Journal of
Theoretical and Applied Electronic Commerce Research

MDPI

Article

Efficient Personalization in E-Commerce: Leveraging Universal Customer Representations with Embeddings

Miguel Alves Gomes [1,*], Philipp Meisen [2] and Tobias Meisen [1]

[1] Institute for Technologies and Management of Digital Transformation, University of Wuppertal, 42119 Wuppertal, Germany; meisen@uni-wuppertal.de
[2] Breinify Inc., San Francisco, CA 94105, USA; philipp.meisen@breinify.com
* Correspondence: alvesgomes@uni-wuppertal.de; Tel.: +49-202-439-1733

Abstract: E-commerce has grown into a billion-dollar industry in recent times with an ever-increasing number of individuals using it regularly. Thus, e-commerce companies can gather interaction data from their customers and analyze it to create focused and personalized marketing campaigns. For large companies, it is possible to tap into these data for personalization using deep learning-based methods that require enormous computing resources. Small companies, on the other hand, cannot afford this. Furthermore, this level of tailor-made addressing necessitates an accurate customer representation. Nevertheless, the exploration of universal representations applicable across various tasks has been limited despite the advantages they offer. We propose a universal customer representation learned only from customer interaction data. We demonstrate that self-supervised trained embeddings of the customer interaction context are a suitable universal customer representation for various e-commerce tasks. To demonstrate the effectiveness of our approach, we conducted experiments comparing four different state-of-the-art approaches and their capabilities in different prediction tasks. Not only do we show that our method outperforms others in most cases, but it also meets other important criteria for real-world applications. It is particularly important to emphasize that our approach does not require a significant amount of resources, and furthermore, is data protection compliant by not using personal information.

Keywords: e-commerce; customer behavior; universal customer representation; behavior prediction; embedding; neural networks

check for updates

Academic Editors: Jiaming Fang, Chao Wen and Benjamin George

Received: 24 October 2024
Revised: 13 January 2025
Accepted: 15 January 2025
Published: 16 January 2025

Citation: Alves Gomes, M.; Meisen, P.; Meisen, T. Efficient Personalization in E-Commerce: Leveraging Universal Customer Representations with Embeddings. *J. Theor. Appl. Electron. Commer. Res.* **2025**, *20*, 12. https://doi.org/10.3390/jtaer20010012

1. Introduction

The exponential growth in the use of the Internet has made e-commerce an indispensable facet of today's society. With the widespread adoption of portable devices, individuals have convenient access to the Internet, making online shopping and digital marketplaces an omnipresent force in the daily lives of consumers worldwide [1]. This unprecedented accessibility is facilitated by search engines, and recommendation systems in particular have become central components of e-commerce companies [2–5].

In this highly competitive and swiftly evolving environment, companies must make precise predictions about consumer behavior to stay ahead of the curve. Personalization offers new opportunities by enabling targeted, individualized interaction with customers through the extensive amount of collected data [6]. Achieving personalization requires the development of a robust customer representation capable of capturing intricate customer behavior patterns and performing predictive analyses. Traditionally, this endeavor has demanded substantial manual effort and domain expertise, often involving feature engineering following extensive data analysis processes [7–10]. Furthermore, this labor-intensive feature engineering must be replicated for various campaigns, as these customer representations are tailored to specific use cases. Consequently, there is a need for a more universal customer representation (UCR) [4,11], whereby 'universal' in this context means that it is applicable to multiple use cases and tasks without reengineering it for a specific use case or task. However, with the ongoing development of deep learning, a change in the methodological approaches of online retail is becoming apparent. Starting from transformer-based models like Bert4Rec [12] for product recommendations or deep-attention-based networks for predicting customer click behavior like DIEN [3], such deep end-to-end approaches show promising performance in dealing with specific tasks. However, these approaches may not be suitable for all companies, particularly smaller ones that lack the necessary expertise, resources, and data. They require simpler, easy-to-implement approaches that offer comparable performance and flexibility.

The following criteria for a UCR are derived from this argumentation for real-world applications with limited resources where various e-commerce tasks exist: (1) Performance comparable to state-of-the-art approaches; (2) Flexibility for the use case, data volume, and data type; (3) Data protection compliance, as with the rise of privacy laws and regulations, access to certain information is no longer feasible. This has led to the emergence of new baseline requirements that prohibit the use of personal customer information [13–15]; (4) Real-time capability is crucial for providing a personalized customer experience and engaging with customers using an appropriate marketing strategy, as noted by Esmeli et al. [10].

In our work, we propose a foundation representation for e-commerce customers based on self-supervised embeddings learned in the context of customer behavior as UCR and can be used as features for arbitrary learning models in order to predict customers' future behavior that meets all four aforementioned requirements. We benchmarked our approach against four other state-of-the-art approaches. For our use case, our approach outperforms the existing methods. To demonstrate the transferability and reproducibility of our approach, we subsequently apply it to three freely available and literature-renowned benchmark datasets. Also, in these use cases, our approach shows good performance. Furthermore, our experiments empirically show that embedding-based approaches represent customer behavior better than manually selected features by domain experts.

The remainder of the paper is structured as follows: In Section 2, we describe the related work regarding different e-commerce tasks and customer representation. In Section 3, we describe our use case in detail and the utilized datasets. Next, the used methodology for the proposed embedding approach is explained in detail in Section 4. The conducted experiments are described in Section 5. Finally, in Section 6 the results of our experiments are presented and critically discussed. Furthermore, we describe our conducted ablation study in Section 7. We summarize our research in Section 8 and give an outlook of potential future directions.

2. Related Work

2.1. Universal Customer Representation Approaches

In related work, the focus is often on generating predictive models for specific e-commerce tasks, rather than learning a universal customer representation. As a result, only a few works address UCR generation using only activity data. In 2018, Ni et al. [4] proposed the Deep User Perception Network (DUPN), an end-to-end Long-Short Term Memory (LSTM) with an embedding input that is trained on multiple tasks. In addition to the work of Ni et al., which comes closest to our perspective, other works enrich their data with text data such as reviews or product descriptions in order to generate customer representations. For example, Gu et al. [16] propose the Self-Supervised User Modeling Network (SUMN) that embeds text (e.g., review, search terms) of customers to model their behavior. A similar approach is proposed by Yang et al. [11]. They present the Lifelong User Representation Model (LURM) which represents customer behavior based on textual features like written reviews, product names, and product categories over all activities in their history. Wu et al. [17] proposed a pre-trained user model (PTUM) that learns customer behavior from textual data which is inspired by pre-trained large language models. Recently, Bertrand et al. [18] introduced an autoencoder-based framework for learning universal customer embeddings from tabular data and using that tabular-embedding for different downstream tasks, such as recommendation or forecasting. However, tabular data and its mix of feature types often requires heavy preprocessing.

2.2. Task-Specific Customer Representation Approaches

Instead of focusing on universal customer representation, a larger number of related works focus on task-specific customer representation approaches. Three e-commerce tasks in particular are addressed more frequently: purchase prediction, churn prediction, and click-through rate (CTR) prediction.

Purchase prediction is the prediction of the customer's intention to purchase. In the literature, the features are typically extracted from clickstream data from historical records, which are mostly from known customers and their demographic information [9,19–24]. A shift in research methodology can be identified around 2018. Approaches started to focus on real-time capability and rely less on customers' demographic information. Esmeli et al. [10] propose an early purchase prediction framework that leverages extracted session-related information from ongoing customer sessions. Sudirman et al. [25] use a particle swarm optimizer to improve the accuracy of a decision tree for a purchase prediction use case. Sheil et al. [26] propose an end-to-end three-layered LSTM with an embedding layer to represent and predict customers' purchase intention at the same time in which different features are encoded into the embedding. Alves Gomes et al. [27] propose a similar but simpler approach that only uses customer interaction as input.

Another task is to predict customer churn or return. It has been shown that the cost of acquiring new customers is five times higher than the cost of retaining existing customers [28]. Here, the approaches found in the literature mainly focus on manual feature engineering [29–34]. Rachid et al. [29] and Berger et al. [30] use a combination of transactional and behavioral/usage features such as number of sessions, session length, and conversion rate over all historical sessions. Friedrich et al. [31] and Perisic et al. [32] extracted RFM-related features [35] to represent the customer. Xiahou and Harada [33] extracted features based on customer activity and the time of the day.

Recently, CTR prediction has received a lot of attention in industry and academia alike. In the literature, the task is to determine if an item, e.g., a retrieved item in a search, recommended product, or ad, is clicked by the customer. For example, Gulhane et al. [36] and Huang et al. [37] propose an approach to predict the CTR for displayed ads. Other authors propose a CTR prediction model to optimize the retrieved items of a search engine [2,38,39]. In contrast to purchase and churn prediction, CTR prediction is typically approached through deep end-to-end approaches where customer representation and task solving are learned by the model as a whole. All learning models consist of an input embedding layer to encode different information from the data [3,39–48]. DIN [42], DIEN [3], TIEN [43], and MARN [44] are approaches that predict CTR for ads or products given a sequence of customer behavior. Lu et al. [49] propose a highly tailored hybrid model combining LightGBM, DeepFM, and DIN leveraging feature engineering, clustering, and temporal feature extraction for CTR prediction. Geng et al. [50] utilize a large language model (LLM) to predict click behavior based on textual customer behavior descriptions. To enhance the efficiency of the LLM prediction, the authors propose an Aggregated Hierarchical Encoding.

3. Use Case and Datasets

3.1. Use Case

In today's e-commerce environment, the offering must be highly personalized to retain customers. This leads to the need of various amounts of highly adaptable use cases and campaigns, covering the different needs for each customer. However, the increasing complexity of personalization efforts is paralleled by growing concerns and stringent regulations surrounding data privacy. As a result, e-commerce companies face the dual challenge of delivering highly personalized experiences while adhering to both local and international data protection laws. This necessitates several requirements for a company, which play a crucial role in the overall system performance, user experience, and regulatory compliance. One such requirement is the assurance of real-time capability, which is imperative for e-commerce businesses seeking to offer personalized customer experiences in real-time, thus maintaining engagement and enhancing conversion rates [10]. Another aspect is the efficient use of resources. E-commerce platforms often operate with limited resources, including computational power, memory, and bandwidth. Consequently, systems must be designed to accommodate varying volumes and types of data while ensuring scalability and efficiency across diverse use cases. This is particularly crucial as customer interactions can vary significantly based on factors such as traffic volume or transaction types. In addition, companies are obligated to ensure data privacy and compliance with regulations specific to various countries [13–15,51]. Consequently, companies are prohibited from gathering personal data or retaining it for extended periods. Given the potential for customers to browse anonymously without logging in, it becomes increasingly challenging to track their behavior without violating privacy laws. Consequently, the development of customer behavior models and personalized strategies must judiciously balance personalization with strict privacy regulations, ensuring that sensitive data are handled appropriately.

Task-specific solutions considering all the aforementioned requirements are therefore no longer economically suitable and manageable, underlining the need for more universal solutions. In our use case, we need to directly address this challenge by developing a solution that enhances customer interactions on online platforms without compromising data privacy. The goal is to improve the customer experience and increase companies' return on investment by precisely predicting customer intent in a privacy-compliant manner. Therefore, we are looking for a UCR that fits all the constraints a company can face.

Figure 1 illustrates the approached use case in this work on a high level. Each interaction is processed by the behavior prediction system, which returns a probability of predefined e-commerce prediction tasks. In consideration of the possibility that customers may choose to remain anonymous, that is to say, not currently logged in, the system's architecture has been designed to rely exclusively on information derived from the ongoing session. This approach not only enhances the system's variability for both known and unknown users but also ensures adherence to data protection regulations by eliminating the need for prolonged data storage. Furthermore, as stated before, the system needs to be real-time capable. According to Miller [52] and Card et al. [53], real-time performance is achieved in under 0.1 s. The prediction of the behavior prediction system is then used to make marketing decisions, such as recommendations, advertisements, etc.

In this work, we will focus on the behavior prediction system and its necessary components, which include the UCR and the learning model to predict future customer intention.

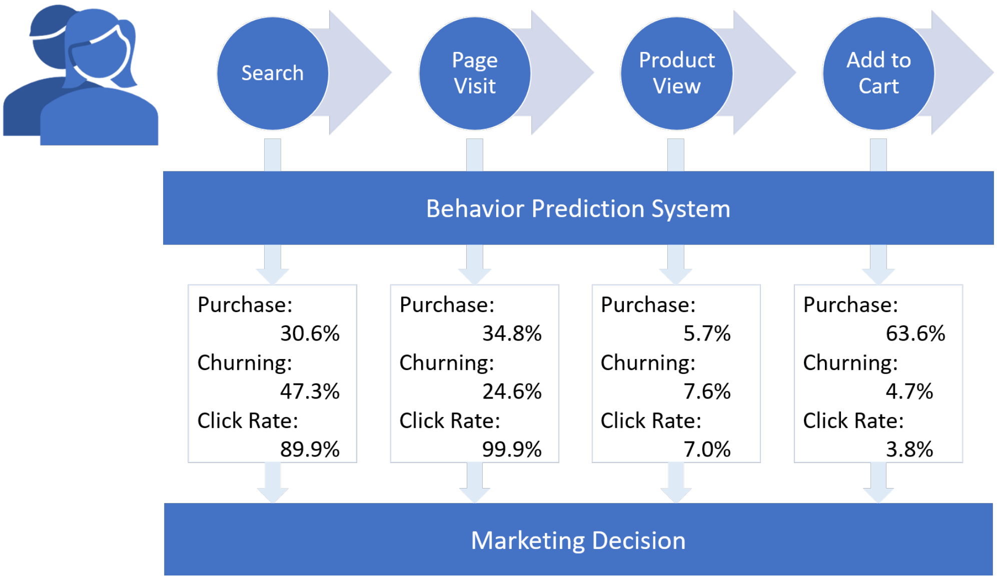

Figure 1. The website's behavior prediction system processes each customer interaction and calculates the probability for predefined prediction tasks, e.g, purchase, churn, and CTR prediction, after each interaction. Subsequently, a customized marketing decision can be derived for the customer in real-time.

3.2. Datasets

For the experiments, we utilized an industrialproprietary dataset collected from a consumer packaged goods retailer situated in the United States between January 2020 and May 2020. The dataset consists of 53.8 million customer events, encompassing 6.2 million sessions. The dataset includes timestamps, session IDs, and various event types (e.g., page view, product view, add-to-cart, remove-from-cart, purchase, and recommendation click), as well as the webpage URL and the customer ID if logged in.

In addition, to showcase the transferability of our methodology and achieve replicability, we conducted experiments on three publicly available datasets containing customer interaction records.

The YooChoose dataset was introduced in 2015 for the RecSys Challenge (https://recsys.acm.org/recsys15/challenge/ accessed on 1 October 2024) and is often used as a benchmark dataset for purchase prediction [8,10,19,20,22,27,54]. The dataset contains information on click and purchase events of a European online retailer including information on the session in which a product was purchased, as well as its price and quantity. It comprises records of approximately 9.2 million sessions with 33 million customer interactions. The records were collected over six months from April 2014 to September 2014. Each event includes details on the session ID, event time, product, and product category. A customer ID is not provided. Moreover, the dataset exclusively comprises view and purchase events.

The RetailRocket dataset serves primarily as a benchmark dataset for churn prediction [30,31]. However, it can also be utilized in certain purchase prediction use cases [26] as well as in CTR prediction [55]. The data were obtained from an e-commerce website and were collected over a period of 4.5 months between May and September of 2015, comprising 2.75 million events from 1.4 million customers. The events consist of five attributes: visitor ID, item ID, timestamp, and event type (view, add-to-cart, and transaction). Furthermore, the dataset provides supplementary details, such as item prices and categories. No session information is included.

The OpenCDP dataset was utilized in the 2020 RecSys tutorial (https://recsys.acm.org/recsys20/tutorials/ accessed on 1 October 2024) and is provided by the REES46 marketing platform (https://rees46.com/de accessed on 1 October 2024). This dataset contains data on customer behavior from a large online store offering multiple categories of products. The data cover the period from October 2019 to April 2020. Each entry in the dataset corresponds to a specific customer event and comprises nine distinct attributes which are session ID, customer ID, event time, event type (view, add-to-cart, or purchase), item ID, product category ID, product brand, product price, and product category brand. It further comprises over 411 million events stemming from approximately 89 million sessions. Not only does this dataset provide rich information but it is also considerably larger than other datasets, qualifying it to test the scalability of the presented approach.

4. Methodology

Unlike the majority of current learning approaches, we opted for a two-stage process instead of an end-to-end approach, as illustrated in Figure 2. In the first step, we generated self-supervised, pre-trained embeddings as the UCR, which serves as a comprehensive profile of the customer, constructed exclusively from interaction data and is then stored. It is hypothesized that the underlying intention of the customer is encoded in their behavior, and therefore, customers with similar behavior have likely a similar intention. This hypothesis guides the subsequent step, wherein a task predictor learns patterns in the customer behavior representation and makes predictions about their future behavior. In the second step, the pre-trained embeddings are utilized as input in a learning model trained for a specific e-commerce task. This modular approach allows for the creation of different predictors tailored to various types of behavior, such as purchase likelihood, churn probability, and CTR prediction using only one customer representation. Each predictor is trained independently, leveraging the UCR to maximize prediction accuracy while maintaining strict adherence to data privacy regulations. Furthermore, this strategy was chosen to circumvent a typical learning behavior found in end-to-end approaches where a task-specific pattern learning is observed. Thereby, extraneous information about the customer is disregarded which is often used in other state-of-the art end-to-end approaches. An alternative approach involves pre-training a model with a general task, such as predicting the subsequent activity in a sequence. However, this method entails pre-training a model with a typically substantial number of parameters, followed by re-training it for the specific target tasks. In this instance, it is more efficient, in terms of both parameter count and training effort, to acquire a suitable UCR and use it in a task-based model. Here, the use of self-supervised pre-trained embeddings becomes crucial. As mentioned before, we want to capture the customer intention by using their activities. Therefore, the context and similarities need to be encoded in the UCR. In the latent vector space, similarities can be expressed through embeddings.

A decade ago, Mikolov et al. [56] demonstrated that the similarity between two words (w_i, w_j) in a given vector space is connected to their respective contexts $(C(w_i), C(w_j))$ in which the context is defined by the surrounding words. With our approach, we will show that this analogy also applies to sequences of customer interactions. For this, we use the skip-gram approach which is shown in Figure 3. Specifically, given a customer activity a_j with $j \in \mathbb{N}$, we describe the context $C(a_j)$ with length $2 \times m$ with $m \in \mathbb{N}$ of this activity a_j. The context $C(a_j)$ is described by the activities $\{a_{j-1}, a_{j-2}, \ldots, a_{j-m}\}$ that happened before activities $\{a_{j+1}, a_{j+2}, \ldots, a_{j+m}\}$ and that happened after activity a_j of the same customer. The goal is to learn the embedding representation e_a for all possible customer activities $a_j \in \{a_0, a_1, \ldots, a_n\}$ with $j, n \in \mathbb{N}, 0 \leq j \leq n$ using a single-layer neural network with trainable parameters θ that maximizes the likelihood of the context $C(a_j)$. This yields the objective function,

$$L(\theta) = \prod_{j=0}^{n} \prod_{\alpha=-m;\alpha\neq 0}^{m} P(a_{j+\alpha}|a_j;\theta). \quad (1)$$

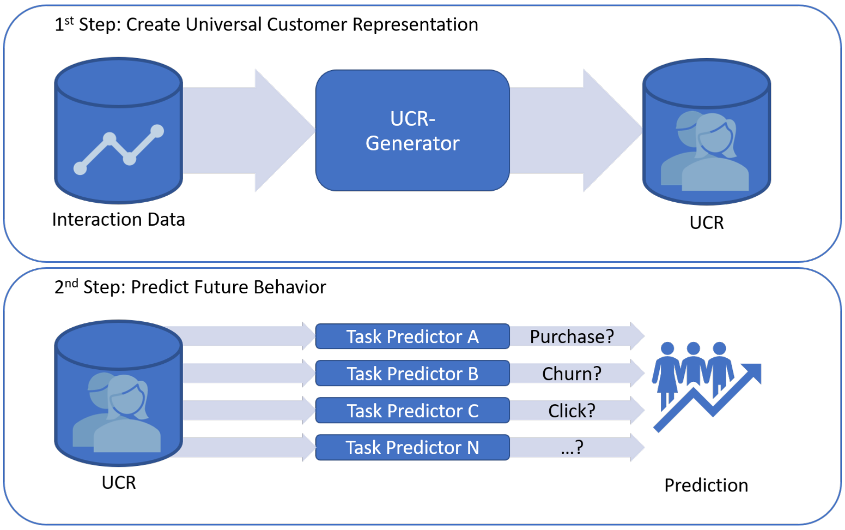

Figure 2. Decoupled, two-stage approach for predicting customer behavior based on historical interaction data: In the first stage, universal customer representations are learned, which are used as input in the second stage to train task-specific prediction models.

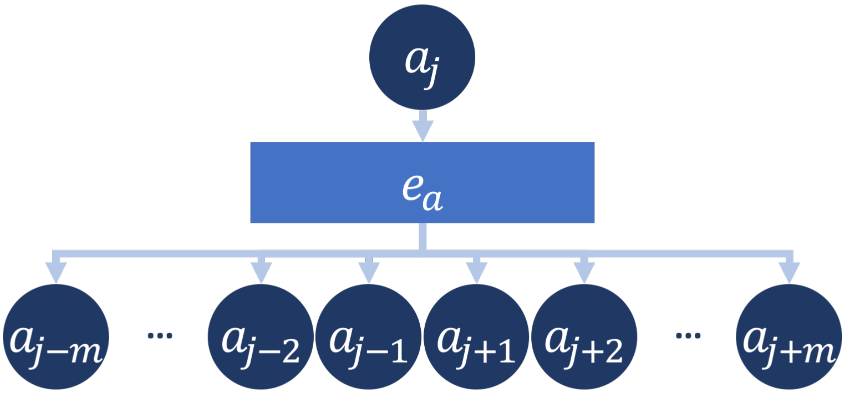

Figure 3. The architecture of our embedding approach. The customer activity a_j is the input to the embedding layer e_a, which then predicts its context $\{a_{j-m}, a_{j-m+1}, \ldots, a_{j+m-1}, a_{j+m}\}/a_j$.

In the second stage of our approach, we require a learning model capable of processing sequences of customer activities. Recurrent Neural Networks (RNNs) are a class of neural networks designed to process sequential data by maintaining a hidden state that evolves over time. However, RNNs suffer from limitations in capturing long-range dependencies due to challenges such as vanishing gradients. LSTMs are well suited for this purpose. They address vanishing gradients by introducing gating mechanisms that allow them to effectively capture dependencies. Therefore, LSTMs use a memory cell *c_t* that can hold information for long periods of time and is controlled by three gates, namely, input, forget, and output, that regulate the flow of information. The forget gate learns what information to discard from the memory cell. The input gate learns what new information to add to the memory cell, and the output gate generates the new hidden state based on the memory cell. The gating mechanisms allow LSTMs to effectively learn long-term dependencies, making them suitable for sequential data [57]. It is important to note that we conducted extensive experiments to identify the best-suited learning model for our tasks, which we will discuss in detail in Section 7. The objective for the learning model is to predict $P(x_i|s_i)$, where s_i is a customer activity sequence with $s_i = \{a_{i_0}, a_{i_1}, \ldots, a_{i_n}\}$ of length $n \in \mathbb{N}$ and x_i is a potential future customer behavior, such as a purchase, click or churn.

5. Experimental Setup

After outlining our embedding approach, we will provide a comprehensive description of the experiments. This is done to ensure reproducibility. Figure 4 depicts the process employed for our approach, commencing with data preparation. Thereafter, a customer representation is derived through the learning of the UCR embedding representation. Thereafter, the model is trained for the given task. Finally, the approach's performance is evaluated. A similar pipeline is used for the tested baseline approaches, which is described in detail at the end of this section.

All approaches were implemented with Python 3.10 [58] using NumPy [59] and Pandas [60] for data preprocessing and PyTorch [61] as the deep learning framework as well as Sci-Kit Learn [62,63] for machine learning tasks and evaluation. Data preprocessing and evaluation were completed on a commercial machine equipped with an Intel i9-10885H CPU and 64GB RAM, operating on Windows 10. The training of the model and hyperparameter search were executed on an x86 architecture with GNU/Linux (Kernel Version 6.5.3) equipped with 96 Intel Xeon Platinum 8168 CPUs @ 2.70 GHz, 756 GB RAM, and eight Nvidia Tesla V100 GPUs. For each approach, ours and the baselines, we performed a systematic hyperparameter search using the Optuna library [64]. It uses a tree-structured Parzen estimator to model the hyperparameter space probabilistically. Optuna adaptively balances exploration and exploitation by suggesting parameter configurations based on past evaluations, ensuring an efficient search for optimal hyperparameters. We performed 10-fold cross-validation to tune the hyperparameters, evaluating up to 100 trials.

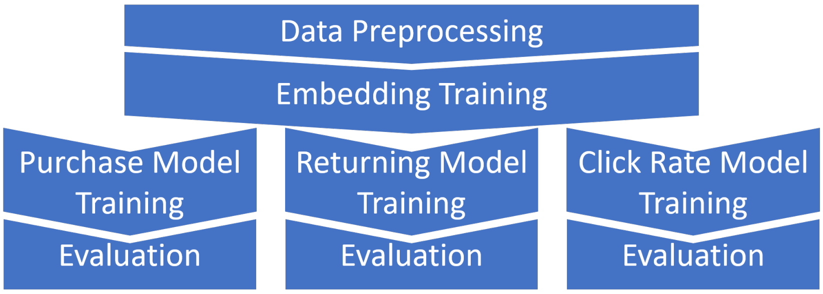

Figure 4. The experimental pipeline conducted in this study to evaluate our approach.

5.1. Data Preprocessing

For data preprocessing, we standardized the structure of all four datasets, eliminated anomalies, and ensured that each dataset contained events with session IDs, interaction objects, and timestamps. A session is defined as a sequence of activities performed by a single customer. The industrial datasets YooChoose and OpenCDP provide this information. However, the RetailRocket dataset does not have any such information. Therefore, we extracted sessions by identifying activities that lasted more than a day for the same customer. Our argument is that a session lasting more than a day provides a more comprehensive view of customer intent, as many purchase decisions involve multiple visits spread over hours within the same day. Furthermore, a short window risks arbitrarily dividing a continuous customer journey into multiple sessions if the customer pauses. Anomalies were eliminated in the subsequent steps. All sessions with less than three interactions or more interactions than the 99.5th percentile were excluded. In a real-world setting, it is not practical to make predictions based on only one or two customer interactions, or on sessions with over one hundred interactions. Regarding our industrial dataset, we used the

URLs rather than product IDs as interactions. Therefore, we excluded user-specific query parameters in the URLs to avoid over-specification. This helps to reduce the number of unique URLs and facilitates the generalization of the trained representation. It is important to note that, in order to demonstrate the effectiveness of customer behavior prediction without relying on personal customer information, we only use customer data to generate ground-truth labels. These data are not used to create the UCR or train the model. To ensure this, we remove all customer information, including customer IDs, from the dataset.

A session is labeled as a purchase if a purchase event exists in that session. There are a variety of ways to generate churn labels. For instance, examining the most recent month of data to identify customer activity, as referenced in [30,31], and categorizing these sessions as either "not churning" or employing more rigorous criteria necessitates the validation of transaction occurrences [65–67]. We consider a session as churn if there is no follow-up session. This method reduces dependency on the previous month's time period and eliminates the necessity of labeling the entire last month as churn and disregarding it. As the customer ID is unavailable in the YooChoose dataset, determining churn in that dataset is not possible. Similarly, to churn we have to assign click labels to the dataset for CTR prediction. For our own industrial dataset, we have recommendation click events that show if a recommendation is clicked by the customer which we use to label a session. This is not given for the other datasets. Therefore, we adhered to the commonly used approach to generate click labels in the literature. The raw dataset has 100% positive click labels. To obtain negative labels, 50% of the sessions have a randomly chosen item added which was not clicked by the customer before [3,4,42,43,68,69].

The final step is to divide the dataset into training and testing sets. To be as close to the real world as possible and to minimize feature leakage, we designated the last month of all datasets as the test set, and therefore, to be used exclusively for evaluating the approaches. Table 1 summarizes the statistical information of all four datasets after preprocessing.

Table 1. Statistics of the four datasets used in our research after preprocessing and splitting into training and test sets.

Dataset		Num. Sessions	Purchase	Churn	Click
YooChoose	train	3,731,708	310,851 (8.3%)	✗	1,865,854 (50%)
	test	676,286	59,675 (8.8%)	✗	338,143 (50%)
RetailRocket	train	165,842	7818 (4.7%)	149,810 (90.3%)	82,921 (50%)
	test	19,741	896 (4.5%)	18,980 (96.1%)	9870 (50%)
OpenCDP	train	32,930,752	3,738,748 (11.3%)	5,829,219 (17.7%)	16,465,376 (50%)
	test	6,121,525	674,628 (11%)	2,604,572 (42.5%)	3,060,762 (50%)
industrial	train	1,288,795	82,710 (6.4%)	1,212,607 (94.1%)	17,632 (1.3%)
	test	119,513	7594 (6.4%)	117,119 (97.8%)	1447 (1.4%)

5.2. Embedding Training

To train our embedding-based UCR as described in Section 4, we generate context data from the training data. Customer interaction is defined by the concatenated event type and URL/product ID as "event_type:URL" or "event_type:product_ID". For the YooChoose dataset which solely comprises view events, the customer activity is restricted to the item ID. Subsequently, trigrams are created consisting of the current interaction a_i and its preceding a_{i-1} and subsequent interaction a_{i+1} that results in a triple (a_i, a_{i+1}, a_{i-1}). To address the initial and final customer activity of a sequence, we introduce "start" and "end". The e-commerce domain is constantly changing with new products or promotions being added frequently, hence the need for the representation to accommodate new and therefore unknown customer activities. It is known as the "Out of Vocabulary" problem in natural language processing. To address this issue, we introduce an unknown token and randomly replace activities of the trigrams in the training process, with a probability equal to the ratio between known and unknown activities which is commonly used in the literature [27,70]. Table 2 presents the statistics of the used context data.

Table 2. Statistics for the training datasets utilized to create a customer representation through embedding training.

Dataset	Num. Trigrams	Num. Activities	Unknown Activities
YooChoose	1,719,739	27,016	2511
RetailRocket	750,056	119,702	8071
OpenCDP	103,881,685	509,286	69,770
industrial	2,812,129	66,205	2771

For the embedding training process, we utilized the cross-entropy loss and Adam optimizer algorithm [71] to predict the context of the activity. As mentioned before, we searched the optimal hyperparameters including the embedding dimension, batch size, and learning rate with the objective of minimizing the training loss. For the embedding dimension, the search of the search space $2^n, 2 \leq n \leq 10$ results in an embedding dimension of 32 for each use case except the OpenCDP dataset in which an embedding dimension of 64 is used. We train the embeddings with a batch size of 1024 and a learning rate of 1.921×10^{-3}. The embeddings are trained for 100 epochs. In order to deal with the high number of distinct activities in the OpenCDP dataset, we used the Adaptive Log Softmax With Loss [72] to speed up the embedding training.

5.3. Model Training

For each task and dataset, we trained a task-specific LSTM that receives the embedded UCR extracted from the sessions as input. The three selected tasks, purchase, churn, and CTR, are all binary classification problems, which allows for utilizing the Binary Cross Entropy With Logit Loss function and Adam as the optimizer. We implemented a consistent architecture across all tasks and use cases while adjusting the hyperparameters. We used the Optuna framework to conduct a hyperparameter search with 10-fold cross-validation maximizing the F1-score. As a result, we implemented a single-layer LSTM with a hidden size equal to that of the embedding size.

As shown in Table 1, our datasets were notably imbalanced for all tasks, except for click labels in the three publicly available datasets. Hence, we utilized three different sampling strategies, which resulted in training each learning model three times for every prediction task. We deployed SMOTE [73] for oversampling, utilized a random undersampler, and did not implement any sampling at all. The outcomes were consistent across all use cases and tasks: undersampling yielded the most favorable results overall. Consequently, we will concentrate solely on analyzing the outcomes acquired through undersampling in subsequent studies.

5.4. Evaluation

The final stage of our experiments involves evaluating the tested approaches. We utilize a variety of metrics to measure if all four aforementioned requirements are fulfilled.

We ran ten training runs, each with random initialization, to eliminate any favorable starting conditions that might affect our results. The reported score metrics represent the average of these ten runs. To evaluate the requirement performance, we use the F1-score and the Area Under the Curve (AUC) score. For the requirement flexibility, we take into account if it is possible to use the approach on all datasets and solve all tasks. To ensure the data protection compliance requirement, we verify that the approach can be applied without personal customer information and for known and unknown customers alike. Finally, for the real-time requirement, we measure how many customer representations and predictions can be completed within 0.1 s without taking into account any parallelization.

5.5. Baseline Approaches

We performed a comparative analysis with alternative baseline approaches for each task and assessed them alongside strategies that generate universal customer models. As a result of our assessment, we carried out preliminary experiments to determine the most effective baseline models for each task. While performance was the primary measure, other essential factors were also taken into consideration, such as the ability of an approach to represent a customer solely based on their activities.

Purchase Baseline: The approach of Esmeli et al. [10] serves as the baseline for purchase prediction because it does not rely on historical customer information and outperforms other approaches like the ones named in Section 2 on the YooChoose dataset. The authors represent customers using twelve features observed during the ongoing session and input the resulting representation into various machine learning classifiers, such as Decision Trees, RF, or Naive Bayes. To solve the prediction tasks, we employed the RF Classifier, which outperformed other models, such as Decision Trees, Naive Bayes, Support Vector Machine, Gradient Boosting, and Multi-Layer Perceptrons.

Churn Baseline: The baseline approach for churn prediction is the approach proposed by Berger et al. [30]. Their customer representation relies on 26 features obtained from all available customer data. However, not all features were feasible to utilize because some rely on historical customer information, which we do not use. For example, features related to purchasing in ongoing sessions (the authors in their research referred to ongoing sessions as the last session since their approach was not applied in real-time) are not available in a real-world scenario. After adjustment, we utilize 18 proposed features comprised of seven session-based features, four purchase-based features extracted from the entire history of customer interactions, three behavior change features, and four application-based interaction features. Similar to the purchase baseline, the RF Classifier is the best-performing model for each task.

CTR Baseline: According to the literature, CTR prediction involves several deep learning-based methods. We implemented these methods using the Deep-CTR library [74], which provides various models for CTR prediction. We evaluate several models on the Amazon Review dataset [75] which is used to benchmark CTR prediction and based on the performance the used baseline approach is the Deep Interest Network (DIN) [42], an end-to-end model containing multiple neural network layers. Its unique component is the activation unit responsible for processing the user behavior sequence. For that approach, the authors categorize the extracted features from the data into user profile features, user behavior features, ad features, and context features. None of the datasets contain personal customer information and therefore we only use behavior features for our experiments, which encompass activity, event type, and product category if the activity relates to an interaction with an item.

UCR Baseline: As the baseline for UCR, we chose the Deep User Perception Network (DUPN) [4]. It employs an end-to-end LSTM with multiple embedding inputs and attention.

Additionally, it obtains customer-related features alongside the sequence information, which we do not incorporate in our experiments as it is not available in our datasets. The remaining components are utilized as previously described. As input, we use activity-property pairs that consist of the activity, event type, timestamp, and category.

6. Results

Table 3 presents the resulting AUC and F1-scores of our experiments. The purchase baseline approach achieves the highest AUC score in predicting customer purchase intent on the YooChoose dataset, the same dataset used in the initial research paper. Similarly, we observed that the churn baseline achieves impressive churn prediction results on the RetailRocket dataset, which serves as the benchmark dataset in its initial research paper. However, the performance of both approaches on other use cases or tasks is only adequate and lags behind the performance of the other approaches investigated. These approaches were not designed for handling UCR and we argue that the use of feature engineering approaches in research presents a widespread issue of over-engineering customer representation, resulting in the approach outperforming the state-of-the-art on the benchmark at the cost of added manual effort and data analysis for every new task and use case. This presents a significant concern in real-world applications. Furthermore, the limited applicability of such approaches is another constraint. The three public datasets demonstrate the challenge of predicting CTRs objectively. Without significant inclusion of features, determining whether a customer will click on the shown product remains difficult.

Table 3. Performance of each tested approach. Bold blue markings indicate the best F1 and AUC scores for each task. Note, that the reported values are the average of ten different initializations and the standard deviation is smaller than 10^{-3}.

		YooChoose		RetailRocket			OpenCDP			Industrial		
Approach	**Score**	**Purchase**	**Click**	**Purchase**	**Churn**	**Click**	**Purchase**	**Churn**	**Click**	**Purchase**	**Churn**	**Click**
Purchase Baseline	F1	0.657	✗	0.585	0.690	✗	0.779	0.556	✗	0.632	0.585	0.446
	AUC	**0.690**	✗	0.578	0.772	✗	0.836	0.508	✗	0.679	0.551	0.527
Churn Baseline	F1	0.597	✗	0.643	0.902	✗	0.522	0.565	✗	0.491	0.662	0.469
	AUC	0.627	✗	0.629	**0.928**	✗	0.563	0.563	✗	0.596	0.629	0.498
CTR Baseline	F1	0.612	0.733	0.663	0.575	**0.661**	0.874	0.528	**0.891**	0.748	0.625	0.933
	AUC	0.609	0.781	**0.724**	0.592	**0.663**	**0.923**	0.488	**0.951**	0.805	0.659	0.963
UCR Baseline	F1	0.671	0.660	0.629	**0.980**	0.357	0.867	0.567	0.021	0.748	0.987	0.934
	AUC	0.679	0.705	0.637	0.667	0.496	0.916	0.537	0.499	0.807	0.713	0.971
Our	F1	**0.676**	**0.777**	**0.674**	**0.980**	0.660	**0.875**	**0.568**	0.612	**0.766**	**0.989**	**0.945**
	AUC	0.683	**0.890**	0.707	0.706	0.659	0.922	**0.567**	0.549	**0.821**	**0.731**	**0.981**

The three approaches for learning a customer representation from data can be effectively implemented in all tested use cases and tasks. Additionally, the approaches showed minimal difficulty in transferring to other use cases and tasks, requiring less effort than the baseline approaches used for purchase and churn prediction.

The CTR baseline shows exceptional F1 and AUC performance for CTR prediction on both the RetailRocket and OpenCDP datasets. In addition, it also delivers exceptionally positive purchase prediction performance on the OpenCDP dataset. The CTR baseline demonstrates robust performance in predicting both CTR and purchases. Nonetheless, it persistently underperforms in the churn prediction task.

The UCR baseline persistently ranks among the top three approaches in almost all experiments. Nevertheless, it underperforms in CTR prediction for the RetailRocket and OpenCDP datasets. However, it excels in our industrial use case, outperforming the CTR baseline.

Our approach consistently outperforms other tested approaches across the tested use cases and tasks, as evidenced by its high F1 and AUC scores. Even in situations where our approach falls short of the top performer, it consistently ranks above the UCR baseline and secures the second-place position. The experiments conducted on our dataset provide further evidence of the superior performance of our approach in comparison to the other approaches in which all other approaches are outperformed by our approach in both the F1 and AUC metrics. Based on the results, we assume that self-supervised pre-training in the context of our approach proves beneficial in obtaining a UCR because we infer similarities between activities rather than learning task-dependent features from the data. Furthermore, our approach prides itself on its simplicity, resulting in a reduced number of parameters and less overfitting, which will be further analyzed and discussed in the next Section 7. Additionally, we hypothesize that our approach's superior performance in each task in our use case results from URLs as activities, which allows for a more nuanced understanding of customer behavior than is available in the other three datasets, which solely provide information on mere product ids. This results in more precise customer representations, enabling a finer distinction between customers and leading to improved predictive capabilities. This advantage promotes our approach to learning and understanding the context of customer activities better.

Table 4 presents the results of our real-time evaluation. Specifically, it shows the number of predictions the approaches can generate within 0.1 s. The purchase and churn baseline values are rounded up to the nearest tenth, while the other three approaches are rounded down to a full thousand.

Table 4. The number of predictions performed within 0.1 s for each approach.

Approach	Number of Predictions in 0.1 s	
Purchase Baseline	170	−194%
Churn Baseline	30	−199%
CTR Baseline	7000	−52%
UCR Baseline	10,000	−18%
Our	12,000	+0%

When choosing an approach for a real-world prediction system, it is essential to factor in all of our requirements beyond performance. In particular, as discussed previously, data privacy has become increasingly important. Therefore, all evaluated approaches rely exclusively on available interaction data for making predictions and are compliant with data protection regulations, following some adjustments we made beforehand. Additionally, all tested approaches can generate predictions within 0.1 s. Nonetheless, the purchase and churn baseline cannot make a sufficient amount of predictions to deal with too many simultaneous customer accesses in real-time without adding additional compute resources and parallelization, which increases operational costs. Both approaches do not have the required flexibility to handle all tasks and datatypes, which is indispensable for our real-world application without extensive modification and manual effort accompanied by data analysis and laborious testing.

On the contrary, the CTR baseline, UCR baseline, and our approach fulfill our flexibility requirement. Adding or removing information can be done with less effort. It should be noted that the degree of ease of implementation depends on the embedding technique used. We have fully implemented all three approaches such that the inclusion of an extra feature is easily accomplished through a configuration file update. Although further effort is required, this process is considerably simpler than the conventional feature engineering approach.

In the context of real-world applications, a critical consideration is the low-resources requirement to minimize costs. The training of an RF can be completed in a brief period, with minimal computational demands. Conversely, neural networks necessitate a greater allocation of resources. The utilization of the CTR baseline considerably extends the training duration, rendering it ill suited for our live prediction system. While this approach demonstrates exceptional performance in specific scenarios, it imposes substantial resource requirements, leading to higher costs for the enterprise.

Table 5 summarizes which approach meets which requirement. The CTR baseline, UCR baseline, and our approach meet all requirements and are therefore suitable for our real-world applications. They are real-time capable and ensure privacy compliance by accommodating unknown customers, and the information gathered during the ongoing session is sufficient for precise predictions. We can achieve UCR through learned embeddings without the need for complex models, which is especially beneficial for smaller companies.

Table 5. Requirements that the tested approaches meet that we need for a prediction system to be applied in a real-world application.

Approach	Performance	Flexibility	Data Protection	Real-Time
Purchase Baseline			✓	
Churn Baseline			✓	
CTR Baseline	✓	✓	✓	✓
UCR Baseline	✓	✓	✓	✓
Our	✓	✓	✓	✓

7. Further Discussion and Analysis

Our two-stage approach has been developed through a series of experiments, during which we investigated which combinations of embeddings and models offer the best performance. We also tested end-to-end approaches. The following investigations were carried out: Use of different features, such as time of interaction, event type, product category, and product price, if given. The utilization of distinct models of simple multilayer perceptrons (MLPs) with varying depth and width, in conjunction with residual connections, LSTM units, and gated recurrent units (GRUs), as well as compact transformer architectures comprising up to six layers, was undertaken. Additionally, the approaches were trained with varying numbers of epochs.

Thereby, our experiments yielded the following insights: End-to-end approaches exhibit overfitting during training, resulting in a decline in test performance as the number of epochs increases. This phenomenon is also observed when a two-stage approach is employed with deep models like transformers. The LSTM offers the optimal performance in our use case. The addition of additional features in the embedding only marginally enhances performance. Datasets lacking event-type information exhibit a positive effect by embedding category or time information of the interaction. However, one disadvantage is that the inference runtime increases with each additional feature. Event types can be encoded in the interaction without any loss of performance and do not lead to any additional runtime during inference. In terms of pre-trained embedding approaches, Skip-Gram is slightly superior to Continuous Bag of Words (CBOW). Embedding pretraining of a few epochs results in a satisfactory representation. However, as the embeddings are trained for longer, their effectiveness increases, reaching the optimal performance level between 60 and 100 epochs depending on the data used, after this point, no discernible improvement in prediction performance could be identified.

In the following, we will examine in more detail why we chose to use an LSTM for task prediction rather than a Transformer, which is the go-to architecture in NLP for large language models and is increasingly popular in computer vision. Table 6 presents the results of three Encoder-Transformer models with one, three, and six layers, respectively. Each model utilizes four attention heads and a 128-dimensional feed-forward layer. For the binary classification head, we apply average pooling over the encoder output. All other configurations follow the original BERT architecture as proposed by Devlin et al. [76].

Table 6. Performance of three different Transformer-Encoder models and LSTM with our proposed UCR.

		YooChoose		**RetailRocket**			**OpenCDP**			**Industrial**		
Approach	**Score**	**Purchase**	**Click**	**Purchase**	**Churn**	**Click**	**Purchase**	**Churn**	**Click**	**Purchase**	**Churn**	**Click**
LSTM	F1	0.676	0.777	0.674	0.980	0.660	0.875	0.568	0.612	0.766	0.989	0.945
	AUC	0.683	0.890	0.707	0.706	0.659	0.922	0.567	0.549	0.821	0.731	0.981
Transformer-1	F1	0.651	0.641	0.635	0.616	0.346	0.827	0.497	0.027	0.754	0.663	0.666
	AUC	0.652	0.625	0.647	0.579	0.497	0.874	0.550	0.499	0.810	0.701	0.494
Transformer-3	F1	0.628	0.646	0.633	0.624	0.323	0.845	0.550	0.022	0.753	0.661	0.605
	AUC	0.663	0.652	0.634	0.588	0.495	0.896	0.554	0.499	0.809	0.706	0.487
Transformer-6	F1	0.592	0.646	0.628	0.626	0.322	0.847	0.564	0.101	0.758	0.661	0.597
	AUC	0.643	0.634	0.632	0.603	0.492	0.906	0.557	0.500	0.812	0.704	0.496

The results of our experiments indicate that using an LSTM for prediction yields more accurate results than the evaluated Transformer-based models. This finding seems unexpected, given the popularity and proven effectiveness of Transformer architectures in a wide range of machine learning tasks, particularly in sequence modeling and natural language processing.

Interestingly, we also observe that smaller Transformer models outperform larger ones on smaller datasets like YooChoose or RetailRocket. We hypothesize that the larger models overfit the training data, leading to poorer generalization on test samples. This hypothesis is supported by examining the training and test logs for all tested approaches, including the UCR-baseline and CTR-baseline. Notably, the UCR-baseline and CTR-baseline achieve near-perfect F1 and AUC scores between 20 and 30 training epochs when validated on the training data. Similar trends are observed for the Transformer-based models, which show peak performance around 40 epochs of training. However, their performance on the test data deteriorates with continued training.

Consequently, a more detailed analysis was conducted. Figure 5 illustrates the AUC scores for purchase prediction on the test data for the UCR-baseline DUPN, the CTR-baseline DIN, the three Transformer models, and LSTM across different training epochs. As is evident, all models demonstrate a decline in performance over time, particularly DUPN and DIN. The observed overfitting can be attributed to the excessive number of parameters in the models. For instance, DIN has a total of 747,186 parameters, while DUPN has 212,710 parameters. In contrast, the one-layer Transformer has only 25,441 parameters, the three-layer Transformer has 50,849 parameters, and the six-layer Transformer has 88,961 parameters. In contrast, the LSTM model utilizes a comparatively modest number of parameters, with a total of 8481, a factor that likely contributes to its superior generalization across diverse use cases and tasks.This behavior is consistent across all the investigated tasks and datasets.

These results raise intriguing questions about the relationship between model complexity, dataset size, and generalization. While the success of Transformers on large-scale datasets is well documented, our findings suggest that on smaller datasets, the simplicity and parameter efficiency of LSTMs can provide a significant advantage. This outcome is particularly noteworthy in light of the prevalent claim that Transformer architectures excel in modeling long-range dependencies more efficiently than LSTMs.

One potential explanation for the observed discrepancy in performance is the inherent characteristics of the datasets themselves. Small datasets may not provide sufficient diverse training examples to fully leverage the capacity of larger models, such as Transformers with multiple layers. In such scenarios, the augmented model capacity may lead to overfitting rather than enhanced generalization. Conversely, the reduced parameter count of the LSTM enables it to achieve a more optimal balance between learning capacity and generalization.

Another consideration is the training dynamics of the two architectures. Transformers rely heavily on self-attention mechanisms, which can be sensitive to noise and sparsity in the data, particularly in smaller datasets. LSTMs, with their sequential structure and built-in gating mechanisms, seem to be more robust to such challenges, allowing them to capture meaningful patterns more effectively in data-constrained scenarios.

Overall, our findings highlight the importance of aligning model architecture and complexity with the characteristics of the dataset. While Transformers are a powerful tool, their advantages may not always manifest in settings with limited data or specific types of sequence prediction tasks. Further research is needed to better understand these dynamics and to develop adaptive approaches that leverage the strengths of both LSTMs and Transformers in a principled manner.

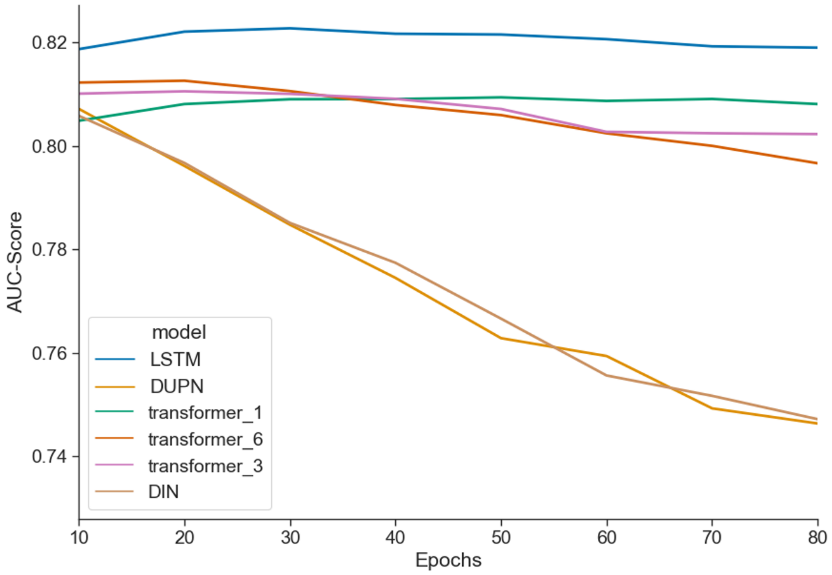

Figure 5. Test AUC performance of the neural learning models for purchase prediction for the industrial dataset over 80 epochs.

A key consideration for practical applications is the scalability of the proposed approach and its relevance to e-commerce businesses of different sizes. Smaller e-commerce businesses, which often lack access to vast amounts of data, could benefit significantly from the findings of this study. Our two-step approach combining embeddings and LSTMs, with its reduced computational and memory requirements, is well suited for deployment in such environments. The proven robustness and efficiency of embeddings make them an attractive option for startups or mid-sized enterprises seeking to implement predictive analytics with a constrained infrastructure.

In contrast, larger e-commerce enterprises, which typically have access to extensive datasets, may find Transformers advantageous due to their capacity to capture complex, long-range dependencies inherent in large-scale interactions. However, while our experiments demonstrate the approach's feasibility on mid-sized datasets, testing the method on very large-scale datasets was beyond the scope of this study. The ability to handle such datasets effectively is vital for large enterprises with high volumes of user interactions.

Therefore, for larger e-commerce enterprises, where extensive datasets are readily available, additional performance considerations must be evaluated.

All research is subject to certain limitations, which must be discussed in order to provide a transparent assessment of the methods employed. Our experiments demonstrate that our approach outperforms other baseline models for prediction tasks across all investigated datasets. This finding, while robust, raises certain limitations that warrant discussion.

Firstly, our results and analysis suggest that model complexity, as characterized by the number of parameters, plays a critical role in generalization, particularly on smaller datasets. Larger models appear to suffer from overfitting despite their architectural advantages in capturing long-range dependencies. However, attributing the superior performance of our approach solely to their smaller parameter count may oversimplify the relationship between model architecture and dataset characteristics. While the smaller datasets, such as YooChoose and RetailRocket, are representative of real-world applications, they may not fully capture the complexity or diversity of interactions found in larger-scale datasets where larger models like Transformer architectures excel. This dataset constraint limits the generalizability of our findings to small-to-medium-scale data scenarios. Testing on more extensive datasets, potentially with billions of interactions, could yield different results and should be investigated in the future. Nonetheless, it is important to acknowledge that the majority of enterprises lack access to such extensive large datasets, highlighting the need for our approach.

Furthermore, we acknowledge a notable limitation in the preparation of the datasets: the need to generate churn and click labels for prediction due to the unavailability of these labels in their original form.

The generation of these labels introduces potential biases and uncertainties. Labeling churn, for example, often relies on assumptions such as a specific duration of inactivity to define a "churned" customer. Similarly, click labels may be derived based on interactions that meet certain thresholds, such as time spent on a page or the number of items clicked during a session. These assumptions, while necessary, might not fully capture the underlying behaviors they aim to represent, potentially leading to noise in the data or misclassification of customer behavior.

The impact of these biases can extend to the predictive models, potentially influencing their performance and generalizability. Models trained on generated labels might learn to overfit to the heuristic patterns rather than the true underlying behaviors. By using a diverse set of datasets, we aimed to reduce over-reliance on any single labeling heuristic. Despite these measures, the inherent limitations of label generation must be acknowledged.

8. Summary and Outlook

E-commerce is an integral part of modern society, opening up a wide range of opportunities. One form that is frequently sought is personalization. In this context, companies with small amounts of data and fewer resources need to be able to establish contact with their customers in e-commerce. Parameter-intensive and therefore data-hungry deep learning models are rarely feasible for such companies. Furthermore, the use of such models is not always advisable. However, our experiments show that a learning approach does not require such complexity to solve various e-commerce tasks. A self-supervised, trained embedding as a universal customer representation and an LSTM prove to be sufficient to compete with state-of-the-art task-solving performance and furthermore fulfill the criteria required by real-world applications. Additionally, our results show that approaches utilizing customer modeling by experts can no longer compete with learning approaches.

In the future, we plan for the system to be evaluated by marketing experts who should evaluate the reasonableness of the representation. In order to assess plausibility, we will investigate visualization and ablation studies that provide insights into the embedding-based UCR and its usage.

Currently, we have only analyzed the ongoing session, but if accessible, we intend to expand our approach to include the prior user history. In order to achieve this, it may be worthwhile to augment the embeddings with time-based activity data. There is existing literature on this topic, such as time2vec [77] or TEE [54].

Continual retraining is a limitation of our approach we want to tackle in the future. Therefore, we plan to investigate approaches on how to incrementally add new activities in the pre-trained embedding space without losing previously learned knowledge.

Another next step is to develop a marketing decision system. Currently, it is tailored for human understanding, utilizing customer intention probabilities that result from the prediction system to make marketing decisions. Theoretically, we could directly link it with the embedding representation to make more efficient and improved decisions. However, as a disadvantage, this would result in increased difficulties when it comes to reasoning, which should be considered.

Author Contributions: M.A.G.: Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing—original draft, writing—review and editing, visualization, project administration; P.M.: Validation, data curation, writing—review and editing, funding acquisition; T.M.: Validation, writing—review and editing, supervision. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by the Open Access Publication Fund of the University of Wuppertal.

Institutional Review Board Statement: Due to the nature of the study, the Ethics Committee approval wasn't required.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: For our research we used four different e-commerce datasets of which three are publicly accessible and one is 3rd party data. The yoochoose dataset is openly available at https://www.kaggle.com/datasets/chadgostopp/recsys-challenge-2015 accessed on 24 December 2024. The retailrocket dataset is openly available at https://www.kaggle.com/datasets/RetailRocket/ecommerce-dataset accessed on 24 December 2024. The openCDP dataset is openly available at https://www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store accessed on 24 December 2024. Restrictions apply to the availability of the industrial data. Data were obtained from Breinify Inc.

Conflicts of Interest: Author Philipp Meisen was employed by the company Breinify Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Rahman, M.S.; Hossain, M.A.; Zaman, M.H.; Mannan, M. E-service quality and trust on customer's patronage intention: Moderation effect of adoption of advanced technologies. *J. Glob. Inf. Manag. (JGIM)* **2020**, *28*, 39–55. [CrossRef]
2. Fan, Z.; Ou, D.; Gu, Y.; Fu, B.; Li, X.; Bao, W.; Dai, X.Y.; Zeng, X.; Zhuang, T.; Liu, Q. Modeling Users' Contextualized Page-Wise Feedback for Click-Through Rate Prediction in E-commerce Search. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event, 21–25 February 2022; Association for Computing Machinery: New York, NY, USA, 2022; pp. 262–270. [CrossRef]
3. Zhou, G.; Mou, N.; Fan, Y.; Pi, Q.; Bian, W.; Zhou, C.; Zhu, X.; Gai, K. Deep Interest Evolution Network for Click-Through Rate Prediction. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 5941–5948. [CrossRef]
4. Ni, Y.; Ou, D.; Liu, S.; Li, X.; Ou, W.; Zeng, A.; Si, L. Perceive Your Users in Depth: Learning Universal User Representations from Multiple E-Commerce Tasks. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '18, London, UK, 19–23 August 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 596–605. [CrossRef]

5. Carmel, D.; Haramaty, E.; Lazerson, A.; Lewin-Eytan, L. Multi-Objective Ranking Optimization for Product Search Using Stochastic Label Aggregation. In Proceedings of the Web Conference 2020, WWW'20, Taipei, Taiwan, 20–24 April 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 373–383. [CrossRef]
6. Alves Gomes, M.; Meisen, T. A review on customer segmentation methods for personalized customer targeting in e-commerce use cases. *Inf. Syst. E-Bus. Manag.* **2023**, *21*, 527–570. [CrossRef]
7. Hong, T.; Kim, E. Segmenting customers in online stores based on factors that affect the customer's intention to purchase. *Expert Syst. Appl.* **2012**, *39*, 2127–2131. [CrossRef]
8. Lin, W.; Milic-Frayling, N.; Zhou, K.; Ch'ng, E. Predicting Outcomes of Active Sessions Using Multi-action Motifs. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Thessaloniki, Greece, 14–17 October 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 9–17. [CrossRef]
9. Martínez, A.; Schmuck, C.; Pereverzyev, S.; Pirker, C.; Haltmeier, M. A machine learning framework for customer purchase prediction in the non-contractual setting. *Eur. J. Oper. Res.* **2020**, *281*, 588–596. [CrossRef]
10. Esmeli, R.; Bader-El-Den, M.; Abdullahi, H. Towards early purchase intention prediction in online session based retailing systems. *Electron. Mark.* **2021**, *31*, 697–715. [CrossRef]
11. Yang, B.; Liu, K.; Xu, X.; Xu, R.; Liu, H.; Xu, H. Learning Universal User Representations via Self-Supervised Lifelong Behaviors Modeling. 2021. Available online: https://openreview.net/forum?id=YTtMaJUN_uc (accessed on 10 September 2024).
12. Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; Jiang, P. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM'19, Beijing, China, 3–7 November 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 1441–1450. [CrossRef]
13. U.S. Department of Justice. Electronic Communications Privacy Act of 1986 (ECPA). 1986. Available online: https://bja.ojp.gov/program/it/privacy-civil-liberties/authorities/statutes/1285 (accessed on 1 October 2024).
14. State of California Department of Justice. California Consumer Privacy Act (CCPA). 2018. Available online: https://oag.ca.gov/privacy/ccpa (accessed on 1 October 2024).
15. European-Parliament. Regulation (EU) 2016/679 of the European Parliament and of the Council. 2016. Available online: https://www.legislation.gov.uk/eur/2016/679/article/1 (accessed on 1 October 2024).
16. Gu, J.; Wang, F.; Sun, Q.; Ye, Z.; Xu, X.; Chen, J.; Zhang, J. Exploiting behavioral consistence for universal user representation. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; Volume 35, pp. 4063–4071.
17. Wu, C.; Wu, F.; Qi, T.; Lian, J.; Huang, Y.; Xie, X. Ptum: Pre-training user model from unlabeled user behaviors via self-supervision. *arXiv* **2020**, arXiv:2010.01494.
18. Bertrand, J.H.; Gargano, J.P.; Mombaerts, L.; Taws, J. Autoencoder-based General Purpose Representation Learning for Customer Embedding. *arXiv* **2024**, arXiv:2402.18164.
19. Romov, P.; Sokolov, E. RecSys Challenge 2015: Ensemble Learning with Categorical Features. In Proceedings of the 2015 International ACM Recommender Systems Challenge, RecSys'15 Challenge, Vienna, Austria, 16–20 September 2015; Association for Computing Machinery: New York, NY, USA, 2015. [CrossRef]
20. Park, C.; Kim, D.; Oh, J.; Yu, H. Predicting User Purchase in E-Commerce by Comprehensive Feature Engineering and Decision Boundary Focused Under-Sampling. In Proceedings of the 2015 International ACM Recommender Systems Challenge, RecSys'15 Challenge, Vienna, Austria, 16–20 September 2015; Association for Computing Machinery: New York, NY, USA, 2015. [CrossRef]
21. Li, Q.; Gu, M.; Zhou, K.; Sun, X. Multi-classes feature engineering with sliding window for purchase prediction in mobile commerce. In Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW), Atlantic City, NJ, USA, 14–17 November 2015; IEEE: New York, NY, USA, 2015; pp. 1048–1054.
22. Mokryn, O.; Bogina, V.; Kuflik, T. Will this session end with a purchase? Inferring current purchase intent of anonymous visitors. *Electron. Commer. Res. Appl.* **2019**, *34*, 100836. [CrossRef]
23. Balyemah, A.J.; Weamie, S.J.; Bin, J.; Janda, K.V.; Joshua, F.J. Predicting Purchasing Behavior on E-Commerce Platforms: A Regression Model Approach for Understanding User Features that Lead to Purchasing. *Int. J. Commun. Netw. Syst. Sci.* **2024**, *17*, 81–103. [CrossRef]
24. Malik, V.; Mittal, R.; Chaudhry, R.; Yadav, S.A. Predicting Purchases and Personalizing the Customer Journey with Artificial Intelligence. In Proceedings of the 2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 14–15 March 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–5.
25. Sudirman, I.D.; Sharif, O.O.; Rahmatillah, I.; Dewi, C.K. Customer Purchase Prediction Using Optimized Decision Tree with Particle Swarm Optimization. In Proceedings of the 2024 International Conference on Data Science and Its Applications (ICoDSA), Kuta, Bali, Indonesia, 10–11 July 2024; pp. 58–63. [CrossRef]
26. Sheil, H.; Rana, O.; Reilly, R. Predicting purchasing intent: Automatic feature learning using recurrent neural networks. *arXiv* **2018**, arXiv:1807.08207.

27. Alves Gomes, M.; Meyes, R.; Meisen, P.; Meisen, T. Will This Online Shopping Session Succeed? Predicting Customer's Purchase Intention Using Embeddings. In Proceedings of the 31st ACM International Conference on Information and Knowledge Management, CIKM'22, Atlanta, GA, USA, 17–21 October 2022; Association for Computing Machinery: New York, NY, USA, 2022; pp. 3707–3716. [CrossRef]
28. Pfeifer, P.E. The optimal ratio of acquisition and retention costs. *J. Target. Meas. Anal. Mark.* **2005**, *13*, 179–188. [CrossRef]
29. Rachid, A.D.; Abdellah, A.; Belaid, B.; Rachid, L. Clustering prediction techniques in defining and predicting customers defection: The case of e-commerce context. *Int. J. Electr. Comput. Eng.* **2018**, *8*, 2367. [CrossRef]
30. Berger, P.; Kompan, M. User Modeling for Churn Prediction in E-Commerce. *IEEE Intell. Syst.* **2019**, *34*, 44–52. [CrossRef]
31. Fridrich, M.; Dostál, P. User Churn Model in E-Commerce Retail. *Sci. Pap. Univ. Pardubic. Ser. D Fac. Econ. Adm.* **2022**, *30*, 1478. [CrossRef]
32. Perišić, A.; Pahor, M. RFM-LIR Feature Framework for Churn Prediction in the Mobile Games Market. *IEEE Trans. Games* **2022**, *14*, 126–137. [CrossRef]
33. Xiahou, X.; Harada, Y. B2C E-Commerce Customer Churn Prediction Based on K-Means and SVM. *J. Theor. Appl. Electron. Commer. Res.* **2022**, *17*, 458–475. [CrossRef]
34. Ehsani, F.; Hosseini, M. Customer churn prediction using a novel meta-classifier: An investigation on transaction, Telecommunication and customer churn datasets. *J. Comb. Optim.* **2024**, *48*, 1–31. [CrossRef]
35. Hughes, A.M. Strategic database marketing: The masterplan for starting and managing a profitable. In *Customer-Based Marketing Program*; Irwin Professional Publishing: Burr Ridge, IL, USA, 1994.
36. Gulhane, P.R.; Kumar, T.S.P. TensorFlow Based Website Click through Rate (CTR) Prediction Using Heat maps. In Proceedings of the 2018 International Conference on Recent Trends in Advance Computing (ICRTAC), Chennai, India, 10–11 September 2018; pp. 97–102. [CrossRef]
37. Huang, G.; Chen, Q.; Deng, C. A New Click-Through Rates Prediction Model Based on Deep & Cross Network. *Algorithms* **2020**, *13*, 342. [CrossRef]
38. Chen, C.; Chen, H.; Zhao, K.; Zhou, J.; He, L.; Deng, H.; Xu, J.; Zheng, B.; Zhang, Y.; Xing, C. EXTR: Click-Through Rate Prediction with Externalities in E-Commerce Sponsored Search. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 14–18 August 2022; Association for Computing Machinery: New York, NY, USA, 2022; pp. 2732–2740. [CrossRef]
39. Ge, T.; Zhao, L.; Zhou, G.; Chen, K.; Liu, S.; Yi, H.; Hu, Z.; Liu, B.; Sun, P.; Liu, H.; et al. Image Matters: Visually Modeling User Behaviors Using Advanced Model Server. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 2087–2095. [CrossRef]
40. Li, F.; Chen, Z.; Wang, P.; Ren, Y.; Zhang, D.; Zhu, X. Graph Intention Network for Click-through Rate Prediction in Sponsored Search. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 961–964. [CrossRef]
41. Wang, F.; Zhao, L. A Hybrid Model for Commercial Brand Marketing Prediction Based on Multiple Features with Image Processing. *Secur. Commun. Netw.* **2022**, *2022*, 1–10. [CrossRef]
42. Zhou, G.; Zhu, X.; Song, C.; Fan, Y.; Zhu, H.; Ma, X.; Yan, Y.; Jin, J.; Li, H.; Gai, K. Deep Interest Network for Click-Through Rate Prediction. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '18, London, UK, 19–23 August 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 1059–1068. [CrossRef]
43. Li, X.; Wang, C.; Tong, B.; Tan, J.; Zeng, X.; Zhuang, T. Deep Time-Aware Item Evolution Network for Click-Through Rate Prediction. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland, 19–23 October 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 785–794. [CrossRef]
44. Li, X.; Wang, C.; Tan, J.; Zeng, X.; Ou, D.; Zheng, B. Adversarial Multimodal Representation Learning for Click-Through Rate Prediction. In Proceedings of the Web Conference 2020, WWW'20, Taipei, Taiwan, 20–24 April 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 827–836. [CrossRef]
45. Sang, L.; Li, H.; Zhang, Y.; Zhang, Y.; Yang, Y. AdaGIN: Adaptive Graph Interaction Network for Click-Through Rate Prediction. *ACM Trans. Inf. Syst.* **2024**, *43*, 1–31. [CrossRef]
46. Wang, Y.; Piao, H.; Dong, D.; Yao, Q.; Zhou, J. Warming Up Cold-Start CTR Prediction by Learning Item-Specific Feature Interactions. *arXiv* **2024**, arXiv:2407.10112.
47. Lee, S.; Hwang, S. Context-aware cross feature attentive network for click-through rate predictions. *Appl. Intell.* **2024**, *54*, 9330–9344. [CrossRef]
48. Huan, Z.; Ding, K.; Li, A.; Zhang, X.; Min, X.; He, Y.; Zhang, L.; Zhou, J.; Mo, L.; Gu, J.; et al. Exploring Multi-Scenario Multi-Modal CTR Prediction with a Large Scale Dataset. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, DC, USA, 14–18 July 2024; pp. 1232–1241.

49. Lu, J.; Long, Y.; Li, X.; Shen, Y.; Wang, X. Hybrid Model Integration of LightGBM, DeepFM, and DIN for Enhanced Purchase Prediction on the Elo Dataset. In Proceedings of the 2024 IEEE 7th International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 27–29 September 2024; pp. 16–20. [CrossRef]
50. Geng, B.; Huan, Z.; Zhang, X.; He, Y.; Zhang, L.; Yuan, F.; Zhou, J.; Mo, L. Breaking the Length Barrier: LLM-Enhanced CTR Prediction in Long Textual User Behaviors. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24, Washington, DC, USA, 14–18 July 2024; Association for Computing Machinery: New York, NY, USA, 2024; pp. 2311–2315. [CrossRef]
51. Burri, M.; Schär, R. The reform of the EU data protection framework: Outlining key changes and assessing their fitness for a data-driven economy. *J. Inf. Policy* **2016**, *6*, 479–511.
52. Miller, R.B. Response Time in Man-Computer Conversational Transactions. In Proceedings of the Fall Joint Computer Conference, Part I, San Francisco CA, USA, 9–11 December 1968; Association for Computing Machinery: New York, NY, USA, 1968; pp. 267–277. [CrossRef]
53. Card, S.K.; Robertson, G.G.; Mackinlay, J.D. The Information Visualizer, an Information Workspace. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '91, Montréal, QC, Canada, 22–27 April 2006; Association for Computing Machinery: New York, NY, USA, 1991; pp. 181–186. [CrossRef]
54. Alves Gomes, M.; Wönkhaus, M.; Meisen, P.; Meisen, T. TEE: Real-Time Purchase Prediction Using Time Extended Embeddings for Representing Customer Behavior. *J. Theor. Appl. Electron. Commer. Res.* **2023**, *18*, 1404–1418. [CrossRef]
55. Zeng, J.; Chen, Y.; Zhu, H.; Tian, F.; Miao, K.; Liu, Y.; Zheng, Q. User Sequential Behavior Classification for Click-Through Rate Prediction. In *Database Systems for Advanced Applications. DASFAA 2020 International Workshops*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 267–280. [CrossRef]
56. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2, NIPS'13, Lake Tahoe, NV, USA, 5–10 December 2013; Curran Associates Inc.: Red Hook, NY, USA, 2013; pp. 3111–3119.
57. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
58. Van Rossum, G.; Drake, F.L., Jr. *Python Reference Manual*; Centrum voor Wiskunde en Informatica: Amsterdam, The Netherlands, 1995.
59. Harris, C.R.; Millman, K.J.; van der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *Nature* **2020**, *585*, 357–362. [CrossRef] [PubMed]
60. Wes McKinney. Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference, Austin, TX, USA, 28 June–3 July 2010; van der Walt, S., Millman, J., Eds.; pp. 56–61. [CrossRef]
61. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32, pp. 8024–8035.
62. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
63. Buitinck, L.; Louppe, G.; Blondel, M.; Pedregosa, F.; Mueller, A.; Grisel, O.; Niculae, V.; Prettenhofer, P.; Gramfort, A.; Grobler, J.; et al. API design for machine learning software: Experiences from the scikit-learn project. In Proceedings of the ECML PKDD Workshop: Languages for Data Mining and Machine Learning, Prague, Czech Republic, 23–27 September 2013; pp. 108–122.
64. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 4–8 August 2019.
65. Chou, Y.C.; Chuang, H.H.C. A predictive investigation of first-time customer retention in online reservation services. *Serv. Bus.* **2018**, *12*, 685–699. [CrossRef]
66. Li, X.; Li, Z. A Hybrid Prediction Model for E-Commerce Customer Churn Based on Logistic Regression and Extreme Gradient Boosting Algorithm. *Ing. Syst. d'Inform.* **2019**, *24*, 525–530. [CrossRef]
67. de la Llave Montiel, M.A.; López, F. Spatial models for online retail churn: Evidence from an online grocery delivery service in Madrid. *Pap. Reg. Sci.* **2020**, *99*, 1643–1665. [CrossRef]
68. Zhou, C.; Bai, J.; Song, J.; Liu, X.; Zhao, Z.; Chen, X.; Gao, J. ATRank: An Attention-Based User Behavior Modeling Framework for Recommendation. *Proc. AAAI Conf. Artif. Intell.* **2018**, *32*. [CrossRef]
69. Ren, K.; Zhang, W.; Rong, Y.; Zhang, H.; Yu, Y.; Wang, J. User Response Learning for Directly Optimizing Campaign Performance in Display Advertising. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, Indianapolis, IN, USA, 24–28 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 679–688. [CrossRef]

70. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. In Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2, NIPS'14, Montreal, QC, Canada, 8–13 December 2014; MIT Press: Cambridge, MA, USA, 2014; pp. 3104–3112.
71. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
72. Grave, É.; Joulin, A.; Cissé, M.; Grangier, D.; Jégou, H. Efficient softmax approximation for GPUs. In Proceedings of the 34th International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; Precup, D., Teh, Y.W., Eds.; Proceedings of Machine Learning Research; Volume 70, pp. 1302–1310.
73. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
74. Shen, W. DeepCTR: Easy-to-Use, Modular and Extendible Package of Deep-Learning Based CTR Models. 2017. Available online: https://github.com/shenweichen/deepctr (accessed on 1 October 2024).
75. Ni, J.; Li, J.; McAuley, J. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3 November 2019; pp. 188–197.
76. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
77. Kazemi, S.M.; Goel, R.; Eghbali, S.; Ramanan, J.; Sahota, J.; Thakur, S.; Wu, S.; Smyth, C.; Poupart, P.; Brubaker, M. Time2vec: Learning a vector representation of time. *arXiv* **2019**, arXiv:1907.05321.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.