A Paper-Project Association Model Based on Academic Social Networks (Postprint)

Authors: Wang Liu, Tang Yong, Yang Zuoxi, Fu Chengzhou, Mao Chengjie, Mao Chaodan

Date: 2019-04-01T00:00:00+00:00

Abstract

For the distinctive users of academic social networks, this paper proposes a collaborative association model for paper and project data based on scholar social networks. First, a two-step feature selection method is employed to preprocess the data, eliminating irrelevant and redundant features to obtain effective features that influence the association between papers and projects. Subsequently, the Text Vector Space Model (TVSM) is utilized to compute text similarity between papers and projects, thereby forming recommendation sets for different papers/projects. The model is implemented and deployed on Scholar Network, a social network platform for researchers, using its real-world data. Online application performance and user feedback demonstrate that the model exhibits favorable accuracy and practicality, enables more comprehensive mining of the rich information embedded between papers and projects, provides users with more efficient and convenient academic research management services, and presents a novel research methodology for analyzing academic big data.

Full Text

Preamble

Association Model of Paper and Project Based on Scholar Social Network

Wang Liu, Tang Yong†, Yang Zuoxi, Fu Chengzhou, Mao Chengjie, Mao Chaodan
(School of Computer Science, South China Normal University, Guangzhou 510631, China)

Abstract: This paper proposes a collaborative association model for paper and project data based on scholar social networks, tailored to the unique user base of

such platforms. The model employs a two-step feature selection method to preprocess data, eliminating irrelevant and redundant features to obtain effective features that influence paper-project associations. It then utilizes a Text Vector Space Model (TVSM) to compute text similarity between papers and projects, generating recommendation sets for different papers/projects. Implemented and deployed on SCHOLAT, a social network for researchers, the model' s online application and user feedback demonstrate its strong accuracy and practicality. The model enables more comprehensive mining of rich information embedded in paper-project relationships, providing users with more efficient and convenient academic research management services, and offering a novel research approach for analyzing academic big data.

Keywords: social network; collaborative association model; feature selection; text similarity; scholat

0 Introduction

In recent years, with the rapid development of the Internet, information data has grown exponentially, making the extraction of effective information from massive datasets a significant challenge in data mining [1]. Social networks face similar issues, as the vast amount of academic achievement information such as papers and projects has led to information overload in scholar social networks [2]. Scholars' demand for mining relationships between papers and projects has also surged. Among these, papers and projects—representing the most significant scholarly achievements—contain rich information resources, occupying a crucial position in scholar social networks. However, users currently struggle to fully extract effective information from these resources, and research on collaborative association models for papers and projects in scholar social networks remains scarce. Therefore, addressing the unique needs of scholar users in social networks, providing accurate and personalized paper/project recommendations to help them better associate papers with projects and fully mine the information between them has become an urgent research problem.

Association rule mining is an important branch of data mining that helps users discover potential relationships in large datasets [3]. Consequently, establishing a collaborative association model for papers and projects is essential for mining their rich scholarly information resources. This paper focuses on constructing an association model between papers and projects to facilitate relationship mining. Given the numerous different feature attributes in papers/projects, data preprocessing and feature selection are required before information mining. Among these features, some are effective for paper-project associations, while others are irrelevant or redundant. To address this, our model employs a two-step feature selection method [4] to preprocess paper and project features, obtaining effective feature sets. Since papers and projects in scholar social networks are predominantly stored in text format, and building upon traditional collaborative filtering recommendation models, we adopt the TVSM model to calculate text similarity between papers and projects, forming different paper neighbor-

hoods/project neighborhoods. Recommendation sets for target papers/projects are then identified by sorting similarity values in descending order. Combined with user requirement inputs, the final recommendation sets are generated to provide more accurate and personalized paper/project association choices, with the collaborative association model ultimately visualized. Through this model, the information contained in papers and projects can be understood more clearly and deeply, their potential relationships discovered, and data mining performed from multiple perspectives, providing users with an accurate, personalized, and practical research information management tool. The proposed paper-project data association model based on scholar social networks has been deployed on SCHOLAT, and user feedback analysis demonstrates its good accuracy and practicality.

1 Related Work

Recent years have seen numerous studies on paper recommendation based on social networks. Huang et al. [5] constructed an academic paper recommendation model based on community division, leveraging the social nature of academic social networks. Their approach uses label propagation for community division and recommends academic papers among users within each community structure. Chen et al. [6] proposed a search scheme for calculating semantic vectors of academic paper documents based on single-word semantic vectors, specifically for the computer science paper corpus on SCHOLAT, and applied it to the platform. Tang et al. [7] proposed a similar paper recommendation algorithm for academic social platforms, which first uses ANSJ for word segmentation and calculates TF-IDF weights, then maps papers to high-dimensional vectors using Word2Vec, and computes similarity using cosine similarity.

These studies only address academic paper search and recommendation, without considering that scholar projects also contain rich scholarly information, nor do they deeply investigate the collaborative association between papers and projects.

For scholar social networks, papers and projects contain many different feature attributes that require preprocessing and feature selection before mining relevant information. In data preprocessing and feature selection, Kira et al. [8] proposed the RELIEF algorithm, a classic feature weighting algorithm based on binary classification. This algorithm assigns different weights to features based on their correlation with categories, removing features with weights below a set threshold, and finally obtains the average weight of each feature. Since RELIEF is limited to binary classification problems, Zhang et al. [9] proposed a new RELIEF feature weighting algorithm by integrating maximum margin and maximum entropy theory, offering better adaptability for two-class, multi-class, and online data. However, this algorithm' s limitation lies in its inability to effectively remove redundant features, as it assigns high weights to all features highly correlated with categories. Ding et al. [4] proposed a two-step feature selection method combining RELIEF and K-means algorithms after analyzing

user feature information in social networks, achieving better feature sets. Experimental results also demonstrated that this algorithm is suitable for complex social network data with good performance.

To address information overload in social networks, recommendation systems have emerged. Among recommendation algorithms, collaborative filtering (CF) is one of the most successful algorithms for personalized recommendation systems. These algorithms primarily use user-item rating matrices for similarity calculation to find neighbor sets for target users. Currently, there are two main types: memory-based CF [10] and model-based CF [11]. Memory-based CF can be divided into user-based CF [12] and item-based CF [13]. These algorithms typically use user-item rating matrices, combined with similarity algorithms, to calculate similarities between different users/items, thereby finding the most similar users/items to form nearest neighbor sets and generate recommendations. In such collaborative algorithms, similarity calculation is a key step. As a measure of difference between two individuals, higher similarity indicates smaller differences, and recommendation quality based on similarity is typically better. Current similarity measurement methods mainly include cosine similarity [14], adjusted cosine similarity [15], and Pearson correlation coefficient [16], with appropriate methods selected based on actual data conditions.

In summary, our proposed model employs the two-step feature selection method combining RELIEF and K-means algorithms, which is more suitable for social network environments. On this basis, considering that paper and project feature attributes are primarily composed of natural language text, and building upon traditional collaborative filtering similarity calculation and traditional Vector Space Model (VSM) [17] feature similarity calculation, we adopt the TVSM model to calculate text similarity between papers and projects, find neighbor sets for target papers/projects, form recommendation sets, and combine them with users' personalized needs for recommendation.

2.1 Feature Selection Method

In the two-step feature selection method, the RELIEF algorithm offers high operational efficiency, noise tolerance, and is unaffected by feature interactions, making it suitable for complex social network data. The first step uses the RELIEF algorithm [8] to remove features irrelevant to paper-project associations from the feature matrix. Let the original dataset be D, with two feature sets in the original data representing papers and projects respectively. A sample R is selected from the training set D, composed of a p-dimensional vector $(x_1, x_2, ..., x_p)$, where p is the number of features and x_{Rj} is the value of the j-th feature of sample R. The distance between two samples R_1 and R_2 regarding feature j is defined as:

a) For non-numeric features:

$$diff(R_{1j}, R_{2j}) = \begin{cases} 0 & \text{if } R_{1j} = R_{2j} \\ 1 & \text{if } R_{1j} \neq R_{2j} \end{cases}$$

b) For numeric features:

$$diff(R_{1j}, R_{2j}) = \frac{|R_{1j} - R_{2j}|}{\max(j) - \min(j)}$$

where $\max(j)$ and $\min(j)$ represent the maximum and minimum values of feature j, respectively.

The algorithm finds the nearest neighbor sample H of the same class as R and the nearest neighbor sample M of a different class, then updates the weight of feature j based on the distance difference between sample R and its two nearest neighbors on feature j, as shown in Equation (3):

$$W_j = W_j - \frac{diff(R_j, H_j)}{m} + \frac{diff(R_j, M_j)}{m}$$

where W_j is the weight of feature j (initially 0), m is the number of randomly sampled instances, and R_i is the i-th sampled instance. After m iterations, the average weight of each feature is obtained. A larger weight value indicates better classification capability and greater representativeness of the category. After algorithm execution, the weight set T is sorted in descending order, and features with weights below a given threshold α are removed.

After the first step of RELIEF feature selection, irrelevant features are filtered out, but the resulting feature set still contains some redundant features. The second step addresses this by combining the K-means clustering algorithm [18]. K-means is a partition-based unsupervised algorithm that can simply and quickly solve clustering problems, offering good scalability and efficiency for large datasets. Through the first step, we obtain feature set F with irrelevant features removed. Given the required number of clusters k, k starting points are randomly selected as far apart as possible to serve as the centroids of k clusters. The remaining points in the dataset are then assigned to the nearest cluster based on their distance to each cluster centroid. For each cluster, the mean of all samples in the cluster is calculated as the new centroid. If the centroids converge, the process ends; otherwise, it continues iteratively, calculating distances from remaining points to new centroids and selecting new centroids until convergence or reaching the iteration limit, yielding the final clustering result. Similarity within the same cluster is high, while similarity between different clusters is low. Combining the feature weights obtained in the first step, redundant features with lower weight values within the same cluster are removed, resulting in the final effective features influencing paper-project associations.

The algorithm flow of the two-step feature selection method combining RELIEF and K-means is shown in Figure 1 [Figure 1: see original paper].

Two-Step Feature Selection Algorithm Pseudocode:

```
Input:
  Training set D = {(x , y ), (x , y ), ..., (x , y )}
  Sample size m
  Selected Rounds of Sample R
  Number of samples
  Threshold
  Feature Weighting Set T
  Feature Weighting W
  distance diff
  Number of clusters k

Process:
  1: T =  ; W = 0
  2: for i = 0 to m do
  3:   Randomly select sample R from D
  4:   Select the neighbor set H of R from samples of the same class
  5:   Select the neighbor set M of R from samples of different class
  6:   for j = 1 to p do
  7:     W  = W  - diff(R , H )/m + diff(R , M )/m
  8:   end for
  9:   T.append(D)
  10: end for
  11: Remove irrelevant features from T where W  <   and return
  12: F = remaining features after removing irrelevant features
  13: Randomly select k features from F as initial centroids
  14: centroids = {c , c , ..., c }
  15: repeat
  16:   for each feature f in F do
  17:     Compute distance between f and each centroid c
  18:     Assign f to the cluster with minimum distance
  19:   end for
  20:   Update centroid of every cluster
  21: until centroids converge or reach maximum iterations
  22: Remove redundant features with lower weights within each cluster

Output: Effective paper and project features
```

2.2 TVSM Model

In our model, to better achieve collaborative association between papers and projects, we need to provide more accurate recommendation results for target papers/projects, enabling users to better select corresponding papers/projects

from the recommendation set for association. After removing irrelevant and redundant features, and considering that paper and project information is primarily composed of natural language, we adopt the TVSM model to calculate similarity between different texts. First, paper and project information are separately segmented into word groups, where w_i represents the total number of words in a word group. Then each word in the group is assigned a weight value to construct a feature vector. The weight value is calculated using the TF-IDF method [19], as shown in Equation (4):

$$w_i = tf_i \times \log\left(\frac{n}{df_i}\right)$$

where tf_i represents the frequency of the current word w_i in a text, df_i represents the number of texts containing w_i, and n represents the total number of texts. The TF-IDF value for each word in each text is obtained, and a vector model is constructed for each text. Cosine similarity [14] is then used to calculate similarity between texts, as shown in Equation (5):

$$sim(D_1, D_2) = \frac{\sum_{i=1}^{m} w_{1i} \times w_{2i}}{\sqrt{\sum_{i=1}^{m} w_{1i}^2} \times \sqrt{\sum_{i=1}^{m} w_{2i}^2}}$$

Based on sorting by similarity from largest to smallest, the nearest neighbor set for target papers/projects can be formed to provide users with selected recommendation sets.

2.3 Collaborative Association Model

The framework of the paper/project collaborative association model is shown in Figure 2 [Figure 2: see original paper]. The main components include: (a) effective paper and project features selected through two-step feature selection; (b) similarity recommendation sets obtained by applying the TVSM model to calculate text similarity between papers and projects; (c) final recommendation results for target papers/projects formed by combining users' personalized needs; and (d) synchronized updating of paper-project associations.

The collaborative association model proposed in this paper is designed for the special scholar users of scholar social networks, providing collaborative association for their most representative academic achievements: papers and projects. Considering that paper and project information contains many irrelevant and redundant features, the RELIEF algorithm first assigns different weight values to features based on their correlation with categories, removing irrelevant features with weights below the threshold. Then, the K-means clustering algorithm divides features into k clusters based on similarity measurement, removing features with lower weights within clusters (i.e., redundant features). Through this two-step feature selection method combining these two algorithms, effective feature

selection and data preprocessing are achieved. Since paper and project information is primarily composed of text, the TVSM model is combined to calculate similarity between text features. Paper information and project information are first segmented, weights are assigned to each word through TF-IDF to construct feature vectors, and cosine similarity is finally used to calculate text similarity. After these calculations, different paper/project similarity recommendation sets can be obtained by sorting according to similarity magnitude. To better provide accurate and personalized recommendations for users, personalized requirement input is provided in SCHOLAT' s practical application, allowing users to find more accurate target projects/papers based on their own needs. On this basis, the paper-project collaborative association model is constructed, users perform collaborative association of papers and projects according to the model, and the system ultimately visualizes and synchronously updates the association results.

2.4 Collaborative Association Model Application

Our model uses paper and project data from SCHOLAT (http://www.scholat.com), an online academic information service platform for scholars that provides functions including scholar online communication, academic information management, and literature retrieval. Taking SCHOLAT' s dataset as an example, papers have more than a dozen features including author, title, source, keywords, abstract, type, etc., while projects have about 9 features. Evidently, both papers and projects have numerous features, among which some are effective and some are redundant for paper-project association relationships. Through the two-step feature selection method, we select the "author," "title," and "source" features from paper data and the "title," "participants," and "source number" features from project data, and apply the model to SCHOLAT to serve the broad scholar user community online. The model graph is shown in Figure 3 [Figure 3: see original paper].

In SCHOLAT' s practical application, users input their specific requirements for associated papers in the personalized requirement input box on the project interface. As shown in Figure 4 [Figure 4: see original paper], based on user-input specific information such as "temporal data management" and "temporal," the system combines the user input fields with the selected paper similarity recommendation set from the model to provide a final matching recommendation list, with results listed in reverse chronological order.

When users perform collaborative association of papers and projects, the association results are synchronized and updated through the paper-project collaborative association model and visualization model. The visualization results of paper interface associated with projects are shown in Figure 5 [Figure 5: see original paper].

Through the above research, the proposed paper-project data collaborative association model based on scholar social networks has been implemented and applied online to SCHOLAT, providing a broad scholar user community with

an accurate, personalized, and practical research information management tool. Through online questionnaires, we analyzed user satisfaction with recommendation results and whether users ultimately selected associated papers/projects. The final survey results show that SCHOLAT users are satisfied with this association function. The model demonstrates good recommendation performance, can provide better results based on users' personalized needs, and has good practicality.

3 Conclusion

This paper proposes a collaborative association model for paper and project data based on scholar social networks. In this model, irrelevant and redundant features from original paper and project data on SCHOLAT are first filtered through a two-step feature selection method combining RELIEF and K-means algorithms. Then, the TVSM model is used to calculate text similarity between papers and projects, forming paper/project similarity recommendation sets, and more accurate recommendations are selected by combining users' personalized needs. Finally, users select required information from the recommendation results for paper-project collaborative association, and the system simultaneously visualizes and updates the paper-project associations on both paper and project interfaces. Online application and user feedback on SCHOLAT demonstrate that the model has good accuracy and practicality, better helping scholar users manage research achievements and more deeply mine the rich resources and information embedded in paper-project relationships. In future work, we can further study academic relationships among scholar users through these associations, construct user knowledge graphs, perform user profiling, and provide better services.

References

[1] Xu R, Wang S, Zheng X, et al. Distributed collaborative filtering with singular ratings for large scale recommendation [J]. Journal of Systems & Software, 2014, 95 (9): 231-241.

[2] Isinkaye F O, Folajimi Y O, Ojokoh B A. Recommendation systems: principles, methods and evaluation [J]. Egyptian Informatics Journal, 2015, 16 (3): 261-273.

[3] Liu Junyu, Jia Xiuyi. Multi-label classification algorithm based on association rule mining [J]. Journal of Software, 2017, 28 (11): 2865-2878.

[4] Ding Rui, Zhu Jia, Tang Yong, et al. A novel feature selection strategy for friends recommendation [C]// Proc of International Conference on Computer Supported Cooperative Work in Design. 2016.

[5] Huang Yonghang, Tang Yong, Li Chunying, et al. Academic paper recommendation model based on community partition [J]. Journal of Computer Applications, 2016, 36 (5): 1279-1283.

[6] Chen Guohua, Tang Yong, Xu Yuying, et al. Research on academic semantic search using word vector representations [J]. Journal of South China Normal University: Natural Science, 2016, 48 (3): 53-58.

[7] Tang Zhikang, Li Chunying, Tang Yong, et al. Paper recommendation method based on scholar social platform [J]. Computer and Digital Engineering, 2017, 45 (2): 221-225.

[8] Kira K, Rendell L A. The feature selection problem: traditional methods and a new algorithm [C]// Proc of the 10th National Conference on Artificial Intelligence. AAAI Press, 1992: 129-134.

[9] Zhang Xiang, Deng Zhaohong, Wang Shitong, et al. Maximum entropy relief feature weighting [J]. Journal of Computer Research and Development, 2011, 48 (6): 1038-1048.

[10] Bellog, Alejandro N, Castells P, et al. Improving memory-based collaborative filtering by neighbour selection based on user preference overlap [C]// Proc of Conference on Open Research Areas in Information Retrieval. 2013: 145-148.

[11] Yin H, Cui B, Li J, et al. Challenging the long tail recommendation [J]. Proceedings of the VLDB Endowment, 2012, 5 (9): 896-907.

[12] Bellogin A, Parapar J. Using graph partitioning techniques for neighbour selection in user-based collaborative filtering [C]// Proc of ACM Conference on Recommender Systems. 2012: 213-216.

[13] Pirasteh P, Jung J J, Hwang D. Item-based collaborative filtering with attribute correlation: a case study on movie recommendation [M]// Intelligent Information and Database Systems. Springer International Publishing, 2014: 245-252.

[14] Dong Yangyi, Li Weihua, Yu Hui. Hierarchical relation mining of Chinese text based on mixed cosine similarity [J]. Journal of Computer Research and Development, 2017, 34 (5): 1406-1409.

[15] Ma Z, Yang Y, Wang F, et al. The SOM based improved K-means clustering collaborative filtering algorithm in TV recommendation system [C]// Proc of the 2nd International Conference on Advanced Cloud and Big Data. IEEE Computer Society, 2014: 288-295.

[16] Bobadilla J, Ortega F, Hernando A. Recommender systems survey [J]. Knowledge-Based Systems, 2013, 46 (1): 109-132.

[17] Sidorov G, Gelbukh A, Gómezadorno H, et al. Soft similarity and soft cosine measure: similarity of features in vector space model [J]. Computación y Sistemas, 2014, 18 (3): 491-504.

[18] Cohen M B, Elder S, Musco C, et al. Dimensionality reduction for k-Means clustering and low rank approximation [C]// Proc of the 47th ACM Symposium on Theory of Computing. 2015: 163-172.

[19] Huang Chenghui, Yin Jian, Hou Fang. A text similarity measurement combining word semantic information with TF-IDF method [J]. Chinese Journal of Computers, 2011, 34 (5): 856-864.

Note: Figure translations are in progress. See original paper for figures.