A Temporal Multi-Dimensional Scholar Recommendation System with Interest Drift Modeling

Haoyu Wu

***Abstract*—The rapid development of social networking platforms has enabled scholars to connect with collaborators who share similar research interests, making scholar recommendation an increasingly prominent topic in recent years. To address the limitations of existing scholar recommendation methods—such as inadequate modeling of temporal dimensions and insufficient capture of dynamic shifts in research interests—this paper proposes an approach, (TD-PCR) temporal dynamic features based personalized research collaborator recommendation, which improves accuracy and interpretability .Transcending the constraints of traditional static modeling, this approach achieves accurate recommendations through a threefold temporal dynamic mechanism: First, it partitions scholars' research trajectories using a sliding time window (2 years) and dynamically weights recent academic outputs with a temporal decay factor. Second, it constructs a research interest trend similarity metric to analyze the matching degree of evolutionary paths in scholars' research directions. Third, it designs a multi-dimensional time-sensitive feature system, encompassing dynamic text similarity, temporally decayed collaboration networks, and evolutionary characteristics of academic influence.In terms of model architecture, the approach innovatively integrates time-aware Doc2Vec encoding, weighted random walk algorithms, and attention-enhanced neural networks. It realizes precise prediction through a five-dimensional feature vector (text similarity, social relevance, average influence, trend similarity, and domain similarity).**

***Index Terms*—Scholar recommendation; Temporal dynamic modeling; Research interest evolution; Attention mechanism; Doc2Vec; Academic collaboration network**

I. Introduction

With the rapid development of academic social platforms, platforms such as ResearchGate and GitHub have emerged as core hubs for scholarly communication and collaboration, facilitating researchers in discovering potential collaborators and sharing research resources. However, amid the exponential growth of their user bases, it has become increasingly challenging for scholars to precisely identify suitable collaborators among massive users, rendering the demand for personalized recommendation systems ever more prominent.

Existing scholar recommendation methods mostly focus on static feature analysis, such as recommendations based on paper text similarity (e.g., TF-IDF, LDA) or social relationship networks. Yet they have two notable limitations: first, they overlook the dynamic evolution of scholars' research interests (i.e., "interest drift"), as a scholar's research directions may differ significantly across periods; second, they lack consideration of the temporal decay effect, often underestimating the impact of recent collaborative relationships or research outputs on recommendations. For instance, traditional methods often treat all of a scholar's papers as a static text collection, failing to reflect their latest research trends and resulting in insufficient timeliness of recommendations.

To address the aforementioned issues, this paper proposes a time-sensitive multi-dimensional scholar recommendation system that integrates interest drift modeling. The core innovations of the system are as follows:

- Dynamic feature capture: Through time window partitioning, extract scholars' research keywords, domains, and text features across different periods to quantify the evolutionary trajectory of their research interests.
- Interest drift modeling: Design a trend similarity calculation method to measure the dynamic matching degree of research directions between scholars, rather than relying on static text similarity.
- Multi-dimensional temporal weighting: Incorporate a temporal decay mechanism into text similarity (based on a time-sensitive Doc2Vec model), social relevance (weighted PageRank algorithm), and academic influence evaluation to enhance the timeliness and accuracy of recommendations.
- Deep learning integration: Construct a neural network model with an attention mechanism to adaptively fuse multi-dimensional features and output personalized recommendation results.

The subsequent structure of this paper is organized as follows: Section 2 reviews related work; Section 3 introduces the background knowledge and related theoretical foundations of the algorithm; Section 4 elaborates on the framework and key algorithms of the recommendation system; Section 5 presents the experimental design and result analysis; Section 6 summarizes the research and outlines future work.

II. Related Work

Based on recent literature reviews, the recommendation algorithms applied in academic recommendation systems can be categorized into the following types, covering directions such as content - based features, collaborative filtering, hybrid strategies, and deep learning algorithms.[1]

CBF recommenders use keywords or topics as key features because they are used todescribe a publication. The creation of a content-based profile of users usually concentrateson the user's preferencemodel, and the user'sinteraction log with

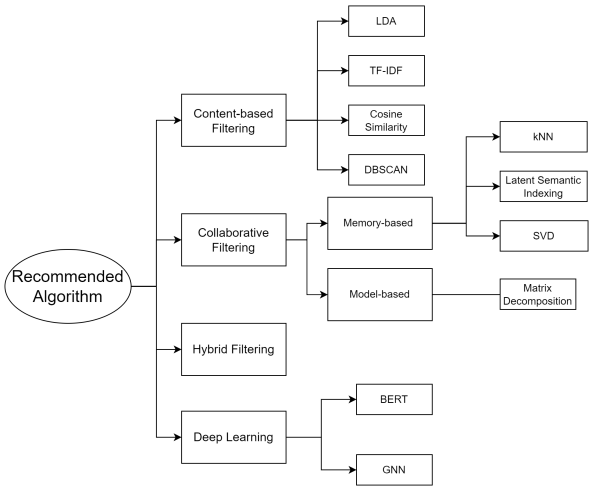

Fig. 1. Recommendation Algorithm

the recommendation systemconverted by a weighted vector of item features. For example, Hong et al.constructed apaper recommendation methodology based on a user profile built with extracted keywords,and calculated the similarity between a given topic and collected papers by using cosinesimilarity to recommend initial publications for each topic.[2]

Common methodologies using a collaborative filtering algorithms can be categorizedinto two groups: model-based and memory-based. The main difference between the twoapproaches is that the model-based approach uses a matrix factorization-based algorithm, inwhich the preferences of users can be calculated by embedding factors. The memory-basedapproach calculates the preferences of users for items based on arithmetic operations (cor-relation coefficients or cosine similarity). Memory-based CF approaches are widely used inscholarly literature recommendation systems, which includes several different approaches,such as k-nearest neighbors(kNN), Latent Semantic Index (LSI), and Singular Value Decom-position (SVD). Pan and Li used the LDA (Latent Dirichlet Allocation) model toconstruct a paper recommendation system using a thematic similarity measurement totransform a topic-based recommendation into a modified version of the item-based rec-ommendation approach. Ha et al.proposed a novel method using SVD for matrixfactorization and rating prediction to recommende newly published papers that have notbeen cited by other papers by predicting the interests of the target researchers.[3]

As a combination of CBF and CF, hybrid recommendation approaches can be categorizedinto four main groups. The first group implements CBF and CF methods separately and thencombine their recommendation results. The second and third groupsincorporate CBF characteristicsinto a CF method or incorpo-rate some CF characteristics into a CBF method. The second and third groupsincorporate CBF characteristicsinto a CF method or incorpo-rate some CF characteristics into a CBF method.

The research on deep learning algorithms is as follows, Wei et al. noted that existing GNN-based recommendation methods overlook multimodal item contents, and while MMGCN attempted to address this, it failed to adaptively capture user preferences. To tackle this, they proposed the Multimodal Graph Attention Network (MGAT), which disentangles personal interests at the modality level by conducting information propagation within individual multimodal interaction graphs and leveraging a gated attention mechanism to identify the importance of different modalities to user preference.[4]

To address the issue that existing paper recommendation systems fail to effectively integrate semantic and structural information in academic resource networks, Wang et al. proposes the HGNN-MAO model[5], which employs a hybrid update embedding method to aggregate low-order (structural) and high-order (semantic) neighborhood information via heterogeneous transformers and meta path-guided aggregators, with validation on academic datasets demonstrating its superiority over advanced techniques in accuracy and training speed.

III. Preliminaries

A. Text Representation Model

Neural networks play a significant role in the field of Natural Language Processing (NLP).In the realm of text representation, the word vector (Word2Vec) model and paragraph vector (Doc2Vec) model, proposed by Mikolov et al., are the most widely adopted; both are unsupervised deep learning algorithms. The key distinction is that Word2Vec typically learns the feature vector representation of individual words from a corpus, whereas Doc2Vec is capable of learning the feature vector representation of text of any length.

The Word2Vec model can predict the next word based on context. The model is defined as follows: for a given set of text sequences, the maximum average logarithmic probability of the sequence serves as the objective function for training Word2Vec. During the training convergence process, the model maps words with similar meanings to analogous positions in the vector space, and text similarity can be computed using vectors of a specific length.

The Doc2Vec model not only leverages word vectors to predict the next word in a text but also incorporates paragraph topic vectors into the next-word prediction task. This renders text semantics more distinct and reliable.

B. Random Walk Algorithm

The random walk algorithm is a graph-based probabilistic traversal method, whose core lies in simulating a memoryless random movement process: in a graph $G = (V, E)$ consisting of nodes (e.g., scholars, papers) and edges (e.g., collaboration or citation relationships), starting from an initial node, each step involves randomly jumping to a neighbor of the current node based on transition probabilities—with simple walks selecting neighbors with equal probability and weighted walks using edge-weight-normalized probabilities. This process generates a random path due to the Markov property (memorylessness), where each step depends solely on the current state. In academic recommendation scenarios, it can

explore indirect associations between nodes: for instance, by traversing scholar collaboration graphs to quantify potential collaboration degrees.

The goal of random walk is to identify correlations between any two nodes. Let $G = (V, E)$ denote an undirected connected graph with V nodes and E edges, whose adjacency matrix is A. If node v_m and node v_n are connected, then $A_{v_m v_n} = 1$; otherwise, $A_{v_m v_n} = 0$. The degree of node v_m is the number of nodes connected to it, denoted as $d(v_m)$:

$$d(v_m) = \sum_{v_n} A_{v_m v_n} \tag{1}$$

C. MLP and Attention Mechanism

A perceptron acts as a binary classifier, using weights and biases to map input information to binary outputs (0 or 1); during training, it minimizes classification errors by fine-tuning these weights. The Multi-Layer Perceptron (MLP), an extension of this basic model, includes multiple neuron layers—hence its alias, Deep Neural Networks (DNN). Formed by linking multiple perceptrons, MLPs handle complex multi-classification tasks, with the simplest structure adding a hidden layer to create a three-layer feedforward network. In practice, they can have any number of nodes and layers (with full connectivity between adjacent layers), where hidden layers process input data and weights via activation functions (e.g., Sigmoid, Tanh, ReLU) before the output layer produces results.

The attention mechanism mimics human visual focus, directing more computational resources to critical information to boost efficiency, and is widely used in fields like image recognition, NLP, and recommendation systems. Its computation involves two key steps: first, calculating attention distribution probabilities for input data; second, computing a weighted average based on these probabilities. For an input set $X = \{x_1, x_2, \ldots, x_N\}$, it assigns distinct weights to each group based on their relevance to the task.

IV. The TD-PCR Approach

The workflow of Mul-RSR is outlined in Section A. The three steps of Mul-RSR are introduced in details in Sections B-D.

A. Overview of TD-PCR

The flow chart of Mul-RSR for personalized scholar recommendation framework is shown in Fig. 2. The framework is divided into three main steps: data collection and processing, multi-dimensional feature extraction, model training and recommendation. The specific implementation process is described as follows.

1) Data collection processing: First, data preprocessing is performed. The dataset we obtained contains information on 3,000 papers, stored in a paper-centric data structure. Since our goal is to conduct scholar recommendation, we convert the dataset into a scholar-centric structure and construct dynamic features for scholars. Next, each scholar's data is divided by time windows: specifically, information such as the scholar's publication outputs and collaboration relationships is segmented using a two-year window, preparing for downstream tasks. This allows for a better exploration of the dynamic changes in each scholar's research interests.

2) Multi-dimensional feature extraction: After obtaining the dynamic features of each scholar, we proceed to calculate their five-dimensional features.

First is text similarity: we extract paper abstracts within each time window to train a Doc2Vec model, vectorize these abstracts, and then compute text similarity as one feature dimension.

Next, we construct a collaboration network where each scholar is a node, and edge weights are determined by collaboration frequency multiplied by a temporal decay factor—reflecting the closeness of scholars' collaborative relationships. With this collaboration network, we can calculate social relevance between scholars using random walk.

Comprehensive influence, also known as academic influence, represents a scholar's influence and contributions in academia or through their papers. It is computed by integrating factors such as temporal decay, journal impact factors, and the number of collaborators.

Then, we extract paper keywords and research fields from scholars' features to calculate similarity in research trends: this involves comparing whether changes in research fields across the latest two time windows are consistent between two scholars, reflecting dynamic changes in their research interests.

Additionally, we compute the degree of overlap in research fields across all papers of two scholars, which reflects their overall similarity in research interests at a macro level.

3) Model training and recommendation: The correlation strength between the five-dimensional features obtained in step 1 is used as the input data of the recommendation model, which is based on the multi-layer perceptron and the attention mechanism. First, we obtain the initial parameters of the model through layer-by-layer greedy pre-training and then carry out the forward propagation training of the model. Finally, the back-ward propagation optimization is done based on the loss function, and the model parameters are learned and updated. Among them, the attention mechanism can continuously adjust the weight of the input data, thereby improving the accuracy of the recommendation model.

B. Data collection and processing

The dataset we provide contains data on 3,000 papers, with each paper's data structure including its title, authors, year of publication, keywords, abstract, publishing journal, impact factor, and research fields in both Chinese and English. We transform this data structure into a scholar-centric format, which facilitates the study of relationships between scholars. Additionally, we divide each scholar's paper data into time windows, with a two-year interval per window, to segment their papers. The final data structure includes the scholar's Chinese and English names, time window, all their papers, and all their collaborators. Each time window aggregates the papers from the corresponding two-year period, encompassing

all the scholar's papers from those two years, as well as their keywords and research fields.

C. Multi-dimensional feature extraction

1) Textual Similarity Calculation: After obtaining the processed scholar data, the next step is to calculate the text similarity.We use the Doc2Vec model to vectorize the paper abstracts of each scholar within their respective time windows. After obtaining these vectors, we calculate the textual similarity between each pair of scholars, thereby constructing a textual similarity matrix $M_{n\times n}^{TS}$.

As an unsupervised algorithm for paragraph vectorization, the Doc2Vec deep text representation model is capable of learning fixed-length feature vectors from text segments of varying lengths. This allows a sentence, paragraph, or even an entire document to be converted into a vector form.

Doc2Vec operates on a core principle: given a paragraph vector and context words $w_{t-k}, \ldots, w_{t+k}$ as input, the probability associated with the central word w_t's vector is yielded as output through averaging or concatenation operations in the hidden layer of the Doc2Vec neural network. Moreover, the context words $w_{t-k}, \ldots, w_{t+k}$ are derived from text paragraphs via a sliding window technique. Notably, paragraph vectors are shared within the context of individual paragraphs but not across distinct paragraphs.

A key step involves pre-training the Doc2Vec model. Within this model, the parameter dm specifies the training algorithm employed. When $dm = 1$, the PV-DM (Distributed Memory Model of Paragraph Vectors) algorithm is utilized. This approach regards a text paragraph as a single lexical unit and takes into account the concatenation of the paragraph vector and word vectors during training. It can retain information about missing content in the current context or paragraph text, making it the standard configuration for Doc2Vec. In contrast, when $dm = 0$, the PV-DBOW (Distributed Bag of Words Version of Paragraph Vector) algorithm is applied. This method disregards input context words and instead trains a neural network to forecast the probability distribution of randomly chosen words within the paragraph.

Doc2Vec's objective function seeks to maximize the average logarithmic probability expressed below:

$$P = \frac{1}{T}\sum_{t=k}^{T-k} \log p(W_t \mid W_{t-k}, \ldots, W_{t+k}) \tag{2}$$

Here, T denotes the length of the training text sequence, k represents the size of the context window, and p stands for the probability of successfully predicting the central word W_t.

The pre-trained Doc2Vec model is then used to transform all papers of each scholar into vectors in the feature space.

Calculating textual similarity involves the following steps: First, sort the time windows—specifically, the time windows of source scholars and target scholars are sorted in descending order by year. After sorting, window similarity calculation is performed: the feature vectors of corresponding windows from the source and target scholars are extracted to calculate their cosine similarity. Next, temporal weighting is applied: the similarity is weighted using the temporal weights of the original windows, where top-ranked windows (i.e., those closer to the present) have larger temporal weights. Finally, normalization is performed to obtain the textual similarity between scholars.The calculation formula for cosine similarity is given as follows:

$$\cos\theta = \frac{A \cdot B}{|A| \cdot |B|} \tag{3}$$

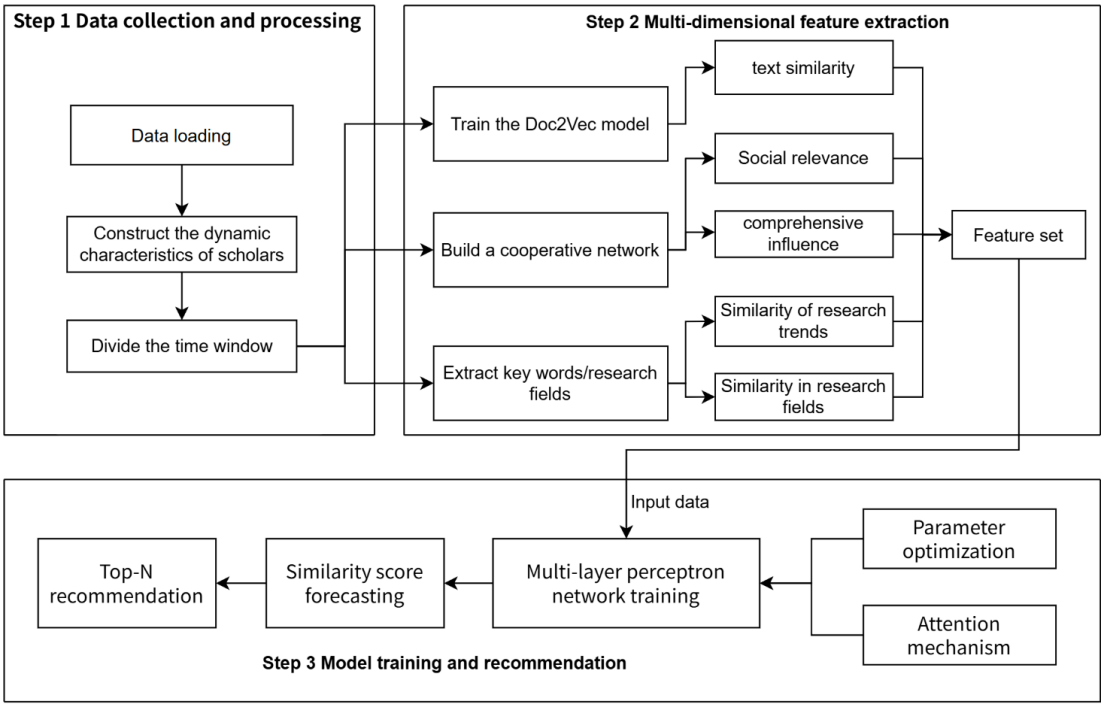

Fig. 2. Overview of TD-PCR

where A and B are the vector representations of two scholars. When the cosine value approaches 1, it indicates that the two vectors are more similar.

2) Social Relevance Calculation: In this step, we use the graph-based random walk algorithm to calculate the socialrelevance between two scholars. A social relevance matrix $M_{n\times n}^{SR}$ will be constructed as the output of this step.

The following relationships among scholars in social networks can be modeled as an undirected graph $G = (V, E)$, where the node set V corresponds to a group of scholars, and the edge set E represents the following connections between them.The core goal of random walk is to gauge the relevance between two nodes: the stronger the relevance, the higher the similarity between the two nodes (scholars). In the context of random walk, the relevance between node v_1 and node v_2 is shaped by the following four factors:

- Factor 1: The number of collaborations between node v_1 and node v_2 in the graph, as well as the year of their most recent collaboration. The greater the number of collaborations and the more recent the collaboration year (relative to the present), the higher the relevance. When the number of collaborations and the collaboration year are the same, Factor 2 shall be referenced.
- Factor 2: The quantity of paths through which node v_1 can reach node v_2 in the graph. A greater quantity of such paths corresponds to higher correlation. For instance, the correlation between (v_1, v_2) is stronger than that between (v_1, v_3) if there are more connecting paths between v_1 and v_2 than between v_1 and v_3. When the number of paths is identical, Factor 3 will be considered.
- Factor 3: The length of paths linking node v_1 to node v_2. Shorter path lengths indicate higher correlation. In scenarios where path lengths are the same, Factor 4 will be referenced.
- Factor 4: The total out-degree of all nodes in the paths that connect node v_1 to node v_2. Here, a node with a larger out-degree can be regarded as having higher visibility and more followers. The smaller the total out-degree of these nodes, the higher the correlation between v_1 and v_2.

The concept of random walk operates as follows: Given a graph G and an initial node v_1 within the graph, the walker has two choices: either stay at the starting node or move on to another node. In the latter case, the walker will randomly select and transition to a node v_2 that is connected to v_1. This process repeats iteratively until the access probability of each node converges to a fixed value [27,33].

In this context, all edge weights in graph G are assumed to be equal. After the random walk process reaches iterative convergence, the access probability of each scholar in the graph can be expressed by the following formula:

$$P_i = \begin{cases} (1-\alpha) + \alpha \sum_{j \in \text{set}(i)} \frac{P_j}{|\text{set}(j)|}, & r = 1 \\ \alpha \sum_{j \in \text{set}(i)} \frac{P_j}{|\text{set}(j)|}, & r = 0 \end{cases} \tag{4}$$

Here, T denotes the length of the training text sequence, k represents the size of the context window, and p stands for the probability of successfully predicting the central word W_t.

Taking Ai LiSha as an example, her collaboration network is illustrated in Fig. 3 . The yellow node represents the scholar himself, blue nodes stand for scholars who have direct collaborations with him, and green nodes denote indirect collaborators with high recommendation scores.

Fig. 3. Cooperation Network

3) Comprehensive influence: We use certain behavioral attributes here to represent scholars' contribution rate indicators, and construct a list L_{PCR}^{n} to store the comprehensive influence of each scholar. This is mainly divided into three contributions:

- Time-decay score: Recent papers are assigned higher weights in terms of their influence, which reflects the timeliness of academic achievements. It prioritizes the reflection of scholars' recent research activity and avoids excessive interference of outdated achievements on their current influence.
- Impact factor score: Papers published in journals with high impact factors are assigned higher weights, which reflects the academic quality of the achievements. It quantifies the academic influence of the publishing platforms for the achievements, with papers in higher-tier journals making more significant contributions.
- Co-author score: The fewer the authors of a paper, the higher the proportion of the scholar's independent contribution in that single paper, which avoids dilution due to collaboration. This score highlights scholars' independent contributions in collaborative research and prevents the inflation of influence caused by "honorary authors" (i.e., authors listed without substantial contributions).

4) Similarity of research trends and research fields: Research trend similarity focuses on changes in research fields within the most recent two time windows to determine the consistency of dynamic trends in scholars' research interests. The process is as follows: first, it is verified that the number of a scholar's time windows is at least two; then, the research fields of the two most recent windows are extracted, and their Jaccard similarity is calculated; finally, the dynamic field alignment between the two scholars is obtained. By iterating

over all scholars, a dynamic field alignment matrix $M_{n\times n}^{RT}$is generated.

Research field similarity covers research fields across all time windows to determine the overall alignment of scholars' research interests. It first extracts all research fields of scholars, calculates the Jaccard similarity between two scholars, and by iterating over all scholars, a static field alignment matrix $M_{n\times n}^{RF}$is obtained.

D. Model training and recommendation

Based on the data processing in the previous section, we have obtained the textual similarity matrix $M_{TS}^{n\times n}$, social relevance matrix $M_{SR}^{n\times n}$, comprehensive influence list bL_{PCR}^{n}, dynamic research field alignment matrix $M_{RT}^{n\times n}$, and static research field alignment matrix $M_{RF}^{n\times n}$. Subsequently, using the scholars' collaboration network, we constructed a collaboration label matrix $M_{SL}^{n\times n}$ based on the collaborative relationships between scholars to store the true collaboration labels among them. I propose TD-PCR—a multi-feature-based dynamic scholar recommendation framework, which is built on the basis of Multi-Layer Perceptron (MLP) combined with an attention mechanism. The values of the feature matrices are used as input to the TD-PCR framework for training, and the output predicted similarity matrix $M_{PS}^{n\times n}$ represents the similarity scores among scholars.

Assuming that in the TD-PCR model, the mapping function between the five-dimensional eigenvalues of scholars and similarity is F, the similarity score matrix is:

$$M_{PS}^{n\times n} = F\left(M_{TS}^{n\times n},\ M_{SR}^{n\times n},\ L_{PCR}^{n},\ M_{RT}^{n\times n},\ M_{RF}^{n\times n}\right) \quad (5)$$

According to the similarity score, Top-N recommendation is performed. The overall process is shown in Algorithm 1.

Algorithm 1 Personalized Scholar Recommendation

Require:Five-dimensional eigenvalues ($M_{TS}^{n\times n}$, $M_{SR}^{n\times n}$, L_{PCR}^{n}, $M_{RT}^{n\times n}$, $M_{RF}^{n\times n}$); Cooperation label matrix ($M_{SL}^{n\times n}$).
Ensure:Prediction matrix ($M_{PS}^{n\times n}$).

1: Initialize the model parameters
2: Forward propagation begins
3: Normalize the eigenvalues and input into the neural network
4: Fit the Cooperation label matrix
5: Output predicted value
6: Backpropagation begins
7: Update model parameters based on BP and SGD
8: Use attention mechanism to adjust the weights again
9: Output similarity score prediction value
10: Perform Top-N recommendation

V. Experiment and Result Analysis

We take scholar Chen Gang as an example to present the experimental results.

Among them, Fig.4 shows the top 10 recommended scholars for Chen Gang, which contains similarity information across various dimensions. The recommended scores are ranked from high to low. Figu.5 presents the best recommended collaborative scholar for Chen Gang, displaying a radar chart of the five-dimensional features between this scholar and Chen Gang.

排名	姓名	推荐分数	文本相似度	趋势相似度	领域相似度
1	Jia Qingxuan	1.0000	0.6541	0.3333	0.4839
2	Fu Yingzhuo	1.0000	0.5568	0.0000	0.2805
3	Li Tong	1.0000	0.7295	0.2316	0.2222
4	Jia ShiYuan	1.0000	0.7333	0.0000	0.2317
5	Fei JunTing	0.9999	0.7487	0.1967	0.2073
6	Huang ZeYuai	0.9997	0.7256	0.1507	0.1702
7	Song JingZhc	0.9997	0.7087	0.2258	0.1739
8	Yuan Bonan	0.9994	0.6075	0.0000	0.1341
9	Jiang Tao	0.9994	0.4259	0.0000	0.1071
10	Duan JiaQi	0.9991	0.7194	0.0000	0.1463

Fig. 4. Recommended collaborators

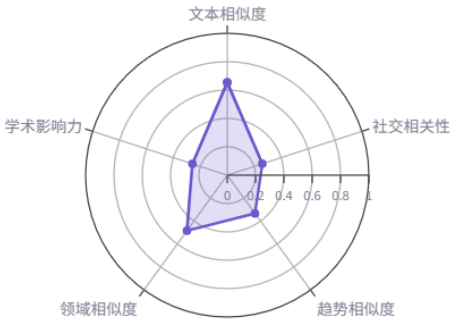

Fig. 5. Five-dimensional radar map

VI. Conclusions and Future Work

Existing recommendation methods cannot meet the requirements of scholar recommendation for strong relevance, high accuracy, interpretability, and the dynamics of scholars' research interests. I propose a personalized dynamic scholar recommendation method based on multi-dimensional features, called TD-PCR. It first divides scholars' data structures by time windows and, in accordance with these time windows, explores the relevance between potential scholars from five aspects: textual similarity of published papers, social relevance, comprehensive influence, research trend similarity, and research field similarity. TD-PCR uses the Doc2Vec text model and random walk algorithm to measure the relevance between scholars. Based on Multi-Layer Perceptron (MLP) and attention mechanism, it can recommend Top-N scholars for each scholar.

Currently, the cold-start problem in the proposed TD-PCR method has not been effectively addressed. Specifically, for newly added scholars—i.e., those lacking sufficient historical data, such as researchers who have just entered the academic

field, have not published enough papers, or have not formed a stable collaborative network—they lack adequate valid data across dimensions including textual features, social relations, comprehensive influence, research trends, and research fields. This makes it difficult for the model to accurately excavate their potential characteristics and research attributes through existing mechanisms. This will be a key focus of future work.

REFERENCES

[1] Z. Zhang, B. G. Patra, A. Yaseen, J. Zhu, R. Sabharwal, K. Roberts, T. Cao, and H. Wu, "Scholarly recommendation systems: a literature survey," *Knowledge and Information Systems*, vol. 65, no. 11, pp. 4433–4478, 2023.

[2] K. Hong, H. Jeon, and C. Jeon, "Userprofile-based personalized research paper recommendation system," in *2012 8th international conference on computing and networking technology (INC, ICCIS and ICMIC)*, pp. 134–138, IEEE, 2012.

[3] J. Ha, S.-W. Kim, C. Faloutsos, and S. Park, "An analysis on information diffusion through blogcast in a blogosphere," *Information sciences*, vol. 290, pp. 45–62, 2015.

[4] Z. Tao, Y. Wei, X. Wang, X. He, X. Huang, and T.-S. Chua, "Mgat: Multimodal graph attention network for recommendation," *Information Processing & Management*, vol. 57, no. 5, p. 102277, 2020.

[5] Z. Wang and T. Jiang, "Heterogeneous graph neural network for metapath aggregation optimization for paper recommendation," in *2024 IEEE 4th International Conference on Software Engineering and Artificial Intelligence (SEAI)*, pp. 365–370, IEEE, 2024.