IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 March 2024, accepted 10 May 2024, date of publication 17 May 2024, date of current version 29 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3402448

RESEARCH ARTICLE

Modeling Long- and Short-Term Project Relationships for Project Management Systems

YONGQIAO ZHANG[1], GUANGYU BAI[2], ZHANCHAO GAO[3], PENGYUAN ZHU[1], AND SHIWEN LI[1]

[1]NARI Technology Company Ltd., Nanjing 211106, China
[2]NARI-TECH Nanjing Control Systems Ltd., Nanjing 210000, China
[3]School of Cyber Security, University of Chinese Academy of Sciences, Beijing 101408, China

Corresponding author: Guangyu Bai (baiguangyu@sgepri.sgcc.com.cn)

This work was supported in part by the Research on the Application of Knowledge-Enhanced Intelligent Analysis Technology in the Project Management of State Grid Corporation under Grant 524608210191.

ABSTRACT In modern power grid project management, project management systems have become indispensable. However, the intricate and mutable nature of relationships among power grid projects presents significant challenges to accurate modeling using traditional approaches. To tackle this problem, we propose a sequential management model that consolidates long-term and short-term potential relationships between power grid projects, aiming to enhance the efficiency of the project management system. Specifically, we develop a project relationship network, which leverages graph convolutional neural networks and attention mechanisms to dynamically capture and integrate project relationship information. This innovative method enables a more refined representation of inter-project relationships within the power grid domain. Furthermore, to account for temporal shifts in project execution, we devise a method incorporating project temporal information to predict project progress. The method employs separate modules for long-term and short-term project execution, allowing us to distinguish between enduring and immediate impacts among power grid projects, thereby enriching the portrayal of project relationships. Experiments on public recommendation system datasets validate the efficacy of our proposed method in the context of power grid project management.

INDEX TERMS Project management, multi-relationships, long-term and short-term relationships.

I. INTRODUCTION

In the digital era, where the Internet reigns supreme, project management systems have ascended as indispensable pillars of technology, playing a crucial role in addressing the "information overload" problem of the big data era effectively. These systems have become an essential component in a wide range of online services, including news, e-commerce, streaming video, and more [2], [5], [21], [23], [35], [36]. Essentially, these systems analyze historical interactions of projects to infer their relationships. However, projects often have multiple types and levels of relationships that can change over time, such as categories, states, and more. Accurately capturing the various relationships of projects and distinguishing between their long-term and short-term relationships is vital, especially in the context of power grid project management.

Various algorithms have been developed to model project relationships in different ways. Collaborative filtering methods [6], [10], [17], [22], [27] predicate relationships by analyzing historical feedback data. These methods primarily capture long-term relationships but ignore sequential features, limiting their ability to model projects' short-term relationships. To capture short-term relationships,

The associate editor coordinating the review of this manuscript and approving it for publication was Anandakumar Haldorai.

sequential methods [11], [31], [35], [37] have been proposed, which model the sequential features using convolutional neural networks or recurrent neural networks.

To capture both long-term and short-term relational dynamics, several methods [2], [24], [33], [34] have been proposed that meticulously demarcate these distinct temporal associations. Zhao et al. [34] ingeniously combine matrix factorization with recurrent neural networks (RNN) to capture the short-term relationship patterns, employing two unique modeling techniques to separately unravel the long-term and short-term dependencies within projects. Concurrently, Yu et al. [33] make improvements upon the conventional LSTM (Long Short-Term Memory) framework to particularly address the modeling of project short-term ties, while resorting to an asymmetric SVD approach [16] to discern the long-term relationships among projects. Attention-based models like DIN [36] and its successor DIEN [35] are capable of discerning intricate project relationships inherent in the current candidate item by gauging the interplay between various projects. Despite this progress, the challenge lies in the fact that project-to-project interaction behaviors typically manifest as implicit feedback within project management systems, rendering the acquisition of labeled data for delineating long-term and short-term relationships a formidable task. Consequently, the independent modeling of these dual facets of project relations suffers from a lack of explicit supervisory signals that could effectively differentiate them.

On the other hand, several studies model relationships by incorporating multiple latent relationship dimensions to more accurately capture the intricacies of complex relationship preferences. Pi et al. [26] decouple the processes of project relationship modeling and click-through rate prediction, thereby introducing the MIMN model. In this approach, two distinct matrices rooted within a memory network architecture are employed to separately house relationship information and the progression of relationship evolution. Li et al. [18] advance the representation of a project using multiple vectors, each encapsulating different facets of its relationships. They specifically design a Multi-Relationships Network with Dynamic Routing (MIND), leveraging the principles of capsule networks [29] to cluster historical behaviors and derive various relationships accordingly. Similarly, Cen et al. [3] propose a controllable multi-relationships sequence recommendation model ComiRec, which captures multiple relationships of projects from their behavior sequences.

The multi-relationships project modeling method can reflect complex relationship preferences. However, a limitation lies in its disregard for temporal sequence data. Project preferences are not merely diverse but also subject to dynamic shifts over time. Long-term and short-term interaction behaviors may have different impacts on a project's current relationships. Therefore, it becomes imperative to discern and isolate the long-term and short-term behavioral patterns within a project, and correspondingly, to model their long-term and short-term relationships with due diligence.

This paper proposes to improve the performance of project management systems by integrating the modeling of multiple relationships along with long-term and short-term relationship preferences. We harness capsule networks [29] to represent multiple relationships, apply attention mechanisms to capture long-term connections and use recurrent neural networks for modeling short-term interactions. Our approach partitions and independently models these long-term and short-term relationships within a multi-relationship framework, thus ensuring a more accurate estimation of project relationship dynamics. We conduct extensive experiments on various datasets, including AMAZON, TAOBAO, and Microsoft's news dataset MIND_NEWS, to demonstrate the effectiveness of our recommendation algorithm that fuses multiple relationships and long-term and short-term relationships.

In summary, the main contributions of this work are as follows.

- We propose a sequential project management model called LSPR-PM that integrates long-term and short-term relationships between projects.
- In our approach, we utilize attention mechanisms to model long-term relationships within projects, while employing recurrent neural networks to model short-term relationships. To enhance the learning process, we ingeniously integrate these long-term and short-term relational attributes of the target items in a manner that is contingent upon their degree of resemblance, thereby fostering an adaptive fusion strategy.
- We conduct experiments on three public recommendation system datasets, demonstrating that LSPR-PM outperforms baseline models and achieves significant performance improvements.

II. RELATED WORK

A. PROJECT MANAGEMENT

Project Management is similar to recommendation systems, which have achieved great commercial success by leveraging project-item feedback information such as click records, purchase records, and other interaction data to estimate project preferences, making them increasingly popular in the era of big data.

Throughout the progression of time, recommendation systems have evolved through diverse phases of innovation, including content-based algorithms, collaborative filtering methodologies, hybrid recommendation strategies, and model-centric approaches. Notably, collaborative filtering, due to its demonstrated potency, has become ubiquitous in recommendation systems as evidenced by seminal works such as Goldberg et al. [6] and Linden et al. [22]. This technique predicates that projects tend to favor items favored by analogous projects, leveraging historical interaction data between projects and items to infer the former's inclinations towards other items. By unraveling the hidden common preferences of projects, recommendation systems can significantly boost their predictive accuracy. Preeminent methods that exemplify this paradigm include Matrix Factorization (MF) [17] and Factorization Machines (FM) [27], which decompose the complex interaction matrices into lower-dimensional latent factors. With the development of deep neural networks, some approaches such as the Neural Factorization Machine (NFM) [9] and Deep Factorization Machine (DeepFM) [8] augment the representational prowess of FM by capturing higher-order, non-linear feature interactions. Building upon this foundation, He et al. [10] make a significant stride with the proposal of Neural Collaborative Filtering (NCF), a framework that leverages deep learning constructs to model the complex interdependencies between projects and items, thereby enriching the field with an advanced technique for recommendation systems.

B. MULTI-RELATIONSHIPS MODELING IN MANAGEMENT

Project multi-relationships modeling constitutes an advanced methodological approach that conceptualizes a project's relationships across multiple latent dimensions, aiming to provide a more nuanced reflection of its multifaceted preference patterns. In traditional recommendation systems, the way projects are related is often shown with a single vector, which might not fully show the many different aspects of these relationships. For example, it might not capture a project's connection to film genres, music styles, or cultural topics. Because of this, researchers are increasingly interested in finding better ways to model the multiple relationships that make up a project, which has become an important area of study in modern recommendation system research.

Nonetheless, the endeavor to model project multi-relationships introduces dual salient challenges to the realm of recommendation systems. Primarily, there exists the quandary of how to effectively mine and embody the various relationships of projects from their sequential interaction histories with items. Secondly, the challenge lies in matching and recommending appropriate items based on the varied relational preferences of a project. To overcome these challenges, researchers develop multi-relationships recommendation models powered by deep learning. For example, Pi et al. and colleagues [26] break away from the usual approach of combining project relationship modeling with click-through rate predictions, instead introducing the MIMN model. In this model, they utilize a memory network architecture that preserves the history of project relationships and their developments in two separate matrices. Li et al. [18] argue that a single representation of projects fails to adequately capture the diverse connections among them. They propose an innovative multi-relationships extraction layer that leverages the dynamic routing mechanism [29] in capsule networks. This mechanism adaptively aggregates a project's historical behavior data into detailed preference representations. Similarly, Cen et al. [3] introduce the ComiRec model, a multi-relationships sequence recommendation framework with controlled labels. This model skillfully identifies and preserves the numerous relationships within a project's behavior sequences, providing a sophisticated and flexible method for generating recommendations.

In conclusion, modeling project multi-relationships can better reflect the project's complex relationship preferences. In the future, with the increase in data volume and the continuous development of recommendation system technology, project multi-relationship modeling will be more widely applied.

C. LONG-TERM AND SHORT-TERM RELATIONSHIPS MODELING IN MANAGEMENT

Project long-term and short-term relationship modeling is based on the project's historical behavior sequence, extracting different relationships of the project, and predicting the preference of candidate items based on the changes and evolution of the relationships. In recommendation systems, project long-term and short-term relationships modeling can improve the accuracy and diversity of recommendations, and meet projects' personalized needs.

To model the project's long-term and short-term relationships, we need to consider the project's behavior sequence, which is the items that the project has recently interacted with. The project's behavior sequence can reflect the changes in the project's relationships over time, where earlier behavior represents the project's long-term relationships, and recent behavior represents the project's short-term relationships. However, traditional Markov chain-based techniques (e.g., [28]) and sophisticated deep learning models (e.g., [11], [14], [19], [20], [25], [30], [31], [35], [37]) often struggle to explicitly differentiate and separately model these temporal relational domains, making a uniform representation insufficient for fully capturing the breadth of a project's relationships. To model the project's long-term and short-term relationships between projects, some methods [2], [7], [12], [24], [33], [34] have been proposed to clearly distinguish between the project's long-term and short-term relationships between projects. For example, Zhao et al. [34] use matrix factorization for long-term relationships and use Recurrent Neural Network (RNN) for short-term relationship modeling. Through two different modeling methods, the project's long-term and short-term relationships between projects are separated. Yu et al. [33] design an LSTM (Long Short-Term Memory) variation to apprehend short-term relationships, complemented by Asymmetric SVD [16] to gauge long-term bonds. Meanwhile, attention-based mechanisms, as exemplified in models such as DIN [36] and DIEN [35], have proven effective in identifying project relationships pertinent to the current candidate item. These models extract and weigh the relevance of each item to the project's historical behavior, thereby constructing a precise representation of the project's relationship preferences.

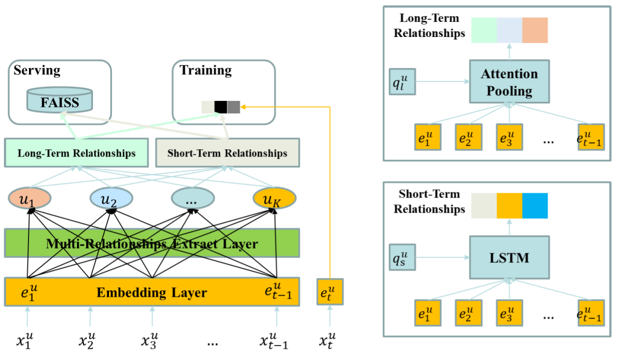

FIGURE 1. **Overall architecture of the proposed method.**

III. PROBLEM FORMALIZATION

In this section, we will introduce the symbols used for formalizing the project of sequential recommendation. Formally, let $u \in \mathcal{U}$ denote a project u from the project set $\mathcal{U}$, and let $i \in \mathcal{I}$ denote an item i from the item set $\mathcal{I}$. For each project, we have a historical sequence of interaction behaviors $(x_1^u, x_2^u, x_3^u, \ldots, x_n^u)$, where the sequence is sorted by time. x_t^u records the t-th item that the project has interacted with in the past. The project of sequence recommendation is to predict the next item that the project may interact with, given the historical sequence of interaction behaviors.

IV. METHOD

In this section, we introduce the LSPR-PM (long-term and short-term Relationships for Power Grid Project Management) model, which ingeniously integrates multi-relationships alongside the long-term and short-term relationships between projects within the power grid landscape. In line with the principles of MIND [18], we employ a capsule network [29] architecture to derive multi-faceted relationship embeddings for power grid projects. Following this, we enhance our framework with a pair of dedicated long-term and short-term relationship encoders that process the obtained multi-relationship representations of the projects. These specialized encoders accurately identify and represent the various relationship patterns of projects across time, differentiating between stable, long-term relationships and temporary, short-term interactions. This comprehensive method provides an exact depiction of the intricate and diverse relationship preferences in power grid projects, thereby improving the overall performance and flexibility of the project management system. The model structure is illustrated in Figure 1.

A. EMBEDDING LAYER

The Embedding Layer is designed to map sparse features to a dense vector representation space. Specifically, for the t-th item x_t^u in the input item sequence, after passing through the Embedding Layer, its corresponding embedding representation e_t^u is obtained.

B. MULTI-RELATIONSHIPS EXTRACTOR LAYER

The Multi relationships Extract Layer is consistent with the baseline model MIND [18]. As shown in Algorithm 1, for an input item sequence $x_1^u, x_2^u, x_3^u, \ldots, x_{t-1}^u$, the sparse feature representation of each item is first passed through the embedding layer to obtain the corresponding dense vector embedding representation $e_1^u, e_2^u, e_3^u, \ldots, e_{t-1}^u$. Then, the dynamic routing algorithm of the capsule network is applied to obtain the project's multi-relationships representation $u_1, u_2, \ldots, u_K$.

Algorithm 1 Dynamic Routing

Input: The embedding representations of items $e_1^u, e_2^u, e_3^u, \ldots, e_{t-1}^u$, the number of iterations r, and the number of relationships K.
Output: Multi-relationships representations $u_1, u_2, \ldots, u_K$.
Initialize behavior capsule i and relationships capsule j, where $b_{ij} \sim N(0, \sigma^2)$.
for k in range(1, r):
  for behavior capsule i, calculate $w_{ij} = softmax(b_{ij}$
  for relationships capsule j, calculate $z_{ij} = \sum w_{ij} e_i^u$, $u_j = squash(z_{ij})$
  update $b_{ij} = b_{ij} + u_j \cdot e_i^u$
return $u_1, u_2, \ldots, u_K$

C. LONG-TERM RELATIONSHIPS REPRESENTATION

As shown in the **Long-Term relationships** module in Figure 1, an attention mechanism is applied to learn the project's long-term relationships. For the multiple project relationships vectors $u_1, u_2, \ldots, u_K$ learned by the Multi relationships Extract Layer, which contains the diverse range of relationships a project maintains across multiple categories, including its long-term relationships, we utilize an attention-based method to isolate these persistent ties. To extract the project's long-term relationships, we apply an attention mechanism. First, the Mean Pooling operation is applied to obtain the long-term relationships query vector $q_l^u = MeanPooling(u_1, u_2, \ldots, u_K)$. This process averages the elements of all relationship vectors, producing a comprehensive overview that highlights the consistent, fundamental connections that define the project's long-term relational characteristics. Then, the similarity between the query vector and the input item embedding representation is calculated:

$$S_i = q_l^u W_Q \cdot e_i^u W^K, \tag{1}$$

where W^Q and W^K are trainable parameters, and then use the softmax function to calculate the weight impact:

$$\alpha_i = softmax(S_i) = \frac{exp(S_i)}{\sum_i (exp(S_i))}. \tag{2}$$

Long-term relationship representation that eventually leads to the fusion of attention mechanisms:

$$u_{long-term} = \sum_{i} \alpha_i e_i^u. \tag{3}$$

D. SHORT-TERM RELATIONSHIPS REPRESENTATION

As shown in the Short-Term Relationships module in Figure 1, a recurrent neural network is applied to capture short-term relationships. For the multi-relationships extraction layer learned by the multi-relationships extraction layer $u_1, u_2, \ldots$ After the u_K is spliced, it is used as the initialization hidden layer of LSTM q_s^u, and then the final hidden layer output is obtained by applying LSTM as a short-term relationships representation of the project:

$$u_{short-term} = LSTM(q_s^u, e_1^u, e_2^u, \ldots, e_{t-1}^u). \tag{4}$$

E. TRAINING AND SERVING

Upon extracting the multi-relationships of a project, the subsequent step involves utilizing both the Long-Term Relationships encoding module and the Short-Term Relationships encoding module. This dual processing results in the generation of the project's long-term relationship representation $u_{long-term}$ and short-term relationship representation $u_{short-term}$.

During the model's training phase, given that the embedding representation of the target item e_t^u, the similarity metrics are computed separately between e_t^u and both $u_{long-term}$ as well as $u_{short-term}$. The outcomes of these calculations enable the construction of the fused long-term and short-term project relationships representation that serves as the basis for model training. This fusion allows the model to understand and incorporate the impact of both enduring and immediate relational contexts when making predictions or recommendations, thereby enhancing its performance and contextual understanding, as shown in the following equation:

$$\begin{aligned} &u_{long-termandshort-term} \\ &= \frac{exp(u_{long-term} \cdot e_t^u)}{exp(u_{long-term} \cdot e_t^u) + exp(u_{short-term} \cdot e_t^u)} \cdot u_{long-term} \\ &\quad + \frac{exp(u_{short-term} \cdot e_t^u)}{exp(u_{long-term} \cdot e_t^u) + exp(u_{short-term} \cdot e_t^u)} \cdot u_{short-term} \end{aligned} \tag{5}$$

Then for predicting the probability of the target item is

$$\begin{aligned} &P(e_t^u | u_{long-termandshort-term}) \\ &\qquad = \sigma(u_{long-termandshort-term} \cdot e_t^u), \end{aligned} \tag{6}$$

where σ is the activation function.

Then the training objective function is.

$$L = \sum logP(e_t^u | u_{long-termandshort-term}). \tag{7}$$

During the inference stage, the learned long-term relationships representation $u_{long-term}$ and short-term relationships representation $u_{short-term}$ are used to search for corresponding items in the vector recall library FAISS,[1] and a recommendation list is generated.

V. EXPERIMENTAL SETUPS

A. DATASETS

For the above-mentioned recommendation algorithm that integrates multi-relationships and long-term and short-term relationships, a large number of experiments have been carried out on the public datasets Amazon,[2] Taobao,[3] and MIND_NEWS,[4] and the statistics of the dataset are shown in Table 1. The Amazon dataset consists of Amazon's product reviews and metadata, and we chose to use the project book purchase data in it, a total of 8,020,294 pieces of data, which contains 543,301 projects, 367,982 book information, and books cover 1,600 categories; Taobao dataset is a Taobao project behavior dataset provided by Alibaba for the study of implicit feedback recommendation problems, which contains all the behaviors of one million random projects (behaviors include clicks, purchases, add-ons, likes), of which the project's purchase behavior records are 2,015,839, covering 672,404 projects, 638,962 products, and 7,097 categories; MIND_NEWS is a large-scale dataset released by Microsoft for news recommendation research, collected from anonymous behavioral logs of Microsoft news websites, containing 2,037,630 browsing records of 91,935 projects, covering 44,908 pieces of news, totaling 17 first-level categories and 248 subcategories.

B. EVALUATION

We use the following metrics to evaluate the performance of our proposed model.

- Recall. The recall measures the proportion of items that have been recommended to a project and that the project has interacted with, out of all the items that the project has interacted with.

$$Recall@N = \frac{1}{|U|} \sum_{u \in U} \frac{\left| \hat{I_{u,N}} \right| \cap I_u}{|I_u|}, \tag{8}$$

  where $\hat{I_{u,N}}$ represents the top-N item set recommended to project u, and I_u is the set of items that project u has interacted with.
- Hit Rate. The hit rate (HR) measures the percentage of recommended item sets that contain at least one item that the project has interacted with [4] and [15].

$$HR@N = \frac{1}{|U|} \sum_{u \in U} \delta(\left| \hat{I_{u,N}} \cap I_u \right| > 0), \tag{9}$$

  where $\delta(\cdot)$ is the indicator function.
- Normalized Discounted Cumulative Gain. The normalized discounted cumulative gain (NDCG) considers the

[1] https://faiss.ai/
[2] http://jmcauley.ucsd.edu/data/amazon/
[3] https://tianchi.aliyun.com/dataset/dataDetail?dataId=649&projectId=1
[4] https://msnews.github.io/

TABLE 1. Dataset statistics.

	#projects	#Items	#Categories	#Samples
AMAZON_BOOK	543,301	367,982	1,600	8,020,294
TAOBAO_BUY	672,404	638,962	7,097	2,015,839
MIND_NEWS	91,935	44,908	17(248 subcategories)	2,037,630

position order of the correctly recommended items in the recommendation list [13].

$$NDCG@N = \frac{1}{Z}DCG@N = \frac{1}{Z}\frac{1}{|U|}\sum_{u\in U}\sum_{k=1}^{N}\frac{\delta(\hat{i_{u,k}}\in I_u)}{log_2(k+1)}, \quad (10)$$

where $\hat{i_{u,k}}$ represents the k-th recommended item to project u, Z is a normalization constant representing the ideal discounted cumulative gain (IDCG@N), which is the maximum possible value of discounted cumulative gain (DCG@N).

C. COMPARING METHODS

- WALS [1]: WALS, short for Weighted Alternating Least Squares, is a classic matrix factorization algorithm that decomposes the project-item interaction matrix into hidden factors for projects and items. Recommendations are made based on the compatibility between the hidden factors of projects and target items.
- YouTube DNN [5]: YouTube DNN is one of the most successful deep-learning methods for industrial recommender systems.
- MaxMF [32]: This method introduces a highly scalable approach for learning non-linear latent factorization to model multiple project relationships.
- MIND [18]: MIND is a Multi-relationships Network with Dynamic Routing designed to model projects' different relationship preferences, with a multi-relationships extraction layer based on a capsule network routing mechanism suitable for clustering project historical behaviors and extracting different relationship preferences.

D. IMPLEMENT DETAILS

The experiments were conducted on a Linux server with an Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processor and three Tesla M40 24GB GPUs. The server runs on CentOS 7.7, with CUDA version 10.2 and Python version 3.9.7. The deep learning framework used is PaddlePaddle 2.2.2.

To train and evaluate the recommendation model, we partitioned the dataset into training, validation, and test sets in a ratio of 8:1:1. The dimensionality of the embedding vectors used by the model is set to 128. We utilized the Adam optimizer with a learning rate of 0.001 and a batch size of 128 for model training, with training epochs set to 100. For model evaluation, we inferred project embeddings from 80% of the project behaviors in the validation and test sets and used these embeddings to predict the remaining 20% of project behaviors. We apply grid search for hyperparameter selection. The parameter K is searched from $\{3, 4, 5, 6\}$, representing the number of vectors used to model the project's relationship preference. The optimal model is selected based on the performance of the validation set, and then we report the results on the test set.

VI. RESULTS

A. OVERALL PERFORMANCE

As shown in Table 2, LSPR-PM represents the proposed model that combines multiple relationships and long short-term relationships, which includes the long-term relationships encoding module and short-term relationships encoding module. LSPR-PM -w/o Short-term removes the short-term relationships encoding module and only models the project's long-term relationships, while LSPR-PM -w/o Long term removes the long-term relationships encoding module and only models the project's short-term relationships.

From the table, we can see that the matrix factorization method WALS is defeated by other methods, thereby illustrating the superior capacity of deep learning algorithms in accurately modeling project relationship preferences and effectively aligning suitable items within recommendation systems. Notably, when deep learning is not in play, MaxMF significantly surpasses WALS, a phenomenon that can be attributed to its ability to generalize the conventional MF approach into a non-linear framework and incorporate multiple project representative vectors. Consequently, it's discernible that methods employing multiple project representation vectors, such as MaxMF, MIND, and LSPR-PM, consistently demonstrate enhanced performance compared to other methodologies like WALS and YouTube DNN.

In addition, our models LSPR-PM, LSPR-PM -w/o Short term, and LSPR-PM -w/o Long term all achieved optimal or suboptimal performance, thus suggesting that through the modeling of multi-relationships within projects, By separating the long-term and short-term relational aspects of projects, we can more accurately track their changing preferences. Incorporating various relationships and considering both long-term and short-term connections enables us to detailedly model the complex interactions among projects, resulting in the most impressive performance results.

B. ABLATION STUDY

As shown in Table 2, we also conducted ablation experiments, where LSPR-PM represents the model that integrates multiple relationships with long-term and short-term relationships between projects, including the long-term relationships encoding module and the short-term relationships encoding module. LSPR-PM -w/o Short-term removes the short-term relationships encoding module and only models the project's

TABLE 2. Overall performance.

		AMAZON_BOOK			TAOBAO_BUY			MIND_NEWS		
		Recall@50	NDCG@50	HR@50	Recall@50	NDCG@50	HR@50	Recall@50	NDCG@50	HR@50
K=3	WALS	0.0397	0.0297	0.0426	0.0876	0.0494	0.0697	0.1073	0.1274	0.3712
	YouTube DNN	0.0483	0.0349	0.0698	0.0955	0.0556	0.0930	0.1195	0.1349	0.4025
	MaxMF	0.0511	0.0387	0.0974	0.1043	0.0599	0.1399	0.1290	0.1537	0.4309
	MIND	0.0600	0.0471	0.1237	0.1134	0.0634	0.1416	0.1456	0.1613	0.4602
	LSPR-PM	0.0658	0.0490	0.1386	0.1155	0.0864	0.1442	0.2018	**0.2159**	**0.5582**
	-w/o Short-term	0.0478	0.0362	0.0976	0.01302	0.0081	0.0194	**0.2038**	0.2081	0.5460
	-w/o Long-term	**0.0680**	**0.0519**	**0.1435**	**0.1329**	**0.1110**	**0.1634**	0.1911	0.2064	0.5404

TABLE 3. Hyperparameter analysis results.

		AMAZON_BOOKS			TAOBAO_BUY			MIND_NEWS		
		Recall@50	NDCG@50	HR@50	Recall@50	NDCG@50	HR@50	Recall@50	NDCG@50	HR@50
MIND	K=3	0.06002	0.0471	0.12369	0.11338	0.06335	0.14163	0.14555	0.16126	0.46015
	K=4	0.04973	0.03944	0.10334	0.10544	0.05604	0.13146	0.13721	0.15374	0.44928
	K=5	0.04982	0.04017	0.10344	0.10544	0.06309	0.13158	0.13366	0.14973	0.43922
	K=6	0.04825	0.03916	0.10078	0.09612	0.05526	0.12045	0.1275	0.14387	0.42781
LSPR-PM	K=3	0.06584	0.049	0.13863	0.11551	0.08644	0.14424	0.20177	**0.21588**	**0.55815**
	K=4	0.06329	0.0478	0.13405	0.11936	0.0938	0.1486	0.19498	0.21079	0.54969
	K=5	0.06358	0.04791	0.13429	0.12018	0.09339	0.14808	0.20164	0.21554	0.55804
	K=6	0.06288	0.04806	0.13279	0.12329	0.09757	0.14968	0.19739	0.21224	0.5534
LSPR-PM -w/o Short-term	K=3	0.04776	0.03622	0.09755	0.01302	0.00812	0.01941	0.20382	0.20805	0.54603
	K=4	0.04713	0.03462	0.09685	0.01283	0.00809	0.01898	0.20781	0.20947	0.54963
	K=5	0.05127	0.03948	0.10456	0.01181	0.00696	0.01764	0.20478	0.20781	0.54542
	K=6	0.05171	0.04806	0.10463	0.01198	0.00728	0.0179	**0.20805**	0.21097	0.5534
LSPR-PM -w/o Long-term	K=3	0.068	0.05189	0.14354	0.13287	**0.11102**	0.16336	0.19114	0.20642	0.5404
	K=4	0.06748	0.0509	0.14195	**0.13539**	0.11009	**0.16364**	0.19406	0.20771	0.54253
	K=5	0.06877	0.05216	0.1443	0.1343	0.10973	0.16265	0.19403	0.21105	0.54783
	K=6	**0.0691**	**0.05225**	**0.14571**	0.13341	0.10962	0.16262	0.19315	0.20983	0.54439

long-term relationships. LSPR-PM -w/o Long-term represents that we remove the long-term relationships encoding module and only model the project's short-term relationships.

From the results of the ablation experiments, we can see that although LSPR-PM, LSPR-PM -w/o Short-term, and LSPR-PM -w/o Long-term achieve optimal or near-optimal performance, the long-term relationships encoding module and the short-term relationships encoding module play different roles in different datasets. On the AMAZON and TAOBAO datasets, we can see that the short-term relationships encoding module plays a more important role, while the long-term relationships encoding module has a negative effect, possibly because these two datasets are based on project purchase behavior, and short-term relationships have a greater impact. In contrast, combining the project's long-term and short-term relationships can achieve better performance on the MIND_NEWS dataset.

The different roles of the long-term and short-term relationships between projects encoding modules on different datasets inspire us to consider the characteristics of project behavior data collected in different application scenarios and to perform targeted optimization to improve the performance of the recommendation system.

C. HYPERPARAMETER ANALYSIS

As shown in Table 3, the hyperparameter $K = [3, 4, 5, 6]$ represents the number of multiple relationships representations for projects. After conducting a large number of sufficient experiments, we found that the hyperparameter K, the long-term relationships capture module, and the short-term relationships capture module play different roles on different datasets. Specifically, as shown in the table, on the AMAZON book purchase dataset, when the project's multiple relationships representation is $K = 6$, adding the short-term relationships capture module results in the optimal performance on all indicators. On the TAOBAO dataset, when $K = 4$, adding the short-term relationships capture module achieves the best results on Recall@50 and HR@50 indicators, and NDCG@50 is only slightly lower than when $K = 3$. On the MIND_NEWS dataset, adding both the long-term and short-term relationships capture modules achieves better performance.

To further analyze the role of long-term and short-term relationships capture modules on different datasets, the results are visualized in Figure 2. When $K = 6$, the MIND_NEWS dataset reflects that the long-term and short-term relationships have almost the same impact on performance. On the TAOBAO dataset, the short-term relationships dominate, and the long-term relationships almost have no effect. On the AMAZON dataset, capturing short-term relationships slightly outperforms capturing long-term relationships.

D. CASE STUDY

In our empirical study, we performed case analysis experiments on the MIND_NEWS, AMAZON, and TAOBAO datasets, as depicted in Figure 3. The horizontal axis of each sub-figure is encoded according to the timestamp,

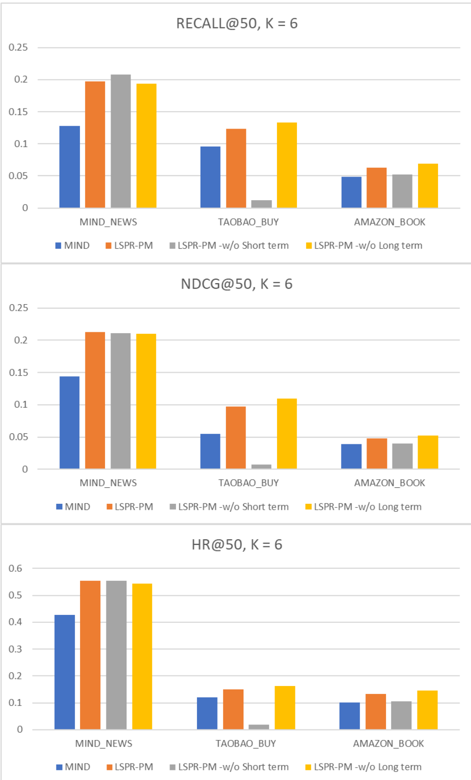

FIGURE 2. Variation of Recall@50, NDCG@50, HR@50 on different data sets.

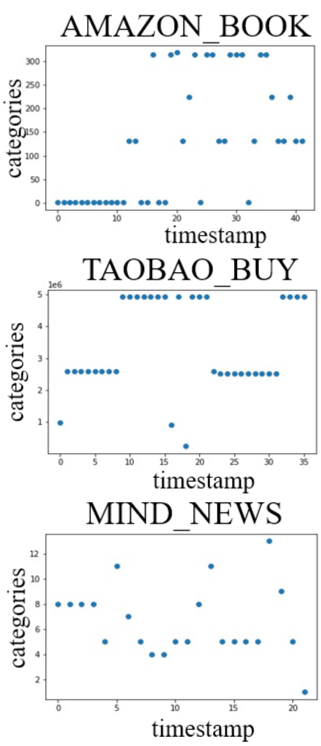

FIGURE 3. Analyzing the multi-relationships vs. long-term and short-term relationships of projects on different datasets.

representing the progression of time, while the vertical axis represents the categories of the items that projects interacted with. A preliminary observation reveals that across all three datasets, projects exhibit interactions with multiple categories of items—a pattern underscored by the distribution of item categories displayed graphically. Moreover, as time unfolds, there is a discernible shift in the types of items that projects interact with, highlighting the inherently dynamic character of project relationships. Upon extracting a project's long-term relationship preference $u_{long-term}$ Equation 3, via Equation 3, and its short-term relationship preference, denoted as $u_{short-term}$, utilizing Equation 4, we proceeded to compute their similarity with the historical sequence of items with which the project has interacted. Our findings show that $u_{long-term}$ bears a closer resemblance to the project's earlier interaction patterns, whereas $u_{long-term}$ aligns more closely with the project's recent behavioral trends. This evidence substantiates that LSPR-PM effectively discriminates between the long-term and short-term relationships within a project, and thereby successfully captures and models the nuanced relationship preferences of a project over time.

VII. CONCLUSION AND FUTURE WORK

A. CONCLUSION

This study explores various modeling techniques and system implementations for managing project relationships in the context of utility grid management. We utilize external knowledge, such as contextual information about grid projects, to aid in modeling project relationships. Furthermore, we investigate the inherent patterns within grid project records to better understand multi-level project dependencies and the long-term and short-term relationships between projects.

1. In the domain of context-augmented project management, the research adeptly utilizes social context data of project teams to strengthen the modeling of project relationships in utility grid management. However, there is a need to better exploit the immediate feedback from project teams within the project management system. To refine the accuracy of project relationship modeling and to facilitate smooth project implementation, future research should focus on optimizing the use of team feedback and also incorporate additional external knowledge sources to address the issues arising from limited data availability.

2. In the integration of multi-level and long-term/short-term relationships in sequence management, the current approach has partially decoupled multi-level project relationships into long-term and short-term relationships but lacks explicit and effective supervisory signals for this decoupling. Therefore, additional research is necessary to develop a self-supervised framework that can differentiate between long-term and short-term project interdependencies within the realm of utility grid management activities.

3. Regarding the development and implementation of a project management system enhanced by multi-dimensional project relationships, its effectiveness has only

been demonstrated through empirical testing on a limited number of datasets. An important future research direction is to scale the system to handle larger datasets, making it genuinely practical and relevant to the intricate challenges of actual utility grid management projects. Additionally, it is essential to thoroughly evaluate the system's performance and robustness in the demanding environments commonly encountered in utility grid project management.

B. FUTURE WORK

This study explores techniques for managing project relationships in utility grid management, leveraging external knowledge and analyzing grid project records to understand dependencies and relationships. However, there are still some shortcomings and areas for improvement, as follows:

1. While the research on socially-enhanced dialogue recommendation effectively utilizes the context of the project to enhance the modeling of project interest preferences, it does not fully leverage the advantage of timely project feedback in dialogue recommendation systems. Further research is required to explore more efficient methods of incorporating and utilizing project feedback data, while also merging external knowledge, to better address issues of data scarcity. This endeavor aims to improve the precision of recommendations and facilitate more interactions within the project context.

2. In the integration of multi-relationships, long-term and short-term relationships sequences, the present method partially partitions project preferences into long-term and short-term segments by leveraging multi-relationships modeling. Nevertheless, the lack of clear and potent supervisory signals hinders the complete realization of this demarcation between temporal interests. Consequently, there is an urgent demand for additional research into the advancement of methods that can self-supervise the disentanglement of long-term and short-term relationships more accurately and efficiently.

3. In the design and implementation of recommendation systems with enhanced multi-relationships, the functionality of such systems has thus far been substantiated only on modestly scaled datasets. Future research needs to further explore how to handle large datasets to make the system truly usable and serviceable to the outside world. Additionally, comprehensive evaluations of the system's computational efficiency and its ability to handle high loads in real-world situations are crucial to confirm their suitability for widespread use in large-scale settings.

REFERENCES

[1] C. R. Aberger, "Recommender: An analysis of collaborative filtering techniques," *Pers. Ubiquitous Comput. J.*, vol. 5, 2014.

[2] M. An, F. Wu, C. Wu, K. Zhang, Z. Liu, and X. Xie, "Neural news recommendation with long- and short-term user representations," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 336–345.

[3] Y. Cen, J. Zhang, X. Zou, C. Zhou, H. Yang, and J. Tang, "Controllable multi-interest framework for recommendation," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 2942–2951.

[4] X. Chen, H. Xu, Y. Zhang, J. Tang, Y. Cao, Z. Qin, and H. Zha, "Sequential recommendation with user memory networks," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, Feb. 2018, pp. 108–116.

[5] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for Youtube recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, Sep. 2016, pp. 191–198.

[6] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Commun. ACM*, vol. 35, no. 12, pp. 61–70, Dec. 1992.

[7] M. Grbovic and H. Cheng, "Real-time personalization using embeddings for search ranking at airbnb," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 311–320.

[8] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "DeepFM: A factorization-machine based neural network for CTR prediction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1725–1731.

[9] X. He and T.-S. Chua, "Neural factorization machines for sparse predictive analytics," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2017, pp. 355–364.

[10] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T. S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.

[11] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," 2015, *arXiv:1511.06939*.

[12] L. Hu, C. Li, C. Shi, C. Yang, and C. Shao, "Graph neural news recommendation with long-term and short-term interest modeling," *Inf. Process. Manag.*, vol. 57, no. 2, Mar. 2020, Art. no. 102142.

[13] K. Järvelin and J. Kekäläinen, "IR evaluation methods for retrieving highly relevant documents," in *Proc. ACM SIGIR Forum*, vol. 51, New York, NY, USA, Aug. 2017, pp. 243–250.

[14] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 197–206.

[15] G. Karypis, "Evaluation of item-based top-N recommendation algorithms," in *Proc. 10th Int. Conf. Inf. Knowl. Manag.*, 2001, pp. 247–254.

[16] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2008, pp. 426–434.

[17] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[18] C. Li, Z. Liu, M. Wu, Y. Xu, H. Zhao, P. Huang, G. Kang, Q. Chen, W. Li, and D. L. Lee, "Multi-interest network with dynamic routing for recommendation at tmall," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, Nov. 2019, pp. 2615–2623.

[19] J. Li, Y. Wang, and J. McAuley, "Time interval aware self-attention for sequential recommendation," in *Proc. 13th Int. Conf. Web Search Data Mining*, Jan. 2020, pp. 322–330.

[20] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, "Neural attentive session-based recommendation," in *Proc. ACM Conf. Inf. Knowl. Manag.*, Nov. 2017, pp. 1419–1428.

[21] Y. Li, M. Liu, J. Yin, C. Cui, X.-S. Xu, and L. Nie, "Routing micro-videos via a temporal graph-guided recommendation system," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1464–1472.

[22] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan. 2003.

[23] J. Liu, P. Dolan, and E. R. Pedersen, "Personalized news recommendation based on click behavior," in *Proc. 15th Int. Conf. Intell. User Interfaces*, Feb. 2010, pp. 31–40.

[24] F. Lv, T. Jin, C. Yu, F. Sun, Q. Lin, K. Yang, and W. Ng, "SDM: Sequential deep matching model for online large-scale recommender system," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, Nov. 2019, pp. 2635–2643.

[25] C. Ma, L. Ma, Y. Zhang, J. Sun, X. Liu, and M. Coates, "Memory augmented graph neural networks for sequential recommendation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 5045–5052.

[26] Q. Pi, W. Bian, G. Zhou, X. Zhu, and K. Gai, "Practice on long sequential user behavior modeling for click-through rate prediction," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 2671–2679.

[27] S. Rendle, "Factorization machines," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2010, pp. 995–1000.

[28] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized Markov chains for next-basket recommendation," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 2010, pp. 811–820.

[29] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[30] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, Nov. 2019, pp. 1441–1450.

[31] J. Tang and K. Wang, "Personalized top-N sequential recommendation via convolutional sequence embedding," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, Feb. 2018, pp. 565–573.

[32] J. Weston, R. J. Weiss, and H. Yee, "Nonlinear latent factorization by embedding multiple user interests," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 65–68.

[33] Z. Yu, J. Lian, A. Mahmoody, G. Liu, and X. Xie, "Adaptive user modeling with long and short-term preferences for personalized recommendation," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 4213–4219.

[34] W. Zhao, B. Wang, J. Ye, Y. Gao, M. Yang, and X. Chen, "PLASTIC: Prioritize long and short-term information in top-N recommendation using adversarial training," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3676–3682.

[35] G. Zhou, N. Mou, Y. Fan, Q. Pi, W. Bian, C. Zhou, X. Zhu, and K. Gai, "Deep interest evolution network for click-through rate prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5941–5948.

[36] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, "Deep interest network for click-through rate prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 1059–1068.

[37] Y. Zhu, H. Li, Y. Liao, B. Wang, Z. Guan, H. Liu, and D. Cai, "What to do next: Modeling user behaviors by time-LSTM," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, vol. 17, Aug. 2017, pp. 3602–3608.

YONGQIAO ZHANG received the M.S. degree in geographic information system from Wuhan University, Wuhan, China, in 2006. He is currently the Director of the Information System Integration Branch, NARI Technology Company Ltd. He has presided more than or participated in several State Grid Science and Technology Projects. His main research interests include grid digitization and smart distribution grids.

GUANGYU BAI received the M.S. degree in electrical power system automation from Nanjing Agricultural University, Nanjing, China, in 2014. He is currently a Project Manager with NARI-TECH Nanjing Control Systems Ltd. He presided more than or participated in several State Grid Science and Technology Projects. His main research interests include grid investment planning, grid digitalization, and grid project management.

ZHANCHAO GAO received the B.S. degree in information security from Huazhong University of Science and Technology, in 2020, and the M.S. degree in computer application technology from the Institute of Information Engineering, Chinese Academy of Sciences, in 2023. He is currently a Computer Engineer. His main research interests include recommender systems, data analysis, and machine learning.

PENGYUAN ZHU received the M.S. degree in computer technology from Nanjing Normal University, Nanjing, China, in 2021. He is currently a Software Research and Development Engineer with the Information System Integration Branch, NARI Technology Company Ltd. He has been responsible for a number of scientific and technological projects of the State Grid Corporation. His main research interests include the application of artificial intelligence technologies, such as machine learning, natural language processing, and data mining in smart grids.

SHIWEN LI received the B.S. degree in communication engineering from Shandong Normal University, Jinan, China, in 2014. He is currently a Software Development Engineer with the Information System Integration Branch, NARI Technology Company Ltd. His main research interests include the application of emerging technologies, such as big data analysis, machine learning, and data mining in relevant business scenarios of state grid.