A Universal Sets-level Optimization Framework for Next Set Recommendation

Yuli Liu
Qinghai University
Australian National University
Xining 810016, China
liuyuli012@gmail.com

Min Liu
Qinghai University
Intelligent Computing and Application
Laboratory of Qinghai Province
Xining 810016, China
liumin5061@gmail.com

Christian Walder
Google DeepMind
Montreal, Canada
cwalder@google.com

Lexing Xie
Australian National University
Canberra, Australia
lexing.xie@anu.edu.au

Abstract

Next Set Recommendation (NSRec), encompassing related tasks such as next basket recommendation and temporal sets prediction, stands as a trending research topic. Although numerous attempts have been made on this topic, there are certain drawbacks: (*i*) Existing studies are still confined to utilizing objective functions commonly found in Next Item Recommendation (NIRec), such as binary cross entropy and BPR, which are calculated based on individual item comparisons; (*ii*) They place emphasis on building sophisticated learning models to capture intricate dependency relationships across sequential sets, but frequently overlook pivotal dependency in their objective functions; (*iii*) Diversity factor within sequential sets is frequently overlooked. In this research, we endeavor to unveil a universal and Sets-level optimization framework for Next Set Recommendation (SNSRec), offering a holistic fusion of diversity distribution and intricate dependency relationships within temporal sets. To realize this, the following contributions are made: (*i*) We directly model the temporal set in a sequence as a cohesive entity, leveraging the Structured Determinantal Point Process (SDPP), wherein the probabilistic DPP distribution prioritizes collections of structures (sequential sets) instead of individual items; (*ii*) We introduce a co-occurrence representation to discern and acknowledge the importance of different sets; (*iii*) We propose a sets-level optimization criterion, which integrates the diversity distribution and dependency relations across the entire sequence of sets, guiding the model to recommend relevant and diversified set. Extensive experiments on real-world datasets show that our approach consistently outperforms previous methods on both relevance and diversity.

CCS Concepts

• **Information systems → Data stream mining**.

Keywords

SDPPs, Optimization Approach, Next Set Prediction

1 Introduction

Next Item Recommendation (NIRec), also widely recognized as sequential recommendation [1, 28, 48], is a predictive model inferring the next likely item of interest from a user's previous temporal item sequence. Extending this concept, Next Set Recommendation (NSRec) aims to predict a subsequent set of items or products by analyzing previous set sequence. Such mechanisms are applicable in various real-world scenarios, including predicting a user's next purchase based on his/her historical transaction records or recommending a new playlist derived from the analysis of his/her daily music listening patterns over several days. Within this field of study, relevant topics encompass Next Basket Recommendation (NBRec) [44, 53] and Temporal Sets Prediction (TSP) [25, 56]. NBRec is derived from the pattern of consumers repeatedly buying assortments of items across different times. In contrast, temporal sets prediction is a newer and broader task that does not solely focus on item baskets but also extends to predicting set sequences in various domains, such as academic courses over semesters or daily online behaviors (clicking or adding products to shopping carts) [47]. This approach is fundamentally a sequential sets to sequential sets learning problem, which initially does not emphasize the personalization aspect inherent in recommendation tasks [15]. However, current research in the field [54, 55] frequently incorporates user preferences into the equation, thereby aligning certain temporal sets prediction tasks with the scope of NSRec.

Figure 1 clearly illustrates the distinction between NIRec and NSRec, with NSRec being the more complex task as it involves predicting the next set based on set-level temporal sequence, compared to NIRec's prediction of the next item from item-level sequence. The complexity of NSRec is further amplified by the presence of an arbitrary number of items within a temporal set, necessitating
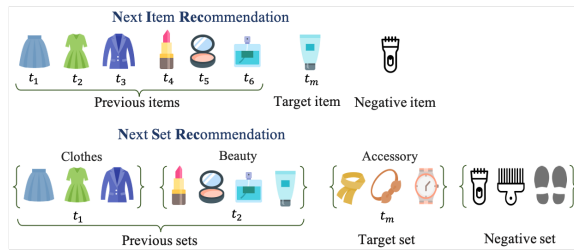
Figure 1: Illustration of next item/set recommendation.

sophisticated models that can discern both dependencies within sets (DinS) and across sets (DacSs). These advanced models typically enhance sequence processing architectures like Recurrent Neural Networks (RNNs) [3, 13, 25] and Transformer [14, 47, 57] to obtain item- and set-level representations. However, it is puzzling that despite the sophisticated model designs, the development of specialized objective functions for NSRec has been overlooked. Instead, there is a tendency to employ objective functions common in NIRec, primarily pointwise loss [10] (*e.g.*, weighted mean square loss [25], binary cross-entropy loss [33, 47, 57], and the multi-label soft margin loss [22, 27, 56]) and pairwise Bayesian Personalized Ranking (BPR) [45, 53], which are all based on individual item comparisons. We argue that these functions may not adequately address the complexities inherent in NSRec. Specifically, individual item-level loss functions presume the estimated items to be mutually independent [36]. For instance, the calculation of pointwise loss is achieved by contrasting the predicted values of items (in target and negative sets) against their true labels, and BPR accumulates the margins between each positive item and its negative counterpart. As Figure 1 shows, such an approach to calculating loss neglects both the DinS within the *Accessory* set and the DacSs across the sequence of *Clothes* → *Beauty* → *Accessory* temporal sets.

In this work, we are committed to the development of an universal optimization framework at the temporal sets level specifically for next set recommendation. Our use of the term 'sets' stems from the intention to capture and represent the dependencies that exist not only within the individual set but also among sets over time. An intuitive approach is to treat the temporal set as a singular element, thus enabling a shift away from individual item comparisons towards direct set entity comparisons. Representing a set, irrespective of the number of items it contains, as a whole inherently accounts for DinS and facilitates the high-level formulation of DacSs. However, three primary challenges arise in the pursuit of this vision:

- Considering a set as an element renders the task of discerning sequence preferences among distinct sets more complex. For example, when faced with set preferences such as {0.3, 0.4, 0.3} and {0.5, 0.2, 0.3}, it becomes a challenge to evaluate which set holds greater importance in the context of preference learning.
- Existing optimization methods in NSRec have centered on item-level relevance comparison, neglecting the diversity metric. Our optimization framework expands this focus to evaluate sets as a whole, thus confronting the dual challenge of modeling set sequences and maintaining their diversity.
- Synthesizing multiple crucial components, such as dependency, relevance, and diversity, into a comprehensive optimization framework presents a formidable challenge, particularly given the intricacies of addressing NSRec.

To address these identified challenges, we make the following key contributions:

- **C1**: Supplementing the sequence preference representation commonly employed in NIRec and NSRec, we introduce an innovative co-occurrence representation. This representation measures the cohesion among items within the temporal set, thereby discerning and validating the differential impact of a specific set.
- **C2**: In line with the objective to conceptualize temporal sets as elements, our optimization framework directly models the structural properties of temporal sets by leveraging the Structured Determinantal Point Process (SDPP), wherein the probabilistic DPP distribution prioritizes collections of structures. A structural similarity measurement approach is proposed for considering diversity in the formulation of optimization criterion.
- **C3**: In response to the complex requirements of NSRec, we make the very first attempt to propose a specialized sets-level optimization criterion, which formulates and captures the DinS and DacSs dependencies. Guided by this criterion, we develop **a universal and Sets-level optimization framework** for Next Set Recommendation (SNSRec), which promotes an informed learning process that synergizes preference and co-occurrence representations, and preserving diversity in the set sequences.
- **C4**: SNSRec stands out for its generality applicability and practicality, allowing mainstream NSRec methods to be deployed within it. This integration, in conjunction with SNSRec's three core components (**C1-C3**), leads to improvements in both accuracy and diversity of NSRec.

2 Preliminary

This section provides some preliminaries, including the formalization of NSRec and DPP.

2.1 Problem Formalization

Next set recommendation can be formalized as follows: Let $\mathbb{U} = \{u_1, \cdots, u_{|U|}\}$ and $\mathbb{V} = \{v_1, \cdots, v_{|V|}\}$ be the entirety of users and items (items or products), respectively. S_t with an arbitrary number of interacted elements is adopted to denote a user interaction set, $S_t \subset \mathbb{V}$. Given a user's historical user-set interactions represented as a temporal sequence of sets $S = \{S_1, S_2, \cdots, S_t\}$, the goal of NSRec is to predict the subsequent set according to the historical records, that is, $\hat{S}_{t+1} = f(S)$.

2.2 Determinantal Point Processes

As elegant probabilistic models, determinantal point processes (DPPs) have been widely employed for diversification [20, 30, 31, 36]. A DPP $\mathcal{P}$ over a ground set $\mathcal{Y} = \{1, 2, \ldots, M\}$, namely the item or product catalog, is a probability measure on $2^{\mathcal{Y}}$ (the set of all subsets of $\mathcal{Y}$). If $Y \subseteq \mathcal{Y}$ is a random subset drawn according to $\mathcal{P}$, the probability is $\mathcal{P}(Y) \propto \det(\mathbf{L}_Y)$, where $\mathbf{L} \in \mathbb{R}^{M \times M}$ is a real, positive semi-definite kernel matrix indexed by the elements of Y. The marginal probability of including one element Y_i is $\mathcal{P}(Y_i) = \mathbf{L}_{ii}$. That is, the diagonal of $\mathbf{L}$ gives the marginal probabilities of inclusion for individual elements of $\mathcal{Y}$. The probability of selecting two items Y_i and Y_j is $\mathbf{L}_{ii}\mathbf{L}_{jj} - \mathbf{L}_{ij}^2 = \mathcal{P}(Y_i)\mathcal{P}(Y_j) - \mathbf{L}_{ij}^2$. The entries of the kernel $\mathbf{L}$ usually measure similarity between pairs of elements in $\mathcal{Y}$. Thus, highly similar elements are unlikely to appear together. In this way, the repulsive characteristic (*i.e.*, diversity) with respect to a similarity measure is captured.

3 Methodology

In this section, we introduce SNSRec by first modeling the temporal set as a structure, and then introduce preference and co-occurrence representations, which are used to measure the structure's quality. Finally, we present our sets-level optimization criterion.

3.1 Structural Temporal Set Modeling

Reviewing the formalization of NSRec, *i.e.*, forecasting the subsequent sets given previous set sequence, the basic component of this prediction task is the set instead of individual items or products, which is also the point that makes this problem apart from NIRec. This implies that standard DPPs are inadequate for the complex relations in set sequence, leading us to consider the structured determinantal point process (SDPP) [30] for its proficiency in modeling sets with inherent structures, such as temporal sets in this work. In SDPP case, elements of a ground set $\mathcal{Y}$ are structures, which means that we will no longer think of $\mathcal{Y} = \{1, 2, \ldots, M\}$; instead, each element $Y_i \in \mathcal{Y}$ is a structure given by a collection of R parts $\left(Y_i^{(1)}, Y_i^{(2)}, \ldots, Y_i^{(R)}\right)$, and each part takes a value from a finite set of M^R possibilities. In the context of NSRec, each structure element is a user interaction set S_t at time t containing R items or products in $\mathbb{V}$. Thus, a temporal set can be explicitly represented as an integrated structure. A random subset drawn from a SDPP $\mathcal{P}$ can be a temporal sequence of sets, *i.e.*, S. In this setting, we can regard NSRec as a problem of drawing salient (accurate and diversified) sets of temporal structures according to the SDPP distribution. SDPP directly treats the temporal set as a united element (DinS) and possesses powerful ability of capturing dependencies across sets (DacSs) incorporating notions of quality and diversity by means of global probabilistic measure, and thus naturally fits the NSRec problem. As a result, introducing SDPP into NSRec will push the corresponding research into a new frontier.

However, we need to note that when the number of elements in $\mathcal{Y}$ is exponential, structured DPPs actually model a distribution over the doubly exponential number of subsets of an exponential $\mathcal{Y}$ [20]. An immediate challenge is that it is difficult to explicitly write down the SDPP kernel $\mathbf{L}$. In this work, we therefore conduct SDPP distribution calculation based on the set sequence specified SDPP kernel $\mathbf{L}^{(S)}$, *i.e.*, a kernel over *sequence ground set* that only involves structures (sets) of a set sequence S instead of over all structures.

3.1.1 Quality Modeling. We use the quality/diversity decomposition to define the entries of SDPP kernel $\mathbf{L}$,

$$L_{ab} = q(S_a)\phi(S_a)^\top \phi(S_b)q(S_b), \tag{1}$$

where $q(S_a)$ is a nonnegative measure of the quality of a structure (set) S_a, and $\phi(S_a)$ represents a vector of diversity features so that $\phi(S_a)^\top \phi(S_b)$ is a measure of the similarity between structures S_a and S_b. SDPPs assume a factorization of the quality score $q(S_a)$ and similarity score $\phi(S_a)^\top \phi(S_b)$ into parts for allowing efficient normalization and sampling. Formally, SDPP decomposes quality multiplicatively and similarity additively:

$$q(S_a) = \prod_{r=1}^{R} q\left(S_a^{(r)}\right) \quad \phi(S_a) = \sum_{r=1}^{R} \phi\left(S_a^{(r)}\right). \tag{2}$$

For temporal sets, the part $S_a^{(r)}$ denotes the rth item in set S_a that contains R items.

To measure the importance or salience of items and the relative strength of relationships between them, we calculate the weight of a temporal set (*i.e.*, quality) for constructing SDPP structure as

$$w(S_a) = \sum_{r=1}^{R} w\left(S_a^{(r)}\right) + \sum_{S_a^{(\alpha)} \in S_a, S_a^{(\beta)} \in S_a} w\left(S_a^{(\alpha)}, S_a^{(\beta)}\right), \tag{3}$$

where $\alpha \neq \beta$. The first term denotes the accumulation of **items' importance**. For NSRec tasks, the importance of part $S_a^{(r)}$, *i.e.* $w\left(S_a^{(r)}\right)$ of an item, can be represented by its predicted relevance score *w.r.t.* the sequence S, which is:

$$\hat{y}_i^P = \mathbf{p}^{(S)\top} \mathbf{e}_i^P, \tag{4}$$

where $\mathbf{p}^{(S)}$ is the final sequence preference representation (underline detailed in Section 3.2.1) of a sequence S, and $\mathbf{e}_i^P$ denotes the preference embedding of item v_i in $\mathbb{V}$ (*i.e.*, the rth part $S_a^{(r)}$ in S_a). The second term in Equation 3 is used to represent the **strength of edges** in S_a. SDPP is originally designed to build structures of threads or paths [20], which means that there is an ordering in the structure. Therefore, only the strength of edges of adjacent elements in the path (structure) is considered. As for NSRec, items in the same set usually do not have an inherent ordering. We therefore propose to capture co-occurrence patterns of items on set level. Specifically, this work considers the cohesion between any two items of a set. In this sense, the edge weight of any two items in a set, *i.e.* $w\left(S_a^{(\alpha)}, S_a^{(\beta)}\right)$, used to represent the strength of cohesion between them,

$$\hat{y}_{\alpha\beta}^C = \mathbf{c}_\alpha^{(S)\top} \mathbf{c}_\beta^{(S)}, \tag{5}$$

where $\mathbf{c}_\alpha^{(S)}$ and $\mathbf{c}_\beta^{(S)}$ are the final co-occurrence representations (underline detailed in Section 3.2.2) of items with respect to temporal sets sequence S, learned based on co-occurrence embeddings.

3.1.2 Diversity Modeling. We have detailed how to calculate the quality of a SDPP structure above. Here we present the approach of structural set similarity measurement ($\phi(S_a)^\top \phi(S_b)$), which contributes to the diversity component of NSRec. Aligning with Equation 1 and Equation 2, the set similarity function can be factorized into the item similarity function as follows (**C2**):

$$Sim(S_a, S_b) = \phi(S_a)^\top \phi(S_b)$$

$$= \left(\sum_{S_a^{(\alpha)} \in S_a} \phi_\alpha\left(S_a^{(\alpha)}\right)^\top\right) \left(\sum_{S_b^{(\beta)} \in S_b} \phi_\beta\left(S_b^{(\beta)}\right)\right) \tag{6}$$

$$= \sum_{\alpha, \beta} \phi_\alpha\left(S_a^{(\alpha)}\right)^\top \phi_\beta\left(S_b^{(\beta)}\right).$$

Here, $\phi_\alpha\left(S_a^{(\alpha)}\right)$ and $\phi_\alpha\left(S_b^{(\beta)}\right)$ represent k-dimensional features of two items belong to S_a and S_b, respectively. This factorization indicates that the diversity measurement of two structures for each SDPP kernel entry is built by accumulating the similarity of any two items from different structures.

To connect the common diversity concept (*i.e.*, scope of categories) in recommendation fields [18, 32, 50, 52] and diversity measurement in SDPP, we propose a category-aware diverse kernel $\mathbf{K}$. The number of items $|\mathbb{V}|$ in NSRec dataset is usually large, which means that learning a nonparametric full-rank diverse kernel $\mathbf{K}$

is computationally expensive. Referring to [19], we use the low-rank factorization of the $N \times N$ **K** matrix as: $\mathbf{K} = \mathbf{A}\mathbf{A}^\top$. In this setting, a learning optimization criterion is built by maximizing the log-likelihood of sampling observed diverse subsets $T^{(+)}$ and minimizing the log-likelihood of sampling negative subsets $T^{(+)}$,

$$\ell = \sum_{(T^{(+)}, T^{(-)}) \in \mathcal{T}} \log \det \left(\mathbf{K}_{T^{(+)}} \right) - \log \det \left(\mathbf{K}_{T^{(-)}} \right), \qquad (7)$$

where $\mathcal{T}$ denotes the collection of paired sets used for training. To obtain the observed diverse subsets, we select as many items belonging to different categories as possible from an observed temporal set. For example, there are four items from three categories (c_1, c_2, c_3) in a set $\{v_5^{c_1}, v_6^{c_2}, v_7^{c_3}, v_8^{c_3}\}$, and two diverse subsets $\{v_5^{c_1}, v_6^{c_2}, v_7^{c_3}\}$ and $\{v_5^{c_1}, v_6^{c_2}, v_8^{c_3}\}$ can be selected. For each selected diverse subset, a same-sized negative subset with randomly selected items will be provided. Considering the dependency among items and computational efficiency, observed temporal sets with item size in the range of 5 to 20 are used for selecting diversified subsets, and a subset should contain at least 3 items. Based on this, a category-aware diverse kernel **K** can be learned, which tends to make the corresponding DPP-distributed samples containing diverse categories. Namely, the items drawn based on pre-learned **K** are diversified. Reviewing the description of DPPs in Section 2.2, the diversity characteristic is contributed by the entries of DPP kernel that measure the similarity between pairs of elements. Since **K** facilitates sampling diversified subsets, we can directly take the corresponding entries of **K** into Equation 6 as the similarity measurement of two items, which means that the corresponding vectors of items in **A** (low-rank factorization) represent their diversity features.

3.2 Representation Learning

As we aim to introduce a universal optimization framework NSRec that optimizes both the novel co-occurrence representations and the common preference representation, we simply introduce the set sequence preference representation and detail exposition of our co-occurrence learning method.

3.2.1 Preference Representation. Learning **pref**erence **r**epresentation (PFR) based on previous set sequence is a critical step to capture relevance between candidate items (that likely appear in next set) and users' dynamic interests. To learn expressive PFR for NSRec, we model preference learning on both item-level and set-level representations. The item-level PFR $\mathbf{h}_v^P$ is learned based on the popular standard multi-head self-attention. Learning set-level PFR means computing a vector representation for each set in the temporal set sequence based on the preference embeddings of items in those sets. The set-level PFR $\mathbf{h}_S^P$ is computed using multi-head attention referring to [47]. To obtain the final sequence preference representation $\mathbf{p}^{(S)} \in \mathbb{R}^d$ *w.r.t.* the temporal sets sequence S for anticipating relevance (**item importance**), we apply a gated fusion module [40, 41, 47] to integrate preference from item level and set level.

3.2.2 Co-occurrence Representation. Temporal set sequence stands apart from other sequence data, as the relationships among items within the set-level element complicate the task of anticipating next set. Previous studies in NSRec focus only on learning PFR of a temporal sequence [25, 44, 56], similar to above $\mathbf{p}^{(S)}$, which is also the common technique of NIRec [28, 36]. Two consequent **limitations** of this approach are evident: (*i*) When we model a set as a

structured element, the relevance of individual items within the set (determined by PFR) is insufficient to distinguish and acknowledge the unique importance of different sets; (*ii*) A crucial factor of the set, *i.e.*, the level of cohesion among the items in it, is neglected, which is a characteristic that is specific to NSRec. For example, in a set where *smartphone* is known with high relevance, and there are other two candidates, *iPad* and *screen protector*, present similar levels of relevance. This similarity makes it challenging to determine which item should be paired with the smartphone to form a set. However, by analyzing existing temporal sets, it could be found that the *screen protector* more frequently co-occurs with the *smartphone* in various baskets or daily activities. This indicates a stronger cohesion within {*smartphone, screen protector*} compared to {*smartphone, iPad*}. Utilizing this insight about cohesive strength enables a more effective differentiation between sets and facilitates completing or complementing an accurate set.

To learn the cohesive strength behind set-level behaviors, we propose a new type of representation (**C1**), *i.e.*, **co-o**ccurrence **r**epresentation (COR), which is learned based on the co-occurrence embedding matrix $\mathbf{E}^C \in \mathbb{R}^{d \times |V|}$ for entire items. Unlike preference representation, COR is not directly associated with personalization, but provides extra evidence in the stage of computing relevance score for a candidate item via estimating the probability of the candidate co-occurring with other ones in the same set. In this work, we propose to learn COR using the item-oriented attention network, which adaptively learns each candidate's representation with respect to sets in the sequence [47, 62]. Co-occurrence embeddings of items in a set sequence are fed into a multi-head self-attention layer to obtain the hidden item representations that consider sequence dependency, *i.e.*, $\mathbf{H}^C = \left[\mathbf{h}_1^C, \cdots, \mathbf{h}_n^C \right] \in \mathbb{R}^{d \times n}$ with n items in a temporal sequence. The co-occurrence representations for all items derived based on the sequence S, from the item-oriented attention layer, can be formulated:

$$\mathbf{C}^{(S)} = \mathbf{W}_I^V \mathbf{H}^C \text{ Softmax} \left(\left(\mathbf{W}_I^K \mathbf{H}^C \right)^\top \mathbf{W}_I^Q \mathbf{E}^C \right), \qquad (8)$$

where $\mathbf{W}_I^Q, \mathbf{W}_I^K, \mathbf{W}_I^V \in \mathbb{R}^{d \times d}$ are learnable matrices; $\mathbf{C}^{(S)} \in \mathbb{R}^{d \times |V|}$ contains sequence S specified item-level co-occurrence representations (final co-occurrence representations) for each candidate in $\mathbb{V}$; and $\mathbf{E}^C$ represents candidates' co-occurrence embeddings.

To introduce COR learning, we can calculate the cohesion strength (**edge strength**) of any two items co-occurring in the same set based on the co-occurrence patterns of sets in training sequences (previous sets and target sets). In this setting, the learning process of COR will be guided to consider the co-occurrence patterns among candidates in entire training sets. In the evaluation procedure, CORs of entire items with respect to a specific set sequence can be obtained from the item-oriented attention network, which will engage in ranking all candidate items for suggesting next sets. The learned CORs are supposed to offer high co-occurrence scores for item pairs that actually co-occur in the subsequent set, and thus provide extra evidence for set recommendation.

3.3 Sets-level Optimization Criterion

In this work, we attempt to capture the dependency in both representation learning and optimization criterion. We therefore define a dedicated sets-level conditional likelihood function by defining
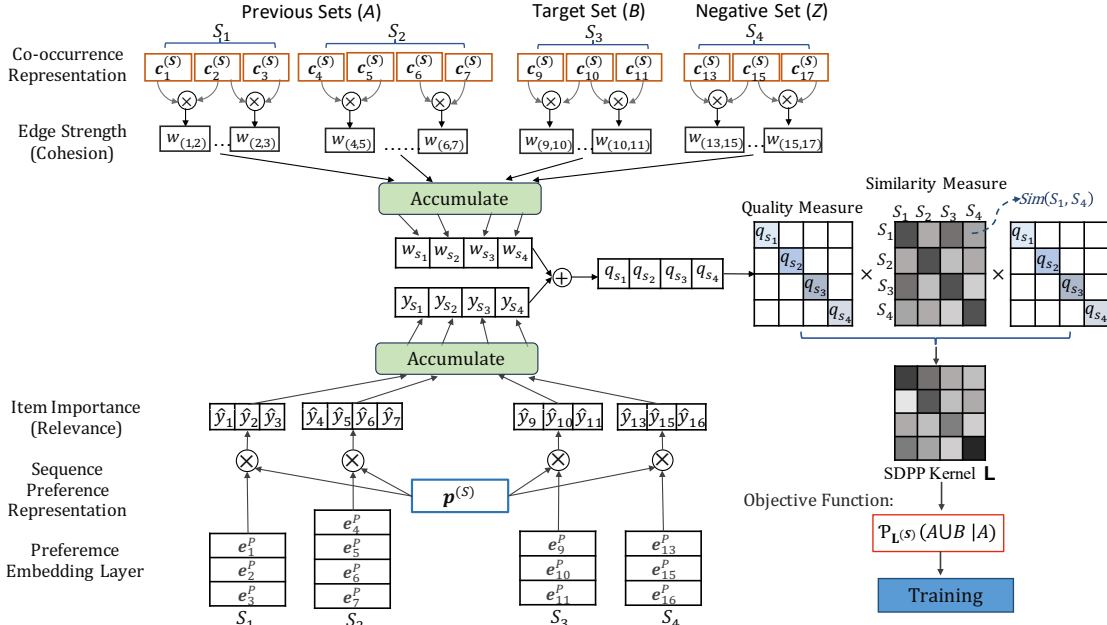
Figure 2: An elaborate depiction for formulating the SNSRec optimization framework.

conditional SDPP probability,

$$\mathcal{P}^{(S)}\left(Y = A \cup B \mid Y = A\right) = \frac{\det\left(\mathbf{L}_{A \cup B}^{(S)}\right)}{\det\left(\mathbf{L}^{(S)} + \mathbf{I}_{\bar{A}}\right)}. \tag{9}$$

In this equation, Y is distributed as a SDPP with temporal sets sequence S specified kernel $\mathbf{L}^{(S)}$; A denotes a subset containing previous sets, and $A \cup B$ represents a subset comprised of sets of the complete training sequence (previous sets and target sets). $\mathcal{P}^{(S)}\left(Y = A \cup B \mid Y = A\right)$ means the probability of if the previous sets is drawn as a SDPP subset of structures, how likely it is that sets of the complete training sequence is SDPP-distributed. This means that the specialized probabilistic function is naturally appropriate to the concept of NSRec, *i.e.*, recommending the next sets conditioned on previous ones. As mentioned in Section 2.2, we cannot afford to construct kernel $\mathbf{L}$ over all possible structures. Fortunately, the training instance in the process of NSRec learning is a definite set sequence. It means that we can form the sequence-specified ground set $\mathcal{Y}^{(S)}$ that only involves the sequence related structures, *i.e.*, previous sets (A), target sets (B), and randomly selected negative sets (Z) that correspond to the target ones. Thus $\mathbf{L}^{(S)}$ represents the sequence-specified kernel on the ground set $\mathcal{Y}^{(S)}$. In the normalization constant, $\mathbf{I}_{\bar{A}}$ is the matrix with ones in the diagonal entries indexed by elements of $\mathcal{Y}^{(S)} - \bar{A}$ and zeros everywhere else.

We then reach the optimization criterion for NSRec training by taking the logarithm of the likelihood function formulated by

$$\mathcal{L}(\Theta) = \prod_{u \in \mathbb{U}} \prod_{S \in \mathcal{D}_u} \mathcal{P}^{(S)} = \sum_{u \in \mathbb{U}} \sum_{S \in \mathcal{D}_u} \log\left(\mathcal{P}^{(S)}\right), \tag{10}$$

where S denotes an observed temporal sets sequence of user u, and $\mathcal{D}_u$ is the compilation of training sequences from u; $\mathcal{P}^{(S)}$ indicates the conditional likelihood (Equation 9) specified by a training sequence of user u. Θ represents model parameters including attention matrices, parameters of gated fusion module, preference embeddings, and co-occurrence embeddings. It can be noted that

DinS and DacSs are both considered in this dedicated optimization criterion, as the corresponding calculation is not solely reliant on independent items or independent sets, but directly treats each temporal set as a SDPP structure and formulates the true set sequence as a SDPP-distributed subset (**C3**). This criterion enables us to transform the goal of NSRec into maximizing the probability of drawing the complete set sequence as a SDPP, given the condition where previous sets are distributed as a SDPP, making the SNSRec framework rather *intuitive*. Looking at the architecture of SNSRec as depicted in Figure 2, it is evident that all preparatory tasks, from learning representations to measuring diversity, are inherently aligned with this intuitive criterion. This suggests that SNSRec is purposefully designed with its goal at the core. Figure 2 also demonstrates that SNSRec is a general framework, as it can be built upon other types of representation forms for constructing SDPP structure, thus reinforcing its adaptability (**C4**).

3.3.1 Optimization. To perform optimization for learning parameters Θ of SNSRec, we can maximize the log-likelihood

$$\mathcal{L}(\Theta) = \sum_{u \in \mathbb{U}} \sum_{S \in \mathcal{D}_u} \left[\log\left(\det\left(\mathbf{L}_{A \cup B}^{(S)}(\Theta)\right)\right) - \log\left(\det\left(\mathbf{L}^{(S)}(\Theta) + \mathbf{I}_{\bar{A}}\right)\right)\right]. \tag{11}$$

Gradient-based learning methods, such as gradient descent and stochastic gradient descent, provide rigorous foundation for optimization because of their theoretical guarantees [2], but require knowledge of the gradient of $\mathcal{L}(\Theta)$. In the discrete DPP setting, this gradient can be computed straightforwardly, and we provide the computation process as follows

$$\frac{\partial \mathcal{L}(\Theta)}{\partial \Theta} = \sum_{u \in \mathbb{U}} \sum_{S \in \mathcal{D}_u} \text{tr}\left(\mathbf{L}_{A \cup B}(\Theta)\right)^{-1} \frac{\partial \mathbf{L}_{A \cup B}(\Theta)}{\partial \Theta}$$
$$- \sum_{u \in \mathbb{U}} \sum_{S \in \mathcal{D}_u} \text{tr}\left(\mathbf{L}(\Theta) + \mathbf{I}_{\bar{A}}\right)^{-1} \frac{\partial \mathbf{L}(\Theta)}{\partial \Theta}. \tag{12}$$

For brevity, we drop the superscript (S) of the sequence specified SDPP kernel. In order to convert the weight function defined in Equation 3 to the multiplicative form, a simple log-linear model is used. Consequently, the entry of SDPP kernel is formulated as

$$L_{ab} = \exp\left(w(S_a)\right) Sim(S_a, S_b) \exp\left(w(S_b)\right). \quad (13)$$

As a structure is regarded as an element in SDPP, we can directly compute the gradient with respect to the quality of an element S_a

$$\frac{\partial L_{ab}}{\partial q_a} = \exp\left(q_a\right) Sim(S_a, S_b) \exp\left(q_b\right). \quad (14)$$

q_a is used to denote the quality of structure S_a.

In Equation 13, we derive similarity measure $Sim(S_a, S_b)$ using diverse kernel **K**, as outlined in Equation 6. This kernel has been pre-learned and does not require further learning during the SNSRec optimization process. Furthermore, the qualities of the structures $(S_a$ and $S_b)$ are computed by accumulating the weights of their elements as shown in Equation 3, a simple process that ensures all components are differentiable and thus allows for the computation of gradients at each location [2]. We can simplify the computation of the gradient of q_a with respect to its parameters within the SDPP kernel by applying the chain rule to decompose the quality weight for calculating gradients of each component [7] . These parameters, which include parameterized preference and co-occurrence representations in Equation 4 and Equation 5, respectively, form the SDPP optimization criterion, denoted as θ_a, and $\frac{\partial q_a}{\partial \theta_a} = \frac{\partial w(S_a)}{\partial \theta_a}$. Substitute decomposed gradients back into the previous gradient formula Equation 12, and we can further receive the final results. That is, the SDPP-based sets-level optimization criterion is differentiable with respect to its parameters. To obtain the gradients of entire model parameters, the attractive backpropagation algorithm [46] can be applied. Finally, the gradient-based algorithms are allowed to be applied for optimization. In experiments, Adam (an variant of stochastic gradient descent) [29] is used.

3.3.2 Prediction. Now we can introduce how to synthetically apply the learned co-occurrence representations $\mathbf{C}^{(S)}$ of entire items and the sequence preference representation $\mathbf{p}^{(S)}$ for predicting the subsequent set of temporal sets sequence S. For an item v_i, the final predicted score for evaluation is

$$\hat{y}_i = (1 - \lambda)\mathbf{p}^{(S)\top}\mathbf{e}_i^P + \lambda \frac{1}{|V|} \sum_{n=1}^{|V|} \mathbf{c}_i^{(S)\top}\mathbf{c}_n^{(S)}. \quad (15)$$

The co-occurrence score of v_i in the subsequent set given previous sequence is measured by averaging the co-occurrence strength between v_i and any one of other items in $\mathbb{V}$. In this setting, we expect the cohesion score for a target item v_t in the subsequent set to be relatively high. This is because CORs are learned to reflect co-occurrence patterns, thereby enhancing the cohesion strength between v_t and other target items of the next set. These cohesion scores are aggregated as shown in Equation 15. Furthermore, we use the parameter λ to balance the contribution of co-occurrence and preference scores when evaluating predictions.

4 Experiments
This section comprehensively evaluates and compares our universal optimization framework, SNSRec.

Table 1: Statistics of the datasets.

Dataset	#Users	#Items	#Sets	#Categories
TaoBao	12.0K	14.2K	85.9K	68
JD-buy	13.5K	19.6K	76.8K	77
JD-add	3.4K	12.1K	16.1K	75
TaFeng	10.3K	8.7K	73.3K	650

4.1 Experimental Settings

4.1.1 Datasets. Experiments are conducted on four real-world datasets from three application scenarios. These datasets are commonly found in two topics (NBRec and TSP) related to NSRec, (*i*) **TaoBao**[1] denotes a public online e-commerce dataset provided by Ant Financial services. It contains the interactions about which items are purchased or clicked by each user with the timestamp. The items purchased in the same day are treated as temporal sets [47, 55] or baskets [8, 45]; (*ii*) **JingDong**[2] is comprised of user action records, including browsing, purchasing, following, commenting and adding to shopping carts, which are widely used in TSP [54, 56] and NBRec [35]. Two datasets are extracted from Jing-Dong, that is, **JD-buy** and **JD-add**, which contain purchasing and adding to shopping carts records respectively. We treat all the items purchased/added in the same day as a set; (*iii*) **TaFeng**[3] dataset contains the transactions about which items are bought by each customer in each basket with the time stamp. This dataset is commonly used in both of NBRec and TSP tasks. We treat products purchased in the same day by the same customer as a set. The statistics of all the datasets are shown in Table 1.

For each user in dataset, we can get a series of training instances. As indicated in Figure 2, a training set instance (*i.e.*, the set sequence specified ground set $\mathcal{Y}^{(S)}$) for the SDPP optimization criterion is comprised of previous sets, target sets, and negative sets, whose sizes (number of sets) are denoted using A, B, and Z, respectively in subsequent discussions. If a temporal set contains only one item, the edge weight is omitted. Following previous TSP or NBRec work [16, 45, 47, 56], each dataset is divided into three parts. The last set, the second last set, and the remaining sets of each user are used for testing, validation, and training, respectively.

4.1.2 Baselines. To validate the superiority of our framework, multiple state-of-the-art models in NBRec and TSP fields built based on different techniques are selected as baselines, which are related to this work and are competitive. **DREAM** [53] is a RNN-based method that uses a pooling layer to represent a basket for next basket recommendation. **CDSL** [36] uses the standard DPP likelihood as the loss function of sequential recommendation. To implement CDSL for NSRec, we treat the preference representation and kernel **K** of SNSRec as the quality and diversity components of DPP in CDSL. **FPMC** [44] designs a hybrid model to take temporal dependencies between items and user's general interests into account. **DIN** [62] proposes a DNN-based method used for click-though rate prediction, which uses attention mechanism to learn the representations of candidate items *w.r.t.* historical behaviors. **Sets2Sets** [25] is the seminal work for TSP. The average pooling operation is used to obtain the representations for sets. **DSNTSP** [47] designs a attention-based network to learn item-level and set-level representations of user sequences for TSP. **ETGNN** [57] is a competitive

TSP work. It uses graph neural network to propagate set-level information for obtaining collaborative signals for temporal sets. **SFCN** [55] anticipates the next set (transactions or daily behaviors) based on specialized neural networks.

4.1.3 Configuration. We implement our optimization framework using PyTorch on a NVIDIA Quadro P2000 GPU. Adam is adopted as the optimizer in our experiments. The number of heads (H) in our representation learning module is set to 4, and 4 trainable queries (K) are used. We tune the hyper parameters in all the compared methods with grid search to choose the best performance for comparison. To keep the comparison fair, the dimension of embeddings in baselines is set to 128, and the PFR and COR are both represented by 64-dimensional vectors.

4.1.4 Evaluation Metrics. To comprehensively evaluate the quality and diversity of SNSRec, three groups of evaluation metrics are adopted: (*i*) **Accuracy**. We evaluate the accuracy of set prediction using Recall@N and NDCG@N. To check the relevance of the predicted set in different ranking positions of the ranking list, we evaluate $N = 20$ and $N = 50$; (*ii*) **Diversity**. The Category Coverage (CC) [42, 52] and Intra-List Distance (ILD) [42, 59] are prevalent and intuitive diversity evaluation metrics, which are appropriate to the diversity concept in recommendation area. (*iii*) **Trade-off**. To evaluate the balance performance between accuracy and diversity, the harmonic metric F-score (F1) [11, 38] is used. F1@$N = 2 \times$ Q@$N \times$ D@N/(Q@N + D@N), where Q@N and D@N are the averaged value of accuracy and diversity metrics of Top-N list.

4.2 Experimental Results

4.2.1 Overall Comparison. The overall comparison is reported in Table 2. The relative improvement (*Improv*) of SNSRec over the highest performance out of baselines is also listed. The main observations obtained from the overall comparison are:

- In most cases, the improvements of quality performance (Recall and NDCG) over the best baselines are greater than 10% (some cases even achieve > 20%, *e.g.*, NDCG@50 on JD-add). The convincing outcomes of our experiments affirm the value of our primary innovations, including direct modelling temporal sets as structures, the formulation of a specialized optimization criterion, and the exploration of DinS and DacSs. These results provide compelling evidence that our proposed SNSRec contributes significantly to improve performances (**C3**) in NSRec.

- SNSRec exhibits superior diversity performance (CC and ILD) compared to the baselines (including CDSL that considers the diversity of individual items). This highlights the effectiveness of utilizing set-level diversity measurements for NSRec tasks (**C2**). Notably, promoting the diversity of the subsequent set holds significant practical value. For instance, in a real-world application scenario such as a retail store, diversifying predictions enables the store to offer bundles with a variety of products, thereby increasing consumers' potential purchasing desire.

- Our method exhibits advantages in both quality and diversity when comparing the overall metric F1. This demonstrates the comprehensive effectiveness of our approach in balancing the relevance and diversity aspects of predictions.

*4.2.2 Generality Study (**C4**).* To validate the universality of SNSRec, we deploy original NSRec models into it. Specifically, we combine
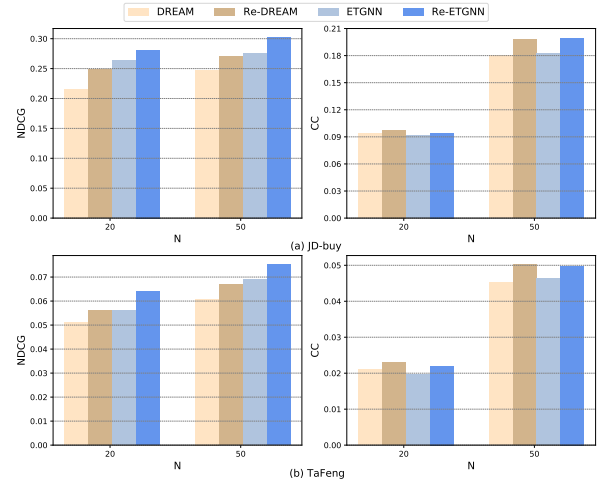

Figure 3: Performance comparison between representative baselines and their SNSRec reworked counterparts.
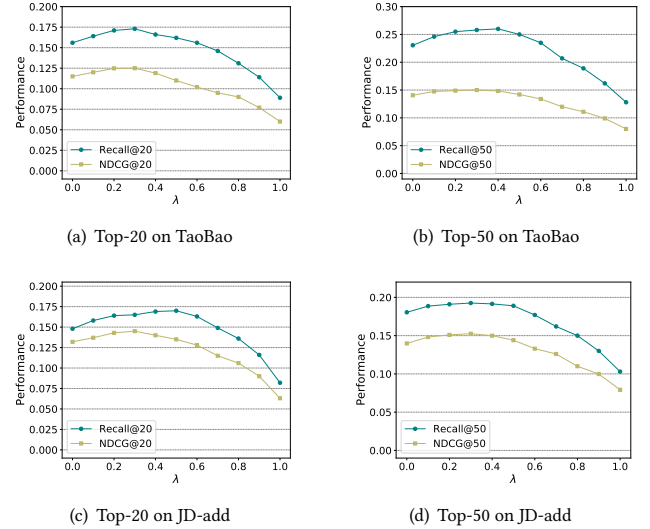

Figure 4: Performance trends with parameter λ.

our co-occurrence learning with an existing NSRec model (*i.e.*, set-level sequence preference representation learning), coupled with the pre-learned diverse kernel to represent sets as SDPP structures, and replace the original model's loss function with our specially designed sets-level optimization criterion, thereby forming a reworked model. We aim to integrate SNSRec with the most popular sequence learning algorithms (attention-based, RNN-based, and neural graph-based methods) to underscore its adaptability and wide applicability in the realm of modern techniques. Given that our preference learning primarily employs attention-based methods with notable effectiveness, we select RNN-based DREAM and graph-based ETGNN as the other two representative models. Figure 3 illustrates the comparison between the original models and and their versions integrated into the SNSRec framework (denoted by 'Re-'). The results indicate a significant enhancement in recommendation performance for both representative models, thereby substantiating the generality and practicality of SNSRec.

Table 2: Overall performance comparisons. The best performing model is boldfaced, while the second best is marked with an asterisk. Improvements over baselines are statistically significant with $p < 0.01$.

Dataset	Metric	DREAM	CDSL	FPMC	DIN	Sets2Sets	ETGNN	DSNTSP	SFCN	SNSRec	*Improv* (%)
	Recall@20	0.1053	0.1306	0.1032	0.1160	0.1286	0.1503	0.1457	0.1544*	**0.1716**	11.14
	Recall@50	0.1526	0.1855	0.1639	0.1795	0.1920	0.2138	0.2046	0.2152*	**0.2549**	18.45
	NDCG@20	0.0885	0.1092	0.0850	0.0915	0.1001	0.1170*	0.1095	0.1084	**0.1247**	6.58
	NDCG@50	0.1033	0.1184	0.1007	0.1090	0.1211	0.1294	0.1363*	0.1343	**0.1490**	9.32
TaoBao	CC@20	0.0905	0.1253*	0.0912	0.1001	0.0986	0.1104	0.1187	0.1126	**0.1360**	8.54
	CC@50	0.1921	0.2201*	0.1938	0.1993	0.1953	0.1870	0.2188	0.1924	**0.2278**	3.50
	ILD@20	0.6613	0.6668*	0.6028	0.6435	0.6492	0.6190	0.6105	0.6400	**0.6957**	4.33
	ILD@50	0.6870	0.7012*	0.6531	0.6720	0.6890	0.6557	0.6617	0.6705	**0.7299**	4.09
	F1@20	0.1541	0.1841	0.1481	0.1622	0.1751	0.1956*	0.1890	0.1948	**0.2185**	11.68
	F1@50	0.1982	0.2285	0.2016	0.2167	0.2312	0.2439	0.2458	0.2487*	**0.2841**	14.21
	Recall@20	0.1047	0.1107	0.0973	0.1027	0.1085	0.1306*	0.1291	0.1250	**0.1640**	25.57
	Recall@50	0.1416	0.1562	0.1332	0.1407	0.1476	0.1734*	0.1616	0.1705	**0.1910**	10.15
	NDCG@20	0.0810	0.0957	0.0737	0.0794	0.0843	0.1027	0.1160*	0.1085	**0.1430**	23.28
	NDCG@50	0.0902	0.1043	0.0850	0.0918	0.0967	0.1165	0.1252*	0.1190	**0.1521**	21.49
JD-add	CC@20	0.1105	0.1118*	0.0937	0.1067	0.0918	0.1085	0.1112	0.1090	**0.1125**	6.30
	CC@50	0.1824	0.1976	0.1780	0.1967	0.1952	0.1812	0.1990*	0.1889	**0.2002**	0.60
	ILD@20	0.6220	0.6384*	0.6152	0.6380	0.6200	0.6214	0.6352	0.6209	**0.6535**	2.37
	ILD@50	0.6434	0.6502	0.6437	0.6513*	0.6503	0.6499	0.6510	0.6431	**0.6918**	6.22
	F1@20	0.1481	0.1619	0.1378	0.1463	0.1517	0.1768	0.1845*	0.1769	**0.2192**	18.78
	F1@50	0.1810	0.1993	0.1724	0.1825	0.1895	0.2149*	0.2144	0.2147	**0.2478**	15.29
	Recall@20	0.3122	0.3106	0.3015	0.3298	0.3158	0.3325	0.3406*	0.3374	**0.3812**	11.92
	Recall@50	0.3460	0.3779	0.3528	0.3829	0.3781	0.4153*	0.4090	0.4146	**0.4634**	11.58
	NDCG@20	0.2157	0.2510	0.2172	0.2453	0.2312	0.2637*	0.2576	0.2619	**0.2804**	6.33
	NDCG@50	0.2465	0.2683	0.2350	0.2627	0.2542	0.2759	0.2825*	0.2702	**0.3080**	9.03
JD-buy	CC@20	0.0942	0.0954*	0.0929	0.0913	0.0950	0.0914	0.0947	0.0910	**0.0974**	2.10
	CC@50	0.1802	0.1936*	0.1783	0.1896	0.1873	0.1826	0.1910	0.1815	**0.1996**	3.10
	ILD@20	0.6411	0.6430	0.6381	0.6443*	0.6276	0.6410	0.6289	0.6340	**0.6712**	4.18
	ILD@50	0.6926	0.6977*	0.6743	0.6831	0.6683	0.6791	0.6705	0.6838	**0.7180**	2.91
	F1@20	0.3073	0.3190	0.3034	0.3228	0.3113	0.3287*	0.3275	0.3281	**0.3555**	8.18
	F1@50	0.3529	0.3746	0.3479	0.3711	0.3636	0.3835	0.3836*	0.3823	**0.4191**	9.25
	Recall@20	0.0593	0.0682	0.0617	0.0652	0.0648	0.0718*	0.0676	0.0703	**0.0814**	13.37
	Recall@50	0.0985	0.1095	0.0979	0.1008	0.1082	0.1103	0.1156*	0.1123	**0.1264**	9.34
	NDCG@20	0.0512	0.0548	0.0516	0.0497	0.0509	0.0562*	0.0531	0.0527	**0.0630**	12.10
	NDCG@50	0.0609	0.0655	0.0617	0.0622	0.0641	0.0691*	0.0680	0.0667	**0.0754**	9.12
TaFeng	CC@20	0.0210	0.0221*	0.0206	0.0220	0.0217	0.0197	0.0213	0.0218	**0.0223**	0.90
	CC@50	0.0452	0.0489	0.0484	0.0497*	0.0475	0.0464	0.0491	0.0420	**0.0500**	0.60
	ILD@20	0.5771	0.5938*	0.5692	0.5837	0.5710	0.5781	0.5812	0.5796	**0.6153**	3.62
	ILD@50	0.6074	0.6126*	0.5934	0.6091	0.5970	0.6098	0.6122	0.6101	**0.6416**	4.73
	F1@20	0.0933	0.1025	0.0950	0.0966	0.0968	0.1054*	0.1006	0.1021	**0.1177**	11.68
	F1@50	0.1281	0.1384	0.1278	0.1307	0.1360	0.1409	0.1437*	0.1404	**0.1562**	8.71

4.2.3 Representations Analysis. We conduct experiments under specialized settings to respectively analyze the importance of preference and co-occurrence representations. Figure 4 shows the trends of quality performance (Recall and NDCG) with parameter λ (introduced in Equation 15) on TaoBao and JD-add. We describe two extreme cases at first, *i.e.*, $\lambda = 0$ and $\lambda = 1$. In the case of $\lambda = 1$, only co-occurrence score is used for calculating prediction score, and thus the learning of preference representation is deprecated from SNSRec. When $\lambda = 0$, the co-occurrence score is not activated, and the learning of co-occurrence representations is not involved in the training process, *i.e.*, the weight of a set (structure) is represented only using relevance. These two extreme settings can serve as a form of ablation study. We aim to use Figure 4 to analyze the following two aspects: (i) **Importance of Co-occurrence**. When λ equals 0 (without using co-occurrence), SNSRec does not reach its full capacity. With the joining of co-occurrence ($\lambda > 0$), impressive improvements are achieved, and SNSRec gradually reaches the peak (when λ is around 0.4) of quality performance. These improvements are obvious evidence of co-occurrence pattern's importance for SNSRec (**C1**). As the involvement of co-occurrence

becomes deeper ($\lambda > 0.4$), the prediction performance declines. This is mainly because that an item's co-occurrence score is determined by calculating its co-occurrence patterns with all other items in $\mathbb{V}$. As the parameter value increases, the co-occurrence score becomes more susceptible to the influence of unrelated items, which in turn, adversely affects the final prediction scores. This adverse effect can potentially overshadow the relevance aspect of the predictions, resulting in a decrease in overall effectiveness. However, it is important to note that we cannot dismiss the role of co-occurrence due to the subsequent decline in performance, as the significant improvement in the earlier stage has already demonstrated the effectiveness and importance of co-occurrence pattern; (ii) **Importance of Preference**. We can treat the performances in the cases of $\lambda = 0$ and $\lambda = 1$ as the capabilities of preference and co-occurrence respectively, as only one of them is involved in SNSRec in the corresponding case. We can see that the preference representations dominate the quality performance of SNSRec, which is a quite common phenomenon in recommendation tasks [26, 28]. This is also the reason why we mention that the co-occurrence pattern is used to complement items for prediction instead of playing a leading role.
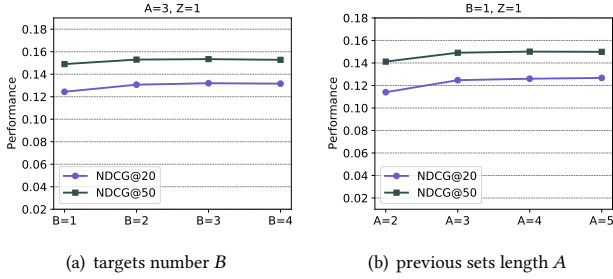
(a) targets number B (b) previous sets length A

Figure 5: Performance trends *w.r.t.* different sizes of $\mathcal{Y}^{(S)}$.

While co-occurrence information provides valuable insights, it is not as influential as preference representations in terms of improving the quality of predictions. This can further explain the reason of subsequent decline, when the λ value becomes excessively large, the co-occurrence component gradually outweighs the preference component, leading to a decline in prediction performance. As the trends on CC with λ is not quite obvious and similar trends can be found on other situations (different datasets and metrics), we omit corresponding presentations to save some space.

4.2.4 Characteristic Analysis. Figure 5 is used to analyze the characteristics of SNSRec, in which NDCG performance trends of Top-20 and Top-50 on TaoBao with varying B and A introduced in Equation 9 are presented. A consistent trend can be observed in both sub-figures of Figure 5, *i.e.*, NDCG performances improve at first and then tend towards stability with the increase of B and A when fixing the other two factors (A and Z in Figure 5(a), and B and Z in the other subfigure). This trend suggests that by considering more target sets or previous sets in our sets-level optimization criterion, more dependencies across the sets sequence can be captured, leading to substantial performance improvements. This effectively validates the importance of sequence dependencies for NSRec and demonstrates that our proposed optimization criterion is effective at capturing these dependencies. To demonstrate the generality and applicability of SNSRec framework, results on different datasets displayed in this section are all obtained with $\lambda = 0.2$, $B = 1$, $A = 3$, and $Z = 1$.

5 RELATED WORK

Temporal Sets Prediction. Unlike widely studied time series forecasting [51, 63] and sequence prediction [6, 12], temporal sets prediction is proposed to untangle complicated relationships among items in the set and dependencies across temporal sets for predicting subsequent sets. Hu et al. [25] first formulate this task and propose a set-to-set method for temporal sets prediction. To capture the dependencies across temporal sets, popular sequence-related models are employed, *e.g.*, multi-heads self-attention [47] and RNN [25]. Recently, some studies have attempted to propagate information between sets to model the complicated relationships. For example, Yu et al. [56] perform graph convolutions on dynamic relationships graph to learn comprehensive set representations. Another study [57] attempts to integrate collaborative signals into temporal sets by propagating element-guided message in a graph format. Although this work claims that the propagation is on set-level, the user-set interaction graph is actually constructed based on interactions between individual items and users instead of directly

via set to set. In addition, previous TSP studies still use the item-level loss function (*e.g.*, weighted mean square loss [25] and binary cross-entropy loss [47, 57]) for training, in which the complicated relationships among sets are ignored.

Next Basket Recommendation. Next Basket Recommendation is a relatively complex task compared to next item recommendation. This research problem originates from real-world shopping scenarios where users purchase a batch of items at a time. Initial studies primarily employed traditional methods such as markov chains [44] and hierarchical representation model [49]. In recent years, sequence learning approaches based on deep learning have become mainstream, including the application and expansion of RNN-based [53], attention-based [5, 60], neural graph-based [39], and hybrid [34, 60] methods, which have further enhanced the effectiveness of NBRec. Additionally, the integration of contrastive learning in basket recommendation has recently garnered attention [24, 43].

DPP for Diversification. Determinantal point process has been a trending research topic in machine learning area, largely because of the repulsive property and efficient algorithms. Applying the maximum a posteriori (MAP) to generate DPP samples as suggestions is the most common and direct manner [17, 23]. However, MAP is NP-hard and is computationally expensive even using the popular greedy algorithm. Chen et al. [9] develop a novel algorithm to accelerate the MAP inference for DPP, which contributes to a flourishing phenomenon of employing DPP MAP in traditional Top-N recommendation models for generating diverse results, such as the studies [38, 52]. Besides using MAP inference, normalized probability and approximate likelihood of DPP distribution are also studied for working as an optimization criterion for item recommendation [36, 37] and basket completion [50], respectively. There also exist multiple models considering using DPP for diversification of sequence data, *e.g.*, addressing the bundle list recommendation problem by the sequence generation approach [4], considering video summarization as a supervised DPP subset selection problem [61], and learning a diversity sampling function to generate a diverse yet likely set of future trajectories [58].

6 Acknowledgments

7 CONCLUSION

In this work, we focus on addressing the problem of next set recommendation via untangling naturally complicated relationships based on means of SDPP. Two types of dependencies (*i.e.*, within a set DinS and across sequence sets DacSs) are formulated in the specialized sets-level optimization framework. To effectively achieve this, we try to explicitly formulate temporal sets as SDPP structures by introducing a novel co-occurrence representation and a pre-learned diversity kernel. The integrated and sets-level optimization criterion contribute to the construction of the general SNSRec framework and to impressive performance on both accuracy and diversity. It would be interesting to extend our framework to other research problems that are constrained in terms of performance due to the structural complexity, *e.g.*, video summarization [21] and bundle list recommendation [4]. The CORs show impressive potential in our experiments, and it is worth the effort to explore further formulation and exploitation of COR for set-level prediction.

References

[1] Muhammad Adnan, Yassaman Ebrahimzadeh Maboud, Divya Mahajan, and Prashant J Nair. 2021. Accelerating recommendation system training by leveraging popular choices. Proceedings of the VLDB Endowment 15, 1 (2021), 127–140.

[2] Raja Hafiz Affandi, Emily Fox, Ryan Adams, and Ben Taskar. 2014. Learning the parameters of determinantal point process kernels. In International Conference on Machine Learning. PMLR, 1224–1232.

[3] Mozhdeh Ariannezhad, Sami Jullien, Ming Li, Min Fang, Sebastian Schelter, and Maarten de Rijke. 2022. ReCANet: A repeat consumption-aware neural network for next basket recommendation in grocery shopping. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. 1240–1250.

[4] Jinze Bai, Chang Zhou, Junshuai Song, Xiaoru Qu, Weiting An, Zhao Li, and Jun Gao. 2019. Personalized bundle list recommendation. In The World Wide Web Conference. 60–71.

[5] Ting Bai, Jian-Yun Nie, Wayne Xin Zhao, Yutao Zhu, Pan Du, and Ji-Rong Wen. 2018. An attribute-aware neural attentive model for next basket recommendation. In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 1201–1204.

[6] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. Advances in neural information processing systems 28 (2015).

[7] Wei-Lun Chao, Boqing Gong, Kristen Grauman, and Fei Sha. 2015. Large-Margin Determinantal Point Processes.. In UAI. 191–200.

[8] Binbin Che, Pengpeng Zhao, Junhua Fang, Lei Zhao, Victor S Sheng, and Zhiming Cui. 2019. Inter-basket and intra-basket adaptive attention network for next basket recommendation. IEEE Access 7 (2019), 80644–80650.

[9] Laming Chen, Guoxin Zhang, and Eric Zhou. 2018. Fast greedy map inference for determinantal point process to improve recommendation diversity. Advances in Neural Information Processing Systems 31 (2018).

[10] Zixuan Chen, Panagiotis Manolios, and Mirek Riedewald. 2023. Why not yet: Fixing a top-k ranking that is not fair to individuals. Proceedings of the VLDB Endowment 16, 9 (2023), 2377–2390.

[11] Peizhe Cheng, Shuaiqiang Wang, Jun Ma, Jiankai Sun, and Hui Xiong. 2017. Learning to recommend accurate and diverse items. In Proceedings of the 26th international conference on World Wide Web. 183–192.

[12] Chung-Cheng Chiu, Tara N Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J Weiss, Kanishka Rao, Ekaterina Gonina, et al. 2018. State-of-the-art speech recognition with sequence-to-sequence models. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 4774–4778.

[13] Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F Stewart, and Jimeng Sun. 2016. Doctor ai: Predicting clinical events via recurrent neural networks. In Machine learning for healthcare conference. PMLR, 301–318.

[14] Yu Hao Chou and Pu Jen Cheng. 2023. Incorporating Co-purchase Correlation for Next-basket Recommendation. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 3823–3827.

[15] Wei Chu and Seung-Taek Park. 2009. Personalized recommendation on dynamic content using predictive bilinear models. In Proceedings of the 18th international conference on World wide web. 691–700.

[16] Mohammed A Fouad, Wedad Hussein, Sherine Rady, S Yu Philip, and Tarek F Gharib. 2022. An efficient approach for rational next-basket recommendation. IEEE Access 10 (2022), 75657–75671.

[17] Lu Gan, Diana Nurbakova, Léa Laporte, and Sylvie Calabretto. 2020. Enhancing recommendation diversity using determinantal point processes on knowledge graphs. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2001–2004.

[18] Mike Gartrell, Ulrich Paquet, and Noam Koenigstein. 2016. Bayesian low-rank determinantal point processes. In Proceedings of the 10th ACM Conference on Recommender Systems. 349–356.

[19] Mike Gartrell, Ulrich Paquet, and Noam Koenigstein. 2017. Low-rank factorization of determinantal point processes. In Thirty-First AAAI Conference on Artificial Intelligence.

[20] Jennifer Gillenwater, Alex Kulesza, and Ben Taskar. 2012. Discovering diverse and salient threads in document collections. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. 710–720.

[21] Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. 2014. Diverse sequential subset selection for supervised video summarization. Advances in neural information processing systems 27 (2014).

[22] Saket Gurukar, Nikil Pancha, Andrew Zhai, Eric Kim, Samson Hu, Srinivasan Parthasarathy, Charles Rosenberg, and Jure Leskovec. 2022. MultiBiSage: A Web-Scale Recommendation System Using Multiple Bipartite Graphs at Pinterest. Proceedings of the VLDB Endowment 16, 4 (2022), 781–789.

[23] Insu Han, Prabhanjan Kambadur, Kyoungsoo Park, and Jinwoo Shin. 2017. Faster greedy MAP inference for determinantal point processes. In International Conference on Machine Learning. PMLR, 1384–1393.

[24] Xinrui He, Tianxin Wei, and Jingrui He. 2023. Robust Basket Recommendation via Noise-tolerated Graph Contrastive Learning. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 709–719.

[25] Haoji Hu and Xiangnan He. 2019. Sets2sets: Learning from sequential sets with neural networks. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1491–1499.

[26] Jin Huang, Wayne Xin Zhao, Hongjian Dou, Ji-Rong Wen, and Edward Y Chang. 2018. Improving sequential recommendation with knowledge-enhanced memory networks. In The 41st international ACM SIGIR conference on research & development in information retrieval. 505–514.

[27] Seungjae Jung, Young-Jin Park, Jisu Jeong, Kyung-Min Kim, Hiun Kim, Minkyu Kim, and Hanock Kwak. 2021. Global-local item embedding for temporal set prediction. In Fifteenth ACM Conference on Recommender Systems. 674–679.

[28] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In 2018 IEEE international conference on data mining (ICDM). IEEE, 197–206.

[29] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).

[30] Alex Kulesza and Ben Taskar. 2010. Structured determinantal point processes. Advances in neural information processing systems 23 (2010).

[31] Alex Kulesza, Ben Taskar, et al. 2012. Determinantal point processes for machine learning. Foundations and Trends® in Machine Learning 5, 2–3 (2012), 123–286.

[32] Yile Liang and Tieyun Qian. 2021. Recommending accurate and diverse items using bilateral branch network. arXiv preprint arXiv:2101.00781 (2021).

[33] Hao Liu, Jindong Han, Yanjie Fu, Jingbo Zhou, Xinjiang Lu, and Hui Xiong. 2020. Multi-modal transportation recommendation with unified route representation learning. Proceedings of the VLDB Endowment 14, 3 (2020), 342–350.

[34] Tong Liu and Bojun Liu. 2022. Next basket recommendation based on graph attention network and transformer. In Journal of Physics: Conference Series, Vol. 2303. IOP Publishing, 012023.

[35] Tong Liu, Xianrui Yin, and Weijian Ni. 2020. Next basket recommendation model based on attribute-aware multi-level attention. IEEE Access 8 (2020), 153872–153880.

[36] Yuli Liu, Christian Walder, and Lexing Xie. 2022. Determinantal point process likelihoods for sequential recommendation. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. 1653–1663.

[37] Yuli Liu, Christian Walder, and Lexing Xie. 2024. Learning k-Determinantal Point Processes for Personalized Ranking. In 2024 IEEE 40th International Conference on Data Engineering (ICDE). IEEE, 1036–1049.

[38] Yong Liu, Yingtai Xiao, Qiong Wu, Chunyan Miao, Juyong Zhang, Binqiang Zhao, and Haihong Tang. 2020. Diversified interactive recommendation with implicit feedback. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34. 4932–4939.

[39] Zhiwei Liu, Mengting Wan, Stephen Guo, Kannan Achan, and Philip S Yu. 2020. Basconv: Aggregating heterogeneous interactions for basket recommendation with graph convolutional neural network. In Proceedings of the 2020 SIAM International Conference on Data Mining. SIAM, 64–72.

[40] Fuyu Lv, Taiwei Jin, Changlong Yu, Fei Sun, Quan Lin, Keping Yang, and Wilfred Ng. 2019. SDM: Sequential deep matching model for online large-scale recommender system. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2635–2643.

[41] Zhiqiang Pan, Fei Cai, Yanxiang Ling, and Maarten de Rijke. 2020. An intent-guided collaborative machine for session-based recommendation. In Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 1833–1836.

[42] Shameem A Puthiya Parambath, Nicolas Usunier, and Yves Grandvalet. 2016. A coverage-based approach to recommendation diversity on similarity graph. In Proceedings of the 10th ACM Conference on Recommender Systems. 15–22.

[43] Yuqi Qin, Pengfei Wang, and Chenliang Li. 2021. The world is binary: Contrastive learning for denoising next basket recommendation. In Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval. 859–868.

[44] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In Proceedings of the 19th international conference on World wide web. 811–820.

[45] Aleksey Romanov, Oleg Lashinin, Marina Ananyeva, and Sergey Kolesnikov. 2023. Time-Aware Item Weighting for the Next Basket Recommendations. In Proceedings of the 17th ACM Conference on Recommender Systems. 985–992.

[46] Frank Rosenblatt. 1961. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Technical Report. Cornell Aeronautical Lab Inc Buffalo NY.

[47] Leilei Sun, Yansong Bai, Bowen Du, Chuanren Liu, Hui Xiong, and Weifeng Lv. 2020. Dual sequential network for temporal sets prediction. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 1439–1448.

[48] Jianling Wang, Kaize Ding, Liangjie Hong, Huan Liu, and James Caverlee. 2020. Next-item recommendation with sequential hypergraphs. In Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 1101–1110.

[49] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2015. Learning hierarchical representation model for nextbasket recommendation. In Proceedings of the 38th International ACM SIGIR conference on Research and Development in Information Retrieval. 403–412.

[50] Romain Warlop, Jérémie Mary, and Mike Gartrell. 2019. Tensorized determinantal point processes for recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1605–1615.

[51] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural Information Processing Systems 34 (2021), 22419–22430.

[52] Qiong Wu, Yong Liu, Chunyan Miao, Binqiang Zhao, Yin Zhao, and Lu Guan. 2019. PD-GAN: Adversarial Learning for Personalized Diversity-Promoting Recommendation.. In IJCAI, Vol. 19. 3870–3876.

[53] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 729–732.

[54] Le Yu, Zihang Liu, Leilei Sun, Bowen Du, Chuanren Liu, and Weifeng Lv. 2023. Continuous-Time User Preference Modelling for Temporal Sets Prediction. IEEE Transactions on Knowledge and Data Engineering (2023).

[55] Le Yu, Zihang Liu, Tongyu Zhu, Leilei Sun, Bowen Du, and Weifeng Lv. 2023. Predicting temporal sets with simplified fully connected networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37. 4835–4844.

[56] Le Yu, Leilei Sun, Bowen Du, Chuanren Liu, Hui Xiong, and Weifeng Lv. 2020. Predicting temporal sets with deep neural networks. In Proceedings of the

[57] Le Yu, Guanghui Wu, Leilei Sun, Bowen Du, and Weifeng Lv. 2022. Element-guided Temporal Graph Representation Learning for Temporal Sets Prediction. In Proceedings of the ACM Web Conference 2022. 1902–1913.

[58] Ye Yuan and Kris Kitani. 2019. Diverse trajectory forecasting with determinantal point processes. arXiv preprint arXiv:1907.04967 (2019).

[59] Mi Zhang and Neil Hurley. 2008. Avoiding monotony: improving the diversity of recommendation lists. In Proceedings of the 2008 ACM conference on Recommender systems. 123–130.

[60] Yuanzhe Zhang, Ling Luo, Jianjia Zhang, Qiang Lu, Yang Wang, and Zhiyong Wang. 2020. Correlation-aware next basket recommendation using graph attention networks. In Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 18–22, 2020, Proceedings, Part IV 27. Springer, 746–753.

[61] Jiping Zheng and Ganfeng Lu. 2021. k-SDPP: fixed-size video summarization via sequential determinantal point processes. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence. 774–781.

[62] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. 1059–1068.

[63] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35. 11106–11115.