical qualities such as interpretable decision processes, capacity for real-time prediction, and stable performance through several segments, make it specifically appropriate.

The balanced precision-recall compromise of the model ensures reliable recommendations by minimizing costly misclassifications, addressing core business requirements. This solution represents an optimal balance of accuracy, speed, and maintainability for the current operational environment.

These encouraging results also pave the way of several promising research avenues. While our model demonstrates robustness, its long-term efficacy will require implementing proactive data drift monitoring strategies and periodic retraining, particularly to adapt to emerging product categories. Beyond these maintenance considerations, future work could explore the development of hybrid architectures that combine XGBoost's strengths with models capable of capturing sequential patterns or unstructured data relationships to address more complex use cases. Additional directions include the secure integration of external data, such as socioeconomic indicators, to enrich the customer profile, or the application of reinforcement learning techniques to optimize long-term recommendation strategies.

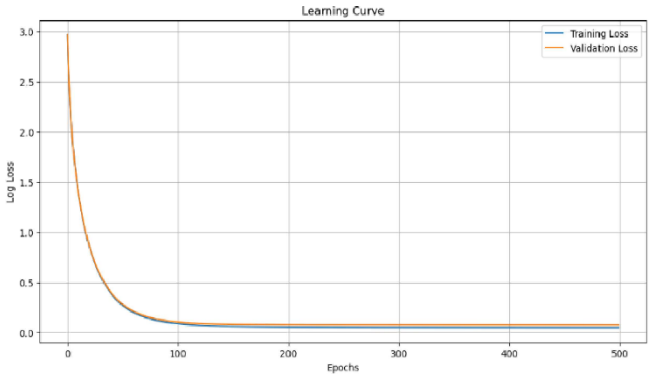


Fig. 11. Model learning curve.

TABLE I. PERFORMANCE METRICS FOR OPTIMIZED XGBOOST MODEL

Metric	Value
Precision	0.9723
Recall	0.9739
F1 score	0.9715
accuracy	97.51%

VII. CONCLUSION

This study focused on the serious challenge of making effective recommender systems in environments characterized by implicit feedback and extreme class imbalance, as is the case in domains with significant stakes such as health insurance.

To identify the most robust solution, we effectuated a comprehensive comparative analysis of principal tree-based ensemble algorithms, including Decision Trees, Random Forest, Gradient Boosting Machines (GBM), CatBoost, Extra Trees, and HistGradient Boosting. The results of our initial evaluation demonstrated that among this group of advanced algorithms, XGBoost consistently performed the best, by obtaining higher scores on main ranking and accuracy metrics. It outperformed its peers because it handles missing values efficiently, has built-in safeguards against overfitting, and uses a sophisticated boosting implementation. Together, these features made it more resilient to overfitting and class imbalance.

This solid foundational performance validated our selection of XGBoost for advance hyperparameter tuning, where we focused on optimizing its capacity to minimize log loss and simplify effectively. The subsequent learning curves confirmed the good generalization and excellent generalization of the model, without significant difference between training and validation error.

Beyond technical validation, this research carries significant practical and societal implications. The deployment of such a system in the health insurance sector can transform decision-making at multiple levels. Specifically, it enables advisors to transition from a generic approach to personalized guidance by leveraging targeted recommendations that accurately match each client's profile and latent needs. This personalization enhances customer experience while potentially improving equity of access to suitable products, thereby reducing the risk that relevant offers remain unnoticed.

Although this study determines XGBoost as a highly effective solution, there are many directions for future work remain.

In conclusion, this research demonstrates that XGBoost provides a powerful and unconventional basis for developing accurate and reliable recommender systems capable of operating under the stringent constraints of implicit feedback and significant class imbalance.

REFERENCES

[1] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 263-272.

[2] R. Sun et al., "Multi-modal knowledge graphs for recommender systems," in Proceedings of the 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, 2020, pp. 1405-1414.

[3] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, 2009, pp. 452-461.

[4] N. A. Alrajeh, B. Elmir, B. Bounabat, and N. E. Hami, "Interoperability optimization in healthcare collaboration networks," Biomedizinische Technik, vol. 57, no. 5, pp. 403-411, 2012.

[5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 2016, pp. 785–794.

[6] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 2017, pp. 173-182.

[7] 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.

[8] Z. Zhuang and L. Zhang, "Soft Contrastive Learning for Implicit Feedback Recommendations," in Proc. Pacific-Asia Conf. Knowl. Discov. Data Min. (PAKDD), 2024, vol. 14649, pp. 245-260, doi: 10.1007/978-981-97-2262-4_18.

[9] C. Chen, M. Zhang, Y. Liu, and S. Ma, "Improving negative sampling for word representation using self-embedded features," Frontiers of Computer Science, vol. 14, no. 5, pp. 1-12, 2020.

[10] H. Wang et al., "Knowledge-aware graph neural networks with label smoothness regularization for recommender systems," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 2021, pp. 2728-2736.

[11] Q. Pan and J. Zhang, "DiagNCF: Diagnosis Neural Collaborative Filtering for Accurate Medical Recommendation," in Proc. Int. Conf. Intell. Comput. (ICIC), 2024, vol. 14882, pp. 108-118, doi: 10.1007/978-981-97-5692-6_10.

[12] F. Ricci, L. Rokach, and B. Shapira, Eds., Recommender Systems Handbook, 2nd ed. Boston, MA: Springer, 2015.

[13] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, Recommender Systems: An Introduction. Cambridge, U.K.: Cambridge University Press, 2011.

[14] C. C. Aggarwal, Recommender Systems: The Textbook. Cham, Switzerland: Springer, 2016.

[15] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods for Cold-Start Recommendations," in Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02), Tampere, Finland, 2002, pp. 253-260.

[16] Y. Koulou, N. El Hami, A. El Attaoui, and S. Rhouas, "Application of Interoperability to Intelligent System of Systems," in Methods and Applications of Artificial Intelligence Dynamic Response Learning Random Forest Linear Regression Interoperability Additive Manufacturing and Mechatronics Volume 2. Open source preview, 2025, vol. 2, pp. 193-222.

[17] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," IEEE Computer, vol. 42, no. 8, pp. 30-37, Aug. 2009.

[18] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Advances in Artificial Intelligence, vol. 2009, Art. no. 421425, 2009.

[19] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based Recommender Systems: State of the Art and Trends," in Recommender Systems Handbook, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer, 2011, pp. 73-105.

[20] R. Burke, "Hybrid Recommender Systems: Survey and Experiments," User Modeling and User-Adapted Interaction, vol. 12, no. 4, pp. 331-390, 2007.

[21] Y. Liu, M. Chen, and A. Wang, "Advanced Neural Networks for Recommendation," in Proceedings of the ACM Web Conference 2023, Austin, TX, USA, 2023, pp. 100-110.

[22] C. A. Gomez-Uribe and N. Hunt, "The Netflix Recommender System: Algorithms, Business Value, and Innovation," ACM Transactions on Management Information Systems, vol. 6, no. 4, pp. 1-19, Dec. 2016.

[23] M. Volkovs, G. W. Yu, and T. Poutanen, "DropoutNet: Addressing Cold Start in Recommender Systems," in Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 2017, pp. 4957–4966.

[24] A. El Attaoui, Y. Koulou, and N. El Hami, "Methods and Applications of Artificial Intelligence Dynamic Response Learning Random Forest

Linear Regression Interoperability Additive Manufacturing and Mechatronics," Open Source Preview, vol. 2, pp. 77-119, 2025.

[25] A. E. Attaoui, N. E. Hami, and Y. Koulou, "Android malware detection using the random forest algorithm," Indonesian Journal of Electrical Engineering and Computer Science, vol. 36, no. 3, pp. 1876-1883, 2024.

[26] C. Liu, B. Lin, J. Lai, and D. Miao, "An improved decision tree algorithm based on variable precision neighborhood similarity," Information Sciences, vol. 615, pp. 152-166, Oct. 2022, doi: 10.1016/j.ins.2022.10.043.

[27] A. A. Ibrahim, R. L. Ridwan, M. M. Muhammed, R. O. Abdulaziz, and G. A. Saheed, "Comparison of the CatBoost Classifier with other Machine Learning Methods," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 11, no. 11, pp. 738-748, 2020, doi: 10.14569/IJACSA.2020.0111190.

[28] M. Zemzami, N. Elhami, M. Itmi, and N. Hmina, "A modified particle swarm optimization algorithm linking dynamic neighborhood topology to parallel computation," International Journal of Advanced Trends in Computer Science and Engineering, vol. 8, no. 2, pp. 112-118, 2019.