Received: 15 December 2024 | Revised: 12 June 2025 | Accepted: 30 June 2025 | Published online: 16 July 2025

RESEARCH ARTICLE

Artificial Intelligence and Applications
2025, Vol. 00(00) 1-15
DOI: 10.47852/bonviewAIA52025020

BON VIEW PUBLISHING

A Framework for Adaptive Recommendation in Online Environments

Rogério Xavier de Azambuja[1,2,3,*], A. Jorge Morais[2,4], and Vítor Filipe[3,4]

[1] *Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS), Brazil*

[2] *Department of Science and Technology, Universidade Aberta (UAb), Portugal*

[3] *School of Science and Technology, Universidade de Trás-os-Montes e Alto Douro (UTAD), Portugal*

[4] *Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência (INESC TEC), Portugal*

Abstract: Recent advancements in deep learning and large language models (LLMs) have led to the development of innovative technologies that enhance recommender systems. Different heuristics, architectures, and techniques for filtering information have been proposed to obtain successful computational models for the recommendation problem; however, several issues must be addressed in online environments. This research focuses on a specific type of recommendation, which combines sequential recommendation with session-based recommendation. The goal is to solve the complex next-item recommendation problem in Web applications, using the wine domain as a case study. This paper describes a framework developed to provide adaptive recommendations by rethinking the initial data modeling to better understand users' dynamic taste profiles. Three main contributions are presented: (a) a novel dataset of wines called X-Wines; (b) an updated recommendation model named X-Model4Rec – eXtensible Model for Recommendation, which utilizes attention and transformer mechanisms central to LLMs; and (c) a collaborative Web platform designed to support adaptive wine recommendations for users in an online environment. The results indicate that the proposed framework can enhance recommendations in online environments and encourage further scientific exploration of this topic.

Keywords: recommender systems, dynamic taste profile, deep neural networks, transformers, attention model

1. Introduction

Owing to their usefulness and current high worldwide interest, recommender systems have gained importance in various fields such as e-business, e-commerce, e-tourism, e-learning, etc. Thus, different heuristics, architectures, and techniques for filtering information have been proposed to obtain successful computational models for the recommendation problem. One of the latest developments in the field is the use of machine learning, specifically deep learning, which has enabled the growth and expansion of sequential recommender systems [1–3]. Various computational recommendation models have been developed using architectures involving deep neural networks (DNNs).

To solve the complex next-item recommendation problem, in which each user desires only one specific target item among all the items [4], state-of-the-art technologies such as machine learning-based attention and pretrained transformer mechanisms [5, 6], which form the backbone of large language models (LLMs), have been studied and used as opportunities to create successful models. However, when working in an online environment like on the Web, specific characteristics need to be considered, because continuous adaptation mechanisms are required to handle new and obsolete sets of users, items, and feedback in a contextual moment on each recurring iteration at time intervals [7]. This presents a significant challenge in development.

***Corresponding author:** Rogério Xavier de Azambuja, Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS), Brazil. Email: rogerio.xavier@farroupilha.ifrs.edu.br

This research uses sequential recommender systems, which process serialized data input to generate recommendations [8–10], working on Web applications in a case study of the wine sector to recommend wines to Web users. The wine domain was chosen as a product of great worldwide appreciation, and it is important for the economies of Portugal and Brazil, especially in the Serra Gaúcha region of Rio Grande do Sul. In this way, a framework was studied, developed, and evaluated in the wine domain by collaborative users to provide adaptive recommendations in online environments.

In addition to creating an unprecedented and consistent dataset in the wine domain called X-Wines, which fills a gap identified in the scarcity of data in this field, state-of-the-art computational models were studied to offer contributions through our model to the next-item recommendation named X-Model4Rec – eXtensible Model for Recommendation. This built model proposes a new modeling for the initial data considering the user's dynamic taste profile (DTP) and an extensible and scalable architecture using state-of-the-art technologies. To support this research in the online environment, a collaborative Web platform with free access by users was built, providing adaptive wine recommendations to Web users registered on this platform.

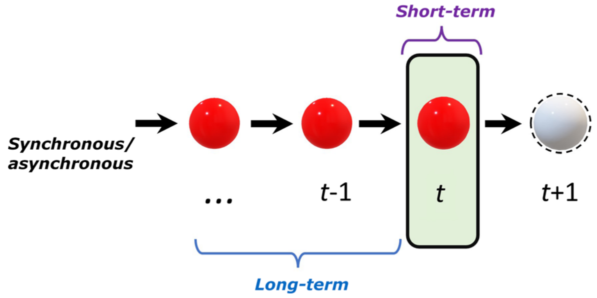
This paper presents an advanced study that combines theory with practice to produce knowledge about adaptive recommendations to minimize the impact of information overload in online environments. First, the research background is characterized, and related works are presented. Then, the aspects of the constructed artifacts and their usability in recommender systems through the conducted experiments are described. After, the results obtained from X-Model4Rec via the novel X-Wine dataset are compared with those of classical and baseline models, and an unprecedented experiment involving wine recommendations for collaborative users in a controlled online environment is presented. Finally, the addressed advances are discussed, and the study is concluded, indicating future research directions.

2. Background

Formally, the recommendation problem can be defined as finding a utility function to recommend one or more relevant items ranked by one output score obtained for each user [7]. Mathematical and statistical methods have been used to add accuracy to human perception, which can be varied to find similarity or diversity in recommended items. In addition, users' preferences can be sparse, scarce (cold start problem), undergo temporal changes, or be estimated by means of prediction mechanisms. The performance of recommender systems is evaluated through specific metrics such as precision, recall, F1 score, and others, which check whether the target user liked or interacted with a particular item among the recommended items [11, 12]. The position in which items are recommended can be considered in the evaluation measured by metrics such as Normalized Discounted Cumulative Gain (NDCG), Mean Average Precision (MAP), and Mean Reciprocal Rank (MRR).

Different approaches to information filtering based on collaboration, content, knowledge, demographics, context, or even hybrid methods [3, 7] usually offer a top set of items containing valuable recommendations as outputs. Sequential recommendation systems, which use serialized data input, and session-based recommendation systems, which utilize grouped data input, have demonstrated significant potential for effective solutions for the recommendation problem, particularly when employing DNN techniques in their development [8, 10, 12]. Specifically in online environments, it becomes necessary to consider recurring data sets available at each contextual moment to generate recommendations. As illustrated in Figure 1, the time intervals *t* (measured in various units such as minutes, hours, days, weeks, and months) can either be uniform or varied, and the treatment of sequential events from the data sets available, which include timestamps, allows us to model user behavior in both short- and long-term dynamics.

Figure 1
Mapping of each user's behavior over a time interval



The next item recommendation is studied in a context where each user desires only one specific target item. On the basis of information presented in the literature [13, 14], as formalized in this investigation, the next-item recommendation problem can be formally defined as the task of accurately predicting a specific next item (denoted as $\hat{\imath}$) for each user from a set of all available items (or classes). This prediction is based on user and item features, as well as contextual information, including a time stamp. The following general formula can be used to express this classification problem:

$$\hat{\imath}^{u}_{t+1} = \operatorname{argmax} p(i \in I \mid U,\ I,\ C(t)) \tag{1}$$

where $I = \{i_1, \ldots, i_{|I|}\}$ denotes a set of items, $U = \{u_1, \ldots, u_{|U|}\}$ denotes a set of users, and $C = \{(U \times I)_1, \ldots, (U \times I)_{|C|}\}$ denotes the sets of user feedback available in a contextual moment $C{=}f(t)$, which vary according to the time interval (t).

To generate a feasible solution for the next-item recommendation problem, several factors must be considered. Each user demonstrates unique and evolving preferences, which difficult the process of discovering, scaling, and generalizing a solution that has a real impact. In this context, the recommender system aims to rank a list of items and identify a single target item from the best k (top@k) recommended items for each user [4, 15, 16], where k is the number of recommended items.

Typically, a standard algorithm for information filtering implements Equation 1 via different techniques and approaches, generally consisting of the steps of executing or predicting the utility function $f(u,\ i)$ from U, I, and C, which relates to the feedback ($U \times I$), recommending the top@k items for each user ranking the highest probabilities (p) found and evaluating the generated recommendation via specific metrics. This problem involves various areas, including classification, ranking, prediction, and data retrieval. It grows with the scalability typically seen in online environments, as any item among all available options could be the next candidate, with a probability of $p = \frac{1}{|I|}$ at each recurring time interval.

Considering real users, the taste profile can be represented systematically by a list of characterizing terms and their relevance values for each user. In the literature, the taste profile is considered important in the recommendation problem. It has not been fully explored by existing computational models, as it is difficult to obtain, and there is a high possibility that people present different tastes over time. Thus, a modeling of the user's DTP while considering temporal variations should be explored to contribute to heuristics that use deep learning techniques to find a feasible solution to the next item recommendation problem.

Furthermore, the search for better use of sequential and session-based recommendations to generate a solution to the next item recommendation problem can be expensive, as it becomes necessary to experiment with different software architectures with different information filtering approaches and heterogeneous data sources with explicit and implicit feedback. Traditionally, explicit feedback has been used, but the dynamics of implicit feedback that can be captured in online environments have required the creation of specific computational models to produce adaptive recommendations in a recurring manner on Web applications.

This investigation focuses on the specialization of existing models to contribute by constructing an artifact designed in this research: its own computational recommendation model and the necessary support for its operation in an online environment. The research follows a design science research methodology, which is structured into iterative stages to guide both the development and evaluation processes. The aim is to evaluate the quality of the recommendation generated and the impact of the proposed modeling on the user's DTP through a constructed framework. It begins with a thorough literature review and replication of advanced recommendation models to identify limitations in web-based contexts. Subsequently, models are developed and experimentally evaluated with various hyperparameters and both implicit and explicit feedback to ensure real-time applicability, performance, and scalability. The final contribution is a novel recommender system emphasizing interpretability and adaptability to dynamic user behavior, including cold-start scenarios, validated through comparative analyses and comprehensive corrective and refutative assessments. The motivation is to discover new solutions that can be replicated, offering concrete insights into the next-item recommendation problem on the Web, especially within the scientific community and small to medium enterprises across various sectors. By providing a reproducible framework that incorporates scalable modeling techniques.

3. Related Works

A comprehensive literature review revealed that recent studies have explored personalized recommendation strategies beyond conventional heuristics. Some heuristics used to understand users' personalized behaviors tend to highly recommend popular items that are well-rated by others. These heuristics often mistakenly assume that past interactions with items increase their relevance, which is not always true. Users may not be aware of all available items, and the next item a user encounters may receive a low rating [17, 18]. The authors surveyed largely agree that users typically interact with only a select few items from the total available options, and there is no straightforward linear correlation that can reliably predict which item will be chosen next. Several common characteristics are observed in most computational recommendation models studied in the literature up to the present moment, such as follows:

1) They are developed in a static manner and through offline processes, without adequate experimentation in an online environment. However, contemporary models attempt to address this issue by employing continual learning strategies.
2) They rely on databases designed for offline experimentation, and often, the taste profiles are either unavailable or used in a stochastic manner. Furthermore, the challenge of data scarcity needs to be considered.
3) They are assessed via limited metrics that frequently do not accurately reflect real user sessions in the contexts where they are applied, such as web applications. Quality metrics remain an area of concern, with researchers advocating for standardized approaches to ensure reliable evaluation [19].

When processing data and metadata to transform them into useful information, several important factors need to be considered, such as the veracity, scalability of volumes, variability of sets, and update rate, as new contextual moments can be formed over time. One significant issue in recommender systems is known as the cold start problem. This problem arises when insufficient data are available for making recommendations, leading to sparse matrices and requiring multiple estimates. This situation typically occurs during the initial phases of the system's operation or when new users or items are added. To address the cold start problem, some researchers are exploring the concept of collective recommendation for user groups [20], including hybrid models that combine collaborative and content-based filtering or apply machine learning via DNN architectures.

In the context of using DNNs, the task involves learning from embedded data that includes timestamp sets. The objective is to develop a function that leverages the user's context and history to classify the most likely next item from among all available items [12, 21]. Such models often use sophisticated embedding techniques to capture temporal patterns, user–item interactions, and contextual signals, helping to model implicit feedback in a more nuanced way. However, creating computational models that can rapidly converge on effective solutions, especially in online environments, presents significant challenges [7]. Most of the models found in the scientific literature are evaluated offline, aiming to establish realistic practices for assessing a recommender's ability to meet user needs. Additionally, some DNN-based models incorporate methods that allow for alternate calculations based on multiple measures of similarity and diversity in information filtering [18, 22, 23].

Existing research on sequential recommender systems typically emphasizes long-term sessions. In contrast, short-term sessions present unique challenges due to the limited contextual information available, which makes predicting the next item particularly difficult [15]. To capture the dynamics of these sequences, models have employed Markov chains to calculate the probability of selecting one specific item from the ordered list of previously chosen items. Additionally, neighborhood-based models have gained popularity for identifying similar users with related interests, often relying on specific heuristics [24]. A comprehensive study presented in [11] compared approaches such as session-based neighborhoods and sequential rule mining with more complex models using DNNs. The findings indicated that the simpler methods performed better in terms of accuracy measurements. In [25], session-based recommendations were generated via Graph Neural Networks (GNNs), which integrate neural networks with graph theory. In this work, the authors utilized structured graphs to represent session-based sequences, allowing them to capture both the current interests of users and their overall preferences within the active session. Studies on RoBERTa [26] and XLNet [27] illustrate advancements in sequential learning, particularly in terms of capturing intricate dependencies across time-sensitive user interactions.

The Self-Attention based Sequential Recommendation (SASRec) model presented in [28] improved upon the models based on Markov chains, Convolutional Neural Networks (CNNs), or Recurrent Neural Networks (RNNs). SASRec employs the economical architecture Multilayer Perceptron (MLP), which offers optimal computational performance. This model can effectively capture both semantic and syntactic patterns over time through recurrent iterations. The self-attention mechanism has been evaluated in long-term scenarios involving dense datasets and in situations with sparse datasets derived from more recent user interactions. Although effective in some cases, newer techniques, such as self-attention mechanisms, offer superior adaptability by recognizing latent user preferences more accurately.

The Adaptively Distilled Exemplar Replay (ADER) model [29], an enhancement of SASRec, focuses on building robust and scalable session-based recommender systems in online environments. It was published at the largest conference on the topic, ACM/RecSys[1], in 2020. The ADER model represents an innovative approach to address the classic problem of catastrophic forgetting related to past data. It dynamically selects a portion of the repeated data examples via a herding method to be reused in the next cycle's training. ADER delivers outstanding results, surpassing models that rely on retraining with all historical data at each time interval. However, practical tests have indicated that the model requires significant computational power, as the training process remains slow. Therefore, deployment at scale may necessitate the use of parallel computing or specialized hardware acceleration.

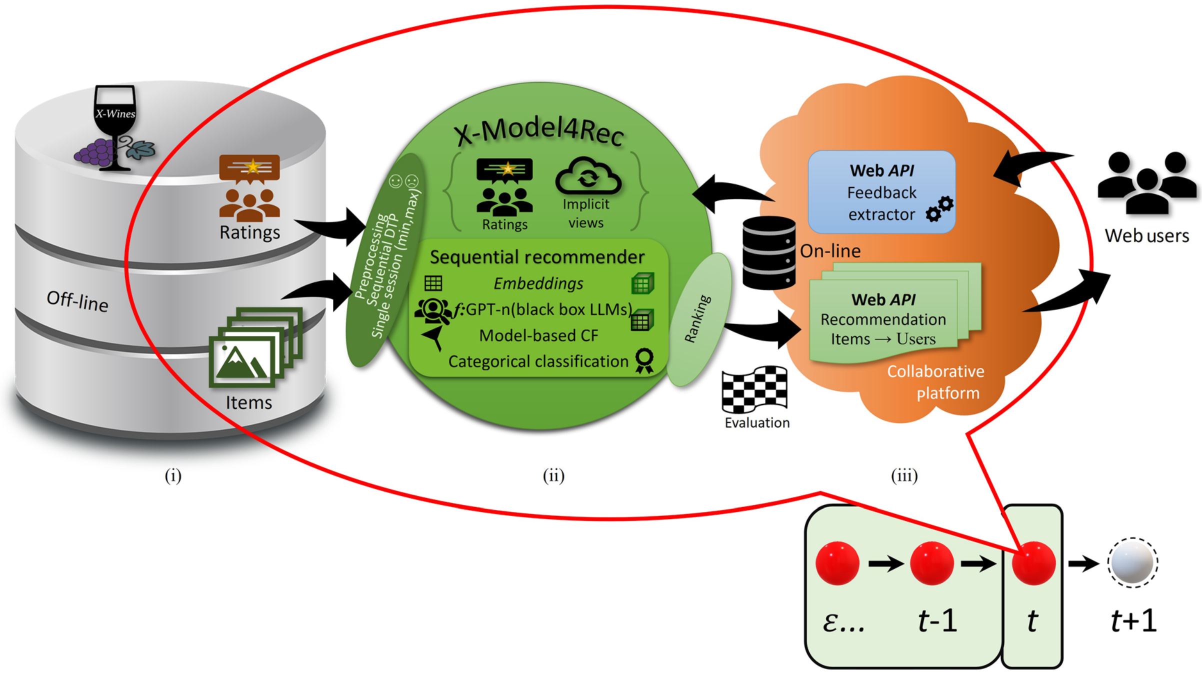
The personalization of time [30], user experience [18], and explanations of recommendations have been studied in the literature, with few studies in the field of wine [2, 31]. In such niche domains, user preferences are highly subjective and context-dependent, requiring models to incorporate additional dimensions such as sensory attributes, origin, and occasion of use to ensure accurate recommendations. Finding a solution that emerges from the initial data is challenging, and the use of generative technologies through pretrained transformer mechanisms [32, 33] has enabled new studies as state-of-the-art technologies in recommender systems. The application of deep learning in recommender systems presents a significant challenge and area of research. Recently, the rise of attention [5] and the adoption of transformer mechanisms [17, 27] have attracted considerable interest. Transformers are now being applied for sequential recommendation prediction, where dialog history and intent understanding play key roles in recommending items to users.

4. The Framework Produced

The recommendation in this research is generated by observing available datasets at different contextual moments, illustrated by the

[1]https://recsys.acm.org

Figure 2
Artifacts built to support adaptive recommendation in an online environment



time intervals [ε, *t*] in Figure 2. The work developed focuses on a specific type of recommendation, which combines sequential recommendation with session-based recommendation. The goal is to solve the next-item recommendation problem on Web applications, using the wine domain as a case study. Computational recommendation models developed using different approaches to information filtering and architectures with DNNs were specialized, and state-of-the-art technologies such as machine learning-based attention and transformer mechanisms have been studied and used. The official research repository entitled "Wine data for wider use" provides support at address https://sites.google.com/farroupilha.ifrs.edu.br/xwines (last accessed 2024/11/11).

Figure 2 illustrates the framework developed for the object of study: adaptive recommendation is based on a new modeling of the initial data to explore the user's DTP to address information overload in online environments. As a contribution to science, the following artifacts are proposed: (a) a novel database of wines called X-Wines, (b) an updated recommender model named X-Model4Rec – eXtensible Model for Recommendation, which is based on modeling the user's DTP, and (c) a collaborative Web platform to support adaptive wine recommendation to users in an online environment. Using the proposed framework, the adaptive recommendation is generated by recurring processes at temporal intervals and updated to the collaborative user participants of the Web platform.

This framework establishes a recurring mechanism that can be installed and configured to provide adaptive and portable recommendations for Web applications. It enables a software package to be imported and widely utilized in online environments. Structured and unstructured historical data are collected through a feedback extractor module. A preprocessing method is executed in the background at timed intervals to model and encode the user's evolving taste profile, generating data at a contextual moment for retraining the proposed model. When X-Model4Rec is used, features that aim to interpret the nonlinear relationships between users, items, and their preferences are prioritized. As the final step of this mechanism, the newly generated recommendations are dynamically updated for users of the Web application, identified on the right in Figure 2. The three artifacts featured are detailed below.

4.1. X-Wines dataset construction

Like most agricultural products, wine typically has a limited data volume, with few elements that restrict users and hinder scientific exploration, especially in recommendation systems. This research revealed the need to build a novel and consistent wine dataset called X-Wines [34], which is composed of wine instances and ratings performed by real users. The data were collected on the open Web in 2022 and preprocessed for wider free use. In the construction of the X-Wines dataset, the FAIR guiding principles [35] and the international standardization of wine[2] were followed, data protection and privacy laws were respected, no private data was used, and no system had its security checked or violated.

The collected data were verified and validated via electronic processes. Extensive preprocessing was performed on the data gathered, resulting in more than 3,000 lines of source code and the elimination of millions of unapproved data entries. Various formats and classifications were validated to ensure that the presented data sets closely align with reality. Some care was taken to establish a common language for representing knowledge in the wine domain, which facilitated data interoperability. In this way, relevant wine attributes and the evaluation sequences from anonymized users were validated for consistency, enhancing the reusability of the data.

To estimate the reliability of the data obtained from the Web and from images of wine labels via optical character recognition processes, a statistical test was performed. A random sample containing 100 wines and their respective attributes was drawn from among the validated wines. The result was found through the average of all proportions of preprocessed data when compared manually to the official datasheet of producers. The overall result of this document-based benchmark obtained a coincidence of 97.75% and a standard deviation of 4.15%. Thus, with a 95% confidence interval, the real average assertiveness is between 96.94% and 98.56%. The documents, websites, and images used in this document-based benchmark can be found openly in the official research repository.

[2]https://www.oiv.int/what-we-do/standards

4.2. X-Model4Rec – eXtensible model for recommendation

During this research, it was found that sequential and session-based recommendations can be used together in a complementary way with DNNs through training and testing methodologies to predict the target item that a user might be interested in based on what they previously interacted with. It was identified that some users look for the same products, whereas others prefer exploring new items and variations. These preferences can change over time, revealing different patterns of behavior. Each user is unique, interacting with products at varying frequencies, and their individual preferences may vary over time. In this identified scenario for the prediction task, heuristics can emerge from the data itself, that is, considering how the datasets present themselves at certain contextual moments and how they undergo temporal changes becomes an object of interest in this research for the next-item recommendation problem using the wine domain.

The proposed X-Model4Rec model [36] uses single sessions of defined length and consists of an extensible architecture adapted from the input data, as demonstrated in the data flow illustrated in Figure 3. Unlike other models, multiple user ratings of the same items are desirable in X-Model4Rec for processing the user's dynamic taste profile. DTP is sensitive to changes in the categories of items that users have interacted with previously, given by Equation 2, which is used at the core of the processing.

$$DTP_u = \bigcup_{1}^{max} \begin{cases} 1, & \text{changed the last category} \\ 0, & \text{kept the last category} \end{cases} \tag{2}$$

model evaluation via metrics commonly used in recommender systems [1, 11, 19], such as precision and recall to compose F1 score, MAP, MRR, and NDCG, among others. The algorithms constituting the modeling and architecture of X-Model4Rec are presented in detail in [36].

The projected architecture uses the initial data modeling obtained by DTP categorization of items according to the continuity or alternation of the type of wine in which the user interacted over time as input. Then, the labeled data are represented by embedding (S^{emb}) to be used in the inner layers of the model with an economical MLP and are captured by linear transformations of the intricate data about the relationships between users and items ($dataDTP^{emb}$). From the spectrum matrix $\left(p_{|U|\times|I|}\right)$ formed in the prediction task, scores are obtained for the estimated connection probability (p) of each target item for each user. These scores are ranked in descending order, and the highest (k) are separated to compose the top@k recommendation items for each user, where k={1, 10, 20, 30,…} are commonly used.

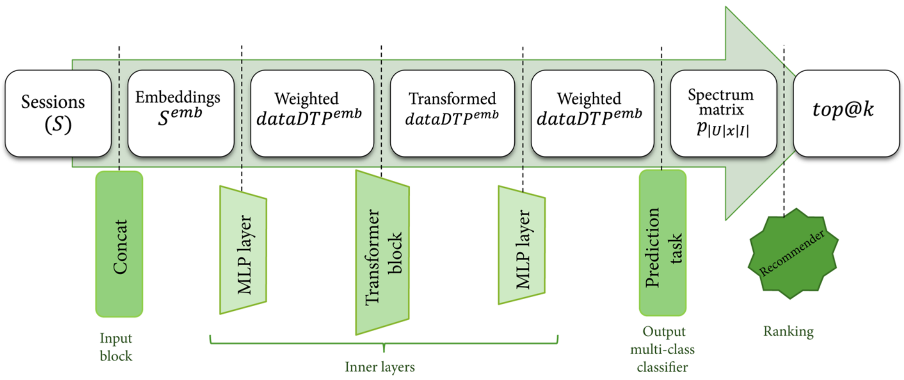
In X-Model4Rec, multi-head attention directed to the next token was used during the model training stage, and attention directed to the last sequential token was used in the prediction stage, that is, generating output scores by applying the concept of attention [5], because, in this way, faster convergence of the proposed model was achieved. Although using different attention heads in data processing may not ensure significant improvements in model performance, multi-head attention has facilitated the discovery of formulations for dynamic objective functions for the prediction of the next-item recommendation problem ($\hat{\imath}$), given by Equation 3.

$$\hat{\imath}_u^{t+1} = argmax\ p\Big(i \in I | multi\_class\_classifier(MLP(Transformer(MLP(S))))^{attention}\Big) \tag{3}$$

Three algorithms were created to compose and evaluate X-Model4Rec: the first algorithm preprocesses the initial datasets to form the training and testing sets using modeling in which sequential data are classified and normalized, resulting in single sessions (S) with a defined length. The second algorithm is designed to train the model, make predictions, and rank the results for the top-k next-item recommendation scenarios. This algorithm uses a DTP for each user to uncover complex relationships within the data. It also monitors the loss rate and the Area Under the Receiver Operating Characteristic Curve (AUC) as performance metrics. Finally, the third algorithm performs

With the extensible functionality of the proposed model, the transformer block at the center of the internal layers can be modified while maintaining the dimensionality of the embedding set. This allows several transformer mechanisms to be attempted. In addition, experiments using the temperature parameter described in [37] were performed on the output classifier to vary the probability distribution considered in the multiclass classification task and obtain a better overall performance of the classifier. To help combat the overfitting problem, in which the training performs much better than the test does, the multiclass classification task uses the weight-binding technique as

Figure 3
Architectural data flow for generating recommendations in X-Model4Rec



a form of regularization [38] applied from the input block to the output classifier. However, underfitting was not detected, in which the training was not satisfactory, which would invalidate the proposed model.

4.3. Collaborative web platform

A Web application was built from software engineering processes to allow free access by real users to the novel X-Wines dataset, with the option for them to receive personalized wine recommendations. X-Model4Rec and other well-known recommendation models in the literature were used to generate recommendations using data in different contextual moments to promote investigations about collaboration in information produced between users in a controlled online environment. The Web application is also symbolized as an orange website in Figure 2, with two possible modules: a feedback extractor and a personalized recommendation presenter.

The project's decision to create its own Web platform was based on the possibility of building a fully manageable online environment with the sole purpose of scientific research. The collaborative platform has made it transparent for the Web user to collaborate and generate explicit and implicit data for scientific research with prior agreement. The user's role was modeled to freely access the collaborative platform in two ways, either fully anonymous with limited functionalities, having access to freely browsing and filtering all the data presented in the X-Wines database or providing little identifying data to allow the user to return in the future and explore all the features of the platform. With logged access, registered users will be able to browse and perform various filters, evaluate wines of interest, monitor with the option to delete their previously registered information, and receive personalized recommendations generated by the studied computational recommendation models. Users of the platform will also be given the option to switch between the two forms of access offered and even disable and completely delete their account on the collaborative Web platform.

User movements on the collaborative platform are mapped to both implicit and explicit data sets at times predefined by the administrator. The administrator's role was designed to monitor the execution of the Web platform, provide support to users when necessary, and enable the extraction and compilation of temporal data, generating new data sets for scientific research. From these generated data sets, combined with the information presented in the X-Wines database and filtered based on user activities, a contextual data sample is created at a specific moment (U, I, and C). These timely generated samples serve as inputs for the computational recommendation model X-Model4Rec, as well as for other selected models. The resulting top@k recommendations are then presented in a personalized way on the collaborative Web platform, as illustrated in Figure 4.

The execution of the recommendation models occurs in the background through an application external to the collaborative Web platform, in which it instantiates the models used with the sample at a given contextual moment. Sequential recommender systems allow the use of a greater variety of features than nonsequential models do. A known disadvantage is the need for a greater volume of data for training sessions, which was provided by merging data collected online with the X-Wines dataset.

As in the construction of the new X-Wines offline database, users of the collaborative Web platform are anonymized during data extraction, preserving their identity. This is because it becomes more important for investigations to record their ratings and movements in a controlled online environment.

5. Experiments and results

Various experiments were conducted in both online and offline environments to test and evaluate the execution performance of X-Model4Rec using the X-Wines dataset, a new and consistent wine dataset produced as part of this investigation. These experiments can be found in the official research repository. The experiments conducted in an online environment are emphasized in this paper, utilizing the collaborative Web platform.

Currently, there are few references to the wine domain in recommender systems. Although some wine datasets can be found in repositories such as Kaggle (https://www.kaggle.com/datasets, last accessed 2024/03/27) and GitHub (https://github.com/datasets, last accessed 2024/03/27), they often lack sufficient relevant data or are not organized with the necessary rigor for scientific exploration. This finding led us to create a large public dataset called X-Wines [33], which is published under a free license for wider free use in the official research repository. The preprocessed data from the open Web refer to the characteristics found on wine labels and consumer evaluations,

Figure 4
Personalized recommendation on the collaborative web platform

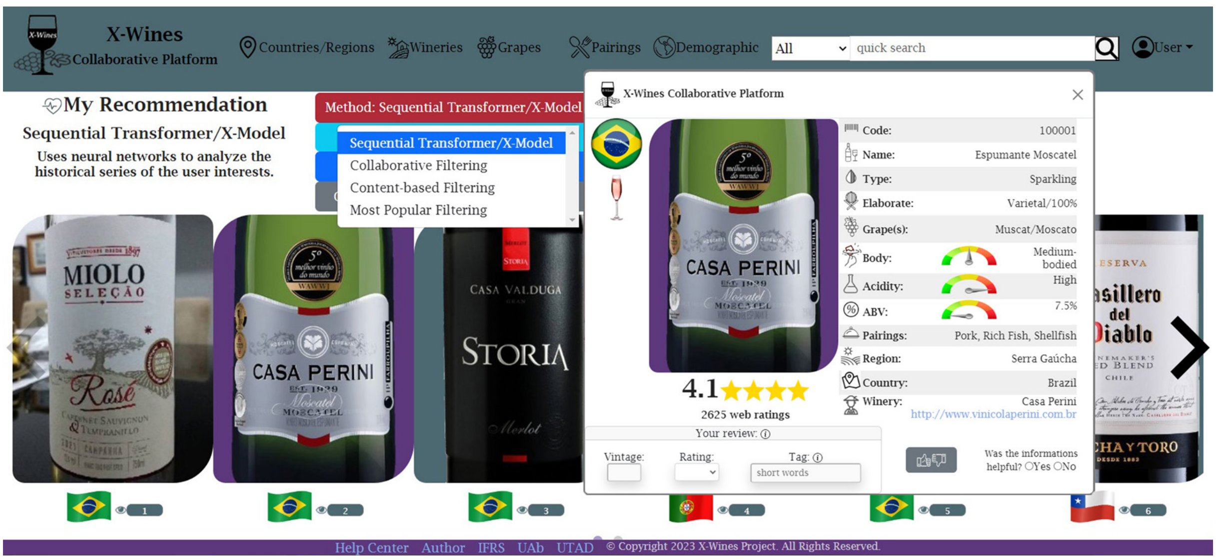


Table 1
Characteristics of X-Wines dataset versions

Version	Wines	Wine types	Wine countries	Users	Ratings	Multiple user wine rating
Full	100,646	6	62	1,056,079	21,013,536	Yes
Slim	1,007	6	31	10,561	150,000	No
Test	100	6	17	636	1,000	No

classified on a five-star scale (1 to 5) over 10 years (2012–2021) for wines that were produced in different countries.

The X-Wines database used in the experiments conducted is available in three versions, one is contained in the other: Full ⊃ Slim ⊃ Test. Full contains all records, which is composed of 100,000 instances and 21 million ratings performed by over 1 million real users; Slim contains a random sample of one percent of instances; and a Test version containing only the 100 wines manually verified by sampling with 1,000 preference ratings randomly selected for the experimentation of the database produced in this research. Table 1 presents the main comparison between the different versions.

Because most of the researched recommendation models are evaluated offline, three experiments are first presented in which the Slim version of the X-Wines dataset is used as initial data. The results of these experiments were significant for calibrating the X-Model4Rec parameter used to generate adaptive recommendations in a controlled online environment, in which the Full version of the X-Wines dataset was used.

5.1. Offline X-Model4Rec calibration

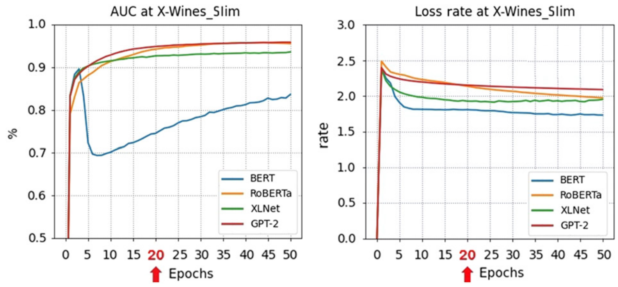
5.1.1. Experiment 1: model extensibility

Owing to the extensible characteristic of X-Model4Rec, which allows the exchange of the transformer block while preserving the dimensionality of the data encoded in the embeddings, at the time of model calibration, four transformer mechanisms were tested using the X-Wines_Slim sample described above. The following transformers were alternated: Bidirectional Encoder Representations from Transformers (BERT) [6], Robustly Optimized BERT Approach (RoBERTa) [26], generalized autoregressive pretrained method LXNet [27], and Generative Pre-trained Transformer (GPT-2) as referenced in [39]. The decision for these was due to their relevance, as reported in the literature [17, 32] and because they were offered freely in black box modules in the framework NVIDIA Merlin[3], which is used in X-Model4Rec encoding. Running a variable distribution as the temperature value [37], the results of this benchmark are shown in Figure 5, and the hyperparameters used were as follows:

1) EMBEDDING_DIM=64; DROPOUT=0.1; HEADS=8; NEURONS=[128, 64]; TEMPERATURE_SCALING=1.0;
2) LOSS=CategoricalCrossEntropy(label_smoothing=0.2, from_logits=True); OPTIMIZER=Adam (learning_rate=0.005);
3) BATCH_SIZE_TRAINING=256; and BATCH_SIZE_TEST=1024.

The AUC provides a value between 0 and 1, indicating how correct the expected predictions are, with a value above 0.5 indicating more correct than incorrect classifications made by the model. The loss rate is directly related to how well the model adjusts to the input data. In this case, when experimenting with different transformer mechanisms, rapid convergence to the smallest error was found, and minimal variation occurred over several epochs. On the basis of the results obtained in this first experiment, GPT-2 was selected for use in the internal block of the

Figure 5
X-Model4Rec training on the X-Wines_Slim dataset



X-Model4Rec architecture, owing to its stability demonstrated in the calibration performed, and 20 epochs were chosen for training.

5.1.2. Experiment 2: X-Model4Rec versus classic approaches

To allow experimentation on the next-item recommendation problem from a defined data sample, typically, the last item that each user interacted with in this sample is previously removed to be compared later with the recommendation generated. Thus, except for the last interaction intentionally removed for each user, all other data in the sample can be used to generate the top@k recommendation, where k is the number of recommended items. Then, the results can be measured via the metrics@k commonly used in recommendation systems [1, 11, 40]. It is verified and expressed on a finite scale to determine whether the top@k items recommended for each user contain a target item with identified interest for the same users, and to consider the positional order in which items are recommended.

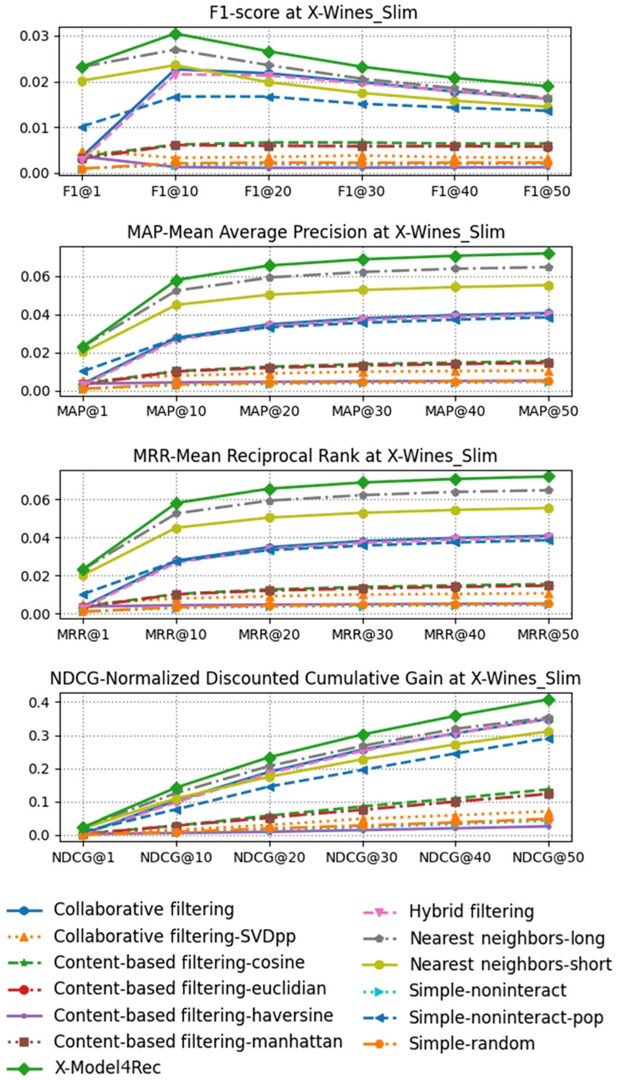
Although it was not possible to cover all the approaches studied in this research, several classic models were tested through different implementations, which are listed in the legend of Figure 6. The models selected for comparison are now offline and are well-known and widely used worldwide, such as Collaborative Filtering, as well as Content-Based Filtering, and implemented through Hybrid Filtering approaches.

Each selected classical model follows specific principles and presents structural variations. For instance, algorithms based on singular value decomposition (SVD) typically reduce matrices to their singular values and then reconstruct them through factorization processes. Additionally, similarity-based algorithms employ different distance measures, such as Euclidean, Cosine, Manhattan, and Haversine; collaboration identifying the short neighborhood, in which only the last user–item interaction is observed, or the long neighborhood observing more interactions, which is equal to 10 in this experiment. These computational recommendation models are not based on random draws but use classical information filtering techniques. To provide a comprehensive analysis, we also included three models identified by the prefix ‘Simple’, which utilize random draws for comparison.

The same hyperparameters as those presented above were used for X-Model4Rec with LLM GPT-2 as the transformer block and the models were trained over 20 epochs. All the models used the same Slim version of the X-Wines dataset as input. Figure 6 presents the results after each compared model is executed.

[3]https://developer.nvidia.com/merlin

Figure 6
Scores obtained during the evaluation of X-Model4Rec and traditional approaches



The F1 score balances the precision and recall metrics, MAP, which is equivalent to MRR in the next-item recommendation problem, that is, searching for a single target item, and the top@k position relevance NDCG metrics were used. The green lines indicate the results of X-Model4Rec in this experiment, extended by many data points produced (Appendix I), which are as good as some classical methodologies found in the literature.

5.1.3. Experiment 3: X-Model4Rec versus baseline models

The results, measured by the area under the ROC curve (AUC), as well as precision and recall evaluation metrics from X-Model4Rec, are compared with four selected baseline models that utilize the original hyperparameters defined by their authors. The related works were selected on the basis of their similarities, including both traditional approaches and those with short- and long-term sequential dynamics. The four baseline models are presented as follows:

1) ATRank [41] uses an attention model, specifically a multi-head attention mechanism, to account for users' heterogeneous behaviors. It incorporates a DNN in both the self-attention layer and the vanilla attention layer to derive user preferences.
2) The one called here as Bias-long short-term memory (LSTM) [42] refers to a bidirectional LSTM network designed to capture sequential implicit data from users through multiple gating mechanisms.
3) Bayesian personalized ranking (BPR)-matrix factorization (MF) as described in [43] is a traditional model that employs BPR alongside the MF method. It is trained on both positive and negative user ratings, as well as the concatenated embeddings of item IDs and categories.
4) The Long- and Short-Term Preference Model [4] uses a trainable matrix to capture long-term preferences from users while integrating short-term preferences to develop comprehensive user profiles for generating personalized online recommendations.

The same hyperparameters as those presented above for X-Model4Rec with LLM GPT-2 as the pretrained transformer block were used. All the models used the same *Slim* version of the X-Wines dataset as input and were trained over 20 epochs. Table 2 shows the results after each model was compared. The values that are highlighted in bold and underlined represent the most favorable outcomes in each column.

The results presented by X-Model4Rec outperformed the scores of the other classical and baseline methods tested in these experiments using wine data samples. It obtained the highest average score for the quantity of recommended wines desired by Web users, and the last line of Table 2 demonstrates a significant improvement. In addition to the experiments developed, the model execution yielded positive results when tested with various data volumes, allowing greater scalability, and rapid convergence with the extensible use of transformer mechanisms.

The multi-head attention mechanisms and pretrained GPT-2 transformer used in X-Model4Rec focused on the subsequent element of the input list with sequential displacement between the elements, which resulted in good results. However, the monitoring performed by the AUC metric in this experiment indicates that it is possible to further improve the proposed model, for example, by observing the external hyperparameters used, including bias adjustment and internal verification of the architecture in new future experiments.

Table 2
AUC, precision, and recall metrics when evaluating on X-Wines_Slim dataset

		Precision@						Recall@					
Model	**AUC**	**1**	**10**	**20**	**30**	**40**	**50**	**1**	**10**	**20**	**30**	**40**	**50**
ATRank	0.8859	0.0180	0.0143	0.0123	0.0108	0.0098	0.0089	0.0180	0.1430	0.2452	0.3253	0.3906	0.4453
Bias-LSTM	0.8830	0.0175	0.0143	0.0122	0.0108	0.0098	0.0089	0.0175	0.1427	0.2443	0.3244	0.3905	0.4463
BPR-MF	0.8161	0.0081	0.0075	0.0070	0.0066	0.0063	0.0061	0.0081	0.0751	0.1391	0.1974	0.2534	0.3063
LSPM	0.8045	0.0086	0.0073	0.0067	0.0063	0.0060	0.0057	0.0086	0.0731	0.1339	0.1898	0.2398	0.2856
X-Model4Rec	0.8444	0.0260	0.0177	0.0144	0.0122	0.0107	0.0097	0.0260	0.1773	0.2889	0.3669	0.4299	0.4842
Percentage	−4.68%	+44.44%	+23.78%	+17.07%	+12.96%	+9.18%	+8.99%	+44.44%	+23.99%	+17.82%	+12.79%	+10.06%	+8.74%

5.2. Evaluating the online adaptive recommendation

To generate adaptive wine recommendations in an online environment at different contextual moments and verify the usability of the produced framework, a specific experiment was conducted. Over six months, the collaborative Web platform was utilized by wine enthusiasts over 18 years of age who voluntarily agreed to participate in this investigation. Several participants shared access to the platform and invited other Web users, creating a cascading effect. The majority of registered users identified Brazil and Portugal as their countries of origin; however, several nationalities were represented in smaller numbers, including Italians, French, English, Germans, North Americans, Chinese, and Koreans.

Out of the 215 accesses identified by potential users on the collaborative Web platform, there were 63 instances where there were insufficient data to establish sequential sessions. Additionally, optional registration data were not available for these users, preventing them from receiving recommendations and participating in the evaluation. These cases involved users who accessed the platform but did not view any wines or engage in navigation, and they chose not to create credentials for registered access. Figure 7 illustrates the steps followed in an unprecedented experiment in a controlled online environment.

Following these steps, explicit and implicit feedback produced from user navigation on the collaborative Web platform were captured (Step 1); the data captured at weekly time intervals (t = 7 days) were merged with the Full version of the X-Wines dataset presented in Table 1 (Step 2) to produce sample instances of the filtered dataset called X-Wines_Online (Step 3), which contains enough data for sequential sessions of fixed length. The dataset used in this experiment contains 15,699 wines, 102,182 users who rated these wines, and 339,448 ratings merged with the data captured from 215 Web users, with 152 used due to collaborative Web platform usage. These users produced interactions in wines, with 1,158 implicit views (V) and only 15 explicit ratings (R). In this online experiment, beyond X-Model4Rec, three other well-known classic models [44] were also tested (Step 4), namely, collaborative filtering by a long neighborhood with a size of 10 (the same value assigned to the maximum length of single sessions in X-Model4Rec, modeled with sessions between two and 10 items), content-based filtering using the cosine similarity method, and a model that uses an approach considered simplistic, filtering among the most popular items. Finally, after receiving the personalized recommendation (Step 5), Web users were invited to participate anonymously in a satisfaction survey, in which the first 100 reviews were counted (Step 6).

The *top@10* recommendations generated in four different models in Step 4, all using the same contextual X-Wines_Online dataset, were presented to Web users in a personalized way. Subsequently, 100 of these Web users anonymously and spontaneously responded via an electronic form with options ranging from zero to 10 to the following question: "Please, for each recommender, how many wines would you like/interact with in each recommendation you received?" The results measured by the weighted average are shown in Table 3, and the official research repository contains all the data, source code, and forms.

Figure 7
The framework in a controlled online environment

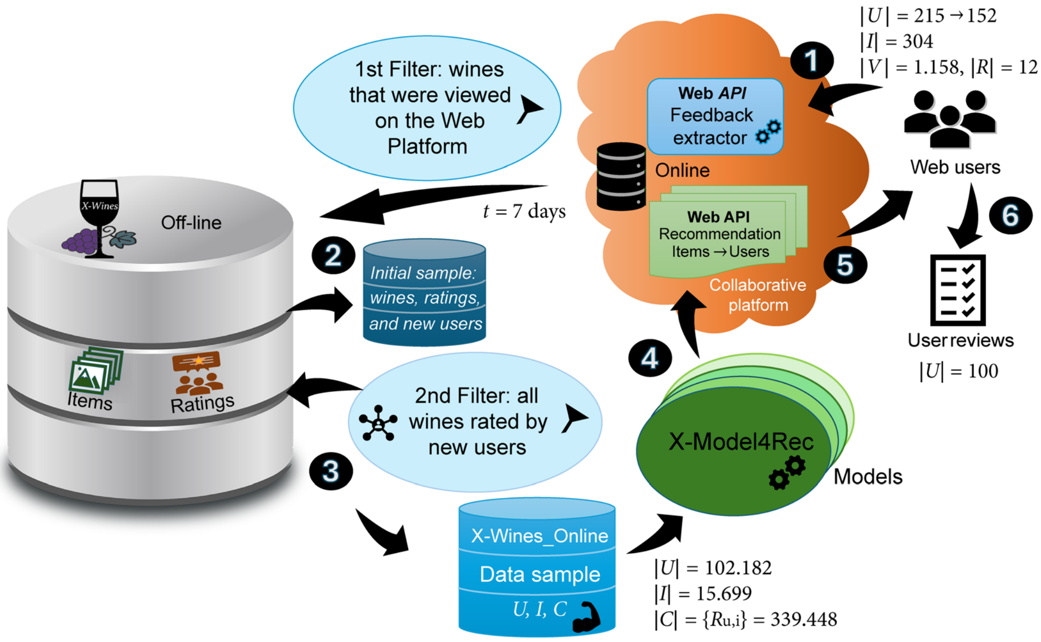


Table 3
Evaluation of the adaptive recommendation by users from the Web platform

	The quantity of wines recommended as a possible next-item recommendation problem											
Model	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**∑**	**x̄**
X-Model4Rec	4	4	5	6	8	17	12	18	13	13	100	6,68
Collaborative filtering	4	8	15	16	13	14	11	7	6	6	100	5,25
Content-based filtering	8	18	12	10	11	12	9	9	3	2	100	4,29
Most popular filtering	21	11	12	10	12	8	6	8	3	3	*100*	3,90

The results produced by X-Model4Rec achieved the highest average score in terms of the quantity of recommended wines that users desired for future interactions, as calculated by the weighted average.

The three algorithms mentioned were implemented with the following versions: Python 3.10.10, Pandas 2.0.1, Numpy 1.24.3, TensorFlow 2.10.0, Transformers 4.31.0, Merlin-Models and NVTabular 23.6.0 from NVIDIA on version 11 of Debian GNU Linux run on a microcomputer with CPU 6-core i7-10750H of 2.60 GHz, integrated 10th Gen GPU UHD Intel, GeForce GPU GTX 1650 NVIDIA, NVMe M.2 SSD, and 16 GB of RAM. Under these conditions, the recommender X-Model4Rec trained faster than some other verified models did, achieving the following runtimes for each input sample verified:

1) X-Wines_Slim: preprocessing in 12.7 s, training and testing in 301.5 s, and running each epoch in 11 s.
2) X-Wines_Online: preprocessing in 101 s, training and testing in 2,285 s, and running each epoch in 112 s.

The execution times of each tested model needed to be hidden from this presentation, as some implementations found in the reference literature have required different versions of the Python programming language and their own libraries and must be executed in different virtual environments to find the results presented. It is important to highlight that for these observed runtimes, this proposed framework guarantees a non-prohibitive computational time for the task of generating recommendations. Moreover, the main scores obtained through the evaluation metrics provide a solid basis for subsequent implementations.

6. Discussion

This research aims to improve the recommendation in online environments for the wine domain, as developed in a thesis work [45]. However, the Web offers a vast amount of information that needs to be properly handled and validated to be useful. Without proper verification and validation, data volumes may be simple simulations and may not reflect reality. Filling a gap detected was the main motivation for building an unprecedented wine database, in addition to making it openly available to the scientific community. The discovery of this need directed this research first to the creation of a large and consistent real dataset, made public and accessible, containing more than 100,000 wine labels, and 21 million five-star ratings classified by 1 million users on the Web.

There are models such as those used in the previous experiments and several others that propose finding or estimating all the classifications, first forming a matrix filled with all the elements U×I and then executing other more specific heuristics. However, most of the models considered only unique relationships, that is, a single user-item classification, disregarding other reviews made by a user on the same items, data that admittedly can be important for the generation of the next item recommendation. In X-Model4Rec, more than one review and negative ratings on target items are welcome to generate the recommendation; As a result, heuristics that recommend only from single reviews, or even reviews considered positive, tend to ignore a large portion of possibilities in items when generating the next item recommendation.

Existing approaches are usually based on identifying user preferences. Among the various possible approaches to the construction of a recommendation system, the excellent results presented by the recommendation model that uses collaborative neighborhood-based filtering can be verified. Even though it is not a guarantee for all possible situations, which certainly no approach will be complete, it presents greater difficulty than other approaches in the treatment of large volumes of data and the cold start problem when there is a lack of feedback from users. Thus, there is a need to observe scalability, which is a limiting factor in large real-world applications.

This research focused primarily on conducting experiments to find a way to “improve the neighborhood”, either by new user–item relationships or by rearranging existing relationships with new weights and weightings that could be exploited in some existing neighborhood algorithms. After conducting several experiments with varied heuristics, these experiments were successful, and it was possible to execute the same neighborhood algorithm with and without the previous transformation of the same initial data, obtaining better results in cases where it was previously tried to “improve the neighborhood.” Later, a new modeling was proposed to identify the alternation or continuity in the categorization of items on which users interacted in the past, forming single sessions of defined length to exploit the so-called user’s dynamic taste profile (DTP). From the experimentation of this new proposed modeling, it was possible to arrive at the computational recommendation model X-Model4Rec – eXtensible Model for Recommendation. State-of-the-art technologies, such as attention mechanisms, which have already been used in recent years, and innovative transformers, which are growing and deserve to be experimented with to contribute to various fields, have allowed the implementation of the paths taken in this research. Transformer mechanisms have many applications focused on problems related to natural language processing; however, the results presented in this study prove their application and the need for continuity in studies related to multiclass classification problems in recommendation systems.

In this way, it is important to emphasize that the use of transformer mechanisms implies the definition of attention modules, in which in this investigation a multi-head attention resource was used that was directed to the next sequential element in the training and the last element of the input sequence in the prediction. The mechanism implements positional feed-forward networks, in which the positional encoding of the initial tokens is performed to be used a posteriori in the proposed architecture of the transformers and the residual connection and normalization are performed in several steps in the network to stabilize the output of the transformed data. In this way, distant trends and relationships are found between users and items in the data and as verified between situations presented in other mechanisms, avoiding deformations and anomalies in the data very different from the others (outliers) during training that contribute to the acceleration of the convergence desired by computer models. This approach can generate good results for a classification problem.

With the architecture defined for the internal layers of the developed model, it was possible to maintain the dimensionality of the data encoded in the embeddings between the layers, allowing the alternation of the transformer block, if desired. As demonstrated in this research, the alternation of the transformer block defined as a component of the model can be useful in solving the recommendation problem in different domains, in which better results can be obtained by experimenting with not just one, but also distinct pretrained transformer mechanisms with varied large-scale datasets and composed of different DNN architectures.

The current definition of hyperparameters presented followed a procedure of varied scientific experimentation to select the best values and algorithmic functions among the possibilities found in the literature. They were considered sufficient to support the execution of the X-Model4Rec built in this investigation. However, even though several ablation tests were performed that allowed good parametric values to be found to generate the verified recommendation, not all possible combinations were tested, which is a limitation of this research. In this sense, new studies may still be conducted to verify the parametric fine-tuning.

In online environments, it is important to consider data dispersion as a problem, because Web users can reach millions and produce sparse data. It was found that, in addition to the efficiency and novelty of a new computational recommendation model that exploits specific characteristics in the user–item relationship, the sequential recommendation system should inevitably process sparse data and be executed in non-prohibitive computational time to update the recommendation generated for Web users. Given that updates in online environments are recurrent, it is crucial to weigh the time intervals really necessary for the retraining of the recommender model and the effective update of the recommendations, which needs to be defined in an equalized way for the proper functioning of the adaptive recommendation in Web applications.

7. Conclusion

This research focused on solving the adaptive next-item recommendation problem in online environments for the wine domain. The resulting artifacts, which include an unprecedented wine dataset and an extensible recommender model using pretrained transformer mechanisms, were verified offline and online with the participation of real users registered on our collaborative Web platform.

This research contributes to showing that efficient generative technology (designed to create content by tokenizing words) can be satisfactorily used for a classification problem, to make predictions in classes (or items) to solve a complex next-item recommendation problem. Additionally, with emphasis on the online environment, pretrained transformer mechanisms offer a promising treatment for dynamicity and recurrence to generate updated recommendations at time intervals for Web users.

The experiments using wine data samples produced promising results for achieving the proposed objective of creating and validating a sequential recommendation model to verify the impact of dynamic taste profiling on efficiently providing scalable session-based recommendations for realistic Web applications. The evaluation responses by collaborative Web users to the recommendation generated in this research presented the best average interaction between the classic and baseline models evaluated. Furthermore, a portable solution for Web applications was produced that can generate recommendations via pretrained transformer mechanisms in a non-prohibitive computational time. This framework has proven to be an effective way to adapt the proposed model to work from the input data, producing satisfactory results. It improves the recommendation in online environments and promotes further scientific research on a specific topic.

Finally, X-Wines was cited by other authors and indicated as a dataset for the production of scientific works on the Recommender Systems International Conference ACM-RecSys (https://github.com/ACMRecSys/recsys-datasets, last accessed 2024/11/11). In future work, new advances may be promoted in enhancing the proposed X-Model4Rec recommender, including bias adjustment, internal check architecture, and a revised setup for new experiments with LLMs mechanisms, as well as performing new evaluations using standard metrics employed in recommender systems.

Recommendations

Training on recommender systems, deep learning, dataset analytics, and the Python programming language is recommended for those who want to use this framework in whole or in part.

Acknowledgement

We thank the Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS) from Brazil for supporting this research.

Ethical Statement

This study does not contain any private human data or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in X-Wines Research Project at https://sites.google.com/farroupilha.ifrs.edu.br/xwines.

Author Contribution Statement

Rogério Xavier de Azambuja: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, and Project administration. **A. Jorge Morais:** Conceptualization, Methodology, Validation, Formal analysis, Data curation, Writing – review & editing, Visualization, Supervision, and Project administration. **Vítor Filipe:** Conceptualization, Methodology, Validation, Formal analysis, Data curation, Writing – review & editing, Visualization, and Project administration.

References

[1] Alhijawi, B., Awajan, A., & Fraihat, S. (2022). Survey on the objectives of recommender systems: Measures, solutions, evaluation methodology, and new perspectives. *ACM Computing Surveys*, *55*(5), 1–38. https://doi.org/10.1145/3527449

[2] Li, Y., Liu, K., Satapathy, R., Wang, S., & Cambria, E. (2024). Recent developments in recommender systems: A survey. *IEEE Computational Intelligence Magazine*, *19*(2), 78–95. https://doi.org/10.1109/MCI.2024.3363984

[3] Shao, B., Li, X., & Bian, G. (2021). A survey of research hotspots and frontier trends of recommendation systems from the perspective of knowledge graph. *Expert Systems with Applications*, *165*, 113764. https://doi.org/10.1016/j.eswa.2020.113764

[4] Du, Y., Liu, H., Qu, Y., & Wu, Z. (2018). Online personalized next-item recommendation via long short term preference learning. In *PRICAI 2018: Trends in Artificial Intelligence: 15th Pacific Rim International Conference on Artificial Intelligence*, *1*(15), 915–927. https://doi.org/10.1007/978-3-319-97304-3_70

[5] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). Attention is all you need. *arXiv Preprint: 1706.03762*. https://doi.org/10.48550/arXiv.1706.03762

[6] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguis-*

tics: Human Language Technologies, *1*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[7] Zhang, Q., Lu, J., & Jin, Y. (2021). Artificial intelligence in recommender systems. *Complex & Intelligent Systems*, *7*(1), 439–457. https://doi.org/10.1007/s40747-020-00212-w

[8] Jannach, D., Mobasher, B., & Berkovsky, S. (2020). Research directions in session-based and sequential recommendation: A preface to the special issue. *User Modeling and User-Adapted Interaction*, *30*, 609–616. https://doi.org/10.1007/s11257-020-09274-4

[9] Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., & Jiang, P. (2019). BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international Conference on Information and Knowledge Management*, 1441–1450. https://doi.org/10.1145/3357384.3357895

[10] Wang, S., Hu, L., Wang, Y., Cao, L., Sheng, Q. Z., & Orgun, M. (2019). Sequential recommender systems: Challenges, progress and prospects. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, *19*, 6332–6338. https://doi.org/10.24963/ijcai.2019/883

[11] Ludewig, M., & Jannach, D. (2018). Evaluation of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction*, *28*(4), 331–390. https://doi.org/10.1007/s11257-018-9209-6

[12] Fang, H., Zhang, D., Shu, Y., & Guo, G. (2020). Deep learning for sequential recommendation: Algorithms, influential factors, and evaluations. *ACM Transactions on Information Systems*, *39*(1), 1–42. https://doi.org/10.1145/3426723

[13] Singh, P. K., Pramanik, P. K. D., Dey, A. K., & Choudhury, P. (2021). Recommender systems: An overview, research trends, and future directions. *International Journal of Business and Systems Research*, *15*(1), 14–52. https://doi.org/10.1504/IJBSR.2021.111753

[14] Zheng, Y., & Wang, D. X. (2022). A survey of recommender systems with multi-objective optimization. *Neurocomputing*, *474*, 141–153. https://doi.org/10.1016/j.neucom.2021.11.041

[15] Song, W., Wang, S., Wang, Y., & Wang, S. (2021). Next-item recommendations in short sessions. In *Proceedings of the 15th ACM Conference on Recommender Systems*, 282–291. https://doi.org/10.1145/3460231.3474238

[16] Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2015). Session-based recommendations with recurrent neural networks. In *International Conference on Learning Representations 2016*. *arXiv preprint arXiv:1511.06939*. https://doi.org/10.48550/arXiv.1511.06939

[17] Sun, W., Liu, Z., Fan, X., Wen, J. R., & Zhao, W. X. (2023). Towards efficient and effective transformers for sequential recommendation. In *International conference on database systems for advanced applications*, 341–356. https://doi.org/10.1007/978-3-031-30672-3_23

[18] Gharahighehi, A., & Vens, C. (2021). Personalizing diversity versus accuracy in session-based recommender systems. *SN Computer Science*, *2*(1), 39. https://doi.org/10.1007/s42979-020-00399-2

[19] Tamm, Y. M., Damdinov, R., & Vasilev, A. (2021). Quality metrics in recommender systems: Do we calculate metrics consistently?. In *Proceedings of the 15th ACM Conference on Recommender Systems*, 708–713. https://doi.org/10.1145/3460231.3478848

[20] Dara, S., Chowdary, C. R., & Kumar, C. (2020). A survey on group recommender systems. *Journal of Intelligent Information Systems*, *54*(2), 271–295. https://doi.org/10.1007/s10844-018-0542-3

[21] Martinez, A. D., Del Ser, J., Villar-Rodriguez, E., Osaba, E., Poyatos, J., Tabik, S., ... & Herrera, F. (2021). Lights and shadows in evolutionary deep learning: Taxonomy, critical methodological analysis, cases of study, learned lessons, recommendations and challenges. *Information Fusion*, *67*, 161–194. https://doi.org/10.1016/j.inffus.2020.10.014

[22] Eberle, O., Büttner, J., Kräutli, F., Müller, K. R., Valleriani, M., & Montavon, G. (2020). Building and interpreting deep similarity models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(3), 1149–1161. https://doi.org/10.1109/TPAMI.2020.3020738

[23] Hiriyannaiah, S., Siddesh, G. M., & Srinivasa, K. G. (2022). Deep visual ensemble similarity (DVESM) approach for visually aware recommendation and search in smart community. *Journal of King Saud University-Computer and Information Sciences*, *34*(6), 2562-2573. https://doi.org/10.1016/j.jksuci.2020.03.009

[24] Nguyen, L. V., Vo, Q. T., & Nguyen, T. H. (2023). Adaptive KNN-based extended collaborative filtering recommendation services. *Big Data and Cognitive Computing*, *7*(2), 106. https://doi.org/10.3390/bdcc7020106

[25] Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., & Tan, T. (2019). Session-based recommendation with graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, *33*(1), 346–353. https://doi.org/10.1609/aaai.v33i01.3301346

[26] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv Preprint:1907.11692*. https://doi.org/10.48550/arXiv.1907.11692

[27] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, *517*, 5753-5763. https://dl.acm.org/doi/10.5555/3454287.3454804

[28] Kang, W. C., & McAuley, J. (2018). Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining*, 197–206. https://doi.org/10.1109/ICDM.2018.00035

[29] Mi, F., Lin, X., & Faltings, B. (2020). Ader: Adaptively distilled exemplar replay towards continual learning for session-based recommendation. In *Proceedings of the 14th ACM Conference on Recommender Systems*, 408–413. https://doi.org/10.1145/3383313.3412218

[30] Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 565–573. https://doi.org/10.1145/3159652.3159656

[31] Shafiee, S. (2024). Unveiling the latest trends and advancements in machine learning algorithms for recommender systems: A literature review. *Procedia CIRP*, *121*, 115–120. https://doi.org/10.1016/j.procir.2023.08.062

[32] Lu, Y., & Volkovs, M. (2023). Robust user engagement modeling with transformers and self supervision. In *Proceedings of the Recommender Systems Challenge 2023*, 23–27. https://doi.org/10.1145/3626221.3627285

[33] Zhang, G. (2023). User-centric conversational recommendation: Adapting the need of user with large language models. In *Proceedings of the 17th ACM Conference on Recommender Systems*, 1349–1354. https://doi.org/10.1145/3604915.3608885

[34] de Azambuja, R. X., Morais, A. J., & Filipe, V. (2023). X-wines: A wine dataset for recommender systems and machine learning. *Big*

Data and Cognitive Computing, *7*(1), 20. https://doi.org/10.3390/bdcc7010020

[35] Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., ... & Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data*, *3*(1), 1–9. https://doi.org/10.1038/sdata.2016.18

[36] de Azambuja, R. X., Morais, A. J., & Filipe, V. (2024). X-Model4Rec: An extensible recommender model based on the user's dynamic taste profile. *Human-Centric Intelligent Systems*, *4*(3), 344–362. https://doi.org/10.1007/s44230-024-00073-3

[37] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In International Conference on Machine Learning, In *International conference on machine learning*, *70*, 1321–1330. https://dl.acm.org/doi/10.5555/3305381.3305518

[38] Moradi, R., Berangi, R., & Minaei, B. (2020). A survey of regularization strategies for deep models. *Artificial Intelligence Review*, *53*(6), 3947–3986. https://doi.org/10.1007/s10462-019-09784-7

[39] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, *33*, 1877–1901. https://dl.acm.org/doi/abs/10.5555/3495724.3495883

[40] De Pauw, J. (2020). Exploratory methods for evaluating recommender systems. In *Proceedings of the 14th ACM Conference on Recommender Systems*, 782–786. https://doi.org/10.1145/3383313.3411456

[41] Zhou, C., Bai, J., Song, J., Liu, X., Zhao, Z., Chen, X., & Gao, J. (2018). Atrank: An attention-based user behavior modeling framework for recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, *32*(1). https://doi.org/10.1609/aaai.v32i1.11618

[42] Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015). An empirical exploration of recurrent network architectures. In *International conference on machine learning*, 2342–2350.

[43] Feng, J., Wang, K., Miao, Q., Xi, Y., & Xia, Z. (2023). Personalized recommendation with hybrid feedback by refining implicit data. *Expert Systems with Applications*, *232*, 120855. https://doi.org/10.1016/j.eswa.2023.120855

[44] Maheswari, M., Geetha, S., & Selva Kumar, S. (2019). Adaptable and proficient Hellinger coefficient based collaborative filtering for recommendation system. *Cluster Computing*, *22*(5), 12325–12338. https://doi.org/10.1007/s10586-017-1616-7

[45] de Azambuja, R. X. (2024). *Recomendação adaptativa em ambientes on-line*. PhD Thesis, University of Trás-os-Montes and Alto Douro.

How to Cite: de Azambuja, R. X., Morais, J. A., & Filipe, V. (2025). A Framework for Adaptive Recommendation in Online Environments. *Artificial Intelligence and Applications, 00*(00), 1-15. https://doi.org/10.47852/bonviewAIA52025020

Appendix

Name	Collaborative filtering	Collaborative filtering-SVDpp	Content-based filtering-Cosine	Content-based filtering-Euclidian	Content-based filtering-Haversine	Content-based filtering-Manhattan	Hybrid filtering	Nearest neighbors-Long	Nearest neighbors-Short	Simple-noninteract	Simple-noninteract-pop	Simple-random	X-Model4Rec
P@1	0.003585 (−549.93%)	0.004899 (−375.607%)	0.003585 (−549.93%)	0.003107 (−649.92%)	0.003585 (−549.93%)	0.003346 (−596.354%)	0.002868 (−712.413%)	0.023181 (−0.513%)	0.020194 (−15.381%)	0.000956 (−2,337.238%)	0.010157 (−129.398%)	0.000956 (−2,337.238%)	0.0233 (*)
R@1	0.003585 (−549.93%)	0.004899 (−375.607%)	0.003585 (−549.93%)	0.003107 (−649.92%)	0.003585 (−549.93%)	0.003346 (−596.354%)	0.002868 (−712.413%)	0.023181 (−0.513%)	0.020194 (−15.381%)	0.000956 (−2,337.238%)	0.010157 (−129.398%)	0.000956 (−2,337.238%)	0.0233 (*)
F1@1	0.003585 (−549.93%)	0.004899 (−375.607%)	0.003585 (−549.93%)	0.003107 (−649.92%)	0.003585 (−549.93%)	0.003346 (−596.354%)	0.002868 (−712.413%)	0.023181 (−0.513%)	0.020194 (−15.381%)	0.000956 (−2,337.238%)	0.010157 (−129.398%)	0.000956 (−2,337.238%)	0.0233 (*)
MAP@1	0.003585 (−549.93%)	0.004899 (−375.607%)	0.003585 (−549.93%)	0.003107 (−649.92%)	0.003585 (−549.93%)	0.003346 (−596.354%)	0.002868 (−712.413%)	0.023181 (−0.513%)	0.020194 (−15.381%)	0.000956 (−2,337.238%)	0.010157 (−129.398%)	0.000956 (−2,337.238%)	0.0233 (*)
MRR@1	0.003585 (−549.93%)	0.004899 (−375.607%)	0.003585 (−549.93%)	0.003107 (−649.92%)	0.003585 (−549.93%)	0.003346 (−596.354%)	0.002868 (−712.413%)	0.023181 (−0.513%)	0.020194 (−15.381%)	0.000956 (−2,337.238%)	0.010157 (−129.398%)	0.000956 (−2,337.238%)	0.0233 (*)
NDCG@1	0.003585 (−549.93%)	0.004899 (−375.607%)	0.003585 (−549.93%)	0.003107 (−649.92%)	0.003585 (−549.93%)	0.003346 (−596.354%)	0.002868 (−712.413%)	0.023181 (−0.513%)	0.020194 (−15.381%)	0.000956 (−2,337.238%)	0.010157 (−129.398%)	0.000956 (−2,337.238%)	0.0233 (*)
P@10	0.012463 (−34.51%)	0.001828 (−817.068%)	0.003417 (−390.606%)	0.003358 (−399.226%)	0.000705 (−2,277.872%)	0.003382 (−395.683%)	0.011853 (−41.433%)	0.014829 (−13.049%)	0.012953 (−29.422%)	0.00098 (−1,610.612%)	0.009177 (−82.674%)	0.001159 (−1,346.419%)	0.016764 (*)
R@10	0.124627 (−34.515%)	0.018282 (−816.978%)	0.034174 (−390.554%)	0.033576 (−399.291%)	0.00705 (−2,277.901%)	0.033815 (−395.762%)	0.118533 (−41.431%)	0.148285 (−13.054%)	0.129526 (−29.427%)	0.009798 (−1,610.982%)	0.091767 (−82.682%)	0.01159 (−1,346.437%)	0.167642 (*)
F1@10	0.022659 (−34.516%)	0.003324 (−816.968%)	0.006213 (−390.584%)	0.006105 (−399.263%)	0.001282 (−2,277.535%)	0.006148 (−395.771%)	0.021551 (−41.432%)	0.026961 (−13.052%)	0.02355 (−29.427%)	0.001781 (−1,611.398%)	0.016685 (−82.679%)	0.002107 (−1,346.607%)	0.03048 (*)
MAP@10	0.027732 (−109.311%)	0.007854 (−639.063%)	0.010222 (−467.854%)	0.010056 (−477.228%)	0.004285 (−1,254.632%)	0.010157 (−471.488%)	0.026735 (−117.116%)	0.052556 (−10.446%)	0.045069 (−28.794%)	0.00284 (−1,943.873%)	0.027561 (−110.609%)	0.003246 (−1,688.232%)	0.058046 (*)
MRR@10	0.027732 (−109.311%)	0.007854 (−639.063%)	0.010222 (−467.854%)	0.010056 (−477.228%)	0.004285 (−1,254.632%)	0.010157 (−471.488%)	0.026735 (−117.116%)	0.052556 (−10.446%)	0.045069 (−28.794%)	0.00284 (−1,943.873%)	0.027561 (−110.609%)	0.003246 (−1,688.232%)	0.058046 (*)
NDCG@10	0.102671 (−38.407%)	0.015665 (−807.143%)	0.028508 (−398.471%)	0.028233 (−403.326%)	0.006129 (−2,218.551%)	0.028406 (−400.261%)	0.097852 (−45.223%)	0.125488 (−13.241%)	0.109458 (−29.825%)	0.008178 (−1,637.638%)	0.076578 (−85.568%)	0.009676 (−1,368.623%)	0.142104 (*)
P@20	0.011441 (−22.035%)	0.001828 (−663.786%)	0.003495 (−299.485%)	0.003119 (−347.643%)	0.00058 (−2,307.241%)	0.003119 (−347.643%)	0.011196 (−24.705%)	0.012373 (−12.842%)	0.010425 (−33.928%)	0.001028 (−1,258.171%)	0.008764 (−59.311%)	0.001195 (−1,068.368%)	0.013962 (*)
R@20	0.228821 (−22.036%)	0.036564 (−663.716%)	0.069901 (−299.486%)	0.062373 (−347.702%)	0.01159 (−2,309.362%)	0.062373 (−347.702%)	0.223922 (−24.706%)	0.247461 (−12.844%)	0.208508 (−33.925%)	0.020552 (−1,258.724%)	0.17529 (−59.305%)	0.023898 (−1,068.487%)	0.279245 (*)
F1@20	0.021792 (−22.04%)	0.003482 (−663.785%)	0.006657 (−299.504%)	0.00594 (−347.727%)	0.001104 (−2,308.967%)	0.00594 (−347.727%)	0.021326 (−24.707%)	0.023568 (−12.844%)	0.019858 (−33.926%)	0.001957 (−1,258.968%)	0.016694 (−59.309%)	0.002276 (−1,068.497%)	0.026595 (*)
MAP@20	0.034828 (−88.38%)	0.009036 (−626.085%)	0.012575 (−421.742%)	0.011998 (−446.833%)	0.004587 (−1,330.325%)	0.012084 (−442.941%)	0.033932 (−93.354%)	0.059318 (−10.606%)	0.050416 (−30.135%)	0.003585 (−1,730.098%)	0.033287 (−97.101%)	0.004085 (−1,506.095%)	0.065609 (*)
MRR@20	0.034828 (−88.38%)	0.009036 (−626.085%)	0.012575 (−421.742%)	0.011998 (−446.833%)	0.004587 (−1,330.325%)	0.012084 (−442.941%)	0.033932 (−93.354%)	0.059318 (−10.606%)	0.050416 (−30.135%)	0.003585 (−1,730.098%)	0.033287 (−97.101%)	0.004085 (−1,506.095%)	0.065609 (*)
NDCG@20	0.18929 (−23.391%)	0.030259 (−671.889%)	0.057547 (−305.87%)	0.051759 (−351.257%)	0.009744 (−2,297.024%)	0.051768 (−351.178%)	0.185109 (−26.178%)	0.206968 (−12.851%)	0.174631 (−33.748%)	0.016971 (−1,276.265%)	0.145206 (−60.851%)	0.019722 (−1,084.292%)	0.233566 (*)

Note. (*) The best values. P=Precision, R=Recall, F1=F1-Score, MAP= Mean Average Precision, MRR = Mean Reciprocal Rank, NDCG = Normalized Discounted Cumulative Gain.

Name	Collaborative filtering	Collaborative filtering-SVDpp	Content-based filtering-Cosine	Content-based filtering-Euclidian	Content-based filtering-Haversine	Content-based filtering-Manhattan	Hybrid Filtering	Nearest neighbors-Long	Nearest neighbors-Short	Simple-noninteract	Simple-noninteract-pop	Simple-random	X-Model4Rec
P@30	0.010272 (−16.871%)	0.001976 (−507.54%)	0.003441 (−248.881%)	0.003011 (−298.705%)	0.000597 (−1,910.888%)	0.003043 (−294.512%)	0.010125 (−18.568%)	0.010646 (−12.765%)	0.009033 (−32.902%)	0.000972 (−1,135.082%)	0.007811 (−53.694%)	0.001155 (−939.394%)	0.012005 (*)
F1@30	0.019881 (−16.87%)	0.003824 (−507.61%)	0.006661 (−248.821%)	0.005828 (−298.679%)	0.001156 (−1,909.948%)	0.00589 (−294.482%)	0.019596 (−18.57%)	0.020606 (−12.758%)	0.017484 (−32.893%)	0.001881 (−1,135.247%)	0.015117 (−53.701%)	0.002236 (−939.132%)	0.023235 (*)
MAP@30	0.038033 (−81.019%)	0.009931 (−593.253%)	0.013912 (−394.875%)	0.013114 (−424.989%)	0.004841 (−1,322.165%)	0.013248 (−419.678%)	0.03712 (−85.471%)	0.062202 (−10.683%)	0.052908 (−30.126%)	0.003933 (−1,650.496%)	0.035623 (−93.266%)	0.004512 (−1,425.864%)	0.068847 (*)
MRR@30	0.038033 (−81.019%)	0.009931 (−593.253%)	0.013912 (−394.875%)	0.013114 (−424.989%)	0.004841 (−1,322.165%)	0.013248 (−419.678%)	0.03712 (−85.471%)	0.062202 (−10.683%)	0.052908 (−30.126%)	0.003933 (−1,650.496%)	0.035623 (−93.266%)	0.004512 (−1,425.864%)	0.068847 (*)
NDCG@30	0.256492 (−17.67%)	0.048815 (−518.283%)	0.08533 (−253.703%)	0.074946 (−302.71%)	0.014889 (−1,927.101%)	0.075727 (−298.557%)	0.252548 (−19.508%)	0.267666 (−12.758%)	0.22711 (−32.894%)	0.024181 (−1,148.149%)	0.19509 (−54.706%)	0.028683 (−952.243%)	0.301815 (*)
P@40	0.009117 (−16.738%)	0.001759 (−505.06%)	0.003295 (−223.005%)	0.002999 (−254.885%)	0.000612 (−1,639.052%)	0.003005 (−254.176%)	0.009147 (−16.355%)	0.009484 (−12.221%)	0.008098 (−31.428%)	0.00101 (−953.762%)	0.007325 (−45.297%)	0.001147 (−827.899%)	0.010643 (*)
R@40	0.364679 (−16.743%)	0.070379 (−504.922%)	0.131796 (−223.028%)	0.119967 (−254.879%)	0.024495 (−1,638.061%)	0.120206 (−254.174%)	0.365874 (−16.362%)	0.379376 (−12.221%)	0.323934 (−31.427%)	0.040387 (−954.146%)	0.292986 (−45.31%)	0.045884 (−827.857%)	0.425738 (*)
F1@40	0.017789 (−16.746%)	0.003433 (−504.952%)	0.006429 (−223.036%)	0.005852 (−254.887%)	0.001195 (−1,637.908%)	0.005864 (−254.161%)	0.017848 (−16.36%)	0.018506 (−12.223%)	0.015802 (−31.426%)	0.00197 (−954.213%)	0.014292 (−45.312%)	0.002238 (−827.971%)	0.020768 (*)
MAP@40	0.039652 (−78.354%)	0.010242 (−590.5%)	0.014722 (−380.376%)	0.013956 (−406.743%)	0.005033 (−1,305.146%)	0.014071 (−402.601%)	0.038908 (−81.765%)	0.063912 (−10.654%)	0.054424 (−29.945%)	0.004249 (−1,564.415%)	0.037306 (−89.57%)	0.004832 (−1,363.597%)	0.070721 (*)
MRR@40	0.039652 (−78.354%)	0.010242 (−590.5%)	0.014722 (−380.376%)	0.013956 (−406.743%)	0.005033 (−1,305.146%)	0.014071 (−402.601%)	0.038908 (−81.765%)	0.063912 (−10.654%)	0.054424 (−29.945%)	0.004249 (−1,564.415%)	0.037306 (−89.57%)	0.004832 (−1,363.597%)	0.070721 (*)
NDCG@40	0.305514 (−17.18%)	0.05859 (−511.029%)	0.109537 (−226.832%)	0.099694 (−259.101%)	0.020357 (−1,658.619%)	0.09993 (−258.253%)	0.306061 (−16.971%)	0.318909 (−12.258%)	0.272219 (−31.512%)	0.033477 (−969.397%)	0.244745 (−46.276%)	0.038093 (−839.81%)	0.358002 (*)
P@50	0.008281 (−16.761%)	0.00169 (−472.13%)	0.003284 (−194.428%)	0.002944 (−228.431%)	0.000629 (−1,437.202%)	0.002961 (−226.545%)	0.008276 (−16.832%)	0.008364 (−15.603%)	0.007384 (−30.945%)	0.001054 (−817.362%)	0.006933 (−39.463%)	0.001159 (−734.254%)	0.009669 (*)
R@50	0.414028 (−16.768%)	0.084478 (−472.28%)	0.164177 (−194.469%)	0.14721 (−228.409%)	0.031425 (−1,438.428%)	0.148046 (−226.555%)	0.413789 (−16.835%)	0.41821 (−15.6%)	0.36922 (−30.938%)	0.052694 (−817.469%)	0.346636 (−39.469%)	0.057952 (−734.227%)	0.483451 (*)
F1@50	0.016236 (−16.771%)	0.003313 (−472.261%)	0.006438 (−194.486%)	0.005773 (−228.408%)	0.001232 (−1,438.88%)	0.005806 (−226.542%)	0.016227 (−16.836%)	0.0164 (−15.604%)	0.014479 (−30.941%)	0.002066 (−817.667%)	0.013594 (−39.466%)	0.002273 (−734.096%)	0.018959 (*)
MAP@50	0.040742 (−76.722%)	0.010554 (−582.206%)	0.015439 (−366.351%)	0.014553 (−394.743%)	0.005185 (−1,288.621%)	0.014681 (−390.43%)	0.039965 (−80.158%)	0.064773 (−11.157%)	0.055433 (−29.887%)	0.00452 (−1,492.92%)	0.038491 (−87.057%)	0.005098 (−1,312.319%)	0.072 (*)
MRR@50	0.040742 (−76.722%)	0.010554 (−582.206%)	0.015439 (−366.351%)	0.014553 (−394.743%)	0.005185 (−1,288.621%)	0.014681 (−390.43%)	0.039965 (−80.158%)	0.064773 (−11.157%)	0.055433 (−29.887%)	0.00452 (−1,492.92%)	0.038491 (−87.057%)	0.005098 (−1,312.319%)	0.072 (*)
NDCG@50	0.348439 (−17.044%)	0.070614 (−477.543%)	0.136865 (−197.977%)	0.122775 (−232.173%)	0.026155 (−1,459.266%)	0.123479 (−230.28%)	0.347988 (−17.195%)	0.353333 (−15.423%)	0.311293 (−31.01%)	0.043766 (−831.833%)	0.290561 (−40.358%)	0.048247 (−745.288%)	0.407826 (*)

Note. (*) The best values. P=Precision, R=Recall, F1=F1-Score, MAP= Mean Average Precision, MRR = Mean Reciprocal Rank, NDCG = Normalized Discounted Cumulative Gain.