Journal of Open Innovation: Technology, Market, and Complexity 10 (2024) 100303

Contents lists available at ScienceDirect

Journal of Open Innovation: Technology, Market, and Complexity

journal homepage: www.sciencedirect.com/journal/journal-of-open-innovation-technology-market-and-complexity

A personalized product recommendation model in e-commerce based on retrieval strategy

Duy-Nghia Nguyen [a,b], Van-Ho Nguyen [a,b,*], Trang Trinh [a,b], Thanh Ho [a,b], Hoanh-Su Le [a,b]

[a] *University of Economics and Law, Ho Chi Minh City, Viet Nam*
[b] *Vietnam National University, Ho Chi Minh City, Viet Nam*

ARTICLE INFO

Keywords:
Conceptual model
Machine learning
ML for business
Personalized recommender system
Product recommendation
Retrieval strategy

ABSTRACT

In recent years, online shopping is one of the routine parts in people's life. It is convenient and takes less effort to purchase it. Regarding the increasing revolution of e-commerce businesses, recommendation engine plays a crucial role in them. Recommendation engines are very popular and easy to implement to their platform nowadays. Due to the extremely high competition of e-commerce businesses, the operation needs to integrate the recommender wisely. This study presents a comprehensive approach to improving user experience and engagement on e-commerce platforms through the implementation of an implicit personalized product recommendation engine. Collaborating with the H&M Group, the research combines the strength of each recommending algorithms which are collaborative filtering, popularity, and Bayesian personalized ranking to develop a robust recommendation system. By leveraging a retrieval strategy that combines multiple algorithmic techniques and evaluating candidates using machine learning models which comprise LightGBM and Deep Neural Network, the study achieves promising results. The authors utilize two popular technical metrics to evaluate their models which are mean average precision at K candidates (MAP@K) and mean average recall at K candidates (MAR@K). The empirical result indicates that the LightGBM model has remarkable performance than Deep Neural Network model, which are 0.06 versus 0.02 respectively in MAP@K and 0.03 versus 0.01 respectively in MAR@K when both recommending ways is at 50 items. Overall, this research contributes a novel framework that addresses the challenges of analyzing large-scale data, cold-start problems, and personalization, thereby enhancing the user experience, and driving sales on e-commerce platforms.

1. Introduction

After the Covid-19 pandemic was over, the economy in every country over the world have both encountered several huge troubles in retaining their customers. It makes enterprises have to excel their business strategies, especially small and medium enterprises (SMEs) must have an extraordinary campaign to appeal customers (Anjar and Anas, 2024). Thus, the knowledge of developing and employing the new strategy is inevitable in this situation which remarkably effects on increasing business adaptation, reducing business loss (Mohammad et al., 2023).

Recent studies have highlighted the crucial role of personalized product recommendations in driving sales and revenue for e-commerce businesses (Novi Fitriani et al., 2023). These recommendations have a substantial impact on key metrics like Average Order Value (AOV) and conversion rates. Engaging with personalized recommendations leads to larger purchase orders and higher completion rates, making them a vital tool for boosting business success.

The impact of personalized and engaging recommendations on purchase behavior is evident in Fig. 1. It shows that when customers interacted with a single personalized recommendation, there was a staggering 369 % increase in Average Order Value (AOV), and this effect continued to rise up to five clicks. This highlights the significant role that personalized and engaging recommendations play in driving larger purchase orders. In fact, product recommendations were found to contribute up to 31 % of e-commerce revenues, with an average of 12 % of sales attributed to recommendations (Barilliance Research, 2018). Furthermore, shoppers who clicked on recommendations exhibited a 4.5x higher likelihood of adding items to their cart and completing their purchase. These findings are consistent with a similar study conducted by SalesForce.

According to a study by PWC (PWC, 2018), 73 % of people consider customer experience to be a crucial factor. Fig. 2 demonstrates that

* Corresponding author at: University of Economics and Law, Ho Chi Minh City, Viet Nam.
E-mail address: honv@uel.edu.vn (V.-H. Nguyen).

https://doi.org/10.1016/j.joitmc.2024.100303
Received 5 March 2024; Received in revised form 4 May 2024; Accepted 14 May 2024
Available online 18 May 2024

customers are likely to be lost if they repeatedly encounter poor service. Given the significance of the product recommendation system in e-commerce, implementing machine learning poses significant challenges:

- Challenges arise from data availability and quality. Despite having the huge volume of datasets and a numerous popular method to acquire the relationship between users and products, the existing methods might be simple and do not consider multiple factors impacting to e-commerce in general such as seasonal shopping, stock of products, etc.
- Another obstacle is the cold-start problem, where new users or products lack sufficient data for the recommendation algorithm to analyze. It is the current issue that led to subpar recommendations and potentially discourage user engagement with the platform. Every recommender should have the ability to suggest suitable products for new users by learning the demographic information and putting them into the most related customer basket who may be in the same favor. Hence, producing the well-structured input for the model is essential and extremely important so that the study is proposed to process the raw data into useful input through multiple layers.

The main contributions and research objectives of this study are researching and proposing the state-of-the-art recommendation system that has the ability of tackling almost all barriers listed. This research first applies retrieval strategies (described in 3.2.) which can acquire the hidden interactions between users and products. It enables the selection of optimal retrieval methods for each user, considering their unique characteristics and preferences. By doing so, well-structured input will be produced which makes proposed models able to be trained with better results.

The authors employ advanced mathematical logic and well-structured input processed before that have been trained on LightGBM and Deep Neural Network techniques. By leveraging these advanced algorithms, the model aims to address all mentioned challenges and to offer personalized product recommendations to users.

This study proposes a conceptual model comprising four main stages:

(1) Pre-analysis
(2) Gathering related products by retrieval strategy implementation
(3) Model constructing
(4) Model evaluation

The study includes 5 sections as follows:

1. Introduction: This section provides an in-depth explanation of the purpose and rationale behind the current research project, including a detailed overview of the research goals and objectives. The chapter also outlines the scope of the work.

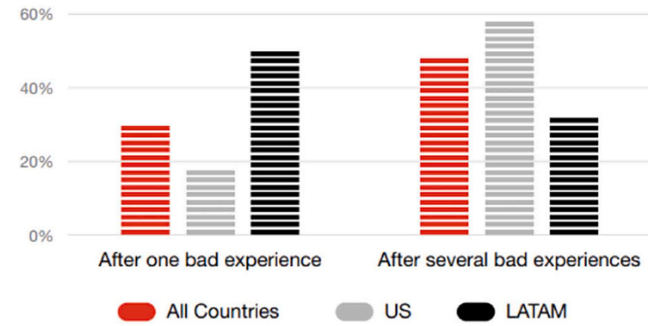

Fig. 2. When do consumers stop interacting with a brand they love? (Source: PwC Future of Customer Experience Survey 2017/18).

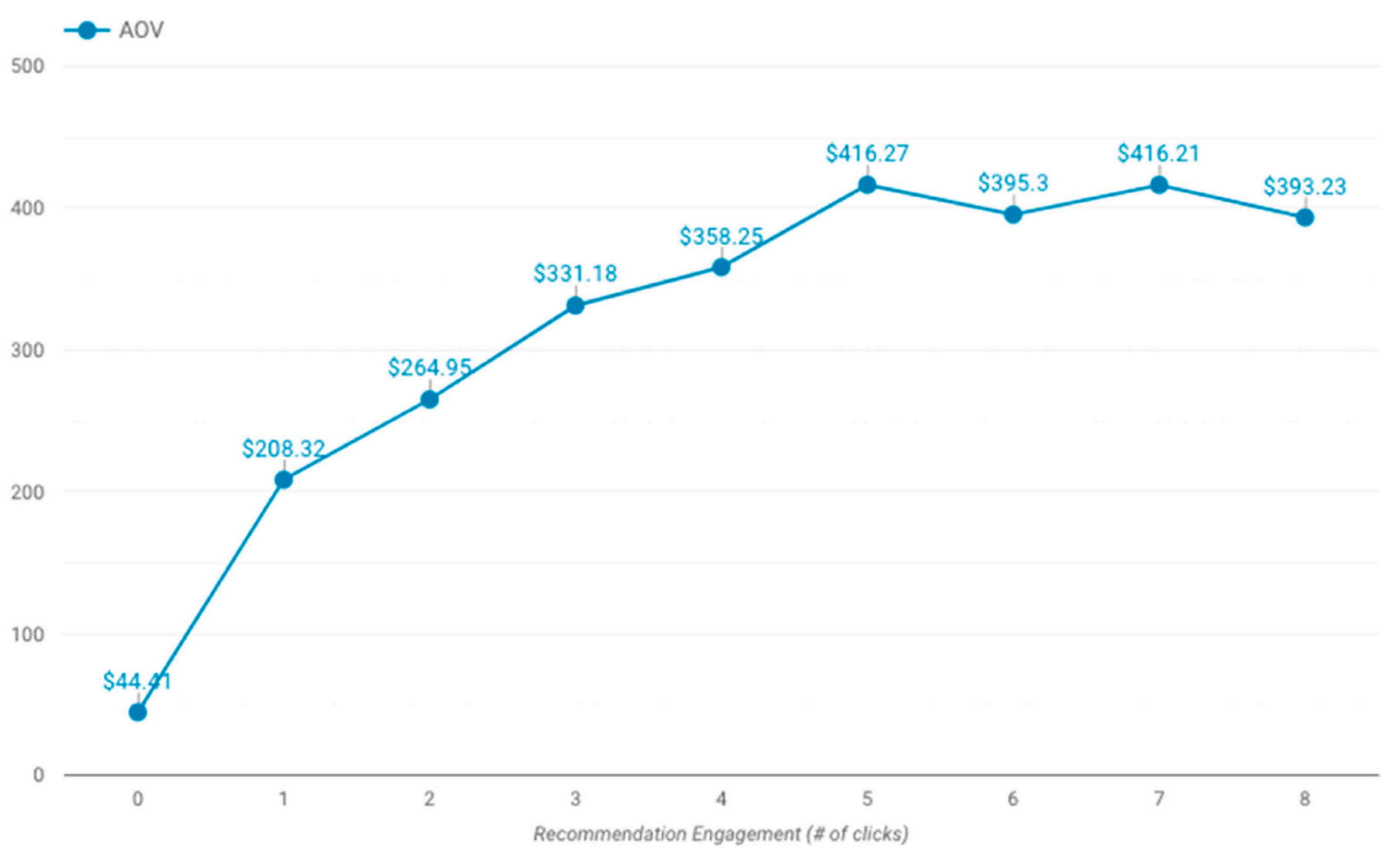

Fig. 1. Personalized Product Recommendations Statistics on Average Order Value (Source: Barilliance Research, 2018).

2. Literature review: This section provides a comprehensive review of the existing literature on the relevant topics, including an in-depth discussion of the various approaches and theories that are applicable to the current study. By examining the literature in detail, the reader gains a thorough understanding of the research context and the theoretical underpinnings of the study.
3. Methodology: In this section, exploratory data analysis is performed to gain a deeper understanding of the raw data and uncover any underlying patterns or insights. This analysis provides valuable information that is used to inform the research findings. In addition to the analysis, this section also includes detailed definitions of the retrieval rules used to extract meaningful data from the raw dataset.
4. Results and discussion: In this section, the reader is introduced to the overall framework of the research, including a detailed description of the experimental process. Additionally, it presents an overview of the architecture of the two proposed models, providing the reader with a clear understanding of their design and functionality.
5. Conclusion: It serves to summarize the findings of the study and provide an evaluation of the proposed models. It draws comparisons between the different models and ultimately presents a conclusive result based on the analysis conducted throughout the study.

Overall, the paper presents a comprehensive solution to the challenges faced in building effective personalized recommender systems. By combining mathematical logics, historical data, and advanced algorithms, the leveraging model offers personalized and relevant recommendations to enhance the user experience and engagement on online platforms. The retrieval strategy approach further improves the accuracy of the recommendations, ensuring that users receive tailored suggestions that align with their preferences.

2. Literature review

2.1. Overview of retrieval strategy

Retrieval strategy refers to the approach used to retrieve information or data from a storage system or database. In information retrieval, a retrieval strategy (described in Fig. 3) is a set of rules or algorithms designed to find and retrieve relevant information from a large pool of data (Vakkari, 2008).

In general, there are several types of retrieval strategies using machine learning models, including:

- *Ranking-based retrieval* (Tie-Yan Liu, 2009) involves using machine learning models, like neural networks, to predict the relevance of products based on user features and previous purchases. The products are then ranked according to their predicted relevance.
- *Clustering-based retrieval* (Christopher et al., 2020) involves grouping products into clusters based on their similarity. Machine learning models like k-means or hierarchical clustering are used to group items based on their features. Users are then recommended the cluster that is most relevant to their behavior
- *Collaborative filtering* (Sarwar et al., 2001) recommends products based on the preferences of similar users. Machine learning models like matrix factorization or nearest neighbor algorithms are used to identify users with similar preferences and suggest popular items among those users.

2.2. Explanation Artificial Intelligence and recommendation systems

Recommendation systems are important on many online platforms because they assist consumers find relevant goods or information based on their interests and previous interactions. Machine learning (ML) and deep learning (DL) approaches have recently transformed the field of recommendation systems, allowing for more accurate and tailored suggestions. Traditional recommendation systems, such as collaborative filtering and content-based filtering, have long been employed to create suggestions. Collaborative filtering approaches examine user-item interactions to identify comparable users or objects and create predictions based on their preferences. In contrast, content-based filtering algorithms employ item attributes and user profiles to propose products that are relevant to the user's interests. While successful, these technologies frequently encounter hurdles such as cold start issues and data scarcity. Machine learning techniques are commonly used to overcome the limitations of traditional recommendation systems (Thomas and John, 2021). Matrix factorization techniques, such as Singular Value Decomposition (SVD) and Alternating Least Squares (ALS), are common methods for learning latent features from user-item interactions and making predictions. In addition, ensemble approaches and probabilistic models have been developed to increase recommendation accuracy and resilience. Deep learning models have shown exceptional performance in a variety of tasks, including recommendation systems. Neural network designs, such as Multi-Layer Perceptrons (MLPs), Recurrent Neural Networks (RNNs), and Convolutional Neural Networks (CNNs), may detect intricate patterns in user behavior and object characteristics. Deep learning-based recommendation systems may develop hierarchical representations of individuals and things, resulting in more precise and personalized recommendations (Zhang et al., 2019; Bobadilla et al., 2020). Explanations in recommender systems assist users in understanding why a suggestion (or a series of recommendations) was created. Explaining suggestions has become a crucial need for increasing customers' trust and satisfaction (Zhong and Negre, 2022). The study of Yin et al., 2023 presented the interpretability of a neural network-based recommendation model that creates visual interpretations that demonstrate the importance of each TV show attribute in forecasting user interests. These interpretations will improve consumers' understanding of neural network learning principles and capture a wide range of user preferences.

2.3. Related works on recommendation systems based on retrieval strategy and research motivation

The authors acknowledge the challenges of studying a novel approach that involves analyzing historical customer purchasing data

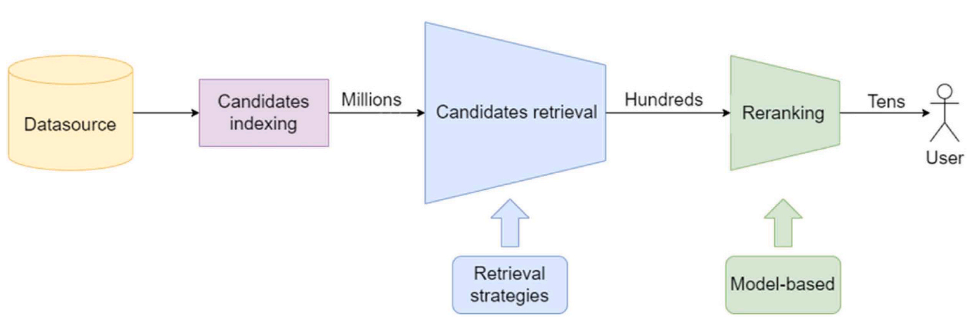

Fig. 3. Overview of retrieval strategy flow (Source: Authors).

and considering the time factor in recommender systems, which has been applied in real business for many years. The study aims to leverage the retrieval strategy's power in determining the relevance of candidates for each customer while incorporating current popular logic and algorithms to create a more personalized shopping experience for customers.

Various approaches to recommend engines have their strengths and weaknesses. One popular method is item-to-item similarity, which is particularly suitable for large datasets with a high number of users, as it exhibits scalability (Sarwar et al., 2001). This approach is transparent and can provide serendipitous recommendations, enhancing the user experience (Sarwar et al., 2001). However, it tends to overlook uncertainties and the temporal correlation between user behavior and time, which can impact the accuracy of recommendations (Nada and Damien, 2022).

Another widely used method in recommendation systems is user-to-user collaborative filtering, which focuses on analyzing user behavior (Ekstrand et al., 2011). Collaborative filtering, as researched by Su and Khoshgoftaar (2009), is a prevalent approach that utilizes the behavior and preferences of similar users to predict an individual user's preferences (Sarwar et al., 2001). Collaborative filtering offers the advantage of simplicity in implementation and comprehension. However, both item-to-item and user-to-user methods have limitations in considering the temporal aspect of item trends and addressing the challenges of cold-start problems, where there is limited or no user data available for new items (Nada and Damien, 2022).

The proposed model incorporates content-based filtering, as introduced by Pazzani and Billsus (2007), which recommends items based on their specific features or characteristics. This approach emphasizes the analysis of item properties and customer interactions with each product. Content-based filtering is particularly suitable for situations where there is a scarcity of user data and effectively addresses cold-start problems, where limited user information is available for new items (Hu et al., 2008). However, one limitation of content-based filtering is the potential occurrence of overspecialization, where it may excessively recommend items that are too similar in nature (Liu and Smyth, 2007).

The study proposed by Koren et al. (2009) presents an effective solution for addressing sparse data and cold-start problems. It introduces interpretable latent factors that capture user preferences and item characteristics. However, this approach has limitations in capturing complex patterns and non-linear relationships, as well as scalability issues when dealing with large datasets. To overcome these limitations, Wang et al. (2015) proposed incorporating matrix factorization and neural network algorithms. This combined approach offers high flexibility and personalization, surpassing previous techniques by mitigating noisy signals and overfitting (Covington et al., 2016). The integration of matrix factorization and neural networks in this manner allows the two techniques to support and complement each other, effectively addressing their respective limitations.

In addition, the study incorporates Bayesian personalized ranking (BPR), a well-known probability-based method, which excels in handling implicit factors like purchase data and effectively tackling cold-start problems by learning latent representations through user-item interactions, enabling personalized recommendations (Rendle et al., 2009). While BPR achieves high scalability through the use of stochastic gradient descent optimization, making it computationally efficient for large-scale recommendation systems, it is generally considered less interpretable compared to other approaches. The primary focus on optimizing ranking accuracy might compromise the model's ability to provide explicit explanations for the recommended items.

Therefore, the authors decide to exploit and try to employ the strength of each reviewed algorithm into 1 approach which is called retrieval strategy. It is used for the purpose of prepare the well-structured input that implies multi-dimensional factors including relationship between users and products; resemblance of users to users and favor; and interaction between seasonal factor and shopping behaviors. Once well-structured input is produced, the authors choose 2 popular machine learning models which are deep neural networks and lightgbm. Deep neural network is a type of artificial neural network that comprises multiple layers of neurons which researchers can easily customize in their study. It is used for a wide range of machine learning usages and is particularly useful for tasks where the input data is high-dimensional. The training process involves adjusting the weights and biases in the network. LightGBM also has a lot of advantages that are widely utilized. One of the main features of lightgbm is its use of the histogram-based algorithm for building decision trees. Instead of using the traditional approach of splitting nodes based on the exact values of the features, lightgbm groups the values into discrete bins, or histograms. This allows it to reduce memory usage and computation time while maintaining accuracy. The histogram-based approach also enables lightgbm to perform fast feature selection by calculating the split gain of each feature using histograms. This means that it can select the most important features to use in each tree, leading to faster training times and better performance.

Hence, building effective recommendation engines requires careful consideration of the strengths and limitations of different approaches. There is no universal solution that applies to all scenarios. Instead, the key lies in understanding the specific needs of the application and leveraging the strengths of each method accordingly (JinHyo and Xiaofei, 2020). By combining and integrating various approaches, taking into account the unique characteristics of the dataset, it becomes possible to develop highly effective and personalized recommendation engines. Such engines not only provide value to users by delivering relevant recommendations but also benefit businesses by enhancing user engagement and satisfaction.

3. Methodology

The study encompasses four primary stages, as illustrated in Fig. 4, to comprehensively address the research objectives. The stage (1), pre-analysis, involves the collection of data and initial exploratory analysis to gain insights into the dataset. Additionally, data processing techniques are applied to enhance the quality and usability of the data. Stage (2), considerable effort is dedicated to building and investigating various retrieval strategies that can be effectively combined to achieve personalized recommendations. These strategies are designed to identify the most relevant products based on user preferences and interests. Stage (3), the retrieved data is further processed and transformed to create suitable inputs for specific models. This includes designing and implementing the necessary modifications to adapt the data to the unique requirements of each model. Stage (4), the experimental results are rigorously evaluated and discussed. This entails a comprehensive analysis of the performance and effectiveness of the implemented approaches.

3.1. Pre-analysis

3.1.1. Data collection

The studied datasets were provided by H&M, they were recorded from September 2018 to September 2020. The customer data is stored about 1371,980 unique customers described in Table 1. The column `FN` stands for foreign national which means the customer is not the citizen of that local retail. The column `Fashion_news_frequency` indicates whether the customer enable notifying fashion news to them or not.

The data of articles products is stored in detail including over 105,000 articles with 25 features described in Table 2.

The transactions are recorded day by day at roughly 32 million transactions, as well as additional information described in Table 3.

3.1.2. Data exploratory

Fig. 5 provides valuable insights into the purchasing behavior of customers over a span of two years, enabling the authors to discern

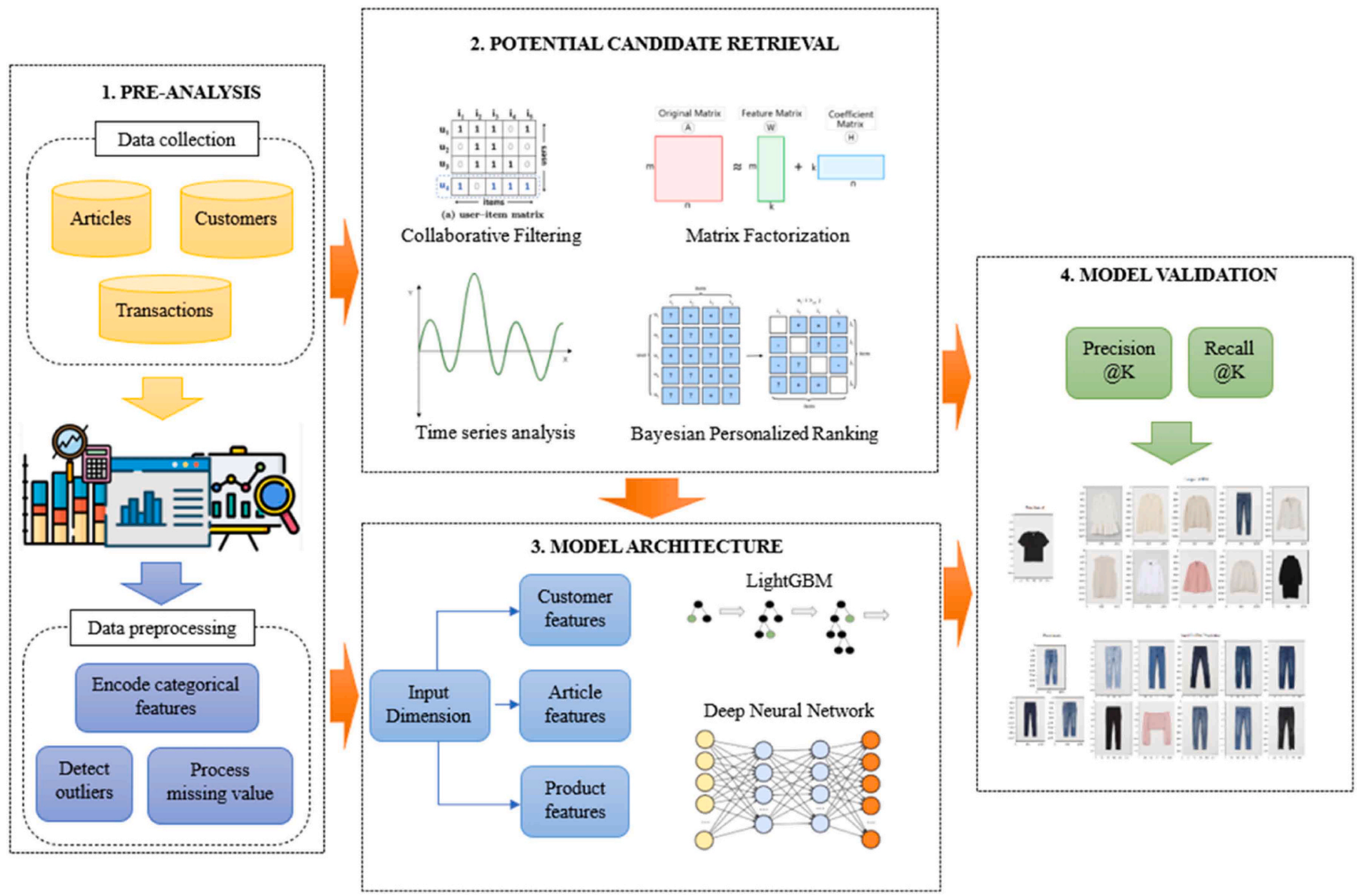

Fig. 4. The proposed recommendation model's workflow (Source: Authors).

Table 1
Example of customer's observations.

Customer_id	FN	Active	Club_member_status	Fashion_news_frequency	Age	Postal code
00000dbaca…			ACTIVE	NONE	49	52043ee2162c…
0000423b00…	1	1	ACTIVE	NONE	26	2973abc54daa…
00007d2de82…	1		PRE-CREATE	Regularly	32	8d6f45050876d…

Table 2
Example of article's observations.

Article_id	Product_code	Prod_name	Product_type_no	Product_type_name	Graphical_appearance_no	…
108775015	108775	Strap top	253	Vest top	1010016	…
108775044	108775	Strap top	253	Vest top	1010016	…
110065001	110065	OP T-shirt (Idro)	306	Bra	1010016	…

Table 3
Example of transaction's observations.

t_dat	customer_id	article_id	price	sales_channel_id
2018–09–20	000058a12d5b43e67…	0663713001	0.0508	2
2018–09–20	000058a12d5b43e67…	0541518023	0.0304	2
2018–09–20	00007d2de826758b65…	0505221004	0.0152	2

trends and patterns in the company's business. Analysis of the data reveals that the period from May to August witnessed the highest number of sales, with a peak occurring in June. Subsequently, there was a significant decline in sales, approximately halving, which persisted for approximately five months.

The primary objective behind understanding this sales pattern is to devise strategies that promote heightened customer engagement among existing customers and maximize the acquisition of new customers. By leveraging implicit methods and leveraging the insights gained from the analysis, the aim is to create an environment that fosters excitement and

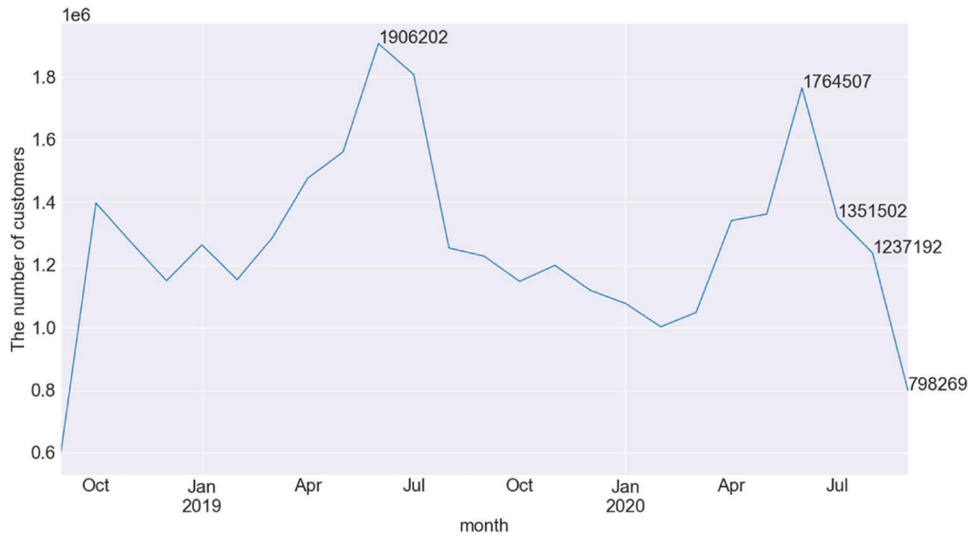

Fig. 5. The number of customers by month (Source: Authors).

engagement for existing customers while attracting a substantial number of new customers to the platform.

3.2. Input preparation and retrieval strategies planning

The dataset spans from 18 September 2018–22 September 2020. To develop an effective recommendation engine, the last week of the dataset (16–22 Sept 2020) is designated as the test set. The preceding four weeks (17 Oct - 16 Sept 2020) will serve as the training set. During this period, retrieval strategies will be implemented. These strategies involve algorithms that determine which items should be recommended to customers based on their preferences.

In more detail, following the guidance of Kaggle Grand Master Jin Zhan, a Japanese Data Scientist, the researcher uses the last week (16–22 Sept 2020) as the test set or the ranking model sample for the current week. The researcher retrieves the top N potential candidates from the previous data using defined rules for each customer. The overlap between the recall data and items purchased in the corresponding week is considered a positive sample, while the remaining data is considered a negative sample.

3.2.1. Input processing

After splitting the data into training and testing sets, the authors proceed to handle missing values and encode categorical features related to article attributes. In particular, they assign tags to articles based on the customer groups they target, such as 0 for unisex, 1 for female, and 2 for male. They then convert these categorical features into discrete values.

Then, the authors design two strategies (also called *simple strategy* and *complex strategy*) to retrieve the potential candidates (items) each week. They study the rules and the way how to implement and retrieve information from the work of W. Zhang et al., 2017.

The result of *simple strategy* is described in Fig. 6, it includes the following rules:

- Retrieve the recent products for each customer in the specific period (3 or 7 days). The pair of each customer and article is scored by different days from it to the last purchased of correspond customer.
- Retrieve the couple of items which customers bought them together. This is commonly known as *market basket analysis* or *association analysis* (Agrawal and Srikant, 1994). This feature supports identify patterns and correlations in customer purchasing behavior.
- Retrieve the recent and the items which customers bought together with decaying function referred to the study of W. Zhang et al., 2017.

$$r = \frac{a}{\sqrt{x}} + b * e^{-c*x} - d \tag{1}$$

Where:

- r is the probability indicating the likelihood specific customers make this order again
- x is the day gap between the last order of specific items purchased by each customer to present

customer_id	article_id	ItemPairRetrieve_1	ItemPairRetrieve_2	ItemPairRetrieve_3	ItemPairRetrieve_4	OrderHistoryDecay_1	OrderHistoryDecay_2	OrderHistory_1	OrderHistory_2
1	15989			1.494154909	1.529558186				
1	16004					-0.600280419	-0.58848416		
1	16024					1.258803359	1.513189668	5.199337583	5.199337583
1	93577	0.482248215	0.478024689	-0.349483006	-0.324253996				
2	46383				-1.70022246				
2	86502	-0.490721237	-0.461214605						
2	87146						-0.002436003	5.199337583	5.199337583
3	1826			-1.236652242	-1.255742874				
3	46352			-0.744694448	-0.55985906				
3	60260					0.227008868	-0.283588313		
3	78504					2.325972458	2.365247647	5.199337583	5.199337583
3	78506	0.186756121	0.231641056	-0.535082815	-0.497809075				
3	92159					-0.36869492	0.169392811		
3	92160					0.035048999	0.497831131		

Fig. 6. Score of candidates retrieved for each customer (Source: Authors).

- $\frac{a}{\sqrt{x}}$ represents a decaying function that decreases as the square root of x increases, including parameter a control the overall scaling of this function
- $b * e^{-c*x}$ represents an exponential decay function that decreases rapidly as x increases. The parameter b controls the overall scaling of this function, and parameter c controls the rate of decay.
- Parameter d represents a constant offset or baseline value that is added to the overall result.

The formula (1) analyzes customer behavior by predicting the likelihood of future purchases based on the date gap between previous purchases. Generally, it suggests that longer date gaps between orders indicate a lower probability of a future purchase. The interpretation of the function depends on the context and parameter values chosen for the analysis, which can be determined based on prior knowledge, experience, or previous analyses. Alternatively, starting values for parameter estimation may be chosen arbitrarily.

- Retrieve the popular items according to age groups over specific time windows (5 days, or 7 days). Then, engineering by using decaying function to get more insight about the effect of time series for these features. Analyzing popular items by customers' age bins over time helps identify frequently consumed products in different age groups during specific periods. This analysis is valuable in various industries, including e-commerce, entertainment, and healthcare. By examining popular items across age groups, trends and patterns in consumer behavior can be identified. For instance, it may reveal preferences of younger customers compared to older ones. Such insights can be utilized to create targeted marketing campaigns, optimize product offerings, and enhance customer engagement and retention.
- Retrieve purchased items per each customer by analyzing time windows, combination between decaying function and time series. By doing this step, the input is enriched by the insight of customers' trend.

The result of *advanced strategy* is described in Fig. 8, it includes the simple rules and advanced mathematical concepts:

- Collaborative Filtering
  - o In collaborative filtering, the system analyzes the historical data of multiple users, such as their past purchases, ratings, or interactions with items (Beladev et al., 2016). It then identifies patterns and similarities between users based on their preferences and behaviors. These patterns are used to make predictions and recommendations for a particular user by leveraging the knowledge gained from similar users.
- Matrix Factorization
  - o This is achieved by producing the product of two rectangular matrices that have lower dimensions. The first of these matrices is referred to as the user matrix, and it is comprised of rows representing users and columns representing latent factors. The second matrix is the item matrix, made up of rows representing latent factors and columns representing items (Fig. 7 and Fig. 8).

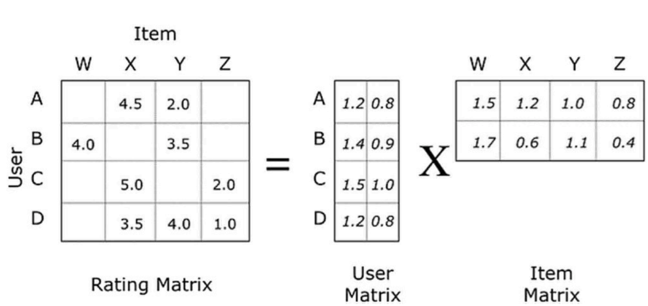

Fig. 7. Decomposition of matrices (Source: Kevin Liao, 2018).

- Bayesian Personalized Ranking
  - o Bayesian Personalized Ranking (BPR) is a recommendation algorithm that uses Bayesian inference to generate personalized rankings of items for users. It focuses on pairwise item comparisons, learning from user-item interactions to predict ranking orders. BPR maximizes the probability of observed rankings by updating latent factors. It is effective for implicit feedback data and has been widely used in recommendation systems.

Once the retrieval process is completed, the input set contains all the features associated with the applied rules. This set records each customer's past behavior in relation to each item. The scores assigned to each observation of the rules vary in scale, and therefore, quantile normalization (Bolstad et al., 2003) is employed as the chosen normalization method.

3.3. Application of machine learning

3.3.1. LightGBM model

Verdict trees are used to map a function from the input space X to the gradient space G. A training set is assumed, consisting of instances x_1 to x_n, where each element is a vector in X with s dimensions. During each iteration of gradient boosting, the negative gradients of a loss function with respect to the output model are represented as g_1 to g_n. The decision tree partitions each node based on the most informative feature, maximizing the gain of evidence. The improvement of data in this model is measured by the variance after the partition. Mathematically, this can be expressed as (Ahamed et al., 2021).

Assume O is a training dataset on a fixed node of a decision tree and the *variance gain* (V) of dividing measure j at a point d for a node is defined in formular (2):

$$V_{j|O}(d) = \frac{1}{n_o}\left(\frac{\left(\sum_{x_i \in O: x_{ij} \le d} g_i\right)^2}{n^j_{l|O}(d)} + \frac{\left(\sum_{x_i \in O: x_{ij} \le d} g_i\right)^2}{n^j_{r|O}(d)} \right) \quad (2)$$

The LightGBM is trained with the following hyperparameters presented in Table 4:

- **Objective**:
  Determines the type of problem and loss function to be minimized. In this case, it is binary classification with binary cross-entropy loss.
- **Boosting Type**:
  Specifies the boosting algorithm used. Here, it is gradient boosting decision trees (GBDT), which creates an ensemble of decision trees.
- **Metric**:
  Evaluation metric used to measure model performance during training. AUC (area under the receiver operating characteristic curve) is used to assess the model's ability to distinguish between positive and negative examples.
- **Max Depth**:
  Sets the maximum depth of each decision tree. A decision tree splits data based on feature values, and the depth determines the number of allowed splits. Here, it is set to 8.
- **Num Leaves**:
  Determines the maximum number of leaves per decision tree. It controls the granularity of the tree, affecting the complexity of decision boundaries. Here, it is set to 128.
- **Learning Rate**:
  Controls the step size for each boosting iteration. It regulates the update of model weights. A lower learning rate, like 0.03 in this case, helps prevent overfitting.
- **Eval At**:

customer_id	article_id	ALS_1	BPR_1	ItemCF_1	ItemCF_2	ItemCF_3	OrderHistoryDecay_1	OrderHistory_1	TimeHistory_1	UGItemCF_1	UGItemCF_2	UGItemCF_3	UGItemCF_4	UGItemCF_5	UGItemCF_6	UGTimeHistory_1	UGTimeHistory_2	UGTimeHistory_3
1	416														0.91023922			
1	12745									0.741950095	0.819215298							
1	13123														0.721347511			
1	13340														0.62133491			
1	14254											0.91023922			0.91023922			
1	14276														0.721347511			
1	15990			2.412167788	2.443765163	1.552721977												
1	16004			6.876390457	7.235980034	3.80315423	330.7990417			2.329368114	2.373625278	1.239822149	1.682308793	1.682308793				
1	16005			3.304775953	3.339805841	2.553906441				0.825465441	0.825465441	0.825465441	1.820058465	1.852193594	1.451132417			
1	16014			4.161669254	4.204386234	2.571413279				1.227013588	1.241400242	1.159153223	1.214276671	1.230509639	1.123043537			
1	16018			4.429381847	4.158313751	3.284989595				0.759946823	0.904575765		0.951517105	0.968628645	0.86550802			
1	16024			3.262604952	3.404777527	1.571296811	28864.88867	0										
1	16025			2.689522982	2.748433113	1.674503326				1.534502983	1.534502983	1.069621801						
1	18295											0.558110654						

Fig. 8. Score of candidates retrieved for each customer (Source: Authors).

Table 4
Hyperparameters of LightGBM model.

Hyperparams	Value
objective	binary
boosting_type	gbdt
metric	auc
max_depth	8
num_leaves	128
learning_rate	0.03
eval_at	12

Specifies the iteration to evaluate validation metrics during training. In LightGBM, it is set to 12, determining when to assess the model's performance.

As limitations of implementing LightGBM described in 4., tunning hyperparameters is extremely complicated so that authors put the default value listed above for this model. The authors also find the optimal learning rate in range between 0.01 and 0.1 by executing optuna library (https://optuna.readthedocs.io/). The changes of learning rate's value compared by auc score are presented in Fig. 9. The authors choose the learning rate at 0.03 because this value reflects the stable learning of this scenario.

3.3.2. Deep neural network model

In the authors' model, the input is divided into three dimensions: Customer dimension, Article dimension, and Product dimension. Each dimension has its own set of layers and corresponding weights, as illustrated in Fig. 10. After concatenating the three dimensions, the model incorporates two dense layers with a swish activation function, which is a non-linear activation function known for its smoothness and effectiveness in capturing complex patterns. Additionally, there is one dense layer for the output with a sigmoid activation function, which is commonly used in binary classification tasks to produce a probability value between 0 and 1. This comprehensive architecture enables the model to effectively process and analyze the input data across multiple dimensions, making it capable of capturing intricate relationships and generating meaningful predictions.

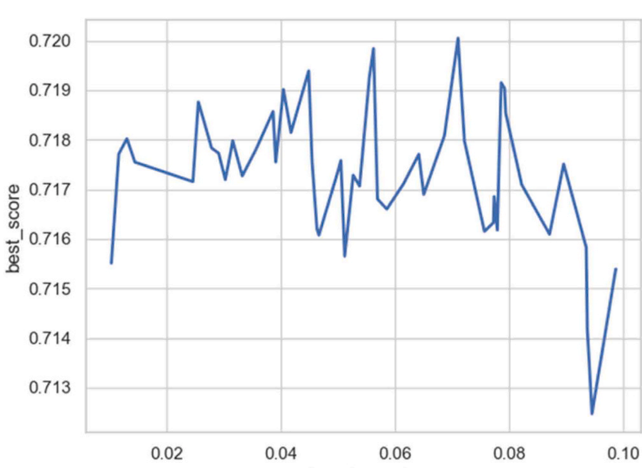

Fig. 9. Learning rate tunning by auc score (Source: Authors).

The *swish* function (Ramachandran et al., 2017) is defined in formula (3):

$$Swish(x) = x * sigmoid(\beta x) \tag{3}$$

Where:

x is the input of the function

β is the hyperparameter that controls the range of values of ouput

The Swish activation function, similar to ReLU, enhances the training of deep neural networks with its smooth curve. It outperforms other activation functions like ReLU, especially in tasks like image recognition and natural language processing. Being non-monotonic, Swish captures intricate input-output relationships that might be overlooked by other activation functions.

3.4. Model evaluation

Once the training phase is completed, the authors proceed to assess their proposed models' performance using the most recent week of the dataset. The simple and advanced input are presented in 3.2.1., these 2 types of input are trained and evaluated as described in Table 5.

This evaluation involves two commonly used metrics in recommender systems: Mean Average Precision (MAP) and Mean Average Recall (MAR) (Ali et al., 2017). The recommended article range for each customer varies from 5 to 50. The performance results for each model and dataset are presented in Fig. 11 and Fig. 12. These figures indicate that the LightGBM model outperforms the Deep Neural Network (DNN) model. In fact, the performance of LightGBM is significantly better, with a value three times larger than that of DNN. This emphasizes the superiority of LightGBM over DNN for recommender systems in the given dataset.

Precision@K (described in the formular 4) (Manning et al., 2008) is a metric that assesses the performance of a recommendation or information retrieval system. It quantifies the percentage of relevant items among the top K items recommended to a user.

$$Precision@K = \frac{Number\ of\ recommended\ items\ @K}{K} \tag{4}$$

Recall@K (described in formular 5) (Khan et al., 2017) is a metric employed to assess the performance of recommendation or information retrieval systems. It gauges the percentage of relevant items that were included in the top K recommendations. This metric is valuable in evaluating the comprehensiveness of recommendation systems, as it indicates the system's capability to retrieve all relevant items for a user.

$$Recall@K = \frac{Number\text{of recommended items @K that are relevant}}{K} \tag{5}$$

Fig. 13. illustrates the recommendations of *small LightGBM* (which is trained with simple input) model that suggests to a random female customer who aged at 27 and have not purchased before on this store

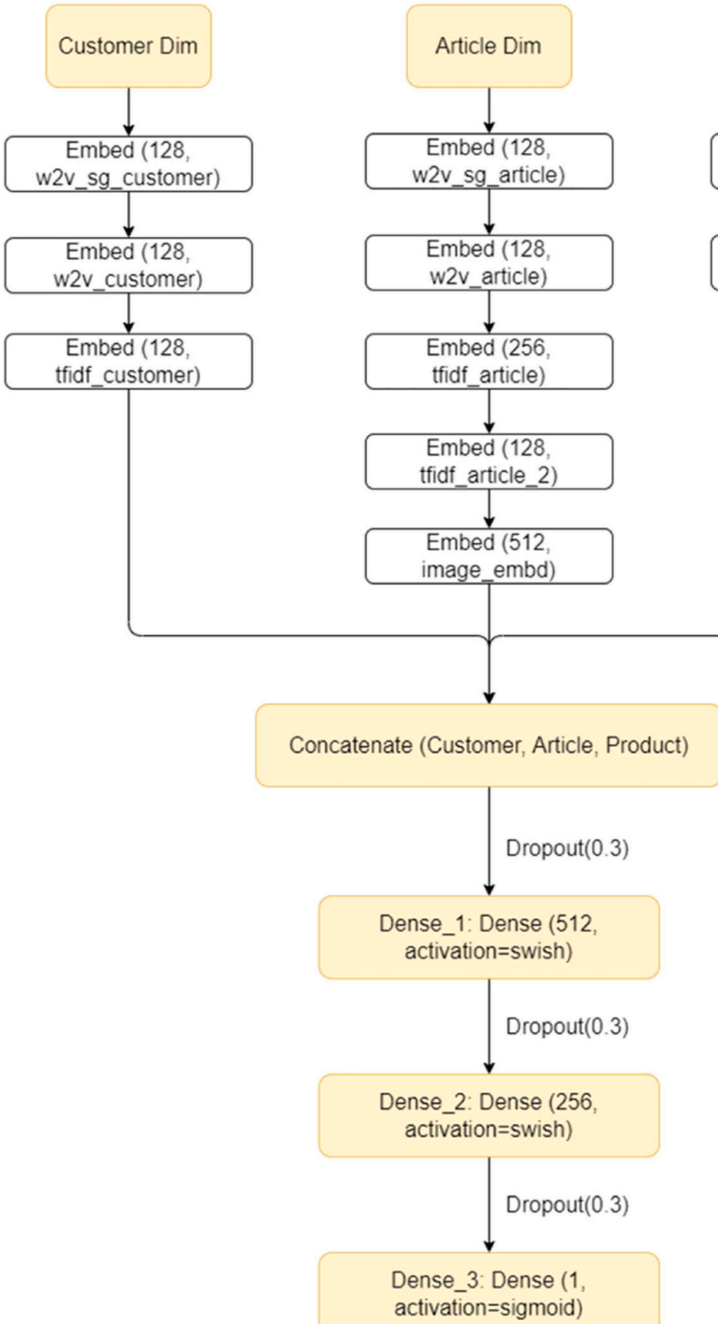

Fig. 10. Proposed deep neural network architecture (Source: Authors).

Table 5
Experimental results of 2 proposed model.

K	LightGBM				Deep neural network			
	Simple input		Advanced input		Simple input		Advanced input	
	MAP@K	MAR@K	MAP@K	MAR@K	MAP@K	MAR@K	MAP@K	MAR@K
10	0.0574	0.0283	0.0546	0.0276	0.0175	0.0066	0.0179	0.0065
20	0.0597	0.0300	0.0571	0.0295	0.0208	0.0083	0.0213	0.0082
25	0.0600	0.0304	0.0575	0.0299	0.0218	0.0089	0.0222	0.0087
30	0.0602	0.0307	0.0577	0.0302	0.0226	0.0094	0.0229	0.0093
40	0.0602	0.3101	0.0579	0.0306	0.0234	0.0099	0.0239	0.0100
50	0.0601	0.0311	0.0579	0.0309	0.0237	0.0102	0.0247	0.0106

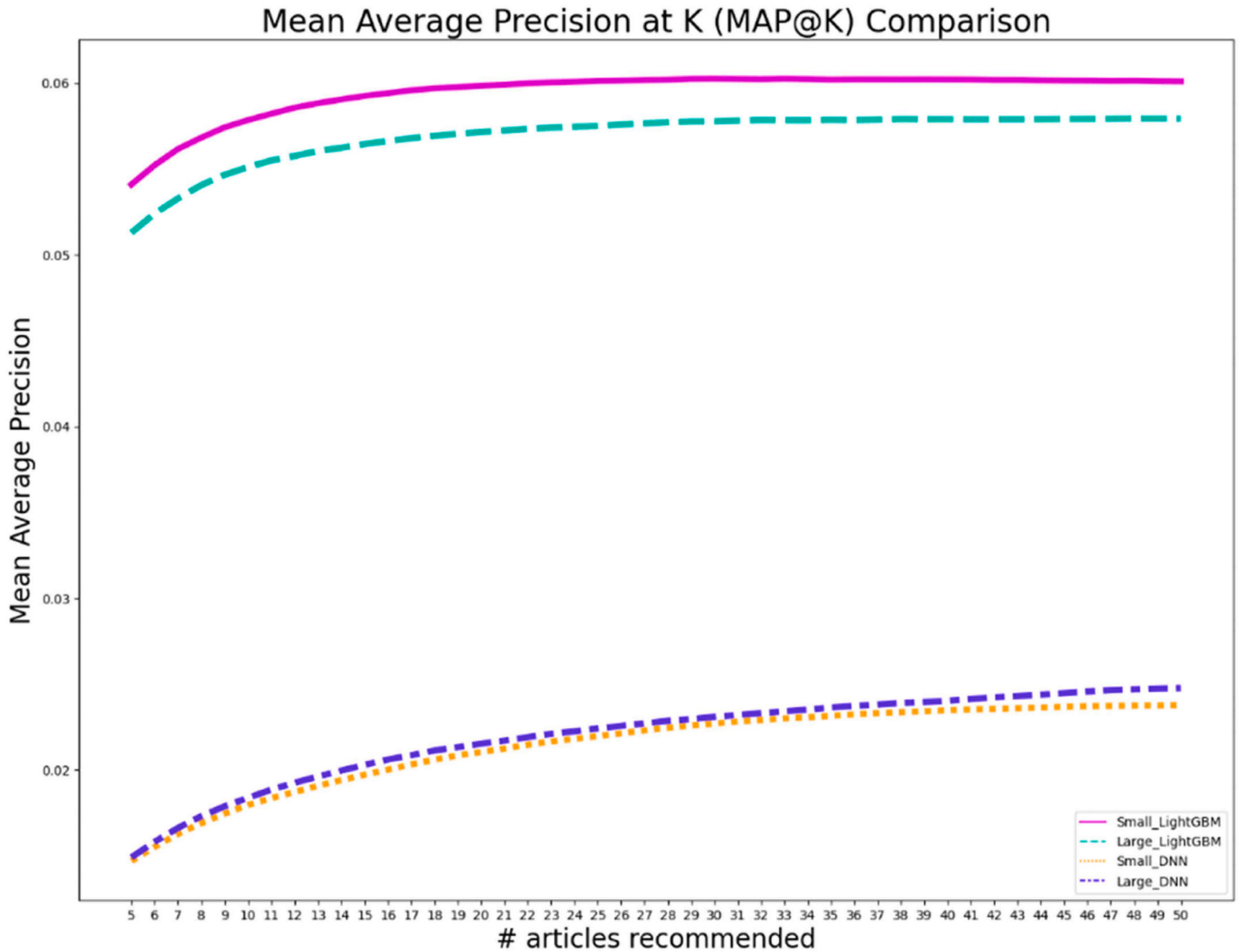

Fig. 11. Mean Average Precision at K Comparison *(Source: Authors)*.

with the number of suggesting items is 50 (k @ 50) but showing top 10 items which have the most relevant score, this customer actually bought 1 item of them in the following week. Fig. 14. also continues presenting top 10 most recommended products in k @50 for this customer applying large LightGBM model. Despite this purchased one has the score less than these top 10, but it is still listed in k @50 products.

Furthermore, Fig. 15. and Fig. 16. indicates another recommendation results of small and large lightgbm respectively to another random female whose age is 29 with k @50.

4. Results and discussion

As described in 3.4., the evaluation score of models is rising while the number of recommendations is changing from 5 to 50 items. However, the effectiveness of models when training 2 different types of input are not much different which means that the simple input also covers this research dataset and current scenario. Due to the limitations of customers' data size, variety of customer demographic, this leads to researchers cannot exploit the full potential of advanced algorithms.

In general, the retrieval rules in this business case proved effective in finding potential candidates by scoring each customer's relevance to items. However, comparing Deep Neural Network and LightGBM performance revealed that the former struggles due to its complex hidden layers. Tuning the Deep Neural Network's hyperparameters is challenging and requires huge computational resources and to conduct for optimal performance.

In addition, recommender systems have become crucial for businesses to enhance customer engagement and drive sales. In this study, LightGBM and DNN algorithms were evaluated using MAP@K and MAR@K metrics on a dataset of recommended articles. LightGBM outperformed DNN significantly, providing three times better results. LightGBM's efficiency in handling large-scale data, optimizing complex objectives, and robustness to categorical features and outliers contributed to its superiority. Besides the numerous advantages of applying LightGBM in this study, it still has several drawbacks of tunning optimal hyperparameters and be prone to overfitting. But the former is solved that described in 3.4. the latter is addressed by using the retrieval approach that implies the algorithm avoid overfitting while training model. These findings emphasize the importance of algorithm selection, considering data characteristics, computational resources, and performance metrics for effective recommender systems implementation.

During researching this study, the authors face several challenges, particularly lacking the time of experiment, the computational resources to conduct and domain knowledge of mathematics. The main limitation is that this study has not been compared its performance to the other baseline models because of the lack of standard evaluation metrics. It currently has a few widely accepted benchmarks for measuring the effectiveness of recommendation systems.

The developed system may be used in fields other than e-commerce with proper adjustments. Here's a description of its possible

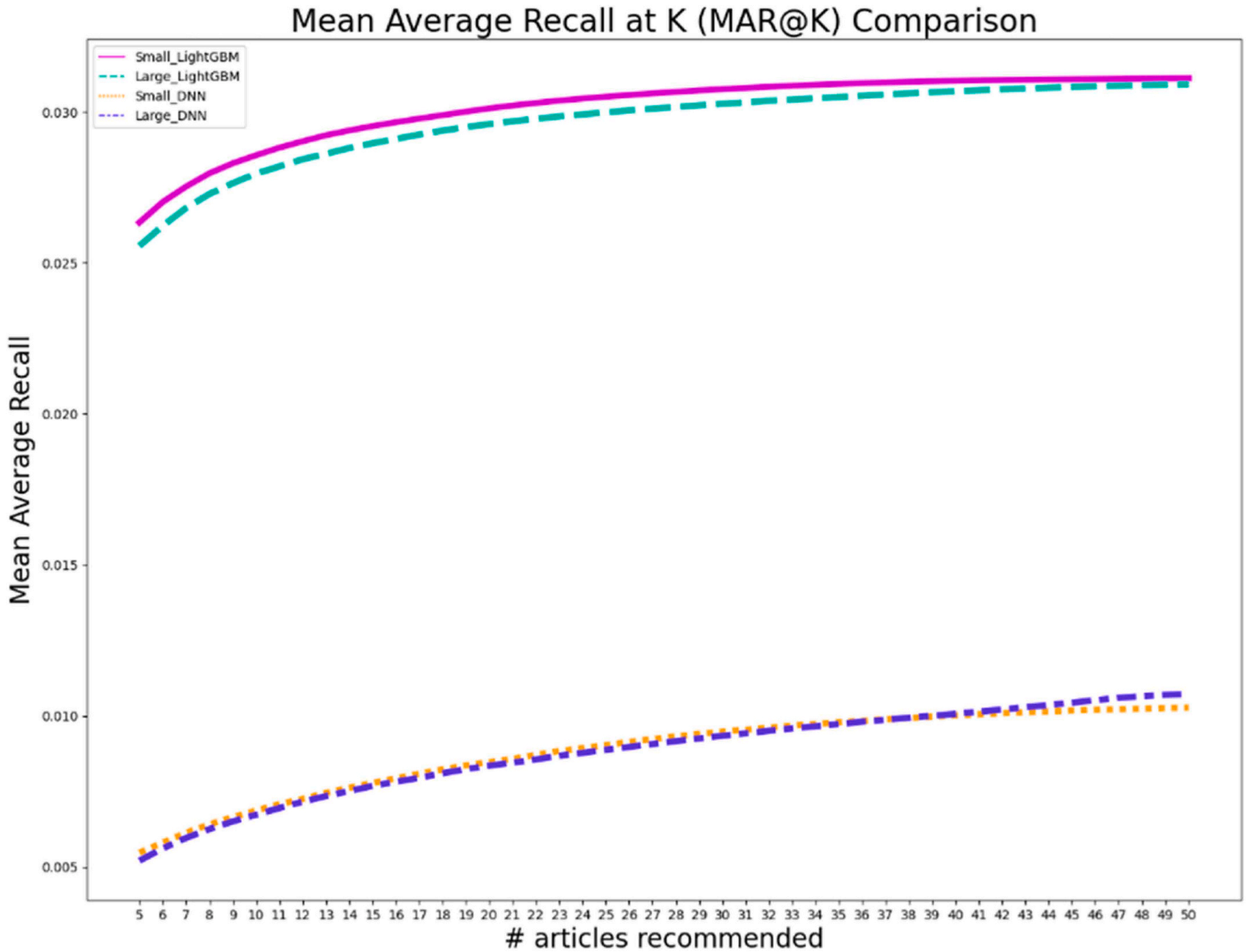

Fig. 12. Mean Average Recall at K Comparison (Source: Authors).

applicability and the adjustments needed for various application domains:

(1) Streaming Services: The recommendation algorithm might be tweaked to propose movies, TV shows, or music depending on user likes and watching habits. Changes might include adding characteristics related to entertainment material, such as genre, director, actors, or music genre. Furthermore, the retrieval approach may need to address the temporal component, proposing information based on the current mood or time of day.
(2) Healthcare Applications: Based on an individual's health profile and medical history, the model might offer individualized treatment regimens, health recommendations, or lifestyle changes. Modifications would include health-related information such as medical problems, symptoms, lifestyle behaviors, and treatment outcomes. The retrieval approach may select therapies according to their efficacy and relevance to the user's health status.
(3) Travel and Hospitality: For travel platforms, the model might propose places, lodgings, or activities based on user interests, previous travel experience, and financial restrictions. Changes might include travel-related characteristics like destination choices, trip dates, lodging preferences, and travel companions. The retrieval approach may prioritize suggestions based on popularity, user ratings, and availability.

In each of these application areas, adjustments would be required to fit the recommendation model to the domain's unique traits and requirements. This may entail adding domain-specific characteristics, altering the retrieval approach to account for important contextual aspects, and fine-tuning the recommendation algorithms to enhance performance in the particular domain. In addition, domain experts' feedback may be necessary to guarantee the model's efficacy and relevance in the target application domain.

5. Conclusion

Our study concludes that LightGBM outperforms DNN in recommender systems. This emphasizes the importance of algorithm selection, considering factors like data characteristics and computational resources. Our findings have significant implications for businesses implementing recommender systems, guiding the selection of suitable algorithms to enhance customer engagement and drive sales.

Firstly, the authors are going to seek e-commerce SMEs, which have adaptive capabilities and are willing to support them to implement and stream new technological strategy and external knowledge to business platforms (Omar et al., 2023). Monitoring and evaluating the performance of model is required by using A/B testing for specific campaigns. Moreover, tuning the hyperparameters is essential while data is enriching every day and highly considering the customers' behavior which motivates them make purchasing. It helps the researchers notice

Fig. 13. Small LightGBM model recommends top 10 items to 1 purchased history (Source: Authors).

Fig. 14. Large LightGBM model recommends top 10 items to 1 purchased history (Source: Authors).

the signal of market and revise their approach appropriately. By doing so, they can enhance the performance of their recommendation systems, improve customer experiences, and achieve sustained growth in the competitive e-commerce industry.

Recommendation systems are vital for personalized suggestions based on user preferences and behaviors. This study acknowledges limitations in time constraints, resources, and domain knowledge. Despite these drawbacks, it provides valuable insights and serves as a foundation for further research. Standardized evaluation metrics and interdisciplinary collaborations are needed to address these limitations and advance the field. Collaborative efforts will drive improvements in recommendation systems and bridge the gap between mathematics and their development.

Funding Acknowledgements

This research is funded by University of Economics and Law, Vietnam National University Ho Chi Minh City, Vietnam

Ethical statement

No applicable because the study does not include research involving animal or human subjects.

Fig. 15. Small LGBM model recommends top 10 items to 3 purchased histories (Source: Authors).

Fig. 16. Large LGBM model recommends top 10 items to 3 purchased histories (Source: Authors).

CRediT authorship contribution statement

Duy-Nghia Nguyen: Writing – review & editing, Writing – original draft, Visualization, Project administration, Methodology, Investigation, Data curation, Conceptualization. **Van-Ho Nguyen:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Funding acquisition, Formal analysis, Conceptualization. **Hoanh-Su Le:** Writing – review & editing, Supervision. **Trang Trinh:** Validation, Data curation. **Thanh Ho:** Writing – review & editing, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In Proceedings of the 20th International Conference on Very Large Data Bases (VLDB).

Ali, M., et al., 2017. MFS-LDA: a multi-feature space tag recommendation model for cold start problem. Program electronic library and information systems. https://doi.org/10.1108/PROG-01-2017-0002.

Anjar, P., Anas, H., 2024. Fostering innovation through learning from digital business ecosystem: a dynamic capability perspective. J. Open Innov. Technol. Mark. Complex. vol.10 (1).

Barilliance Research, 2018. [Guide] Personalized Product Recommendations Tactics for Profits. Retrieved from https://www.barilliance.com/personalized-product-recommendations-stats/#t-1620936380991.

Beladev, M., Rokach, L., Shapira, B., 2016. Recommender systems for product bundling R. Knowl. Based Syst. 111, 193–206. https://doi.org/10.1016/j.knosys.2016.08.013.

Bobadilla, J., Alonso, S., Hernando, A., 2020. Deep learning architecture for collaborative filtering recommender systems. Appl. Sci. 10 (7), 2441.

Bolstad, B.M., Irizarry, R.A., Åstrand, M., Speed, T.P., 2003. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias, 2003 Bioinformatics 19 (2), 185–193. https://doi.org/10.1093/bioinformatics/19.2.185.

Christopher D.Manning, Prabhakar Raghavan, and Hinrich Schütze (2020). Introduction to Information Retrieval.

Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 191-198).
Ekstrand, M.D., Riedl, J.T., Konstan, J.A., 2011. Collaborative Filtering Recommender Systems. Now Publishers Inc.
Fitriani, Novi, et al., 2023. Does social media affect performance in e-commerce business? The role of customer management. J. Open Innov. Technol. Mark. Complex. vol.9 (4).
Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining.
JinHyo, J.Y., Xiaofei, Z., 2020. Business model innovation through a rectangular compass: from the perspective of open innovation with mechanism design. J. Open Innov. Technol. Mark. Complex. vol.6 (4).
Khan, H.U., Daud, A., Ishfaq, U., Amjad, T., Aljohani, N., Abbasi, R.A., Alowibdi, J.S., 2017. Modelling to identify influential bloggers in the blogosphere: a survey. Comput. Hum. Behav. 68, 64–82.
Koren, Y., Bell, R., Volinsky, C., 2009. Matrix factorization techniques for recommender systems. IEEE Comput. 42 (8), 30–37.
Liu, Q., & Smyth, B. (2007). Personalized news recommendation based on click behavior. In Proceedings of the 15th international conference on Multimedia.
Manning, C.D., Raghavan, P., Schatze, H., 2008. Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA.
Mohammad, I., Erma, Y., Mukhammad, K.M., Endang, S.A., 2023. Linking knowledge management to tourism business of SMEs in aftermath of disaster: implications for open innovation. J. Open Innov. Technol. Mark. Complex. vol.9 (2).
Nada, Damien, 2022. Probabalistic approach for recommendation systems. IEEE ICMLA.
Omar, C.C., Domingo, G.P.L., Mauricio, C.V., 2023. Impact of innovation strategy, absorptive capacity, and open innovation on SME performance: a Chilean case study. J. Open Innov. Technol. Mark. Complex. vol.9 (2).
Pazzani, M.J., Billsus, D., 2007. Content-Based Recommendation Systems. In The adaptive web. Springer, Berlin, Heidelberg, pp. 325–341.
PWC, 2018. Experience is everything. Retrieved from https://www.pwc.com/us/en/advisory-services/publications/consumer-intelligence-series/pwc-consumer-intelligence-series-customer-experience.pdf.
Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L., 2009. BPR: bayesian personalized ranking from implicit feedback. Proc. 25th Conf. Uncertain. Artif. Intell. 452–461.
Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web.
Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web (WWW '01), 285-295.
Su, X., Khoshgoftaar, T.M., 2009. A survey of collaborative filtering techniques. Adv. Artif. Intell. 2009.
Thomas, B., & John, A.K. (2021, February). Machine learning techniques for recommender systems–a comparative case analysis. In IOP Conference Series: Materials Science and Engineering (Vol. 1085, No. 1, p. 012011). IOP Publishing.
Tie-Yan Liu (2009). Learning to Rank for Information Retrieval.
Vakkari, P., 2008. Task complexity, problem structure and information actions: Integrating studies on information seeking and retrieval. -13 Inf. Res. 13 (3), 13. https://doi.org/10.47989/iris.v13i3.344.
Wang, H., Wang, N., & Yeung, D.Y. (2015). Collaborative deep learning for recommender systems. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1235-1244).
Yin, F., Fu, R., Feng, X., Xing, T., Ji, M., 2023. An interpretable neural network TV program recommendation based on SHAP. Int. J. Mach. Learn. Cybern. 14 (10), 3561–3574.
Zhang, S., Yao, L., Sun, A., Tay, Y., 2019. Deep learning based recommender system: a survey and new perspectives. ACM Comput. Surv. 52 (1), 1–38.
Zhong, J., & Negre, E. (2022, April). Shap-enhanced counterfactual explanations for recommendations. In Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing (pp. 1365-1372).

Nghia Nguyen Duy (B.S.) received the B.S. degree in Management Information System (MIS) from the Faculty of Information Systems, University of Economics and Law (VNU–HCM), Vietnam, in 2023. He is currently a Regional Data Engineer at Debt Collection Agency across 7 countries in Asia. His research interests include data analytics, and machine learning.

Van-Ho Nguyen (M.A.) received the B.S. degree in Management Information System (MIS) from the Faculty of Information Systems, University of Economics and Law (VNU–HCM), Vietnam, in 2015, and the Master's degree in MIS from the University of Economics Ho Chi Minh City, Vietnam, in 2020. He is currently a Lecturer at the Faculty of Information Systems, University of Economics and Law, VNU-HCM. His current research interests include business analytics, business intelligence, data analytics, and machine learning.

Trang Trinh (B.S.) is currently pursuing the degree in management of information systems with the Faculty of Information Systems, University of Economics and Law, VNU-HCM. She was a Student with excellent academic performance and active participation in extracurricular activities. She has received many encouraging scholarships; award-winning the champion of the "Business Intelligence" surpassing approximately 300 teams. She was also a Delegate Representative of Vietnam who participated in "the 7th Asian Future Leaders Summit" in Malaysia. She was a Core Organizer of the project "Data Analytics and Data Privacy" funded by the American Government for helping Vietnamese citizens acquire data skills.

Thanh Ho (Ph.D.) received M.S. degree in Information Technology from University of Information Technology, VNU–HCM, Vietnam in 2009 and Ph.D. degree in Information Technology from University of Information Technology, VNU-HCM, Vietnam in 2018. He is currently an Associate Professor, Senior Lecturer in Faculty of Information Systems, University of Economics and Law, VNU–HCM, Vietnam. His research interests are data mining, data analytics, business intelligence, social network analysis, machine learning and big data.

Hoanh-Su Le (Ph.D.) received the B.E in Electronics and Telecommunication, MSc in MIS and MBA degrees from Vietnam National University HCM City in 2009 and 2011. He received Ph.D. degree in E-business from Pukyong National University, Republic of Korea in 2016. Since 2011 he has been a faculty member and currently Dean (Tenure) of Faculty of Information Systems at University of Economics and Law, Vietnam National University Ho Chi Minh City (VNU HCM). His research interests are data analytics, artificial intelligence and fintech.