Massive Scientific Paper Mining: Modeling, Design and Implementation

Yang Zhou, Shufan Ji and Ke Xu

State Key Lab. of Software Development Environment
Beihang University, Beijing, 100191, P.R.China
{buaazhouyang@foxmail.com,jishufan@buaa.edu.cn,kexu@nlsde.buaa.edu.cn}

Abstract. With dramatic increasing of scientific research papers, scientific paper mining systems have become more popular for efficient paper retrieval and analysis. However, existing keyword based search engines, language or topic model based mining systems cannot provide customized queries according to various user requirements. Hence, in this paper, we are motivated to propose a novel TAIL (Time-Author-Institute-Literature) model to capture the relationships among literature, authors, institutes and time stamps. Based on the TAIL model, we implement the Massive Scientific Paper Mining (MSPM) system and set up a B/S (Browser/Server) structure for web services. The evaluation results on large real data show that our MSPM system could deliver desirable mining results, providing valuable data supports for scientific research cooperations.

1 Introduction

Scientific papers, which deliver the latest scientific research progress and achievement, are of great importance for scientific researchers to share ideas, find interested topics, identify potential research directions, and evaluate academic achievement. Nowadays, there are more than 5 million papers published each year, with an annual increase of 7% - 8%, which makes it impossible for researchers to "manually" track all relevant papers of their interested topics from such massive datasets. Therefore, scientific paper mining has been proposed for efficient paper retrieval and analysis.

Language model [1] is one of the most straightforward methods for paper retrieval, hence is widely adopted in scientific paper mining systems. Nowadays, there exist quite a few scientific search and analysis engines for scientific paper mining, such as Google Scholar [2] and Microsoft Academic Search [3]. However, those engines are usually limited to "keyword" search, which cannot satisfy various query demands. Even with the same keyword query, different users may have different expectations. For example, a user might want to get the most related papers that exactly contain the keyword, while another user might want to search for the papers under the topic about the keyword. However, the "keyword" search cannot capture the difference.

As language models usually search papers based on keyword frequency, ignoring the relationships among synonyms and the topics/themes of papers, topic models are proposed to effectively associate keywords and topics. Latent Dirichlet Allocation (LDA) [4] initiates the study of topic models. Then many researchers extend the topic models in different perspectives. Steyvers et al. [5] built probabilistic author-topic models to analyze the relationships between authors and topics. Wang et al. [6] introduced a topic-over-time (TOT) model to capture the time's effect on topic trend. More recently, Tang et al. [7] propose a patent mining method with a combination of the topic model and the language model. However, this work employs the product of the two models' values, thus users cannot balance the tradeoff degree of the two models according to their query expectations.

To satisfy various query expectations, in this paper, we propose a novel TAIL(Time-Author-Institute-Literature) model to capture the relationships among literature, authors, institutes and time stamps. The TAIL model is a combination of three models: Customized Model (CM), Author Model (AM), Institute Model (IM). The CM is a tradeoff-balanced combination of language model and probabilistic topic model, which could deliver various customized paper queries; while AM and IM could identify authors/institutes that are specialties at some hot research topics, as well as generate hot topic lists that are being studied by the authors/institutes, providing valuable data supports for scientific research cooperations. Based on the TAIL model, we implement the Massive Scientific Paper Mining (MSPM) system and set up a B/S (Browser/Server) structure for web services. The evaluation results on large real data show that our MSPM system could deliver desirable mining results for various user query expectations.

The rest of this paper is organized as follows. Section 2 introduces some preliminaries of this paper. Section 3 proposes the TAIL model and Section 4 introduces the implementation structure of TAIL-based MSPM system. We evaluate the performance of MSPM system in Section 5, and draw conclusions in Section 6.

2 Preliminaries

Here, we firstly define some notions that we will use in the paper, and then introduce the language model [1] and topic models [4][6] that we adopted.

2.1 Dictionary

The valid information of the papers is processed to generate multiple dictionaries, including the word dictionary W, the author dictionary A and the institute dictionary I. The elements in each dictionary are distinct. The notions relevant to those dictionaries are defined in Table 1.

Notation	Description
N_D	The number of papers.
N_W	The number of distinct words in all papers.
N_Z	The number of topics in the topic model.
N_A	The number of distinct authors in all papers.
N_I	The number of distinct institutes in all papers.
i, j, k, l, m	The indexes of the papers, the words, the topics, the authors and the institutes, respectively $(i = 1, 2, ..., N_D)(j = 1, 2, ..., N_W)(k = 1, 2, ..., N_Z)(l = 1, 2, ..., N_A)(m = 1, 2, ..., N_I)$.
$\mathbf{D} = \{d_1, d_2, ..., d_{N_D}\}$	The set of papers, where d_i refers to the i-th paper.
$\mathbf{W} = \{w_1, w_2, ..., w_{N_W}\}$	The set of distinct words, where w_j refers to the j-th word.
$\mathbf{Z} = \{z_1, z_2, ..., z_{N_Z}\}$	The set of topics, where z_k refers to the k-th topic.
$\mathbf{A} = \{a_1, a_2, ..., a_{N_A}\}$	The set of authors, where a_l refers to the l-th author.
$\mathbf{I} = \{i_1, i_2, ..., i_{N_I}\}$	The set of institutes, where i_m refers to the m-th institute.
N_i	The number of distinct words in the i-th paper.
N^j	The frequency of the j-th word in all the papers.
N_i^j	The frequency of the j-th word in the i-th paper.
n_{ik}	The frequency of the k-th topic in the i-th paper.
n_{kj}	The frequency of the j-th word assigned to the k-th topic.
ξ	Customized factor for the combination of language model and topic model.

Table 1. Notation List

2.2 Language Model

The language model is usually associated with a document in a collection. With a query Q as input, retrieved documents are ranked based on the probability that the document's language model would generate the terms of the query. According to the language model in [1], given a set of papers and a keyword, the relevance between the keyword and the specific paper can be calculated by Eq. (1).

$$\begin{aligned} p_{lm}(w_j|d_i) &= \frac{N_i}{N_i+\sigma} \cdot \frac{N_i^j}{N_i} + (1 - \frac{N_i}{N_i+\sigma}) \cdot \frac{N^j}{N_W} \\ &= \frac{N_i^j}{N_i+\sigma} + (1 - \frac{N_i}{N_i+\sigma}) \cdot \frac{N^j}{N_W} \end{aligned} \tag{1}$$

where N_i is the number of distinct words in the i-th paper, N_i^j is the frequency of the j-th word in the i-th paper, N^j is the frequency of the j-th word in all papers, N_W is the number of distinct words in all papers. σ is the Dirichlet smoothing factor and its value is set according to the average length of the papers in the database [1].

Generally, a query q typed in by users is often composed of multiple single words. Then the probability of a specific paper d_j generating a query q can be

calculated by Eq. (2).

$$p_{lm}(q|d_i) = \prod_{\{\forall j|w_j \in q\}} p_{lm}(w_j|d_i) \tag{2}$$

2.3 Topic Model

Probabilistic topic models are important tools for scientific paper mining, which identify the latent topics/themes of massive unstructured documents. In topic models, papers can be seen as random mixtures over various topics, each of which can be characterized by a distribution over words. According to the LDA model [4], the paper-topic distribution and the topic-word distribution, Θ and Φ, can be estimated by Eq. (3) and (4).

$$\theta_{ik} = \frac{n_{ik} + \alpha}{\sum_{k=1}^{N_Z}(n_{ik} + \alpha)} \tag{3}$$

$$\varphi_{kj} = \frac{n_{kj} + \beta}{\sum_{j=1}^{N_W}(n_{kj} + \beta)} \tag{4}$$

where n_{ik} is the frequency of the k-th topic in the i-th paper, n_{kj} is the frequency of the j-th word assigned to the k-th topic, α and β are the hyper parameters in the LDA model.

After getting Θ and Φ, we can derive the relevance between a word and a paper by Eq. (5).

$$p_{lda}(w_j|d_i) = \varphi_{\hat{k}j} \cdot \theta_{i\hat{k}} \tag{5}$$

where

$$\hat{k} = \arg\max_k \varphi_{kj} \tag{6}$$

Then the relevance between a query q and a specific paper can be derived by Eq. (7).

$$p_{lda}(q|d_i) = \prod_{\{\forall j|w_j \in q\}} p_{lda}(w_j|d_i) \tag{7}$$

To capture the effect of time on the trend of the topics, we adopted TOT [6] into our model. With TOT, the θ_{ik} and φ_{kj} defined in Eq. (3) and (4) become the definitions in Eq. (8) and (9).

$$\theta'_{ik} = \frac{n_{ik}^t + \alpha + \tau(n_{ik}^{t-1} + \alpha)}{\sum_{k=1}^{N_Z}(n_{ik}^t + \alpha) + \tau(\sum_{k=1}^{N_Z}(n_{ik}^{t-1} + \alpha))} \tag{8}$$

$$\varphi'_{kj} = \frac{n_{kj}^t + \beta + \tau(n_{kj}^{t-1} + \beta)}{\sum_{j=1}^{N_W}(n_{kj}^t + \beta) + \tau(\sum_{j=1}^{N_W}(n_{kj}^{t-1} + \beta))} \tag{9}$$

where the superscript ^{t} refers to the values at time t, and τ is the parameter that controls the effect of the values in the previous time on that of the current time.

3 TAIL Model

In this section, we will introduce the TAIL Model that contains the Customized Model (CM), the Author Model (AM), and the Institute Model (IM). Our proposed TAIL Model could well capture the correlation of topics, time stamps, authors, institutes and literatures.

3.1 Customized Model

As the language models are usually limited to keyword search, we combine the topic model with the language model for topic/theme related queries. Different from the combined models in [7], which employ the product of language model value and topic model value, we define a customized factor $\xi \in [0, 1]$ to balance the tradeoff of the two models. Users could set different ξ values to balance the tradeoff degree, according to their special query requirements. Thus, the relevance between a keyword and a paper under CM is defined in Eq. (10).

$$p_{cm}(w_j|d_i) = \xi \cdot p_{lm}(w_j|d_i) + (1 - \xi) \cdot p_{lda}(w_j|d_i) \tag{10}$$

It should be noted that the language model and topic model are the special cases of CM where $\xi = 1$ and $\xi = 0$, respectively. Similarly, the relevance between a query and a paper is defined in Eq. (11).

$$p_{cm}(q|d_i) = \prod_{\{\forall j|w_j \in q\}} p_{cm}(w_j|d_i) \tag{11}$$

3.2 Author Model and Institute Model

To capture the authors' and the institutes' expertise on specific research areas, we propose model-based analysis of authors and institutes by the Author Model (AM) and the Institute Model (IM), respectively.

To get the ranking list of authors/institutes at some hot research topics, we should measure the relevance score of an author/institue and the papers under specific topics, as well as generate hot topics associated with each author/institute, indicating which topics are the author's/institute's specialties. The methodologies of these two models are similarly summarized as follows.

First, the author/institute information is extracted from each paper. We define ad and id in Eq. (12) to record whether an author/institute is associated with a paper.

$$\begin{aligned} ad_{li} &= \begin{cases} 1, & \text{if } a_l \text{ is among the authors of } d_i; \\ 0, & \text{otherwise.} \end{cases} \\ id_{mi} &= \begin{cases} 1, & \text{if } i_m \text{ is among the institutes of } d_i; \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{12}$$

Second, two matrices Θ and Φ are calculated by the LDA model[4], which are the paper-topic distribution and the topic-word distribution, respectively.

Third, the relevance scores of the author a_l/institute i_m on the topic z_k, are calculated by Eq. (13) and Eq. (14), respectively.

$$p_A^{lk} = \sum_{i=1}^{N_D} ad_{li} \cdot \theta_{ik} \tag{13}$$

$$p_I^{mk} = \sum_{i=1}^{N_D} id_{mi} \cdot \theta_{ik} \tag{14}$$

Finally, as topics are anonymous and abstract concepts, we associate the topics with users' queries. The relevance score of an author a_l on a query q is derived by Eq. (15).

$$p_A^l(q) = \prod_{\{\forall j | w_j \in q\}} p_A^{l\hat{k}} \tag{15}$$

where

$$\hat{k} = \arg\max_k \varphi_{kj} \tag{16}$$

Similarly, the score of an institute i_m on a query q is derived by Eq. (17).

$$p_I^m(q) = \prod_{\{\forall j | w_j \in q\}} p_I^{m\hat{k}} \tag{17}$$

With the AM and IM, we could analyze massive scientific paper resources to accurately deliver authors/institutes that are specialties at some hot research topics, as well as generate hot topic lists that are being studied by the authors/institutes, providing valuable data supports for scientific research cooperations.

4 Massive Scientific Paper Mining (MSPM) System

In this section, we will demonstrate the implementation structure of our TAIL model-based Massive Scientific Paper Mining(MSPM) System. As is shown in Fig. 1, MSPM system is set up as a B/S structure, divided into four levels: data plane, model plane, application plane and user interface.

In the data plane, MSPM system keeps crawling meta data from the Internet. At the same time, valuable information from the paper meta data is extracted, cleaned, and stored as the valid data for model-based analysis.

In the model plane, the valid data from the data plane are imported into the TAIL model, processed for the application plane. Firstly, the words are assigned to LM, LDA and CM for data preprocessing. Then AM and IM help obtain the ranking lists of authors and institutes based on the relevance scores, while the TOT model involving time stamps helps generate the topic trends. Moreover, the coauthor networks and cooperation networks among institutes could also be clearly identified.

In the application plane, various mining applications are provided based on the modeling data. Customized requirements from the user interface are delivers to the model plane, while the mining results are delivered to the user interface.

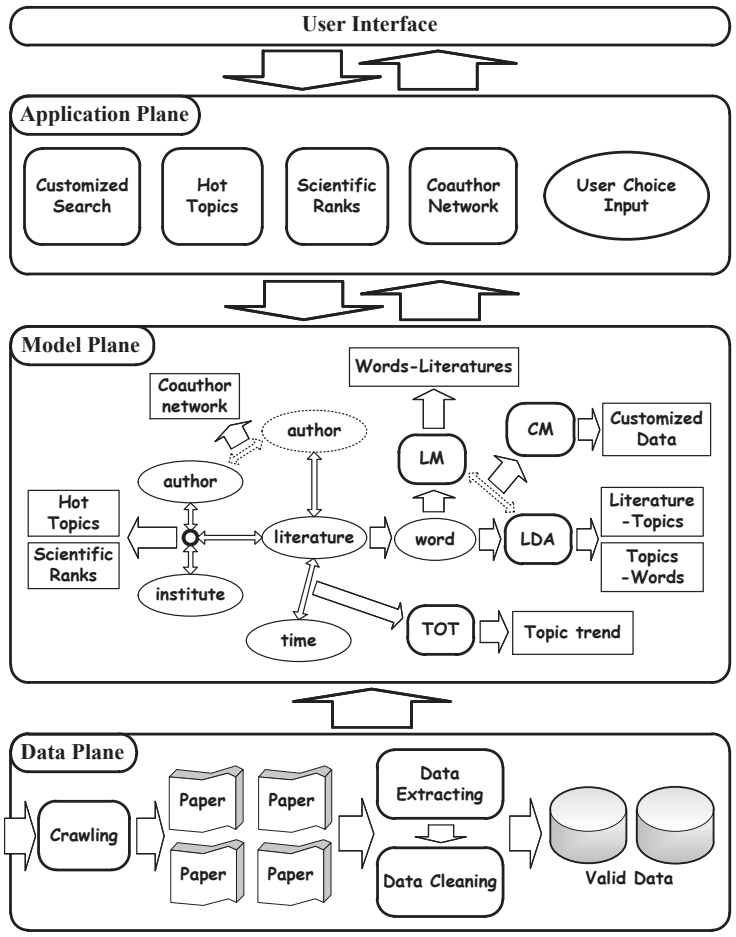

Fig. 1. System Design

5 Evaluation

To evaluate the performance of MSPM system, we experimented on large real data, containing more than 2.76 million papers from 6877 journals, published from 2005 to 2011. Each paper has structured information of title, keywords, abstract, authors, institutes, and etc.

5.1 Customized Paper Query

The basic function of MSPM system is the customized paper query. Users could provide the keywords and set the customized factor ξ, according to their query expectations. Table 2 shows the top 10 query results of the keyword "social

Rank	Paper Title
1	Social Network Type and Subjective Well-being in a National Sample of Older Americans
2	An Experience-Sampling Study of Depressive Symptoms and Their Social Context
3	Social outcomes after temporal or extra temporal epilepsy surgery: A systematic review
4	Social dysmetria' in first-episode psychosis patients
5	Constraining heterogeneity: the social brain and its development in autism spectrum disorder
6	Viscous democracy for social networks
7	Self-concept and psychopathology in deaf adolescents: preliminary support for moderating effects of deafness-related characteristics and peer problems
8	Comparison of Anxiety-Related Traits Between Generalized and Non generalized Subtypes of Social Anxiety Disorder
9	Controllability of Boolean control networks with time delays in states
10	Minimal social network effects evident in cancer screening behavior

Table 2. Sample of Paper Query Results (keywords='social network' and $\xi = 0.2$)

network" with $\xi = 0.2$, from which we can see that MSPM system could deliver closely related query results.

In addition, Fig.2 shows the trends of paper ranks with ξ changing from 0 to 1. It is clear that the ranks of some papers change dramatically with the change of ξ, while those of the other papers have no obvious changes. Thus, the customized factor could well distinguish papers of different properties.

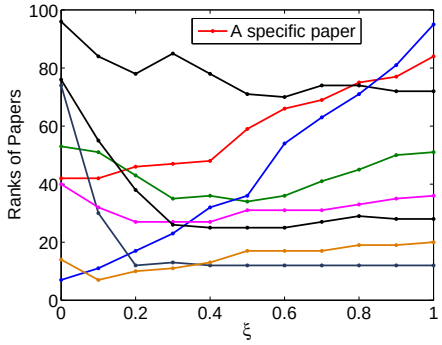

Fig. 2. Ranks of Sample Papers with Various ξ

5.2 Author/Institute Mining

With specific keyword queries, users could get the top authors and institutes with the highest relevance scores. The authors and institutes in the resulting lists are known to be the specialties in the keyword related research domains. Table 3 shows the sample of top authors and institutes mining results with the keyword "social network". Note that the results are for reference only due to data set limitations.

Top Authors	**Top Institutes**
Kleinberg J	The Hebrew University in Jerusalem
Jackson M	Cornell University
Wellman B	University of Maryland
Faloutsos C	Stanford University
Newman M	Carnegie Mellon University

Table 3. Sample of Top Authors/Institutes Mining Results (keywords='social network')

5.3 Precision of Query Results

To evaluate the performance of customized query, 5 graduate students are invite to judge whether the papers, authors and institutes in the returned results are relevant to their query expectations. If a paper/author/institute is rejected by more than one student, it will be regarded as irrelevant and imprecise result. We launch 50 queries with different keywords for each model, and the average precision is calculated based on the students' judgements. As for our CM model, the query precision is further optimized with different ξ, denoted by CM_Opt.

Table 4 shows the average precisions of different models, with #N representing the precision in the top N papers. It is shown that the precision of CM is superior to that of LM [1] and LDA [4]. Moreover, CM models with larger N have lower precisions, which indicates that focusing on fewer query results will lead to better query performance. In addition, the precisions of AM and IM indicate that users could get desirable results from the author and institute mining.

6 Conclusion and Future Work

In this paper, we propose a novel TAIL model to capture the correlation of topics, time stamps, authors, institutes and literatures for massive scientific paper mining. The TAIL model defines a customized factor to balance the tradeoff of the language model and the topic model, providing customized paper queries for users. Based on the TAIL model, we implement the Massive Scientific Paper

Model	**AP**	**Model**	**AP**
LM#10	0.720	CM_Opt#5	0.852
LDA#10	0.684	CM_Opt#10	0.850
AM#10	0.768	CM_Opt#20	0.819
IM#10	0.734	CM_Opt#50	0.804

Table 4. Precision of Different Models (**AP**=Average Precision)

Mining (MSPM) system and set up a B/S structure to provide web services. The evaluation results on large real data show that our MSPM system could deliver desirable mining results for various user query expectations. As for future work, our model could be further optimized by measuring paper quality and popularity based on citations, which would result in more interesting returned papers and author/institute ranking lists.

Acknowledgment

This work was supported by the National 863 Program of China (No. 2012AA011005) and Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20111102110019).

References

1. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: Proceedings of the 24th International ACM SIGIR conference on Research and development in information retrieval, ACM (2001) 334–342
2. : Google scholar. `http://scholar.google.com/`
3. : Microsoft academic search. `http://academic.research.microsoft.com/`
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3** (2003) 993–1022
5. Steyvers, M., Smyth, P., Rosen-Zvi, M., Griffiths, T.: Probabilistic author-topic models for information discovery. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2004) 306–315
6. Wang, X., McCallum, A.: Topics over time: a non-markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2006) 424–433
7. Tang, J., Wang, B., Yang, Y., Hu, P., Zhao, Y., Yan, X., Gao, B., Huang, M., Xu, P., Li, W., Usadi, A.K.: Patentminer: topic-driven patent analysis and mining. In: Proceedings of the 18th ACM SIGKDD, ACM (2012) 1366–1374