Learning Self-Modulating Attention in Continuous Time Space with Applications to Sequential Recommendation

Chao Chen [1] **Haoyu Geng** [1 2] **Nianzu Yang** [1 2] **Junchi Yan** [1 2] **Daiyue Xue** [3] **Jianping Yu** [3] **Xiaokang Yang** [1 2]

Abstract

User interests are usually dynamic in the real world, which poses both theoretical and practical challenges for learning accurate preferences from rich behavior data. Among existing user behavior modeling solutions, attention networks are widely adopted for its effectiveness and relative simplicity. Despite being extensively studied, existing attentions still suffer from two limitations: i) conventional attentions mainly take into account the spatial correlation between user behaviors, regardless the distance between those behaviors in the continuous time space; and ii) these attentions mostly provide a dense and undistinguished distribution over all past behaviors then attentively encode them into the output latent representations. This is however not suitable in practical scenarios where a user's future actions are relevant to a small subset of her/his historical behaviors. In this paper, we propose a novel attention network, named *self-modulating attention*, that models the complex and non-linearly evolving dynamic user preferences. We empirically demonstrate the effectiveness of our method on top-N sequential recommendation tasks, and the results on three large-scale real-world datasets show that our model can achieve state-of-the-art performance.

1. Introduction

Preference learning is of critical importance in modern machine learning applications, such as online advertising, E-commerce and social media. Often, users' preferences are not static and evolve over time due to a variety of reasons. For example, a user's interest in a particular brand might fade away due to maturity or a change in lifestyle. Nonetheless, it is difficult to model all causal aspects explicitly (Widmer & Kubat, 1996; Koren, 2009; Wu et al., 2017).

To model the dynamics in user preferences, many research works (Kang & McAuley, 2018; Ying et al., 2018; Zhou et al., 2018; Zhang et al., 2019) develop different attention mechanisms to learn the sequential patterns of how users' future actions are interacted with previous behaviors, and so far have achieved state-of-the-art performance on many benchmark datasets. The majority of these attentions (implicitly) assume that the underlying sequential patterns are time independent(Kumar et al., 2019; Li et al., 2020; Chang et al., 2020). This is however contradicted with the fact that more recent behaviors will have more impact on the future actions, or more specifically identical behavior sequences of two users with different time intervals should have different implications on their current and future decisions.

To tackle the challenging problem, recent approaches generalize the notion of positional embedding to continuous time (Xu et al., 2019; Shukla & Marlin, 2021). The idea behind these approaches is to encode periodic patterns dependent on the progression of time into a high-dimensional vector space (i.e., time embedding), then by using translation-invariant random Fourier features (Rahimi & Recht, 2007; Yu et al., 2016) to approximately measure the temporal distance between user behaviors in the sequence.

Orthogonal to prior studies, we focus on two pertinent questions fundamental to attention-based preference learning in continuous time space: (i) *how can attentions model the impact of both sequential positions and continuous timestamps in an explicit manner?* — This matters a lot for business, as it helps to better understand users and their needs (Zhang & Chen, 2020; Arrieta et al., 2020; Chen et al., 2021a); and (ii) *how can attentions identify sparse behaviors interacted with future actions?* — Existing techniques in this direction mainly base on sparsemax (Martins & Astudillo, 2016; Niculae & Blondel, 2017) to yield sparse distributions over past behaviors. These approaches tend to preserve very limited temporal information which leads to the difficulty to capture complex and fine-grained temporal dynamics.

In this paper, we propose a novel attention mechanism,

[1] MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, Shanghai, China [2] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China [3] Meituan, Beijing, China. Correspondence to: Junchi Yan <yanjunchi@sjtu.edu.cn>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

named self-modulating attention, to model the complex evolution of user preferences over time. More specifically, our attention explicitly digests temporal dynamics via a conditional intensity function that specifies whether the observed behaviors are relevant with users' future interests at a given time point: high (low) intensities of the behaviors amplify (attenuate) their contributions in the output representations. Here the term *self-modulating* is meant to encompass above positive and negative effects of a single sequence to compute the representation of the same sequence. We further propose continuous time regularization to penalize intensities, as the source of supervised signal for preference learning is mostly coming from the behavior data that is independent of time.

Intuition. Existing attention mechanism is based on the correlation between the query and keys (or between the past and current events for sequential behavior modeling), while such correlation is not modeled as a function of time. However, for real-world behavior modeling e.g. sequential recommendation in the continuous time space, a user's behavior pattern may vary over time.

Theoretical Motivation. More interestingly, in this paper we theoretically show (see details in Theorem 1) under a bounded impact of the past behavior to future actions, more sparse behaviors can lead to higher generalization ability for a prediction model. This theory motivates us to design a mechanism for more effectively uncovering such truly useful past behaviors in a more discriminative manner.

Technical Implementation. Based on the above intuitive and theoretical analysis, in this paper we aim to develop a time sensitive attention mechanism to adaptively and predictively re-weight the past behaviors for their impact to current scoring. The temporal point process is used for its interpretable and rigorous Bayesian nature. The conditional intensity function is used to modulate and soften the time-varying attention in continuous time space.

The key contributions of this paper are as follows:

- We establish a theoretical bound for attention based approaches in preference learning, showing the benefits of bounded and sparsified impact of behavior history.
- To better explore the behavior history, we develop a notion of self-modulating attention, with the classic attention as a special case of our model. Our work concerns modeling attentions informed by time-dependent uncertainties.
- To effectively incorporate the temporal model into preference learning, we develop a continuous time regularizer to fill the gap between traditional preference learning and continuous-time temporal process modeling.
- Top-N recommendation results on benchmark datasets demonstrate that our self-modulating attention outperforms state-of-the-art methods by a notable margin, and it can also be reused as an orthogonal plug-in for existing methods.

2. Related Work

User modeling is of crucial importance in many online applications such as advertising, E-commerce and social media. In general, the goal of user modeling is to learn user representations from the complex and abundant behavior data (Adomavicius & Tuzhilin, 2005; Su & Khoshgoftaar, 2009).

Traditional Recommendation Approaches. Traditional approaches usually assume that user preference is static, and collaborative filtering (CF) (Herlocker et al., 1999; Sarwar et al., 2001) is one of the most popular approaches due to ease of implementation and quality of recommendation. Among existing CF solutions, matrix approximation based methods (Koren, 2008; Koren et al., 2009; Chen et al., 2015; Li et al., 2017; 2021) have achieved state-of-the-art performance in many benchmark datasets, especial for Netflix prize data (Bennett & Lanning, 2007). However, in practice, user preferences often drift over time due to various reasons.

Sequential Recommendation Approaches. To address the problem, FPMC (He & McAuley, 2016) and its hierarchical version HRM (Wang et al., 2015), propose to use Markov chains to model sequential patterns by learning user-specific transition matrices. These approaches have been proved to be able to capture short-term patterns. The major concern behind them is the potential state space explosion intractability in face of different possible sequences over items (He et al., 2017a; Wu et al., 2017; Chen et al., 2021b).

Another emerging line focuses on modeling sequential patterns by using recurrent neural networks. For example, RRN (Wu et al., 2017) and GRU4REC (Hidasi et al., 2016) exploit LSTM (Hochreiter & Schmidhuber, 1997) or GRU (Chung et al., 2014) module to capture dynamic user preferences dependent on sequence positions. Meanwhile, SHAN (Ying et al., 2018), RUM (Chen et al., 2018), DIN (Zhou et al., 2018) and SASREC (Kang & McAuley, 2018) introduce attention mechanisms to sequential recommendation problems. Compared with recurrent networks, attentions have shorter path to access distant positions, and thereby achieve superior performance in many cases (Vaswani et al., 2017).

Temporal Recommendation Approaches. Compared to sequential recommendation problem that has been well studied, relatively less attention has been paid to temporal recommendation algorithms. The prominent work timeSVD++ (Koren, 2009) proposes the use of explicit temporal bias to model the discrete time dynamics. Apart from hand engineered features, the authors in (Lu et al., 2016) develop a vector auto-regressive model to predict future user preferences. Empirical results have shown state-of-the-art performance on many benchmark datasets, but the costly computational overhead and memory use make it hard to support large-scale industrial applications.

Another line of works follow the idea of positional embedding which designs time embedding to capture temporal information, with attention mechanisms for end-to-end learning (Liu et al., 2020). The authors in (Li et al., 2020) divide the time into intervals and project them to low-dimensional space. This treatment discretizes the time and reduces the power of modeling fine-grained temporal information (Trivedi et al., 2019; Chang et al., 2020). By contrast, the works (Xu et al., 2019; Shukla & Marlin, 2021) learn functions of continuous time to capture the dynamics.

Distinct from the above approaches, our method involves re-weighting of attention coefficients according to the intensity function of temporal point processes (Daley & Vere-Jones, 2003), such that self-modulation can be achieved based on continuous time information. Thanks to the intensity powered by neural networks, our self-modulating attention is enabled to model complex evolving dynamics in continuous time space.

Temporal point process has been shown powerful for temporal modeling. The work (Zhou et al., 2013) deals with mutual-excitation of event sequences, with a regularizer on the sparseness and low-rank assumption of mutual correlations. As the rise of deep learning, recurrent neural networks are widely adopted to model temporal dynamics (Mei & Eisner, 2017; Trivedi et al., 2017; Xiao et al., 2017), which are further replaced or enhanced by attentions (Zhang et al., 2020; Zuo et al., 2020). We argue that these works are in general orthogonal to ours, as they focus on the modeling of event times rather than preference learning. In fact, our work can enjoy the advances in (Zhang et al., 2020; Zuo et al., 2020) to improve the expressiveness.

This paper focuses on developing a new attention dependent on both sequential positions and continuous timestamps. More specifically, we generalize the formulation of existing attentions by introducing the intensity function, which specifies if user tastes on the items are changed or not. Compared to existing sequential recommendation models, the difference lies in the ability to model temporal dynamics in the continuous time space.

3. Notations and Preliminaries

Throughout this paper, we denote scalars by either lowercase or uppercase letters, vectors by boldface lowercase letters, and matrices by boldface uppercase letters. Unless otherwise specified, all the vectors are column vectors.

Let $\mathcal{S} = \{(t_j, i_j)\}_{j=1}^{L}$ be a user behavior sequence, each pair (t_j, i_j) denotes that item i_j is rated at time t_j. Then the attentions can be generally defined by, given any query vector $\mathbf{q}$ and the sequence $\mathcal{S}$:

$$\mathrm{Att}(\mathbf{q}\big|\mathcal{S}) = \sum_{j'=1}^{n} p(\mathbf{v}_{j'}\big|\mathcal{S})\mathbf{v}_{j'} = \mathbb{E}_{p(\mathbf{v}|\mathcal{S})}\mathbf{v}, \tag{1}$$

where n is the number of items and the distribution $p(\mathbf{v}|\mathcal{S})$ attentively aggregates all the value vectors $\{\mathbf{v}_{j'}\}_{j'=1}^{n}$.

In the scaled dot-product attention (Vaswani et al., 2017), $p(\mathbf{v}_{\mathbf{j}'}|\mathcal{S})$ equals to $\eta \exp(\mathbf{q}^\top \mathbf{k}_{j'}/\sqrt{d})$ if the item j' is in the sequence $\mathcal{S}$ and equals to zero otherwise, where η is used for normalization i.e., $\eta^{-1} = \sum_{(t,i)\in\mathcal{S}} \exp(\mathbf{q}^\top \mathbf{k}_i/\sqrt{d})$. In the masked version, we should use $p(\mathbf{v}_{\mathbf{j}'}|\mathcal{H}_t)$ where $\mathcal{H}_t = \{(t_j, i_j)\big|t_j < t\}$ signifies the history up to time t.

In the vast literature of sequential recommendation, attention mechanisms have been widely adopted to capture the dynamic user preferences. Most of these approaches (Ying et al., 2018; Kang & McAuley, 2018; Ma et al., 2019) can be generally formulated as follows:

$$\hat{\mathbf{R}} = \mathbf{PVB} \quad \text{and} \quad \mathbf{P}_{u,i} = p(\mathbf{v}_i\big|\mathcal{H}_t^{(u)}), \tag{2}$$

where $\hat{\mathbf{R}}$ denotes the estimate of the underlying matrix $\mathbf{R}$ with m users and n items, $\mathbf{P} \in \mathbb{R}^{m\times n}$ signifies the correlations between the past behaviors and future actions, $\mathbf{V} \in \mathbb{R}^{n\times d}$ denotes the value matrix, and $\mathbf{B} \in \mathbb{R}^{d\times n}$ acts as the output transformation matrix and can also be viewed as item feature matrix. Notably, the nonlinear projection $\phi(\mathbf{PV}, \mathbf{B})$ as in (He et al., 2017b), can also be formed as Eq. (2). This is because the ranking score is usually a non-negative measure, so that it can be approximated by the inner product of random Fourier feature matrices (Rahimi & Recht, 2007; Yu et al., 2016; Choromanski et al., 2020).

Remark. We continue to study the generalization bound of attention-based models in the form of Eq. (2). Mathematically, we set ρ to bound the number of non-zero entries in $\mathbf{P}$ that characterizes sequential patterns, and μ to bound the impact of each historical behavior. As proved in Theorem 1, the generalization error of these models decreases with both ρ and μ. This result is consistent with (Lee et al., 2013; Li et al., 2016) — penalizing the model parameters with ℓ_2-norm can usually improve the model performance by avoiding overfitting. Perhaps more importantly, the theorem shows that it is beneficial to limit a user's future action correlated with a small subset of past behaviors.

Theorem 1 (Generalization Bound). *Suppose that the loss function ℓ is L-Lipschitz, and for the estimate $\hat{\mathbf{R}}$ on an random example set Ω we bound $\rho|\Omega| = \sup_{\mathbf{P}\in\mathcal{F}} \sum_{(u,i)\in\Omega} \| \mathbf{P}_{i,*} \|_0$ and $\mu = \sup_{(u,i,k)\in\Omega} |\mathbf{P}_{u,k}(\mathbf{VB})_{k,j}|$, then with probability at least $1-\delta$, we have the bound:*

$$\mathbf{E}\Big[\ell(\mathbf{R}, \hat{\mathbf{R}})\Big] \leq \mathbf{E}_\Omega \Big[\ell(\mathbf{R}, \hat{\mathbf{R}})\Big] + \mathcal{O}\Big(L\mu\sqrt{\frac{C\rho \ln|\Omega|}{|\Omega|}} + \sqrt{\frac{\ln(1/\delta)}{|\Omega|}}\Big) \tag{3}$$

where $C = d(m+n)\log(48emn)$.

4. Methodology

In this section, we first formulate the generalized attention in continuous time space, which has been rarely studied before. Then we introduce a specific implementation, called self-modulating layer (SMLayer), followed by a continuous time regularization for preference learning, making it more tailored to the sequential recommendation problem.

4.1. Notations

Recall that $\mathcal{S} = \{(t_j, i_j)\}_{j=1}^{L}$ denote a user's behavior sequence of size L, in which each pair (t_j, i_j) means that the given user clicks/likes/views item i_j at time t_j. Then, we further define $\mathcal{H}_t = \{(t_j, i_j) | t_j < t\}$ that signifies the history up to time t. The goal of self-modulating attention is to model the impact of sequential positions as well as continuous timestamps.

We recall that user preferences evolve over time and the interests on the items might wane, such that some past behaviors should not make contributions to the output user representations at some time, even though they are semantically correlated. To remedy such challenge, one straightforward idea is to directly inform the correlation $p(\mathbf{v}_{i_j} \big| \mathcal{H}_t)$ with the expected number of occurrence $\mathbb{E}[\mathrm{N}(t, t + \mathrm{d}t) \big| \mathcal{H}_{t-}, \mathbf{v}_{i_j}]$ conditional on the history $\mathcal{H}_{t-}$, where $\mathcal{H}_{t-} = \mathcal{H}_t \cup \{t_{j+1} \notin (t_j, t)\}$ and $\mathrm{N}(t, t + \mathrm{d}t) \in \{0, 1\}$ denotes the number of occurrences for item i_j in an infinitesimal interval.

4.2. Generalizing Attention to Continuous Time Space

Following this idea, we make an adjustment for classic attentions (Chorowski et al., 2015; Vaswani et al., 2017; Xu et al., 2019) to capture both sequential and temporal dynamics, which is devised as follows:

$$
\begin{aligned}
&\mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\left[\mathbb{E}\Big[\mathrm{N}(t, t+\mathrm{d}t)\big|\mathcal{H}_{t-}, \mathbf{v}\Big]\mathbf{v}\right] \\
\overset{(a)}{=}&\mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\left[p\Big(t_{j+1} \in [t, t+\mathrm{d}t]\big|\mathcal{H}_{t-}, \mathbf{v}\Big)\mathbf{v}\right] \\
\overset{(b)}{=}&\mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\frac{p\Big(t_{j+1} \in [t, t+\mathrm{d}t]\big|\mathcal{H}_t, \mathbf{v}\Big)}{p\Big(t_{j+1} \notin (t_j, t)\big|\mathcal{H}_t, \mathbf{v}\Big)}\mathbf{v} \\
\overset{(c)}{=}&\mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\frac{f(t\big|\mathcal{H}_t, \mathbf{v})\mathrm{d}t}{S(t\big|\mathcal{H}_t, \mathbf{v})}\mathbf{v} \\
\overset{(d)}{=}&\mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\mathbf{v}\lambda^*(t\big|\mathcal{H}_t, \mathbf{v})\mathrm{d}t,
\end{aligned}
\tag{4}
$$

in which (a) holds due to the fact that $\mathrm{N}(t_j, t_j + \mathrm{d}t)$ is either zero or one in the infinitesimal interval; (b) holds due to Bayes' theorem; and (c) - (d) hold due to

$$
\begin{aligned}
p\Big(t_{j+1} \in [t, t+\mathrm{d}t]\big|\mathcal{H}_t, \mathbf{v}\Big) &= f(t\big|\mathcal{H}_t, \mathbf{v})\mathrm{d}t \\
p\Big(t_{j+1} \notin (t_j, t)\big|\mathcal{H}_t, \mathbf{v}\Big) &= S(t\big|\mathcal{H}_t, \mathbf{v}) \\
\frac{f(t\big|\mathcal{H}_t, \mathbf{v})}{S(t\big|\mathcal{H}_t, \mathbf{v})}. &= \lambda^*(t\big|\mathcal{H}_t, \mathbf{v}),
\end{aligned}
\tag{5}
$$

where $S(t\big|\mathcal{H}_t, \mathbf{v})$ and $\lambda^*(t\big|\mathcal{H}_t, \mathbf{v})$ are called survival function and conditional intensity function respectively, in the literature of temporal point process (Aalen et al., 2008; Kleinbaum & Klein, 2010).

In general, dynamic processes will be characterized by the conditional intensity function $\lambda^*(t\big|\mathcal{H}_t)$. Within a short time window $[t, t + dt)$, $\lambda^*(t)$ represents the occurrence rate of a new event given history $\mathcal{H}_t$ that

$$
\lambda^*(t\big|\mathcal{H}_t) = \frac{p(\mathrm{N}(t + \mathrm{d}t) - \mathrm{N}(t) = 1|\mathcal{H}_t)}{\mathrm{d}t}, \tag{6}
$$

where $\mathrm{N}(t)$ is the counting process and $*$ reminds it is history dependent. Here we omit $\mathbf{v}$ for its generality.

Using the generalized formulation of conventional attentions, we can deal with the uncertainty over time and are able to predict future user preferences at time t by

$$
\widetilde{\mathrm{Att}}(\mathbf{q}, |\mathcal{H}_t) = \mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\mathbf{v}\lambda^*(t\big|\mathcal{H}_t, \mathbf{v}), \tag{7}
$$

or future user preferences during time interval $(t_j, t]$ by

$$
\widetilde{\mathrm{Att}}(\mathbf{q}, |\mathcal{H}_t) = \mathbb{E}_{p(\mathbf{v}|\mathcal{H}_t)}\mathbf{v}\int_{t_j}^{t} \lambda^*(s\big|\mathcal{H}_t, \mathbf{v})\,\mathrm{d}s, \tag{8}
$$

where the integral is challenging to compute and proper approximations are needed which will be discussed later.

Remark. We here discuss the characteristics of our proposed self-modulating attention: (i) the conditional intensity function $\lambda^*(t\big|\mathcal{H}_t, \mathbf{v})$ models temporal dynamics in an explicit manner, which acts as a role of time-dependent modulating factor, amplifying or attenuating the impact of past behaviors. This helps to derive the explanations behind the model predictions and the results are highlighted in the experiment sections; (ii) The intensities equal to zero can help to detect the sparse structures in the correlations between past behaviors and future actions, while the intensities less than one, can help to reduce the contribution of the behaviors when computing the predictions. As shown in Theorem 1, these two properties provide improved model generalization ability.

4.3. Self-modulating Layer

The conditional intensity of a temporal point process is often designed to capture the phenomena of interest, for example Poisson process, Hawkes process (Hawkes, 1971; Mei & Eisner, 2017; Zhou et al., 2013) and Self-correction process (Isham & Westcott, 1979). In our case, we implement the intensity function as a neural layer, named self-modulating layer (SMLayer), which can be stacked on the attentions. By doing so, intermediate representations of the attentions can be reused for computational efficiency, and moreover this makes SMLayer a general building block for the broader applications.

We use $\mathbf{Y} \in \mathbb{R}^{L\times d}$ to denote the embeddings of the input sequence $\mathcal{S} = \{(t_j, i_j)\}_{j=1}^{L}$. As our model still needs to be aware of positional information, we choose to use the positional encoding as adopted in (Vaswani et al., 2017):

$$\mathbf{Z}_{i,j} = \begin{cases} \sin(i/10000^{j/d}), & \text{if } i \text{ is even,} \\ \cos(i/10000^{j-1/d}), & \text{if } i \text{ is odd.} \end{cases} \quad (9)$$

After the initial embedding and positional encoding layer, we will have the input embedding $\mathbf{X}$, which is then passed to the self-attention module to compute the intermediate representations

$$\mathbf{X} = \text{concat}\big(\,[\mathbf{Y}, \mathbf{Z}]\,\big)$$
$$\mathbf{Q} = \mathbf{X}\mathbf{W}^{Q}, \quad \mathbf{K} = \mathbf{X}\mathbf{W}^{K}, \quad \mathbf{V}^{\text{seq}} = \mathbf{X}\mathbf{W}^{V}$$
$$\mathbf{H} = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^{\top}}{\sqrt{d}})\mathbf{V}^{\text{seq}}.$$

where all $\mathbf{W}$ are projection matrices, $\mathbf{Q}, \mathbf{K}, \mathbf{V}^{\text{seq}}$ are separately the query, key and values matrices obtained by different transformations of the input $\mathbf{X}$, and $\mathbf{H}$ is the output representations of conventional attentions, where each row corresponds to a particular user behavior. Note that to ensure causality, we mask the future information.

Before the next step, we first denote the j^{th} row of the self-attention output $\mathbf{H}$ by $\mathbf{h}(t_j)$, which corresponds to the representation of the sub-sequence $\{(i_{j'}, t_{j'})\}_{j'=1}^{j}$ up to the time t_j. To model the continuous time temporal dynamics, the intermediate representation $\mathbf{h}(t_j)$ is fed through item-wise feed-forward neural network

$$\mathbf{g}_k(t) = \sigma\left(\underbrace{\mathbf{W}_k^{G}\mathbf{h}(t_j)}_{\text{Endogenous}} + \underbrace{\mathbf{b}_k^{G}(t - t_j)}_{\text{Exogenous}}\right), \quad (10)$$

where the endogenous correlation with the potential next item k is captured by the term $\mathbf{W}_k^{G}\mathbf{h}(t_j)$, which evolves in the embedding space with respect to its previous behaviors in the sequence and not in a random fashion. In the meantime, $\mathbf{b}_k^{G}(t-t_j)$ are used to capture the exogenous force that reacts on the potential next item k during the time interval $[t_j, t)$. By combining $\mathbf{W}_k^{G}\mathbf{h}(t_j)$ and $\mathbf{b}_k^{G}(t - t_j)$ together, the output representation $\mathbf{g}_k(t)$ is expressive to better characterize both of the sequential and temporal dynamics in the continuous time space.

Lastly, the conditional intensity of each potential next item k can be computed in the following function

$$\lambda^*(t|\mathcal{H}_t, \mathbf{v}_k) = f_k(\mathbf{w}_k^{\top}\mathbf{g}_k(t) + \mu_k), \quad (11)$$

where $\mathbf{w}_k$ is the model parameter that serves for the item k, μ_k signifies the base rate that the item k appears in the future, and f_k is the activation function. We notice that the choice of activation function should consider two critical criteria: i) the intensity should be non-negative and ii) the dynamics on different items evolve at different scales. To account for this, we adopt the softplus function

$$f_k(x) = \phi_k \log\left(1 + \exp(x/\phi_k)\right), \quad (12)$$

where the parameter ϕ_k captures the timescale difference.

Remark. In practice because the number of items is always very large, Eq. (10 - 11) require considerable computational time and memory use, not suitable for large-scale online systems. One of the feasible solutions is to cluster the items into groups using spectral clustering techniques (Shi & Malik, 2000; Chen et al., 2010), and replace the item-wise feed-forward neural networks with group-wise feed-forward neural networks. This makes sense because the items in the same group are densely connected to each other but sparsely connected to the items from different groups.

4.4. Continuous Time Regularization for Preference Learning

In preference learning, the model parameters are usually optimized by minimizing the reconstruction error (Mackey et al., 2011; Lee et al., 2013; Li et al., 2016; Chen et al., 2016). The only source of supervised signal is from the behavior data (i.e., $\mathbf{R}$) that is independent of time. Hence, the intensity function learned in such protocol might probably diverge from the complex continuous-time patterns contained in the data.

To remedy the problem, we propose the continuous time regularization $R(\Theta)$ which maximizes the log-likelihood of the timestamps $\{t_j\}_{j=1}^{L}$ in the given sequence, defined by

$$R(\Theta; u) = \sum_{j=1}^{L} \log \lambda_{k_j}^*(t_j|\mathcal{H}_j^{(u)}) - \int_{t_1}^{t_L} \lambda^*(t_j|\mathcal{H}_j^{(u)})\mathrm{d}t$$

where $\mathcal{H}_j^{(u)} = \{(t_i, i_i)|t_i < t_j\}$ is the subsequence up to time t_j. The first term in the right hand signifies the log-likelihood of the behavior history, where k_j signifies the identity of item i_j or the group it belongs to. The second term represents the log-survival probabilities where $\lambda^*(t_j|\mathcal{H}_{t_j}^{(u)}) = \sum_k \lambda_k^*(t_j|\mathcal{H}_j^{(u)})$. Concretely, if we are given the sequences of m users, the model parameters can be optimized by minimizing the following objective function

$$\min_{\Theta} \ell(\mathbf{R}, \hat{\mathbf{R}}) - \gamma\mathbb{E}_{u\in[1,m]}\, R(\Theta; u), \quad (13)$$

Table 1. Statistics of the public datasets for evaluation.

Dataset	#Users	#Items	#Interactions
Amazon	211,384	18,490	1.6M
Koubei	212,831	10,213	1.8M
Tmall	320,497	21,876	7,6M

where γ is the regularization parameter.

The challenge in optimizing Eq. (13) is to compute the integral $\Lambda = \int_{t_1}^{t_L} \lambda^*(t_j|\mathcal{H}_j^{(u)})\mathrm{d}t$. Due to the softplus used in Eq. (11), there is no closed-form solution for this integral.

We consider two techniques to approximate the integral Λ.

1) Monte Carlo integration (Metropolis & Ulam, 1949; Robert & Casella, 2013). It randomly draws a set of samples $\{v_i\}_{i=1}^{N}$ in each interval (t_{j-1}, t_j) to yield an unbiased estimation of Λ, i.e., $\mathbb{E}\,\widetilde{\Lambda}_{\mathrm{MC}} = \Lambda$

$$\widetilde{\Lambda}_{\mathrm{MC}} = \sum_{j=2}^{L}(t_j - t_{j-1})\Big(\frac{1}{N}\sum_{i=1}^{N}\lambda^*(v_i|\mathcal{H}_j^{(u)})\Big); \qquad (14)$$

2) Numerical integration method (Stoer & Bulirsch, 2013). It is usually biased but fast due to the elimination of sampling. For example, trapezoidal rule approximates the integral using linear functions

$$\widetilde{\Lambda}_{\mathrm{NU}} = \sum_{j=2}^{L}\frac{t_j - t_{j-1}}{2}\Big(\lambda^*(t_j|\mathcal{H}_j^{(u)}) + \lambda^*(t_{j-1}|\mathcal{H}_{j-1}^{(u)})\Big). \qquad (15)$$

In our experiments, we find that the approximation defined in Eq. (15) performs comparable to Eq. (14) while consuming less training time. Hence we choose the numerical integration in our main approach. We conjecture the reason is that the softplus function is highly smoothed and the bias introduced by linear approximations is relatively small.

5. Experiments

This section evaluates the quantitative and qualitative results of our self-modulating attention against many state-of-the-art baselines on three large-scale real-world datasets. All experiments are run on a machine with E5-2678 CPU, RTX 2080 and 188G RAM.

5.1. Benchmark Datasets

We use three real-world datasets which are processed in line with (Weimer et al., 2007; Liang et al., 2018; Steck, 2019): (1) Amazon Electronics[1] is a user review dataset, where we binarize the explicit data and only keep users and items who have at least 5 historical records; (2) Koubei[2] is a user behavior dataset, where we keep users with at least 5 records and items that have been purchased by at least 100 users; and (3) Tmall[3] is a user click dataset, where we keep users who click at least 10 items and items which have been seen by at least 200 users. Table 1 shows the statistics.

5.2. Evaluation Protocol

We study the performance using strong generalization protocol (Weimer et al., 2007; Liang et al., 2018; Steck, 2019), where the models are evaluated on the users that are not present at training time. To do so, we split the users into training/validation/test sets with the ratio $8 : 1 : 1$. Then, we use all the data from the training users to optimize the model parameters, whereas for validation/test users, we keep the last record of each user to test and the rest to learn necessary representations for the model. In addition, we select model hyper-parameters and architectures using the validation users, and we report the results on test users for the model that achieves the best results on the validation users.

Two ranking-based metrics are used: Hit Rate (HR) and normalized discounted cumulative gain (NDCG). Notably, in our experiment protocol, the metrics Precision and Recall have the same tendency with Hit Rate. In the following, we report HR@10 and NDCG@10 for all the models.

5.3. Comparing Methods

We compare our proposed attention to state-of-the-art baselines, where we select the continuous time regularization parameter over {1e-4, 1e-5, 1e-6}. For the sake of fairness, the embedding size of all attentions are set to 50. In addition, we summarize the implementation details of the compared baselines as follows:

- **GRU4REC** (Hidasi et al., 2016) is one of state-the-of-art sequential recommendation approaches, which relies on GRU module (Chung et al., 2014) to capture the sequential patterns in user preferences. We use the program provided by the author[4]. Grid search of factor size over {50, 100, ..., 300}, learning rate over {1e-4, 1e-3, 1e-2} and dropout rate over {0.1, ... ,0.5} is performed to select the best parameters.
- **DIN** (Zhou et al., 2018) is among the best sequential recommendation models that build upon attentions. The implementation is public and provided by the authors[5]. We finetune the model by grid search learning rate and dropout rate over {0.5, 1.0, 1.5, 2.0} and {0.1, ... ,0.5}.
- **SHAN** (Ying et al., 2018) proposes hierarchical attentions

[1] http://jmcauley.ucsd.edu/data/amazon/

[2] https://tianchi.aliyun.com/dataset/dataDetail?dataId=53

[3] https://tianchi.aliyun.com/dataset/dataDetail?dataId=35680

[4] https://github.com/hidasib/GRU4Rec

[5] https://github.com/zhougr1993/DeepInterestNetwork

Table 2. Ablation study on the Amazon, Koubei and Tmall datasets. The proposed self-modulating layer (SMLayer) and continuous-time regularization (CTReg) are adapted to attention-based DIN (Zhou et al., 2018) and SASREC (Kang & McAuley, 2018) models. The performance is evaluated in terms of HR@10 and NDCG@10.

		DIN			**SASREC**		
	Dataset	Origin	+SMLayer	+CTReg	Origin	+SMLayer	+CTReg
HR	**Amazon**	0.21955	**0.22065**	0.21985	0.25595	**0.26545**	0.26058
	Koubei	0.32665	**0.33940**	0.33780	0.35455	0.36194	**0.36235**
	Tmall	0.48460	0.49033	**0.49157**	0.50433	0.51218	**0.51347**
NDCG	**Amazon**	0.13443	**0.13383**	0.13296	0.16131	0.16475	**0.16529**
	Koubei	0.24186	**0.25444**	0.25411	0.27070	0.27862	**0.28083**
	Tmall	0.33855	0.34580	**0.35062**	0.34326	0.35214	**0.35811**

Table 3. Performance comparison between the baselines and our proposed method on the Amazon, Koubei and Tmall datasets in terms of HR@10 and NDCG@10. Boldfaces mean that the method performs statistically significantly better under t-tests, at the level of 95% confidence level. We emphasize the comparison against SASREC+, a variant of SASREC equipped with functional time embedding (Xu et al., 2019) which captures continuous-time temporal dynamics.

	Amazon		**Koubei**		**Tmall**	
Model	HR	NDCG	HR	NDCG	HR	NDCG
SHAN (Ying et al., 2018)	0.19250	0.11724	0.28150	0.20256	0.37316	0.25840
DIN (Zhou et al., 2018)	0.21955	0.13443	0.32665	0.24186	0.48460	0.33855
GRU4REC (Hidasi et al., 2016)	0.24380	0.15822	0.32655	0.27052	0.46877	0.33746
SASREC (Kang & McAuley, 2018)	0.25595	0.16131	0.35455	0.27070	0.50433	0.34326
SASREC+ (Xu et al., 2019)	0.25820	0.16204	0.35690	0.27148	0.50607	0.34328
SASREC w/ ours.	**0.26545**	**0.16529**	**0.36235**	**0.28083**	**0.51347**	**0.35811**

to model multi-level user representations. The source code is also provided by the authors[6]. After grid search, we find the parameter settings in the original paper provide the best results in our experiments.

- **SASREC** (Kang & McAuley, 2018) applies the Transformer (Vaswani et al., 2017) into the sequential recommendation task and has achieved state-of-the-art performance in many benchmark datasets. We use the software provided by the authors in our experiments[7]. We adopt the default architecture used in its original paper where three transformer blocks are used. We further search the optimal hyper-parameters by ranging learning rate over {1e-4, 5e-4, ..., 1e-2} and dropout rate from 0.1 to 0.5.

- **SASREC+** (Xu et al., 2019) equips SASREC with time embedding[8], analogous to positional embedding. This approach models continuous-time dynamics in the form of random Fourier features. In our experiments, we concatenate both time and positional embedding with the attention input embedding for the improved model accuracy. We emphasize the comparison to SASREC+, since it is probably most closely related to our work.

5.4. Ablation Study

We evaluate each of the two components of self-modulating attention in continuous time space, namely continuous time regularization (CTReg) and self-modulating layer (SMLayer). Ideally, our attention can be used to improve any attention mechanisms by simply stacking SMLayer and adjusting loss function with CTReg. To demonstrate the adaptability of our proposed attention, we equip DIN and SASREC progressively with SMLayer and CTReg.

Table 3 reports the results of HR@10 and NDCG@10 on the Amazon, Koubei and Tmall datasets. We can see that SMLayer achieves the consistent improvement in both HR@10 and NDCG@10 over both DIN and SASREC across all the three datasets. Especially, SMLayer helps to improve HR@10 and NDCG@10 of DIN by a margin greater than

[6] https://github.com/uctoronto/SHAN

[7] https://github.com/kang205/SASRec

[8] https://github.com/StatsDLMathsRecomSys/Self-attention-with-Functional-Time-Representation-Learning

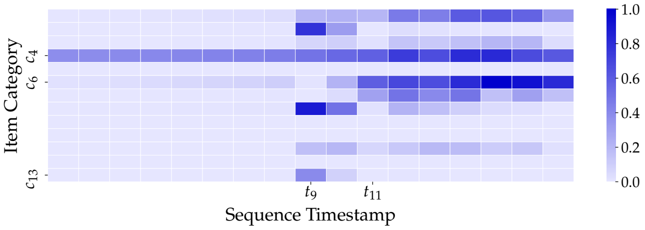

Figure 1. Example of user preference intensities on Koubei. The user usually purchases the items from category c_4 for all the time, and he/she starts to repeatedly buy the items from category c_6 after time t_9. The darker color corresponds to the higher intensity.

0.01. On the other hand, CTReg can also boost the performance in most cases. One exception is on the Amazon dataset. The reason is perhaps due to the fact that Amazon is made of user reviews, and there are nearly 5.39% users wrote the reviews of all the purchased items at the same time. Meanwhile, with using CTReg, the conditional intensity has to be enforced to fit such nuisance patterns. This will inevitably make the optimization procedure prone to overfitting, leading to the degradation of the performance.

The above experiments indicate that both the continuous time self-modulating layer and regularization are flexible and have the ability to improve the model performance. In our analysis, the reasons why our proposed attentions can further improve the recommendation accuracy is mainly due to (1) continuous time self-modulating layer models the impact of both sequential positions and continuous timestamps, so that the model expressiveness is improved; and (2) continuous time regularization enforces the intensity function dependent on the sequence timestamps, so that the intensities can accurately modulate historical behaviors to make attentive contributions to the predictions of the model.

5.5. Quantitative Study

This study evaluates the accuracy of the proposed method by comparing it with five state-of-the-art methods summarized previously, i.e., SHAN (Ying et al., 2018), DIN (Zhou et al., 2018), GRU4REC (Hidasi et al., 2016), SASREC (Kang & McAuley, 2018) and SASREC+ (Xu et al., 2019). Each of the methods is configured by using default parameters provided in the original paper or optimal parameters which produce the best results in grid search. We note again that for the sake of fairness, the embedding size d of query, key, value matrix is set to 50 for all attentions.

Table 3 presents HR@10 and NDCG@10 of all models on the Amazon, Koubei and Tmall datasets. This study shows that the SASREC method with our self-modulating attention outperforms on all the datasets. Note that DIN and SHAN perform worse than the results reported in their origin paper. The difference is caused by the experiment protocol. We evaluate the models on the users that are not present at training time. This setup is more difficult than that where test users and training users are overlapped.

More importantly, we emphasize the comparison between our method and SASREC+ which employs functional time embedding (Xu et al., 2019) to capture temporal dynamics. The empirical results demonstrate the superiority of our self-modulating attention. We believe this is owed to the design of intensity function, powered by neural networks, which is expressive enough to model complex evolving dynamics in the continuous time space.

5.6. Qualitative Study

We also qualitatively analyze the self-modulating attention outputs. Fig. 1 shows the intensities of a user on Koubei evolve alone the time. We note that the reported user usually purchases the items from the category c_4 for all the time, start to repeatedly buy the items from category c_6 after t_9.

At the first glimpse, the intensities are relatively sparse and most of them are less than one. We recall that intensities less than one will attenuate the corresponding past behavior making contributions to the output ranking score, whereas intensities equal to zero can help to derive sparse interactions between the past behaviors and future actions. As demonstrated in Theorem 1, this makes potentials for the improved generalization capacity.

One can see that for category c_4, intensity function assigns larger weights to more recent behaviors. This coincides with a common experience that more recent behaviors have more impact on the future actions. On the other hand, the intensities for category c_{13} remain zero for a long time, then increase to nearly 0.4 during (t_9, t_{11}), and finally diminish back to zero after time t_{11}. This demonstrates the our self-modulating attention takes the timestamps into account and has the ability to cancel the impact of the items that the user might not be interested in during certain time interval.

6. Conclusion

We have proposed a novel attention mechanism in continuous time space, entitled self-modulating attention, to model the non-linearly evolving dynamics, especially for user preferences. The key contribution lies in the generalized formulation of conventional attentions, and the development of self-modulating layer and continuous time regularization. We also provide theoretical generalization bound showing the advantage of our method. Extensive experiments on the top-N recommendation task have demonstrated the effectiveness and flexibility of our proposed self-modulating attention mechanism in continuous time space.

Acknowledgements

This research was supported by Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), NSFC (U19B2035, 61972250, 72061127003), and Meituan.

References

Aalen, O., Borgan, O., and Gjessing, H. *Survival and event history analysis: a process point of view*. Springer Science & Business Media, 2008.

Adomavicius, G. and Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engeering*, 17(6):734–749, 2005.

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020.

Bennett, J. and Lanning, S. The netflix prize. In *KDD Cup*, pp. 35–35, 2007.

Chang, X., Liu, X., Wen, J., Li, S., Fang, Y., Song, L., and Qi, Y. Continuous-time dynamic graph learning via neural interaction processes. In *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM '20)*, pp. 145–154, 2020.

Chen, C., Li, D., Zhao, Y., Lv, Q., and Shang, L. Wemarec: Accurate and scalable recommendation through weighted and ensemble matrix approximation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*, pp. 303–312, 2015.

Chen, C., Li, D., Lv, Q., Yan, J., Chu, S. M., and Shang, L. MPMA: Mixture probabilistic matrix approximation for collaborative filtering. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI '16)*, pp. 1382–1388, 2016.

Chen, C., Li, D., Yan, J., Huang, H., and Yang, X. Scalable and explainable 1-bit matrix completion via graph signal learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI '21)*, 2021a.

Chen, C., Li, D., Yan, J., and Yang, X. Modeling dynamic user preference via dictionary learning for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021b. doi: 10.1109/TKDE.2021.3050407.

Chen, W.-Y., Song, Y., Bai, H., Lin, C.-J., and Chang, E. Y. Parallel spectral clustering in distributed systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):568–586, 2010.

Chen, X., Xu, H., Zhang, Y., Tang, J., Cao, Y., Qin, Z., and Zha, H. Sequential recommendation with user memory networks. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM '18)*, pp. 108–116, 2018.

Choromanski, K., Likhosherstov, V., Dohan, D., Song, X., Gane, A., Sarlos, T., Hawkins, P., Davis, J., Mohiuddin, A., Kaiser, L., et al. Rethinking attention with performers. In *Proceedings of the International Conference on Learning Representations (ICLR '21)*, 2020.

Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K., and Bengio, Y. Attention-based models for speech recognition. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '15)*, pp. 577–585, 2015.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS Workshop on Deep Learning*, 2014.

Daley, D. J. and Vere-Jones, D. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

Hawkes, A. G. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

He, R. and McAuley, J. Fusing similarity models with markov chains for sparse sequential recommendation. In *Proceedings of the IEEE International Conference on Data Mining (ICDM '16)*, pp. 191–200, 2016.

He, R., Kang, W.-C., and McAuley, J. Translation-based recommendation. In *Proceedings of the ACM Conference on Recommender Systems (RecSys '17)*, pp. 161–169, 2017a.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T.-S. Neural collaborative filtering. In *Proceedings of the International Conference on World Wide Web (WWW '17)*, pp. 173–182, 2017b.

Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. An algorithmic framework for performing collaborative filtering. In *Proceedings of the International ACM SIGIR Conference on Research and D in Information Retrieval (SIGIR '99)*, pp. 230–237, 1999.

Hidasi, B., Karatzoglou, A., Baltrunas, L., and Tikk, D. Session-based recommendations with recurrent neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR '16)*, 2016.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Isham, V. and Westcott, M. A self-correcting point process. *Stochastic Processes and Their Applications*, 8(3):335–347, 1979.

Kang, W.-C. and McAuley, J. Self-attentive sequential recommendation. In *Proceedings of the IEEE International Conference on Data Mining (ICDM '18)*, pp. 197–206. IEEE, 2018.

Kleinbaum, D. G. and Klein, M. *Survival analysis*. Springer, 2010.

Koren, Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08)*, pp. 426–434, 2008.

Koren, Y. Collaborative filtering with temporal dynamics. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*, pp. 447–456, 2009.

Koren, Y., Bell, R., and Volinsky, C. Matrix factorization techniques for recommender systems. *Computer*, 42(8): 30–37, 2009.

Kumar, S., Zhang, X., and Leskovec, J. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '19)*, 2019.

Lee, J., Kim, S., Lebanon, G., and Singer, Y. Local low-rank matrix approximation. In *Proceedings of the International Conference on Machine Learning (ICML '13)*, pp. 82–90, 2013.

Li, D., Chen, C., Lv, Q., Yan, J., Shang, L., and Chu, S. Low-rank matrix approximation with stability. In *Proceedings of the International Conference on Machine Learning (ICML '16)*, pp. 295–303, 2016.

Li, D., Chen, C., Liu, W., Lu, T., Gu, N., and Chu, S. Mixture-rank matrix approximation for collaborative filtering. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '17)*, pp. 477–485, 2017.

Li, D., Chen, C., Lu, T., Chu, S., and Gu, N. Mixture matrix approximation for collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, 33 (06):2640–2653, 2021.

Li, J., Wang, Y., and McAuley, J. Time interval aware self-attention for sequential recommendation. In *Proceedings of the International Conference on Web Search and Data Mining (WSDM '20)*, pp. 322–330, 2020.

Liang, D., Krishnan, R. G., Hoffman, M. D., and Jebara, T. Variational autoencoders for collaborative filtering. In *Proceedings of the Web Conference (WWW '18)*, pp. 689–698, 2018.

Liu, H., Lu, J., Zhao, X., Xu, S., Peng, H., Liu, Y., Zhang, Z., Li, J., Jin, J., Bao, Y., et al. Kalman filtering attention for user behavior modeling in ctr prediction. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS '20)*, 2020.

Lu, Z., Pan, S. J., Li, Y., Jiang, J., and Yang, Q. Collaborative evolution for user profiling in recommender systems. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI '16)*, pp. 3804–3810, 2016.

Ma, C., Kang, P., and Liu, X. Hierarchical gating networks for sequential recommendation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '19)*, pp. 825–833, 2019.

Mackey, L. W., Jordan, M. I., and Talwalkar, A. Divide-and-conquer matrix factorization. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '13)*, pp. 1134–1142, 2011.

Martins, A. and Astudillo, R. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proceedings of the International Conference on Machine Learning (ICML '16)*, pp. 1614–1623, 2016.

Mei, H. and Eisner, J. The neural hawkes process: a neurally self-modulating multivariate point process. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '17)*, pp. 6757–6767, 2017.

Metropolis, N. and Ulam, S. The monte carlo method. *Journal of the American Statistical Association*, 44(247): 335–341, 1949.

Niculae, V. and Blondel, M. A regularized framework for sparse and structured neural attention. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '17)*, pp. 3340–3350, 2017.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '07)*, pp. 1177–1184, 2007.

Robert, C. and Casella, G. *Monte Carlo statistical methods*. Springer Science & Business Media, 2013.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the International Conference on World Wide Web (WWW '01)*, pp. 285–295, 2001.

Shi, J. and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Shukla, S. N. and Marlin, B. M. Multi-time attention networks for irregularly sampled time series. In *Proceedings of the International Conference on Learning Representations (ICLR '21)*, 2021.

Steck, H. Markov random fields for collaborative filtering. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS '19)*, pp. 5474–5485, 2019.

Stoer, J. and Bulirsch, R. *Introduction to numerical analysis*, volume 12. Springer Science & Business Media, 2013.

Su, X. and Khoshgoftaar, T. M. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009.

Trivedi, R., Dai, H., Wang, Y., and Song, L. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *Proceedings of the International Conference on Machine Learning (ICML '17)*, pp. 3462–3471, 2017.

Trivedi, R., Farajtabar, M., Biswal, P., and Zha, H. Dyrep: Learning representations over dynamic graphs. In *Proceedings of the International Conference on Learning Representations (ICLR '19)*, 2019.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '17)*, pp. 6000–6010, 2017.

Wang, P., Guo, J., Lan, Y., Xu, J., Wan, S., and Cheng, X. Learning hierarchical representation model for nextbasket recommendation. In *Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '15)*, pp. 403–412, 2015.

Weimer, M., Karatzoglou, A., Le, Q., and Smola, A. Cofirank-maximum margin matrix factorization for collaborative ranking. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '07)*, pp. 222–230, 2007.

Widmer, G. and Kubat, M. Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23 (1):69–101, 1996.

Wu, C.-Y., Ahmed, A., Beutel, A., Smola, A. J., and Jing, H. Recurrent recommender networks. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM '17)*, pp. 495–503, 2017.

Xiao, S., Yan, J., Yang, X., Zha, H., and Chu, S. Modeling the intensity function of point process via recurrent neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI '17)*, 2017.

Xu, D., Ruan, C., Kumar, S., Korpeoglu, E., and Achan, K. Self-attention with functional time representation learning. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS '19)*, pp. 15915–15925, 2019.

Ying, H., Zhuang, F., Zhang, F., Liu, Y., Xu, G., Xie, X., Xiong, H., and Wu, J. Sequential recommender system based on hierarchical attention networks. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI '18)*, 2018.

Yu, F. X., Suresh, A. T., Choromanski, K., Holtmann-Rice, D., and Kumar, S. Orthogonal random features. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '16)*, pp. 1983–1991, 2016.

Zhang, Q., Lipani, A., Kirnap, O., and Yilmaz, E. Self-attentive hawkes process. In *Proceedings of the International Conference on Machine Learning (ICML '20)*, pp. 11183–11193, 2020.

Zhang, T., Zhao, P., Liu, Y., Sheng, V. S., Xu, J., Wang, D., Liu, G., and Zhou, X. Feature-level deeper self-attention network for sequential recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'19)*, pp. 4320–4326, 2019.

Zhang, Y. and Chen, X. Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, 14(1):1–101, 2020.

Zhou, G., Zhu, X., Song, C., Fan, Y., Zhu, H., Ma, X., Yan, Y., Jin, J., Li, H., and Gai, K. Deep interest network for click-through rate prediction. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '18)*, pp. 1059–1068, 2018.

Zhou, K., Zha, H., and Song, L. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS '13)*, pp. 641–649, 2013.

Zuo, S., Jiang, H., Li, Z., Zhao, T., and Zha, H. Transformer hawkes process. In *Proceedings of the International Conference on Machine Learning (ICML '20)*, pp. 11692–11702, 2020.