Time-Aware Deep Learning for Predicting the Next Product in Banking Customer Journeys

1st Wilson Geronimo
Department of Mathematics and Statistics
Thompson Rivers University
British Columbia, Canada
geronimorodriguezw22@mytru.ca

2nd Dr. Stan Miles
Faculty of Economics
Thompson Rivers University
British Columbia, Canada
stanmiles@tru.ca

3rd Dr. Dahae Jeong
Faculty of Marketing
Thompson Rivers University
British Columbia, Canada
djeong@tru.ca

***Abstract*—This study presents a sequential deep learning approach for personalized multi-product recommendation in commercial banking, leveraging customers' historical transaction behavior and demographic attributes. Real-world banking data is preprocessed to extract time-aware sequences of product purchases enriched with socio-demographic and transactional features, which are fed into a two-layer LSTM neural network and benchmarked against tree-based classifiers to predict the next likely product a customer may acquire. Evaluated on actual customer data, the model achieves a Top-1 Accuracy of 54%, rising to 96% at Top-10, demonstrating strong capability in ranking relevant products and capturing behavioral patterns. These results highlight the model's effectiveness for real-world deployment in digital banking platforms, offering a scalable foundation for intelligent recommendation systems that support cross-selling, real-time personalization, and targeted marketing strategies. The complete implementation and data pipeline are publicly available online [1].**

***Index Terms*—Recommender Systems, Deep Learning, LSTM, Sequential Modeling, Multi-Product Recommendation, Customer Behavior Modeling, Financial Services, Transaction Data, Top-K Evaluation, Unsupervised Representation Learning, Embedded Feature Learning, Data Mining, Behavioral Trajectories**

I. Introduction

In today's data-driven economy, financial institutions are increasingly turning to personalized recommendation systems to better understand customer needs and improve cross-selling strategies. Recommender systems, once primarily confined to e-commerce platforms like Amazon and Netflix, are now being applied in domains such as digital banking, where they offer the potential to enhance customer experience and financial inclusion by tailoring financial product offerings to individual behaviors and preferences [2], [3]. As financial portfolios and customer data become more complex and multidimensional, banks face the challenge of making timely, relevant product suggestions without overwhelming users with information overload [4]. This has led to a surge of research on applying machine learning techniques — particularly deep learning — to capture complex, time-dependent behavior from customer transaction histories and demographic features.

Traditional recommendation models such as collaborative filtering and content-based systems have shown promise in predicting user preferences across domains [5], [6]. However, these approaches often fall short in the financial services sector due to the sequential, sparse, and context-sensitive nature of financial product interactions. Unlike consumer goods, financial products are often adopted in a temporal and strategic sequence, influenced by life events, income, and behavioral patterns. Recent studies have begun to explore sequence-aware and context-aware recommender systems for banking, utilizing models such as recurrent neural networks (RNNs), attention mechanisms, and hybrid clustering techniques [2], [7]. These advanced models are designed to understand not just what products a customer prefers, but when and in what order they are likely to adopt them.

A particularly promising direction is the use of LSTM-based neural networks to model user behavior over time, as they are well-suited for capturing long-term dependencies in sequential data. By encoding past financial interactions into a structured format and feeding them into deep sequence models, it is possible to make highly accurate predictions about the next product a customer is likely to acquire. This approach has been applied in various studies with significant results, including outperforming traditional tree-based models and matrix factorization methods in Top-K accuracy metrics [7], [8]. In line with these findings, the current study proposes a sequential product recommendation model for commercial banks that leverages LSTM layers to learn from historical transaction sequences, enriched with demographic and transactional features.

Our proposed model was evaluated on real-world banking data, demonstrating strong performance in product prediction tasks. Specifically, the model achieved a Top-1 Accuracy of 54%, with performance increasing to 96% at Top-10, confirming its effectiveness in ranking relevant products. These results indicate that the model not only understands customer behavior patterns but is also capable of generating actionable recommendations in a banking context. Compared to conventional recommendation systems, our approach supports dynamic product targeting strategies and improves the efficiency of personalized marketing campaigns.

This paper contributes to the growing body of literature on AI-driven banking solutions by presenting an end-to-end LSTM-based recommendation architecture tailored for multi-product financial environments. Building upon prior work in unsupervised learning, customer profiling, and temporal modeling, this research aims to bridge the gap between academic recommender systems and their practical deployment in financial institutions [3], [4]. By capturing sequential behavior and real-time signals, our approach lays the groundwork for more adaptive, accurate, and user-centric recommendation systems in the financial services sector.

Despite these advances, many recent studies in banking recommendation still focus on overall predictive accuracy or binary classification metrics, such as accuracy or AUC, without assessing the effectiveness of ranked recommendation outputs. For instance, Mavundla et al. [9], Boustani et al. [10], and Met et al. [11] primarily report binary or Top-1 classification results, neglecting metrics such as Precision@K or Recall@K that are more actionable in practical deployment. In contrast, this work contributes to the field by explicitly evaluating recommendation performance across multiple K values and highlighting performance trade-offs that can guide the design of cross-selling and personalization strategies in digital banking.

II. Literature Review

The rapid growth of customer data availability in the financial sector has driven a surge of interest in personalized product recommendation systems. Traditional approaches to recommender systems, such as collaborative filtering (CF) and content-based methods, have proven effective in domains like e-commerce and entertainment platforms [5], [6]. However, their effectiveness in the financial domain is more limited due to the sparse and temporally sensitive nature of banking interactions [7]. While these classic techniques model static relationships between users and items, they often fall short in understanding the sequential patterns and contextual dependencies inherent in financial behavior [8].

To address this limitation, research has increasingly shifted toward more dynamic recommendation models, particularly those involving deep learning. In the banking domain, deep learning models allow for the integration of both behavioral sequences and user metadata, enabling more adaptive and personalized recommendations [4]. Recurrent Neural Networks (RNNs), and especially Long Short-Term Memory (LSTM) networks, have gained traction due to their ability to model long-term dependencies in sequential purchase patterns [2]. For example, Shah [7] proposed a model that leverages financial performance metrics and customer traits to predict the next best product, achieving promising results compared to rule-based systems.

In parallel, attention-based and Transformer models have become increasingly relevant in recommendation systems due to their superior ability to model complex temporal and contextual relationships. Liu et al. [2] introduced a self-attention-based encoder model (BRec) that outperforms existing deep learning benchmarks by capturing sequential dependencies in banking product acquisition. The model incorporates a novel temporal representation of user data and achieves top-1 precision of 98.9% on a Santander dataset, demonstrating the power of attention mechanisms over traditional RNN-based approaches.

Collaborative filtering continues to be widely applied, particularly in hybrid and graph-based systems. Ghiye et al. [12] proposed an adaptive CF model that incorporates personalized time decay functions, acknowledging the non-stationarity of user preferences in financial domains. Their approach discounts older user-product interactions based on learned decay rates, resulting in more timely and accurate predictions. Similarly, Leng et al. [13] proposed a graph-based collaborative filtering framework for financial product recommendation that leverages high-order relations between users and categories to enhance recommendation quality.

Other studies have combined cloud-based infrastructure with machine learning models to enable scalable and real-time financial product recommendations. Madasamy [14] presented a hybrid recommender system architecture that utilizes CNNs for feature extraction and collaborative filtering to enhance personalization. This approach focuses on improving accessibility for banking and insurance recommendations by integrating user behavior with cloud-enhanced prediction engines. Meanwhile, Liu et al. [2] also emphasized the importance of ethical considerations and fairness when deploying AI-driven recommender systems in banking environments, where financial impact and customer trust are highly sensitive.

Despite these advances, most of the existing literature still focuses on either e-commerce settings or individual customers, with limited attention to sequence-aware prediction in banking environments where product adoption follows complex temporal patterns. As noted in recent work [3], the vast majority of banking recommendation models are not optimized for handling the sequential nature of customer behavior trajectories or integrating diverse customer metadata. Moreover, small and medium enterprises (SMEs) and corporate customers remain underrepresented in recommendation studies [11].

A notable limitation in the existing literature is the overreliance on binary classification metrics, often overlooking ranked recommendation evaluations. Mavundla et al. [9] use only accuracy and F1-score to evaluate their cross-sell prediction models in health insurance, while Boustani et al. [10] employ deep learning for consumer loan recommendations but stop short of analyzing ranked recommendation performance. Similarly, Met et al. [11] focus exclusively on predictive accuracy, with no attention to recommendation list structure or size.

In parallel, several studies have adopted sequential deep learning models in banking. Liu et al. [15] and Valendin et al. [16] explore Transformer and RNN-based architectures for predicting product adoption patterns, while Kalkan and Şahin [17] show how attention-enhanced RNNs significantly improve cross-sell prediction accuracy in retail settings. However, most of these works either focus on narrow Top-1 metrics or do not evaluate precision-recall trade-offs across different list lengths. This gap motivates the current study's focus on both sequential modeling and practical recommendation list evaluation.

Building on these foundations, the present study proposes a sequential LSTM-based model designed specifically for commercial banks. Unlike traditional CF or attention-based methods, our approach directly captures customer behavior through ordered sequences of product interactions, combined with socio-demographic and transactional features. Evaluated on real-world banking data, the model achieves strong performance on Top-K accuracy metrics demonstrating its effectiveness in modeling both behavioral patterns and timing of financial product adoption.

III. Identified Research Gaps and Study Contribution

While recommendation models are increasingly applied in banking, important methodological and practical gaps persist in the literature. First, many studies prioritize classification accuracy or AUC, which provide limited insight into the effectiveness of ranked lists used in real recommendation systems. Recent work by Mavundla et al. [9], Boustani et al. [10], and Met et al. [11] demonstrates this limitation, as they omit metrics such as Precision@K or Recall@K that are critical for cross-sell campaigns and targeted marketing.

Second, there is limited guidance on how many products should be recommended to each customer. While some models, like those from Liu et al. [15], demonstrate Top-3 recommendation performance, they do not systematically analyze the trade-offs associated with different recommendation list lengths. Studies such as Arocha Prada [18] consider heuristic or collaborative filtering methods but do not examine variation in K for managerial insight.

Third, despite the success of deep learning in other domains, its application to banking product recommendation remains sparse. While sequential models such as LSTM and GRU have been used in customer modeling [16], [19], many implementations lack integration of demographic and transactional data or do not evaluate across multiple K thresholds. This study addresses these gaps by proposing an LSTM-based model trained on real customer-product sequences and evaluating it using Top-K Accuracy, Precision@K, and Recall@K metrics across multiple recommendation list sizes.

IV. Problem Definition and Objectives

In the context of digital banking, customers interact with a variety of financial products over time, forming behavioral trajectories that reflect their evolving financial needs. The goal of this research is to develop an intelligent recommendation system capable of predicting the next product a customer is likely to adopt, based on both their past product usage sequences and their demographic and transactional profiles.

Formally, let each customer c be represented by a sequence of previously acquired products $P_c = \{p_1, p_2, \ldots, p_n\}$ and an associated feature vector X_c containing demographic attributes (e.g., age group, education level) and transactional characteristics (e.g., transaction count, amount, days since last purchase). The task is to learn a function $f(P_c, X_c) \rightarrow \hat{Y}_c$, where $\hat{Y}_c$ is a ranked list of predicted products, with the true next product ideally appearing within the top-K positions.

This problem is approached as a recommendation task with sequence-aware modeling, where the aim is to generate a Top-K list of product recommendations for each user, prioritizing relevance and timing. The predictive model should leverage both the temporal structure of product adoption and the static behavioral features to make personalized and context-aware predictions.

The primary objectives of this study are as follows:

- To design a machine learning model capable of capturing customer behavior sequences using deep learning techniques, particularly LSTM networks.
- To integrate demographic and transaction-based user data to enhance the personalization of recommendations.
- To maximize the model's performance using Top-K accuracy metrics across multiple K values.
- To evaluate the proposed system using real-world banking data in order to ensure practical applicability.

V. Proposed Methodology

The proposed recommendation model leverages a sequential deep learning architecture based on Long Short-Term Memory (LSTM) networks to predict the next financial product a customer is likely to acquire. The approach combines past product usage sequences with demographic and transactional features to generate personalized Top-K recommendations.

A. Input Representation

Each customer is represented by a chronological sequence of previously acquired banking products, alongside an associated set of static and behavioral features. These features include demographic attributes (e.g., age group, education level, income group), product metadata (e.g., application type), and transaction-based variables (e.g., number of transactions, average transaction amount, days since last transaction, days since product opened). All continuous features are normalized using z-score standardization.

To model customer behavior as a temporal process, product sequences are grouped per customer and ordered by product opening date. Customers with only one product are excluded to ensure meaningful temporal sequences. Each input sequence X is a 2D matrix of shape (T, F), where T is the number of time steps (products acquired before the prediction point) and F is the number of features per product instance. Sequences are padded post-hoc to ensure equal length across the dataset.

B. Mathematical Background of LSTM

Long Short-Term Memory (LSTM) networks, introduced by Hochreiter and Schmidhuber in 1997 [20], are a type of Recurrent Neural Network (RNN) designed to model long-range dependencies in sequential data. Unlike traditional RNNs, which suffer from vanishing and exploding gradient problems, LSTMs introduce gated mechanisms that control the flow of information through time, enabling them to retain relevant context over longer sequences.

An LSTM cell operates through three main gates:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{4}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

Here, x_t is the input at time t, h_t is the hidden state, and C_t is the cell state. The gates (f_t, i_t, o_t) control how much information is forgotten, updated, and passed to the next state. These mechanisms enable LSTMs to learn temporal dependencies across variable-length sequences, which is particularly useful in modeling customer product adoption over time.

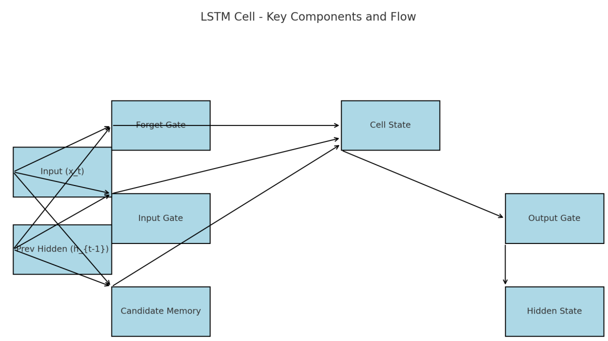

Fig. 1: Schematic of an LSTM cell showing its internal components and data flow. Information from the previous hidden state and current input is processed through forget, input, and output gates to update the cell state and produce a new hidden state.

C. Model Architecture

The model architecture consists of two stacked LSTM layers followed by fully connected (dense) layers. The LSTM layers are used to learn temporal dependencies in the sequential data, capturing patterns in the customer's purchase trajectory. The final hidden state of the LSTM is passed to a dense layer with ReLU activation, followed by a softmax output layer that returns a probability distribution over all possible product classes.

- Layer 1: LSTM (64 units, return sequences=True)
- Layer 2: LSTM (32 units)
- Dense: Fully connected layer with 64 units and ReLU activation
- Output: The final dense layer applies a softmax activation across 26 output units, corresponding to the 26 unique financial products present in the dataset.

D. Training Procedure

The model is trained using the `sparse_categorical_crossentropy` loss function, as the target is a single integer class representing the next product. The Adam optimizer is used with a learning rate of 0.001. The training is conducted over 20 epochs with a batch size of 32.

Model performance is monitored using Top-K accuracy metrics across $K = \{1, 3, 5, 10\}$ on a hold-out test set.

The model is implemented in TensorFlow/Keras, and all experiments are performed using padded input sequences to ensure consistent tensor dimensions. Early stopping and dropout were explored but not included in the final configuration, as the model did not exhibit signs of overfitting within the given training window.

VI. Data Description and Variables

The dataset used in this study originates from structured banking records, consolidated from multiple SQL sources including customer account data, product metadata, and transactional history. The dataset reflects real-world behavior and consists of records representing individual customer-product relationships across savings, credit, and insurance services. Each row corresponds to a single product acquired by a customer, enriched with demographic and transactional variables.

The dataset includes the following variables:

- customer_id: A unique identifier for each customer.
- income_group: Estimated customer income, grouped into brackets.
- age_group: Customer's age bracket at the time of product acquisition.
- education_level: Education level (e.g., Secondary, Master).
- aplicacion: High-level product category (e.g., CUENTAS, TDC, CERTIFICADOS).
- producto: The specific financial product acquired (e.g., Ahorro TDD, INFINITE, GOLD). The dataset includes 26 distinct products across various banking categories.
- fecha_apertura: The date on which the product was opened.
- fecha_transaccion: The most recent transaction date associated with the product.
- cantidad_transacciones: Number of recorded transactions on the product.
- monto_transaccion: Total monetary volume of all recorded transactions.

Several important filtering criteria were applied to ensure consistency and relevance of the data:

- All products acquired throughout a customer's lifetime are considered in the analysis, providing a full historical trajectory of their financial behavior.
- Only products linked to personal banking customers are included; products associated with corporate or enterprise entities are excluded.
- Customers who are also employees of the institution are filtered out to avoid internal behavior patterns that may differ from typical consumer behavior.
- Customers with only one product are excluded from modeling, since sequential modeling requires at least two time steps.

The final dataset spans thousands of customers with multiple product interactions and a rich set of behavioral and demographic signals. These records form the basis for the sequential modeling approach described in the following sections.

VII. Data Preparation and Feature Engineering

To train a sequential deep learning model on customer product behavior, extensive data preprocessing and transformation steps were performed. These steps ensured that the raw dataset could be transformed into structured, numerical input suitable for time-aware modeling using LSTM networks.

A. Categorical Encoding

All categorical variables were transformed into numeric representations to enable model consumption. Specifically:

- producto, aplicacion, and education_level were encoded using integer label encoding, maintaining one-to-one mappings between original values and numeric identifiers.

B. Numerical Feature Normalization

To ensure that numerical features contributed proportionally during training, z-score standardization was applied to all continuous variables:

- income_group, age_group, cantidad_transacciones, and monto_transaccion were scaled using the mean and standard deviation of each variable.

C. Temporal Feature Engineering

Two new temporal features were derived to capture recency and account tenure:

- days_since_last_transaction: Computed as the number of days between a fixed evaluation date and the most recent transaction for each product.
- days_since_product_opened: Measured as the number of days since the product was originally opened.

Both were also standardized using z-score normalization. Missing or zero values in transaction-related dates were treated as NaN and filled using median imputation before transformation.

D. Sequence Construction and Padding

To capture behavioral progression, the data was grouped chronologically by customer and sorted by fecha_apertura to construct temporal product acquisition sequences. For each customer, all but the final product record were used as input, and the final product served as the prediction target.

These sequences varied in length depending on the number of products acquired. To handle this variability and maintain consistent tensor shapes for training, sequences were padded with zeros post-hoc to the maximum observed sequence length using Keras' `pad_sequences()` function.

E. Filtering Criteria

Customers with only one product record were excluded, as sequential prediction requires at least two time steps. Furthermore, missing demographic or transaction-related data was either imputed or filtered to preserve consistency across the training dataset. The final dataset includes 84,815 customers and 232,920 product interactions.

The final result was a three-dimensional tensor of shape (N, T, F) where:

- N is the number of customers,
- T is the maximum sequence length (time steps),
- F is the number of features per product interaction.

This structure served as the input for the LSTM model, allowing it to learn from both the temporal order of product usage and the embedded behavioral context of each transaction.

VIII. Experimental Setup and Results

This section presents the experimental design used to evaluate the proposed sequential recommendation model. The dataset was split into training and test sets by stratified customer sampling, ensuring that customer-level sequences were not split across datasets. Each customer's input sequence includes all but their final product interaction, which serves as the ground-truth prediction target.

The model's performance is evaluated using Top-K accuracy metrics for $K = \{1, 3, 5, 8, 10\}$, assessing the proportion of test

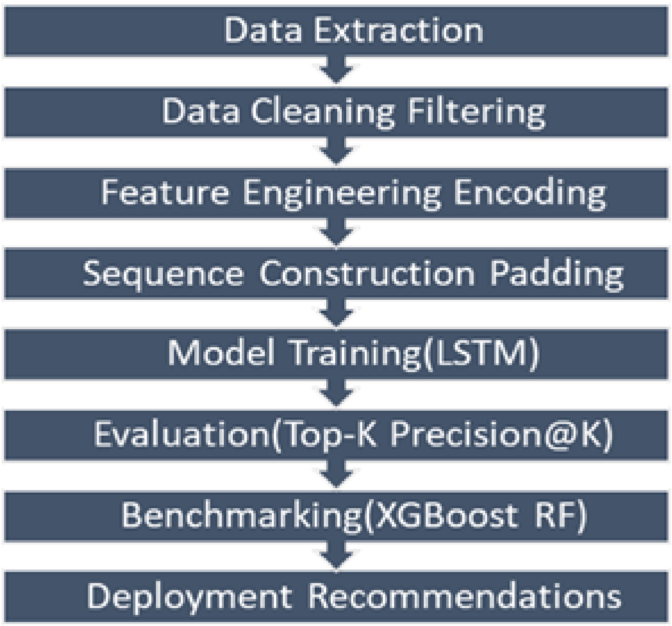

Fig. 2: End-to-end workflow of the project, including data processing, feature engineering, sequence modeling, model training, evaluation, and deployment.

cases in which the correct next product appears within the top K predicted classes. These metrics are highly relevant for real-world banking recommendation scenarios, where suggesting a short list of relevant products is more practical than requiring an exact top-1 match.

A. Model Training Configuration

The model was implemented in TensorFlow/Keras and trained on padded sequences of shape (T, F), where T is the sequence length and F is the number of features per time step. To optimize model performance, a range of hyperparameter values were explored using manual tuning based on validation performance. Table I summarizes the hyperparameter options considered and the final values selected.

TABLE I: LSTM Model Hyperparameters.

Hyperparameter	Values Tried	Selected
LSTM Units (Layer 1)	32, 64, 128	64
LSTM Units (Layer 2)	16, 32, 64	32
Dense Units	32, 64, 128	64
Learning Rate	0.01, 0.001, 0.0005	0.001
Batch Size	16, 32, 64	32
Epochs	10, 15, 20, 25	20
Loss Function	CC, SCC	SCC
Optimizer	SGD, RMSprop, Adam	Adam

Note: CC = `categorical_crossentropy`, SCC = `sparse_categorical_crossentropy`

B. Top-K Accuracy Results

The model's Top-K accuracy was evaluated on the test dataset. As shown in Table II, the model achieves strong performance across all K values. The Top-1 accuracy is 54%, indicating that the model predicts the correct next product more than half the time as its top recommendation. As K increases, performance improves significantly, with Top-10 accuracy reaching 96%, demonstrating the model's effectiveness in narrowing down the most relevant product options.

TABLE II: Top-K Accuracy on the test set.

Top-K	Accuracy@K
1	54%
3	77%
5	86%
8	93%
10	96%

These results validate the model's ability to accurately recommend a concise set of next-best products based on customer purchase behavior and profile data. Such predictive power can support targeted marketing campaigns, cross-selling strategies, and improved digital customer engagement.

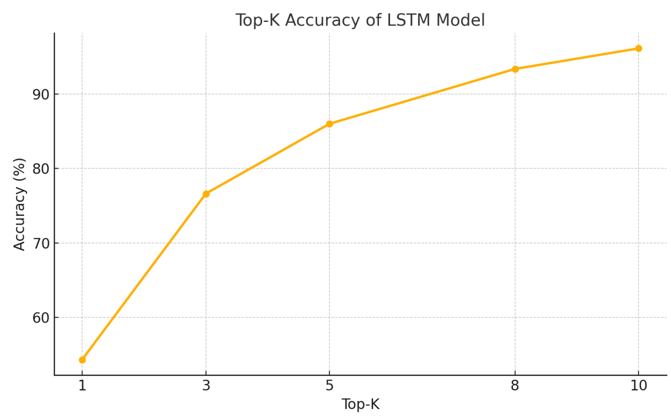

Fig. 3: Top-K Accuracy achieved by the proposed LSTM model. Accuracy increases steadily with higher values of K, reaching over 96% at Top-10.

C. Baseline Models and Comparative Evaluation

To benchmark the performance of the proposed LSTM-based recommendation model, two widely-used classical machine learning algorithms were also implemented and evaluated: XGBoost and Random Forest. These models were trained on a flattened and aggregated version of the dataset, where each customer's product history and behavioral features were transformed into a single-row representation.

Features included product acquisition history (one-hot encoded), aggregated transaction statistics (sum and mean), recency of last transaction, and demographic attributes. The last acquired product was used as the prediction target. Both models were trained using default hyperparameters with minor tuning, and evaluated using Top-1 and Top-5 Accuracy on the same test set used for the LSTM model.

TABLE III: Comparison of Model Performance

Model	Top-1 Accuracy	Top-5 Accuracy
LSTM (Proposed Model)	54%	86%
XGBoost	56%	**89%**
Random Forest	54%	86%

While XGBoost achieves slightly higher Top-1 and Top-5 accuracy compared to the LSTM model, the proposed architecture is designed to model the temporal dynamics of product adoption, which is not captured by tree-based methods. Additionally, LSTM benefits from modeling sequential dependencies that may be

more valuable in long-term prediction scenarios and product lifecycle modeling. These findings suggest that both deep learning and ensemble methods are competitive for financial product recommendation, and may be combined in future hybrid systems.

Impact of Outlier Treatment. To evaluate the robustness of the model, we applied an outlier handling strategy by capping the most extreme values in key numerical features—`income_group`, `age_group`, `cantidad_transacciones`, and `monto_transaccion`—at their respective 99th percentiles. The `fecha_apertura` field was also adjusted by setting a minimum year of 2011, as previously discussed.

Table IV shows a side-by-side comparison of the Top-K accuracy results with and without outlier capping. The differences are minimal across all metrics, demonstrating that the model's performance is stable and not overly affected by extreme values.

TABLE IV: Top-K Accuracy with and without outlier capping.

Top-K	Original Accuracy	With Outlier Capping
1	54%	54%
3	77%	77%
5	86%	86%
8	93%	94%
10	96%	96%

This confirms that the predictive insights remain consistent even after addressing the influence of potential outliers in the dataset.

IX. Evaluation Beyond Accuracy

While Top-K Accuracy is a widely used metric to assess recommender systems, it does not fully capture the quality or usefulness of the ranked predictions. In real-world banking environments, particularly in digital channels where product recommendations are surfaced in sets (e.g., Top-3 or Top-5 suggestions), other evaluation metrics such as Precision@K and Recall@K are more indicative of the system's practical value.

Precision@K measures the proportion of recommended products in the top K that are actually relevant, while Recall@K captures the proportion of relevant products that were included in the top K recommendations. Since each test instance in this study involves predicting a single next product, Recall@K simply indicates whether the true next product is within the top K predictions, and Precision@K reflects how sparse the relevant predictions are in the top list.

TABLE V: Precision@K and Recall@K for various values of K

Top-K	Precision@K	Recall@K
1	54%	54%
3	26%	77%
5	17%	86%
8	12%	93%
10	10%	96%

This analysis contrasts with studies like Boustani et al. [10] and Met et al. [11], which evaluate only Top-1 or binary accuracy and provide no insights into how list length affects performance. Our approach shows that the LSTM model offers both high recall and acceptable precision even as K increases, indicating its robustness in ranking relevant products. Moreover, while Transformer-based methods like those of Liu et al. [15] demonstrate high precision in short recommendation lists, they do not explore performance at larger K, leaving a practical gap for marketing use cases that this study directly addresses.

The results reveal that the proposed model maintains high recall across all values of K, especially at Top-10, where it reaches 96%. This means the model almost always includes the correct product in its top suggestions. Although Precision@K decreases with larger K — as expected — the performance remains within acceptable ranges. In practice, this trade-off is highly favorable: customers presented with 5 or 10 highly relevant options are more likely to engage, even if the correct one is not ranked first. This aligns with the goal of digital recommendation systems in commercial banking — to assist, not dictate, customer decision-making.

From a business perspective, high Recall@K enhances the effectiveness of cross-selling strategies, while modest Precision@K allows for diversity in recommendations. This balance is particularly useful in banking, where offering multiple personalized options can drive greater customer satisfaction, reduce churn, and promote financial inclusion.

X. Risks, Limitations, and Ethical Considerations

While the proposed system shows promising results, it is important to consider its limitations and ethical implications, particularly in a sensitive domain like finance. First, the model is trained on historical customer data that may encode latent biases — such as income-based access patterns or age-related product preferences — which could lead to unintended reinforcement of socioeconomic inequalities.

The dataset includes only personal banking customers and explicitly excludes corporate clients and employees. While this decision ensures a consistent and homogeneous sample, it may also limit the generalizability of the results to broader segments. Furthermore, customers with only one product were excluded from training, potentially introducing survivorship bias by modeling only those with richer financial histories.

Another key limitation is the reliance on static demographic and transactional features. While these features are useful, they do not account for changes in life circumstances, preferences, or goals — factors that can significantly influence financial behavior. Incorporating real-time behavioral signals or feedback loops could improve responsiveness and personalization.

Ethically, the model raises important considerations around fairness and transparency. Using education or income data for predictions, while predictive, must be handled carefully to avoid unfair profiling. Institutions deploying such models should adopt transparent practices and offer customers the ability to understand or contest recommendation logic, in line with explainable AI (XAI) principles.

There is also the question of privacy. Although the model works with anonymized and aggregated data, real-world implementations must prioritize data security, consent, and adherence to regulatory standards such as GDPR. Care must also be taken to avoid over-personalization, which could lead to overly aggressive or manipulative marketing.

Finally, there is the challenge of model drift: customer behavior and financial products evolve over time. Models trained on historical data may become stale, requiring continuous monitoring, retraining, and validation to remain effective and ethical.

Key limitations of this study include:

- Exclusion of customers with only one historical product, potentially limiting cold-start applicability.
- Use of primarily static or pre-aggregated features, without dynamic real-time user signals.
- Focus solely on personal banking customers, with no inclusion of corporate or enterprise clients.
- Temporal drift risk, as customer behaviors and product offerings may evolve over time, requiring regular retraining.

XI. Concluding Remarks and Future Directions

This study demonstrates that time-aware deep learning models can effectively predict the next product in a customer's banking journey, leveraging temporal product sequences and enriched user profiles. The model utilizes a stacked LSTM architecture to capture behavioral dependencies and shows strong Top-K accuracy across multiple thresholds, achieving 96% at Top-10. It demonstrates that deep learning can be effectively applied to banking data when combined with proper sequence modeling and preprocessing techniques.

Unlike most existing studies that emphasize classification accuracy, this research provides a broader view of model performance using Top-K Accuracy, Precision@K, and Recall@K. These metrics reflect real-world conditions in digital banking environments, where customers are presented with ranked product suggestions rather than a single option.

Our sequential model not only captures behavioral dependencies over time but also supports effective ranking of next-best product options with high recall at $K = 5$ and $K = 10$. This offers practical value for marketing teams aiming to increase product uptake through targeted and personalized recommendations.

Furthermore, by evaluating model performance across different list sizes, this study provides concrete managerial insight on how many product recommendations are optimal under different campaign strategies—an aspect rarely addressed in the literature [15], [18]. The integration of sequential product histories, demographic profiles, and multiple evaluation metrics advances the current state of recommender systems for financial services and provides a replicable framework for future studies.

Future work may explore the integration of collaborative filtering signals to incorporate customer-to-customer similarity, as well as attention-based mechanisms to improve model interpretability. Real-time personalization through reinforcement learning or multi-armed bandits could enhance adaptability or leveraging Transformer-based architectures, such as BERT4Rec, for long-term sequence modeling. Additionally, future studies could extend the recommendation framework to include corporate and business banking products. Unlike individual customers, business entities exhibit more complex financial behaviors and product needs—such as managing payroll, working capital, credit lines, or investment accounts—which require richer feature sets and different modeling strategies. Expanding the model to support business segments would demonstrate its scalability and significantly increase its practical impact in enterprise banking environments. Finally, coupling recommendations with customer satisfaction metrics or business KPIs could help align technical accuracy with strategic goals, further bridging the gap between AI performance and measurable value in digital banking platforms.

Data and Code Availability

All datasets used in this study are derived from anonymized financial product records and cannot be publicly distributed in raw form due to privacy restrictions. However, the complete data preprocessing pipeline, complete implementation and evaluation code are publicly available online [1].

References

[1] W. Geronimo, "Time-aware deep learning for predicting the next product in banking customer journeys - code repository," https://github.com/wgeronimor/next-bank-customer-product-prediction, 2025.

[2] D. Liu, G. P. Farajalla, and A. Boulenger, "Brec the bank: Context-aware self-attentive encoder for banking products recommendation," 2022.

[3] D. Wu and X. Li, "A systematic literature review of financial product recommendation systems," *Information*, vol. 16, no. 3, p. 196, 2025.

[4] X. Liu, "Unsupervised deeply embedded online multi-product recommendation of commercial banks based on behavior data trajectory," 2024.

[5] S. C. Mana and T. Sasipraba, "A machine learning based implementation of product and service recommendation models," pp. IEEE: 1–5, 2021.

[6] M. Maqbool, S. S. Zia, M. Naseem, and S. A. Ali, "Product-pair recommendation for customers using machine learning technique," *Journal of Information, Communication and Technology Robotics Applications*, vol. 11, no. 1, pp. 38–46, 2020.

[7] P. U. Shah, "Customer-specific product recommendation based on financial performance," 2024.

[8] B. Barreau, "Machine learning for financial products recommendation," 2020.

[9] K. Mavundla, S. Thakur, E. Adetiba, and A. Abayomi, "Predicting cross-selling health insurance products using machine-learning techniques," *Journal of Computer Information Systems*, pp. 1–12, 2024.

[10] N. Boustani, A. Emrouznejad, R. Gholami, and O. Despić, "Improving the predictive accuracy of the cross-selling of consumer loans using deep learning networks," *Annals of Operations Research*, vol. 339, no. 1, pp. 613–630, 2024.

[11] I. Met, A. Erkoç, S. E. Seker, M. A. Erturk, and B. Ulug, "Product recommendation system with machine learning algorithms for sme banking," *International Journal of Intelligent Systems*, vol. 2024, no. Article ID 5585575, pp. 1–21, 2024.

[12] A. Ghiye, B. Barreau, L. Carlier, and M. Vazirgiannis, "Adaptive collaborative filtering with personalized time decay functions for financial product recommendation," *In Proceedings of the 17th ACM Conference on Recommender Systems (RecSys '23)*, 2023.

[13] Z. Leng, J. Liu, X. Wang, and L. Zheng, "Customer-category interest model: A graph-based collaborative filtering model with applications in finance," 2022.

[14] S. Madasamy, "Cloud-enhanced ai for personalized banking and insurance product recommendations," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 06, 2024.

[15] D. Liu, G. P. Farajalla, and A. Boulenger, "Transformer-based banking products recommender system," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 2641–2645.

[16] J. Valendin, T. Reutterer, M. Platzer, and K. Kalcher, "Customer base analysis with recurrent neural networks," *International Journal of Research in Marketing*, vol. 39, no. 4, pp. 988–1018, 2022.

[17] E. Kalkan and C. Şahin, "Evaluating cross-selling opportunities with recurrent neural networks on retail marketing," *Neural Computing and Applications*, vol. 35, no. 6, pp. 6247–6263, 2023.

[18] D. E. A. Prada, "Sistemas de recomendación de productos para banca empresarial y corporativa: Un análisis comparativo," Master's thesis, Universidad de los Andes, 2018.

[19] M. Sarkar and A. D. Bruyn, "Lstm response models for direct marketing analytics: Replacing feature engineering with deep learning," *Journal of Interactive Marketing*, vol. 53, pp. 80–95, 2021.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.