

Increasing Serendipity of Recommender System with Ranking Topic Model

Zhibo Xiao^{1,*}, Feng Che¹, Enuo Miao² and Mingyu Lu^{1,*}

¹ Information Science Technology Department, Dalian Maritime University, Dalian, 116026, China

² Computer Teaching Department, Dalian Naval Academy, Dalian, 116018, China

Received: 1 Sep. 2013, Revised: 4 Dec. 2013, Accepted: 5 Dec. 2013

Published online: 1 Jul. 2014

Abstract: There are thousands of academic paper published each year, it is quite hard for researchers who enters a new field to discover relevant paper and novel paper to read, which we characterize as choice overload problem. Recommender system can help to alleviate the problem, but recommender system suffers from the intention gap problem which is the incapability of the system to accurately guess users' intentions. We proposed a ranking topic model based semantic recommendation framework which helps to introduce serendipity to the system. First, the proposed ranking topic model reorders learnt topic distributions according to users' intentions. Then, learnt ordered topics are used as features to rank papers in the library according to the relevancy to user query. At the same time, ranked topics also provide novelty to the results. Since there is little work on how to evaluate the serendipity degree of recommender system, we proposed two measure to evaluate this metric. We performed empirical experiments to test the efficiency of proposed framework with state-of-the-art counterparts, the comparison results revealed the superiority of our proposed algorithms. In the end, we illustrated our algorithms with an example and pointed out future research directions.

Keywords: Academic paper recommender system, ranking topic model, serendipity, serendipity evaluation

1 Introduction

[1,2] first proposed choice overload effect. It means that people usually are drown to large candidate choice set to seek more values, but such large candidate set would cause trouble making the choice, which ultimately leads to lower satisfaction and choice delay. The choice overload effect has brought serious troubles for researchers when reading academic papers and selecting academic papers. As suggested by DBLP statistics, each year, academic publications grows dramatically (as is shown in figure 1) [3]. For example, papers published in year 2010 are three times published in year 2000. Besides, each paper would cite around 10 to 20 references. Facing such humongous amount of papers, a new comer entering a research field would have little clue selecting relevant papers to read. Normally, one would resort to search engine, but the result of search engine also goes beyond the normal comprehension capacity of a person. With limited prior knowledge, it is also very difficult for new comer to compare the qualities of different papers and decide which one to read.

Meanwhile, reading low-level papers for a long time would increase the frustration caused by stagnation. Under these circumstances, we propose to use recommendation system to help new researchers to alleviate the academic paper choice overload problem.

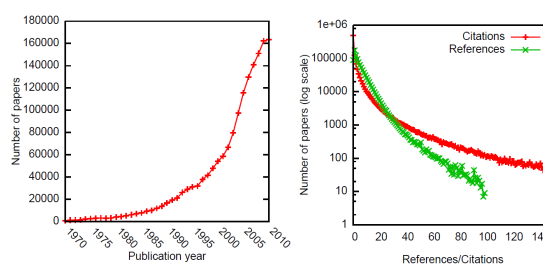


Fig. 1: Annual publication number (left), Reference number (right) from DBLP

* Corresponding author e-mail: xiaozhibo@gmail.com, lumingyu@dlmu.edu.cn

When candidate set grows, it gets more attractive, this is because the overall profit rate (sum of individual profit rate) increases. Three factors contribute to the total opportunity cost:

1. choice burden brought by comparing all the candidates (comparing more products brings more work);
2. prospect disappointment rate (since overall product number is larger);
3. expectations thanks to larger candidate set (people usually expect better goods in a larger candidate set).

The total opportunity cost cannot simply be balanced by the profit provided by large candidate set owing to above three factors, which further amplify the psychology cost of choice making. Hence, recommender system is desperately needed.

For academic reading process, good recommendation can alleviate the choice overload problem by greatly *reducing the time input* in academic reading process.

1. **Overcome language barrier:** For most non-English speaking researchers, it is quite hard to judge whether a paper is relevant or not and whether it is with high quality without reading the majority of the paper, and this would really cause a lot of time. Since our proposed recommender system analyse paper corpus using topic models which performs topic abstraction process usually requires years of specific experience, even for English mother-tones. At the same time, our model rank the topic relevance which heavily relied on academic literacy, the overall effect to the non-English speaking researchers is quite helpful.
2. **Quality control:** Beginners usually don't possess the ability to judge a paper's quality. Since our proposed system stems from topic analysis, with the help of ranking, it can automatically exclude non-relevant topics and prevailing-but-meaningless topics, hence making relevant topics more prominent. If a reader reads an irrelevant paper, then he lost time that can otherwise read a better one. This contributes to the frustration caused by stagnation. Our proposed recommender system can prevent this from the very first beginning part of academic reading process.
3. **Lower the expectation:** If a system can output stable relevant and yet novelty paper to users, then users will ease the expectation to read high level paper, since he/she constantly get the expected papers.

It is well known that Latent Dirichlet Allocation (LDA)[4] is de facto method to analyse latent topics of text corpus after its release. While three years before LDA, a model very much like LDA, which described a model-based clustering method for using multilocus genotype data to infer population structure and assign individuals to populations, was independently invented in [5]. If a researcher in text mining would have read this paper and introduced this model to text mining community, the landscape of text mining might be different. Though from a foreign area, model proposed in

[5] works well in text mining as was proved by LDA and its following variants. Discovering an effective model originated from a foreign area is what researchers might call serendipity. In this paper, we define paper provides **novelty content with high relevance** as serendipity paper. Without doubt, serendipity paper is what researchers expected from an academic paper recommender system.

In current recommender system researches, the majority of systems adopt error-based metric to evaluate the performance of the system, while in most cases, expecting serendipity is the sole purpose of using recommender system for most users [6]. Error-based metric can only evaluate whether an already known item is useful to user, it cannot measure the extra value recommender system provides. Besides, error-based metric has too much uncertainty. Other than user himself/herself, explicit and implicit ratings gathered by recommender system are far more adequate to depict users' interests, we call this difference the **intention gap**. Users' likings, interests and intentions also vary with objective and subjective effects that beyond recommender system's capacity. **Error-based metric cannot fully evaluate a recommender system's performance, and would further enlarge the intention gap between user and recommender system.** If recommender system cannot perfectly depict users' likings, providing results with diversity would be a nice choice [7].

Academic papers contain far more semantic information than regular commodity descriptions and reviews. Academic paper itself is the semantic carrier, the commodity description and its subsidiary comment can only served as supplementary material for recommender system. This is the biggest differences between commodity recommendation and academic paper recommendation. For this reason, extra methods need to be incorporated to process and utilize valuable semantic information. Topic models have been applied in various scenarios analysing the semantics of the corpus, not to mention recommender system [8,9]. For the past 10 years, topic model algorithms blossomed in various research fields since LDA. Topic models have been extended both in theory and application. However, as [10] put it in his review, topic models is not the full stop, they should be used to produce other more meaningful results.

Topic models have been developed with information engineering applications in mind. As a statistical model, however, topic models should be able to tell us something, or help us form a hypothesis, about the data. What can we learn about the language (and other data) based on the topic model posterior?

A crucial problem that hinders topic model is that the learnt topic distributions have no orders, they can not be distinguished between each other, as is shown in figure 2.

Ranking has been studied extensively in information retrieval. According to The Probability Ranking Principle proposed by [11, 12]:

If retrieved documents are ordered by decreasing probability of relevance on the data available, then the system's effectiveness is the best that can be obtained for the data.

It is natural to deduct that if topic distributions are ordered, not only the effectiveness of the model can be boosted, but also can help to calculate the relevance of each topic to a query paper or a query topic. Specifically, topics can be used as a high-level feature to re-organize the corpus. As is shown in figure 2, in most topic models, topics are not ordered according to the query. Ranked topics can not only highlight relevant topics to the query, but also hold back the non-relevant ones. The few highlighted/selected relevant topics then can be served as higher level semantic features to recommend papers. This is clearly different from query-by-keyword or query-by-example retrieval scheme usually adopted in information retrieval and recommender system. Since a paper usually covers multiple topics, selected papers according to some topics can bring about other relevant topics to the retrieved ones, in this way, novel concepts are introduced, hence serendipity of the recommender system is enhanced. With ordered topics, only observed data is need to introduce concepts, furthermore, extra information can also be incorporated into the model. On the other hand, as is in our definition of serendipity, relevance is one of the two ingredients, learning an ordered topic distribution list can assuredly benefit serendipity, hence to enhance the performance of recommender system.

Ranking topic models can not only discover latent topic distributions in the corpus, but can also rank them according to user's intention by order them to the query. Since topic is an abstract concept which positioned in the middle of low level term frequency and high level term meaning, they can express certain amount of meanings and they can be displayed and further built on based on learnt probability distributions. In this way, topics is used as soft clustering labels to organize documents, we call them *soft* because they are not hand-appointed by person. Furthermore, it is because they are different from people's choices to represent the middle level features of corpus that they can bring novelty into systems. In ranking topic models, ranking schemes are then used to re-order topic distributions according to users' intentions given specific queries or inexplicit ones.

So the problems boil down to how to learn an ordered topic distributions given a certain topic or a certain concept. In order to solve this problem, a query-based ranking topic model is proposed in this paper. The basic idea is to make use of the posterior distributions learnt by topic models and then re-order the topics according to their relationships and relevance to the query paper.

The main contributions of this paper are:

Topic 247		Topic 5		Topic 43		Topic 56	
word	prob.	word	prob.	word	prob.	word	prob.
DRUGS	.069	RED	.202	MIND	.081	DOCTOR	.074
DRUG	.060	BLUE	.099	THOUGHT	.066	DR.	.063
MEDICINE	.027	GREEN	.096	REMEMBER	.064	PATIENT	.061
EFFECTS	.026	YELLOW	.073	MEMORY	.037	HOSPITAL	.049
BODY	.023	WHITE	.048	THINKING	.030	CARE	.046
MEDICINES	.019	COLOR	.048	PROFESSOR	.028	MEDICAL	.042
PAIN	.016	BRIGHT	.030	FELT	.025	NURSE	.031
PERSON	.016	COLORS	.029	REMEMBERED	.022	PATIENTS	.029
MARIJUANA	.014	ORANGE	.027	THOUGHTS	.020	DOCTORS	.028
LABEL	.012	BROWN	.027	FORGOTTEN	.020	HEALTH	.025
ALCOHOL	.012	PINK	.017	MOMENT	.020	MEDICINE	.017
DANGEROUS	.011	LOOK	.017	THINK	.019	NURSING	.017
ABUSE	.009	BLACK	.016	THING	.016	DENTAL	.015
EFFECT	.009	PURPLE	.015	WONDER	.014	NURSES	.013
KNOWN	.008	CROSS	.011	FORGET	.012	PHYSICIAN	.012
PILLS	.008	COLORED	.009	RECALL	.012	HOSPITALS	.011

Fig. 2: Topics derived from Latent Dirichlet Allocation, as can be seen, they can't be distinguished from each other. Topic number only serves as topic index.

1. Paper-user relation model is proposed based on item-user relation model to suite the academic paper recommendation;
2. Topic ranking based semantic recommendation framework is proposed, the framework can not only provide extra semantic to recommendation but also increase the reliability of the system;
3. A serendipity-based evaluation method is proposed.

In chapter 2, we review related researches on serendipity problem in recommender system and ranking topic models. In chapter 3, we propose our model of semantic recommendation framework based on ranking topic models, since there is no consensus measure on evaluating the degree of serendipity in recommender system, we propose our own serendipity measure in chapter 4. In chapter 5, we perform comparison experiments to test the effectiveness of our methods. In chapter 6, we provide an example to further illustrate the effectiveness of our proposed method. In the last chapter, we conclude our work and point out future research directions.

2 Related Works

2.1 Serendipity in Recommender System

[13] discussed choice overload effect from information overload perspective, they defined choice overload effect as the choice difficulty when faced with large candidate set, they pointed out that choice overload problem and its internal psychology process have been intensively studied in many other fields [14], but few works discussed its effect in recommender system, let alone a comprehension framework. [8] initially proposed the concept of diversity in recommender system, the evaluate diversity alone with accuracy, as the higher accuracy is, the lower diversity goes. [15] further clarified the concept of diversity in recommender system. Specially, they pointed out the deficit of using error-based metric in evaluating academic paper recommendation. Since users can always get papers without recommender system, what they expect from the

system is the novel results. [16] provided the evidence that recommendation techniques can be a way to solve the information overload problem, which further provides siding evidence for our work. [17] defined top-N recommendation, and further pointed out that error-based evaluation methods were inappropriate to evaluate top-N recommendation results. They all had some helpful discussion on how to bring diversity into recommender system, but didn't provide an unified framework. [18] evaluated five precision metrics and three recommender systems, they found that result derived from error-based and precision-based metric has no direct connection, and it is questionable to use precision-based metric to evaluate recommender system, which also raise the need to adopt serendipity-based evaluation methods.

2.2 Ranking topic models

Topic model algorithms get wide recognition since LDA model [4]. w represents all the words in the corpus, which is the only observed variable, θ is the latent variable representing the topic distribution of each document, z is the variable which denotes the topic assignment probability of each topic to each term, α, β are model parameters. For each document, there is a θ_m governs the proportion of each topic in this document. In every document, each term corresponds to a z_n , deciding which topic this term w belongs to. LDA learns the distribution of latent variables θ, z with techniques like variational inference or gibbs sampling with fixed α, β . Then α, β is learnt given θ, z .

Compared with conventional topic analysis techniques, such as TF-IDF [19], LSI [20], pLSI [21], LDA avoids "overfitting" problem. Since then, a sequence of work, like Correlated topic models [22], Online LDA [23], Hierarchical Dirichlet Process [24], MedLDA [25] continues to push the development of topic models. LDA based models become the de facto problem when analysing topics.

Topic over time proposed [26] and dynamic topic models [27] both considered the time-stamp of documents by incorporating new random variables in the models, these two models can be seen as re-ranking topic distributions by time. [28] proposed the problem of re-ranking topic distributions according to their importance. They defined important topics and irrelevant topics in three different manners, and then used weighted scores derived from three manners to rank topic distributions accordingly. They first raised the problem that an ordered topic list with ranking is necessary for the model. [29] proposed methods to select the appropriate words to represent topics, this can be seen as re-ranking terms in each topic. They proposed a series of features to depict the importance of terms in topic, and selecting important terms in topic via features is more suitable than just using probability information. Though from different perspectives, the work by Alsumait and Lau revealed the

fact that a ranked topic list is necessary. [30] proposed four topic ranking methods and did a thorough job to evaluate the importance of topics. [31] proposed a correlated topic model based ranking topic model, then the proposed model was used to perform multi-document summarization, the experiments showed that the proposed model outperformed summarization based on topic model without ranking techniques, proved that ranking topic models can actually boost the performance of other related tasks and also prove the effectiveness of ranking topic models. [32] is an improvement of [33], both works investigated how link structures can influence topic discovery. The drawback of their works are clear that they need link information between documents to work.

In academic paper recommendation, getting a top-N recommendation list which orders in a decreasing fashion according to a query paper or a query topic is the basic requirement. Given the affluent semantic hidden in academic papers, using topic model to analyse their latent meaning is a natural choice. Considering both points, designing a ranking topic model which can get ordered topic distributions matches perfectly to the requirement of academic paper recommendation.

3 Semantic recommendation framework based on topic ranking

Since the semantic in academic paper is far more abundant than other commodity items, we propose a three-tier semantic recommendation framework, as is shown in figure 3. The highest tier is the contribution to serendipity from paper itself, including the novelty to user, popularity of the paper and the quality of the paper. The middle tier is the contribution to serendipity from topics, including the correlation between topics and the distance between topics. The lowest tier is the terms' contribution to the serendipity, specifically, terms with topic information is considered to evaluate its contribution to the serendipity. The proposed three-tier semantic recommendation framework can be further classified into two categories: the first category concerns paper's attributes, the novelty part in serendipity is mainly evaluated by this category; the second category is the content of the paper, including the topics and terms of the paper, which are the middle and the lowest tiers of the model, these two tiers have the closest connection, the relationships between term and topic greatly rely on the selection of the topic when calculating the serendipity score of each paper, choosing the appropriate topic among all the topic is the key problem to increase the serendipity of the recommendation.

In this section, we will introduce our semantic recommendation framework focusing on paper novelty and paper relevance.

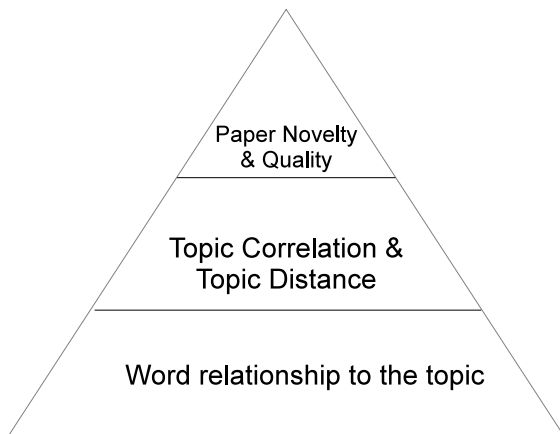


Fig. 3: Three-tiers semantic recommendation framework

3.1 Paper Novelty

The state-of-the-art novelty metrics are largely based on popularity, two novelty metrics are defined in [6], one is by taking the opposite of popularity score, and the other way to measure novelty is to calculate the distance between items. These two metrics didn't take semantic content into consideration. In academic paper recommendation scenario, the content of the papers are represented by topics, so we only need to calculate the novelty of one paper against each user, if a paper is not read by this user, it is novel to this user.

[34] first brings about "Harry Potter" effect in recommender system, the same problem exists in academic paper recommendation. Famous tutorial and survey papers are usually serve as entry point for most researchers, they are also the easiest result often returned by search engine, so in order to increase serendipity in our system, we have to deal with this problem first.

For dataset containing user reading history, for the i th user u_i in all N users, and the j th paper c_j in all M papers, we define the novelty of c_j towards u_i as:

$$p_{nov}(ij) = 1 - \mathcal{R}[ij] \times popularity(j) \quad (1)$$

If u_i hasn't read paper c_j , $\mathcal{R}[ij]$ is the indicator function $\mathcal{R}[ij] = \mathbb{1}[ij] = 1$; if user u_i has read c_j before, $\mathcal{R}[ij] = 1 - 1/t$, here t indicates the time gap between recommendation point and when user has read this paper, we call this time gap "ageing factor". For computation simplicity, if the time gap is less than a year, we approximately set $t = 1$. The longer user has read a paper, the more likely user will forget about he/she has read this paper, which feel more novel to this user; on the contrary, the novel degree will be quite small if user has just read it. A paper c_j 's novelty to a user is defined as:

$$popularity(j) = \frac{\sum_i \sum_j \{1 - \mathbb{1}[ij]\}}{N} \quad (2)$$

Given users' reading history, the "ageing factor" can help mitigating the "Harry Potter" effect to some extend, since famous and high-cited-numbers are usually read first by users and then users go into detailed areas to read papers on more specific areas.

Almost all academic paper databases and indexing databases, like Google Scholar, Microsoft Academic Search, IEEE Xplore Digital Library, provide statistics about each paper, like downloaded times. Downloaded times is a great indicator to reflect the popularity of one paper. We defined c_j 's downloaded popularity as:

$$p_d(j) = \frac{d_j}{\sum_{j \in M} d_j} \quad (3)$$

d_j is the downloaded time of paper c_j .

Combined with above two metrics, we can finally define the novelty of paper c_j to user u_i :

$$p_{nov} = p_{nov}(ij) + p_d(j) \quad (4)$$

As is discussed in [35], since we only concern the relevant ranking of each paper, the aggregation method used to combine p_{nov} and p_d is meant to provide a score to distinguish each other, then we simply adding these two score to avoid miscellaneous design.

3.2 Paper Relevance

As is mentioned above, the semantic of papers are represented by topics, in order to calculate the correlation between topics, all the papers are processed by topic models. It is worth mentioning that derived topic distributions by most topic models cannot distinguish each other by any measure, let alone adjust the ordering according to users' needs. Topic distributions without ranking is suitable for tasks like browsing corpus, since recommendation system need to select only a few papers for each user, so papers must be ordered according to certain rule. In our academic paper recommendation system, we adopted topic model to perform topic analysis, which makes it inevitable to solve the topic ranking problem in order to recommend.

First, we train a topic model on the corpus, and then we can derive the topic distribution for each paper c_j . Normally, an academic paper would have no more than three topics, if indeed there are more than three topics in one paper, there some topics are meant to be more dominant. Here we make the assumption that each paper has three majority topics. We choose the top three topics with highest probabilities, then these three topics will be served as recommendation seed to select semantically similar papers. For example, the proportion of the top three topic in c_j are $t_1 : 50\%, t_2 : 20\%, t_3 : 10\%$, then we select 5 most similar topics to t_1 , 2 most similar topics to t_2 , and 1 most similar topic to t_3 . In this way, we find more topics than the original seed, and bring more

diversity to increase serendipity. At the same time, this method can prevent the recommended paper being too similar to the original one due to limited topic numbers.

We use weighted topic coverage to rank topics. Originated from the idea of TF-IDF, weighted topic coverage presume topics with more probabilities are more important than others, but prevailing topics with high probabilities in lots of papers make themselves naturally lack novelty, which is useless to create serendipity. Weighted topic coverage is taken here to prevent this problem, in this case, we not only consider the correlations between topics but also take term-topic relationship into consideration. In this way, the important topics which covers the main ideas of the document can be highlighted. For surveys and tutorials, they usually covers many topics and each topic only take a small amount of contents, their novelty level is relatively low under weighted topic coverage. Together with paper novelty measure in equation (4), “Harry Potter” effect is further eased.

The weighted topic coverage is defined as:

$$\mu(z_k) = \frac{\sum_{m=1}^M N_m \cdot \theta_{m,k}}{\sum_{m=1}^M N_m} \quad (5)$$

The differences between topics are defined as:

$$\sigma(z_k) = \sqrt{\frac{\sum_{m=1}^M N_m \cdot (\theta_{m,k} - (\mu(z_k)))^2}{\sum_{m=1}^M N_m}} \quad (6)$$

N_m is the length of paper, z_k is the topic-term assignment probability in topic models as convention goes, θ is the paper-topic proportion in topic models. Then the ordering of topic k is defined as:

$$O_k \triangleq (\mu(z_k)) \cdot (\sigma(z_k)) \quad (7)$$

[30] proposed other topic ranking techniques such as Laplacian score and topic similarity, since they require dataset have labeling information, they are not always suitable for academic paper recommendation.

After getting the top three topics t_{j1}, t_{j2}, t_{j3} for paper c_j , we can derive the relevant topics for each three topics, and find the corresponding papers containing the relevant topics. Since the sum of probabilities of top three topics are not bigger than 1, the final number of derived topics are not bigger than 10. Choosing the highest probability paper in each topic can get the ultimate recommendation list.

In the end, we summary our proposed recommender framework as follows:

4 Serendipity Evaluation Metrics

In the field of information retrieval, Kendall’s τ is often used to compute the correlation between two ranked lists. [36] defined ranking 1:

Require: Corpus matrix, $N_{n \times m}$;
 Query paper, c_j ;
 User reading history;

Ensure: Recommended paper list, L ;

- 1: Given $N_{n \times m}$, train LDA model;
- 2: Learn query paper c_j ’s topic proportion with learnt LDA model;
- 3: Rank the corpus topics according to query paper’s weighted topic coverage;
- 4: Calculate the paper novelty according to user’s reading list;
- 5: Recommend paper according to paper relevance and paper novelty;
- 6: Calculate the serendipity value of the recommendation result.
- 7: **return** L ;

Algorithm 1: Framework of semantic recommendation based on ranking topic model

Definition 1 (Ranking) Distance σ generates a permutation of all the objects, so-called ranking, where the objects are ordered according to their distances to query.

Each topic distribution is a permutation of vocabulary, in this way, each topic is a degenerated ranking list.

As discussed in section 2.1 there is no verdict on how to evaluate serendipity. Specifically, in academic paper recommendation, there is little work around, so we adapt two well-known metrics and propose two evaluation metrics evaluate serendipity in academic paper recommendation: similarity between recommendations and re-recommend matching degree.

4.1 Recommendation Correlation

In the field of information retrieval, Kendall’s τ is often used to compute the correlation between two ranked lists. Given two lists with N items, there are C pairs of items that have the same rank in both rankings, D pairs have the opposite rankings, the the Kendall’s τ value between two lists is defined as:

$$\tau = \frac{C - D}{N(N - 1)/2} \quad (8)$$

In most times, discrepancies among those items having high rankings are more important than those with lower rankings. The Kendall’s τ does not make such distinctions and equally penalizes errors both at high and low rankings. The AP correlation (τ_{ap}), that is based on average precision and has a probabilistic statistic gives more weight to the errors at high rankings and has nice mathematical properties with makes it easier to interpret.

Let $L1$ and $L2$ be two lists both with N items, $L2$ is the actual ranking and $L1$ is the ranking of items whose

correlations with $L2$ we would like to compute. Pick any random item in $L1$ other than the top ranking item, which we denoted as l_i . Next pick any item l_j which has higher ranking than l_i if these two items are in the same relative order in $L2$, then return 1, otherwise, return 0. Then the expected outcome can be written as:

$$p' = \frac{1}{N-1} \cdot \sum_{i=1}^N \frac{C(i)}{(i-1)} \quad (9)$$

where $C(i)$ denotes the number of items above rank i and correctly ranked with respect to the item at rank i in $L1$. The AP correlation coefficient τ_{ap} is defined as:

$$\tau_{ap} = p' - (1-p)' = 2p' - 1 = \frac{2}{N-1} \cdot \sum_{i=2}^N \left(\frac{C(i)}{i-1} \right) - 1 \quad (10)$$

The AP correlation coefficient is not a symmetric statistic. AP correlation coefficient is suitable for situation where an actual ranking exists. In some cases, one would like to compute the correlation among two ranked lists where we do not have the notion of "actual" rankings. In such case, a symmetric version of the statistics could be used, which is defined as:

$$symm\tau_{ap}(L1, L2) = \frac{\tau_{ap}(L1 | L2) + \tau_{ap}(L2 | L1)}{2} \quad (11)$$

The essence of AP correlation is to accumulate the mismatching ranks compare to the "actual" rank. In our situation, there is no way to get the "actual" rank without asking the user, and the "actual" rank may vary even to the same user given different time and context. Built from the idea of aggregation of mismatching ranks, we treat seed paper as a degenerated list and calculate the distance between recommended paper to seed paper as mismatching degree to seed paper.

We denote similarity between recommendations of seed paper c_j as sim_j . sim_j can serve as a method to measure the novelty level of the recommendation of paper c_j . We use cosine measure to calculate the similarity of two paper, the reason is two folds: First, in $symm\tau_{ap}(L1, L2)$, the symmetric measure is not reflexive given two lists. When comparing two papers, the calculation sequence is irrelevant, the angle between the two document vectors is important, not the actual Euclidean distance. Second, because the document vectors are very sparse, documents under the cosine measure are efficiently indexable by the inverted index.

$$sim_j = \frac{\sum_{l=1}^{N_j} \sum_{m=1, m \neq l}^{N_j} \cos(c_{jl}, c_{jm})}{2 \cdot N_j} \quad (12)$$

here, N_j is the top- N recommendation number for paper c_j which is predefined by system or requested by user, in our paper, N_j corresponds to our four settings as $N_j = \{5, 10, 15, 20\}$. $l, m \in N_j$ is the index for papers in recommendation pool. Since sim_j calculates the similarity between papers twice, we balance it in the denominator.

4.2 Re-recommend matching degree

Besides recommendation correlations, recommendation diversity and the robustness of the recommend are important parts in serendipity.

Given a recommend seed paper c_j , its recommended K papers form a cluster. In this cluster, the distance between recommended paper to the seed paper represents the serendipity degree between them. With these K papers as seed to recommend, another K clusters can be formed. Some may contain the original seed paper c_j , some are not, as is shown in figure 4. The central dot is the original

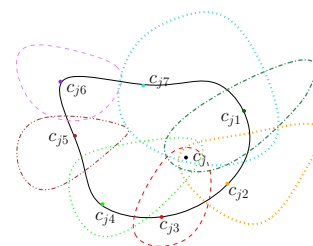


Fig. 4: Re-recommend degree : paper c_j as recommend seed, after the cluster is formed, each paper is used as seed to form another recommend cluster. Cluster formed by paper $c_{j1}, c_{j2}, c_{j3}, c_{j4}, c_{j7}$ contain original seed c_j , while the other clusters don't contain it. Since recommended papers each contain several topics which are not the same with the seed paper, their serendipities to the seed paper are all different, which results each cluster not regularly shaped.

seed paper c_j , we illustrated that 7 papers were recommended using c_j , denoted as c_{j1}, \dots, c_{j7} . Solid line is used to form a cluster using these 7 dots. Using these papers as seed, another round of recommend can form another 7 clusters(denoted in different colours and line styles). In these 7 clusters, some contains the original seed, like cluster $c_{j1}, c_{j2}, c_{j3}, c_{j4}, c_{j7}$; cluster c_{j5}, c_{j6} don't contain c_j .¹

Since our recommendation scheme is based on topics, after two times of recommendation, the topics may drift apart to the seed paper, this results the non-overlapping clusters like c_{j5}, c_{j6} . These kinds of clusters provide the serendipity of the system. For cluster $c_{j1}, c_{j2}, c_{j3}, c_{j4}, c_{j7}$, they provide the robustness of the system.

Re-recommend matching degree counts counting how many items matches the original recommendation seed, and is denoted as D_{re} .

$$D_{re} = \frac{1}{R \cdot S} \sum_{i=1}^R \sum_{k=1}^S \Delta(c_j, c_{ik}) \quad (13)$$

In above equation, R denotes the first time recommendation number, i.e. how many paper are

¹ Cluster is denoted in bold font, like c_{j4} . Paper is denoted in plain font, as c_j .

recommended with seed paper c_j , and they are denoted as c_{ji} , ($i = 1, \dots, R$), S is the second time recommendation number, i.e. how many paper are recommended with seed paper c_{ji} , which are denoted as c_{ik} , ($k = 1, \dots, S$). The function $\Delta(c_j, c_{ik})$ measure how many times second time recommendation agrees with the original paper c_j . If c_j appears in the cluster c_{ik} , then $\Delta(c_j, c_{ik}) = 1$; if After getting the recommendations for paper c_j , using the recommendation list as new input to get further recommend, then calculating how many times paper c_j appears in the new recommendation lists.

Proposed re-recommending degree is similar to the ball-overlap factor (BOF) proposed by [37], which is defined as

$$BOF_k(\mathbb{S}, d) = \frac{2}{|\mathbb{S}^*| * (|\mathbb{S}^*| - 1)} \sum_{\forall o_i, o_j \in \mathbb{S}^*, i > j} \text{sgn} \left(((o_i, \delta(o_i, kNN(o_i))) \bar{\cap} (o_j, \delta(o_j, kNN(o_j)))) \right)$$

where $\delta(o_i, kNN(o_i))$ is the distance to o_i 's k th nearest neighbor in a sample of the database \mathbb{S}^* and $(o_i, \delta(o_i, kNN(o_i)))$ is the ball in metric space centered in o_i of radius $\delta(o_i, kNN(o_i))$. The statement $\text{sgn}((\cdot, \cdot) \bar{\cap} (\cdot, \cdot))$ returns 1 if the two balls geometrically overlap, and 0 if they do not. The BOF_k calculates the ratio of overlaps between ball regions, where each region is made up of an object (from the database sample) and of a covering radius that guarantees k data objects are located inside the ball. The overlap ratio then predicts the likelihood that two arbitrary ball-shaped regions will overlap or not.

Different from BOF, in D_{re} , clusters formed by paper are not organized in ball shape. The target of D_{re} is solely the original paper c_j , so there is no need to count in kNN region.

It is easy to prove that $\Delta(c_j, c_{ik})$ satisfies reflexivity, non-negativity and symmetry, but doesn't satisfy triangle inequality, which makes $\Delta(c_j, c_{ik})$ a non-metric distance. Being non-metric distance, $\Delta(c_j, c_{ik})$ can provide better *robustness* [37]. $\Delta(c_j, c_{ik})$ is resistant to outliers, anomalous and "noisy" objects.

5 Experimental results and discussions

5.1 Dataset

To prove the efficiency of our proposed framework, we use CiteULike dataset in [9]. The dataset contains CiteULike users' profile and users' reading information from 2004 to 2010. After removing redundant and void information and users who have read less than 10 papers, the dataset contains 5551 users and 16980 papers, there are 204986 user-paper pair. On average, each user keeps a 37 paper reading list, the least number is 10, the biggest number is 403, over 93% of the users read less than 100

papers. For each paper, the dataset only keeps the title and the abstract (since CiteULike didn't provide full content), after removing stopwords, there are over 8000 unique terms.

5.2 Experiment Design

For topic model algorithm, we use Online variational Bayes for LDA and batch LDA algorithm proposed by [38] and we adopted the implementation provided in Gensim by [39]. Three sets of experiments are carried out to evaluate our algorithm.

1. We first discuss the effect of different topic numbers, recommendation number to the ranking scheme to decide the optimal setting of the experiment.
2. Afterwards, we compare the result of our work with CTR algorithm proposed in [9], since we adopted their dataset. Also CTR is based on topic model, which makes the comparison just.
3. In the end, we compare our algorithm with the degenerated algorithm that has no ranking scheme, to further test the effect of the ranking scheme.

We train topic models with topic number to be 50, 100 and 150. With the implementation of Gensim and given dataset, training a topic model with 150 topics will requires three days on a Pentium E7400 2.8 Ghz CPU 3G RAM machine, so we didn't train models with larger topic number. The intention of this section is to evaluate how different topic number would affect the final result, the optimal topic number for the dataset is beyond the scope of this paper. With trained models, we get the recommendation list, the top-N numbers are chosen to be 5, 10, 15, 20. We randomly choose 20 papers from the dataset and run the whole experiment and then report the averaged result. The number 20 is an empirical number we choose.

5.2.1 Effect of Different Topic Number on Serendipity

We first test how different topic number will affect the serendipity of the recommendation. Experimental algorithm is our proposed recommendation algorithm based on ranking topic model.

In figure 5, the horizontal axes is the recommended paper number, the vertical axes is the average similarity between recommendations, which means that the similarity is averaged on TopN, in this way, numbers in different settings can be compared. From figure 5, we can see that the similarity between recommender papers are all stays at a very low level, as recommended paper number increases, the similarity decrease accordingly, which proves that our proposed framework indeed brings about serendipity to the recommender system. When there are 150 topics in trained model and the output recommend paper number is 5, due to this imbalance

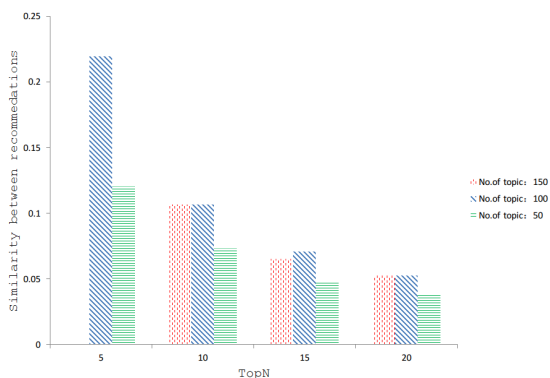


Fig. 5: Different topic number’s effect on recommended paper similarity

setting, the recommendation result is not satisfactory. Setting the recommended paper number fixed, we can clearly see that models with high topic number’s similarity is higher than models with small topic number, in three reported setting, models with 100 topics and 150 topics are almost the same, this states that for this dataset, higher topic number will get better result, but when topic number reaches a certain level, training model’s topic number doesn’t affect the recommendation result.

In figure 6, as TopN number increases, the averaged matching degree decreases as expected in all settings. Given fix TopN number, training model with higher topic number has larger matching degree, which means larger topic number makes recommendation more robust.

5.2.2 Effects on different topic models

CTR(Collaborative Topic Regression) model is a recommendation model combined with traditional collaborative filtering with topic modelling. CTR represents users with topic interests and assumes that documents are generated by a topic model. Besides CTR

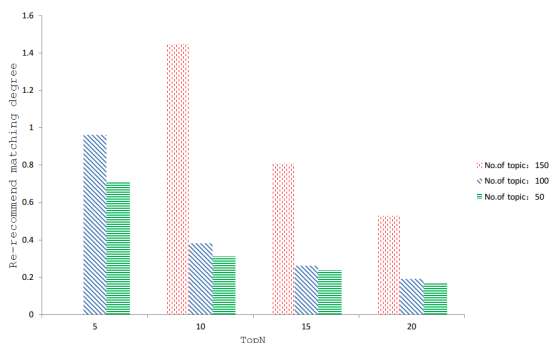


Fig. 6: Topic number’s effect on re-recommendation matching degree

model users’ rating with a latent variable which interacts with topic modelling. Based on the result of previous section, we set the topic number as 150, then compare the results of two algorithms.

In figure 7, the blue dotted bar represent the result of our algorithm, the red dashed bar is the result of CTR. As is shown, our algorithm constantly outperforms CTR when TopN recommend number increase on recommend paper similarity.

In figure 8, the the blue dotted bar represent the result of our algorithm, the red dashed bar is the result of CTR. Our algorithm also performs better than CTR on re-recommending matching degree.

In both two figures, CTR’s performance stays at a relatively low level. Although RTM leads CTR in paper similarity measure, but the margin is not too much, especially when TopN number increases, the differences between two algorithms get smaller. Although on paper similarity, two algorithms performs almost equally well. On re-recommending degree, RTM great outperforms CTR, which indicates RTM can provide more serendipity than CTR. Since both models are based on topic models, the results indicates that re-order topic distributions can indeed highlights important topics in relation to the query paper, and ignores the irrelevant topics concerning query paper.

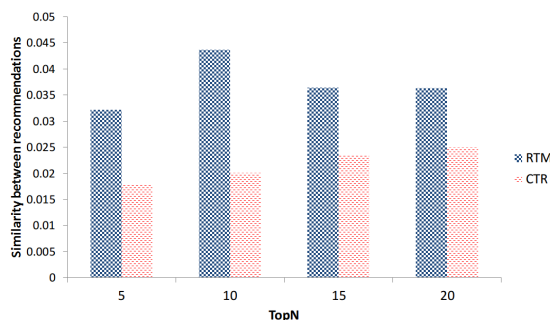


Fig. 7: Comparison result on recommended paper similarity

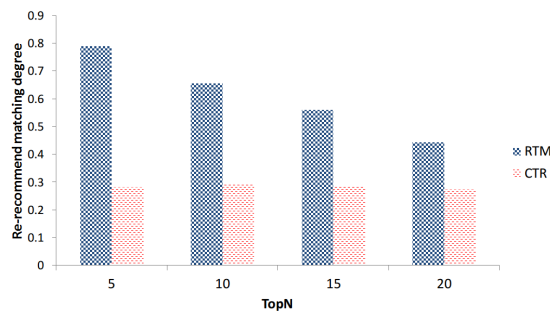


Fig. 8: Compression result on re-recommendation matching degree

5.2.3 Effect on Ranking Scheme on Serendipity

Based on results of previous sections, we further compare the result on how ranking scheme affect the recommendation. Since we only concern ranking factor, we make other factors fixed. We train a 100 topic models and recommend 5, 10, 15, 20 papers each. We also run our experiment on 20 different input papers and report the averaged number. From figure 9, we can see that

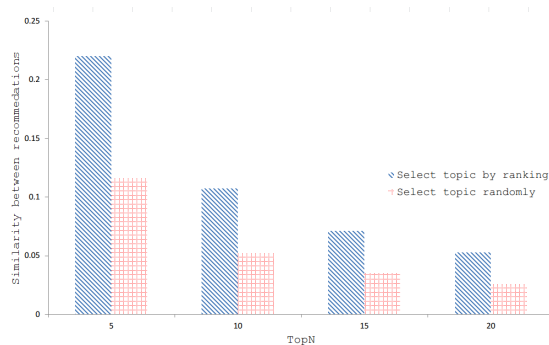


Fig. 9: Recommended paper similarity on different ranking scheme

similarities of both models are all stays at a low level, which proves that topic models can improve the serendipity of recommendation system. Models with ranking topic models clearly outperform models with randomly selected topics, this indicates that ranking scheme can not only bring about novelty but also can make recommendation more stable. From figure 10, we

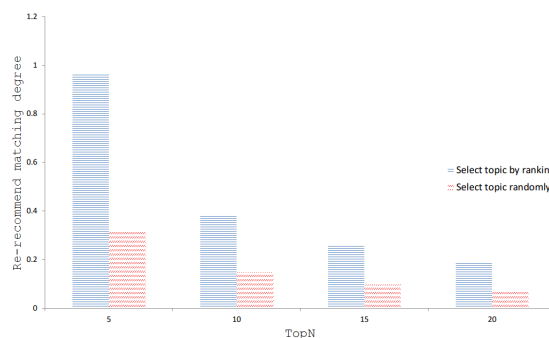


Fig. 10: Re-recommend matching degree on different ranking scheme

can also clearly see that models with ranking topic model outperform models with randomly selected topics. Figure 10 shows that re-recommend matching degree of ranking topic model recommendation is higher, which shows that the recommendation is more stable.

6 An example

To illustrate our proposed algorithm, we provide an example which uses our proposed algorithm to recommend. The seed paper is called “Exploring complex networks” and we present recommended paper based on it. We highlight words from different topic with different colours. In the seed paper, we colour words in topic “network” with red, words in topic “physics” blue, words in topic “biology” with green.

The original abstract of the seed paper is:

*The study of **networks** pervades all of science, from neurobiology to statistical **physics**. The most basic issues are structural: how does one characterize the wiring diagram of a food **web** or the Internet or the metabolic network of the **bacterium** *Escherichia coli*? Are there any unifying principles underlying their **topology**? From the perspective of nonlinear dynamics, we would also like to understand how an enormous **networks** of interacting dynamical systems—be they **neurons**, power stations or lasers—will behave collectively, given their individual dynamics and coupling architecture. Researchers are only now beginning to unravel the structure and dynamics of complex networks.*

The recommended documents with our proposed algorithm are as follows:

Large n field theories **string theory** and **gravity**
 Exploiting generative models in discriminative classifiers
 Motifs in **brain** networks
Spike timing dependent synaptic plasticity depends on dendritic location
 The structure and function of **complex networks**
Spike timing dependent plasticity from synapse to **perception**
 Collective dynamics of small world **networks**
 Activity dependent scaling of quantal amplitude in **neocortical neurons**
 Hierarchical organization in **complex networks**
 Scalefree **topology** of **email networks**
 Point process models of single **neuron** discharges

As can be clearly seen, the recommendation provided with our results covers all the topics and each recommendation result is relevant to the seed paper. What is even more appealing is that in some recommendation, multiple topics exist, which can provide serendipity to users.

Below is the recommendation results without ranking schemes:

As can be seen that, only several keywords are covered compared to our proposed scheme and there are clearly irrelevant results provided. As is defined previously, serendipity results are results with relevancy and novelty. Some results in the above list are clearly novelty but without relevance to the seed paper. These kinds of results are not serendipity, they are not relevant.

Above comparison clearly shows that recommender system with ranking topic model scheme clearly outperform regular recommender system.

7 Conclusion and Future Work

In this paper, an academic paper recommender system is proposed to solve the choice overload problem most researchers faced when they enter a new research field. Specifically, we proposed to build recommender system based on ranking topic models, we improved and expanded item-user relation model, proposed a paper-user relation model suitable for academic paper recommendation. In the end, we proposed two novel serendipity evaluation metrics. We performed comparison experiments with state-of-the-art counterparts, the results showed that recommender system with ranking topic model outperforms models without ranking schemes, and when training topic model's topic number increases at a certain level, training topic models' topic number doesn't affect the ranking scheme.

While designing our framework and carrying out experiments, we discovered that further studies needs to be done:

1. Human judgement and relevance feedback. Since topic models are unsupervised learning algorithms, and our analysis object is text, there is no clear label can be used. Recommender system tries to guess user's intention, although some objective evaluation methods can be used to evaluate the result, but still, user's direct judgement is the best feedback to further optimize the algorithm.
2. Incorporating ranking and topic discovery in a holistic process. Although current ranking scheme apply to any topic model, but now these two processes are separated, we plan to merge these two processes, to directly output ranked topic distributions.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No. 61272369, 61073133, 61175053, 61105117, 61033012; Science and Technology Planning Project of Dalian City (2011A17GX073; 2010E15SF153) and the Fundamental Research Funds for the Central Universities No. 3132013335.

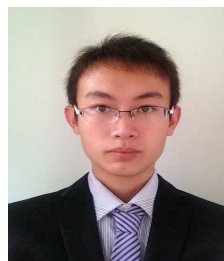
References

- [1] Sheena S Iyengar, Gur Huberman, and Wei Jiang. How much choice is too much? contributions to 401 (k) retirement plans. *Pension design and structure: New lessons from behavioral finance*, 83–96 (2004).
- [2] Sheena S Iyengar, Mark R Lepper, et al. When choice is demotivating: Can one desire too much of a good thing? *Journal of personality and social psychology*, **79**, 995–1006 (2000).
- [3] Onur Küçüktunç, Erik Saule, Kamer Kaya, and Ümit V Çatalyürek. Diversifying citation recommendations. *arXiv preprint arXiv:1209.5809*, (2012).
- [4] David M Blei, Andrew Y Ng, and Michael Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, **3**, 993–1022 (2003).
- [5] Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, **155**, 945–959 (2000).
- [6] Saül Vargas and Pablo Castells. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, ACM, 109–116 (2011).
- [7] Daniel Fleder and Kartik Hosanagar. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management science*, **55**, 697–712 (2009).
- [8] Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, ACM, 22–32 (2005).
- [9] Chong Wang and David M Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 448–456 (2011).
- [10] David M Blei. Probabilistic topic models. *Communications of the ACM*, **55**, 77–84 (2012).
- [11] Stephen E Robertson. The probability ranking principle in ir. *Journal of documentation*, **33**, 294–304 (1977).
- [12] Stephen Robertson and Hugo Zaragoza. *The Probabilistic Relevance Framework*, Now Publishers Inc, **3**, (2009).
- [13] Dirk Bollen, Bart P Knijnenburg, Martijn C Willemsen, and Mark Graus. Understanding choice overload in recommender systems. In *Proceedings of the fourth ACM conference on Recommender systems*, ACM, 63–70 (2010).
- [14] Benjamin Scheibehenne, Rainer Greifeneder, and Peter M Todd. Can there ever be too many options? a meta-analytic review of choice overload. *Journal of Consumer Research*, **37**, 409–425 (2010).
- [15] Mi Zhang and Neil Hurley. Avoiding monotony: improving the diversity of recommendation lists. In *Proceedings of the 2008 ACM conference on Recommender systems*, ACM, 123–130 (2008).
- [16] Toine Bogers and Antal Van den Bosch. Recommending scientific articles using citeulike. In *Proceedings of the 2008 ACM conference on Recommender systems*, ACM, 287–290 (2008).
- [17] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth*

- ACM conference on Recommender systems, ACM, 39–46 (2010).
- [18] Alejandro Bellogin, Pablo Castells, and Ivan Cantador. Precision-oriented evaluation of recommender systems: an algorithmic comparison. In *Proceedings of the fifth ACM conference on Recommender systems*, ACM, 333–336 (2011).
- [19] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, (1986).
- [20] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, **41**, 391–407 (1990).
- [21] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 50–57 (1999).
- [22] David M. Blei and John D Lafferty. Correlated topic models. In *Advances in Neural Information Processing Systems 18*, (2006).
- [23] Matthew Hoffman, David Blei, and Francis Bach. Online learning for latent dirichlet allocation. In *Advances in Neural Information Processing Systems 23*, 856–864 (2010).
- [24] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, **101**, 1566–1581 (2006).
- [25] Jun Zhu, Amr Ahmed, and Eric Xing. Medlda: Maximum margin supervised topic models. *Journal of Machine Learning Research*, **1**, 1–48 (2010).
- [26] Xuerui Wang and Andrew McCallum. 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, 424–433 (2006).
- [27] David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, ACM, 113–120 (2006).
- [28] Loulwah AlSumait, Daniel Barbará, James Gentle, and Carlotta Domeniconi. Topic significance ranking of lda generative models. In *Machine Learning and Knowledge Discovery in Databases*, Springer, 67–82 (2009).
- [29] Jey Han Lau, David Newman, Sarvnaz Karimi, and Timothy Baldwin. Best topic word selection for topic labelling. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, Association for Computational Linguistics, 605–613 (2010).
- [30] Yangqiu Song, Shimei Pan, Shixia Liu, Michelle X Zhou, and Weihong Qian. Topic and keyword re-ranking for lda-based topic modeling. In *Proceedings of the 18th ACM conference on Information and knowledge management*, ACM, 1757–1760 (2009).
- [31] Zhibo Xiao, Di Wu, Qingfeng Li, Yuting Xu, and Mingyu Lu. Corsum: multi-document summarization algorithm based on ranking topic models. In *CCML 2013*, (2013).
- [32] Dongsheng Duan, Yuhua Li, Ruixuan Li, Rui Zhang, and Aiming Wen. Ranktopic: Ranking based topic modeling. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, IEEE, 211–220 (2012).
- [33] Yizhou Sun, Jiawei Han, Jing Gao, and Yintao Yu. itopicmodel: Information network-integrated topic modeling. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*, IEEE, 493–502 (2009).
- [34] Greg Linden. Early amazon: Similarities. <http://glinden.blogspot.com/2006/03/early-amazon-similarities.html>, (2006).
- [35] Ammar Ammar and Devavrat Shah. Ranking: Compare, don't score. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, IEEE, 776–783 (2011).
- [36] Tomáš Skopal and Benjamin Bustos. On nonmetric similarity search problems in complex domains. *ACM Computing Surveys (CSUR)*, **43**, 34 (2011).
- [37] Tomáš Skopal. Unified framework for fast exact and approximate search in dissimilarity spaces. *ACM Transactions on Database Systems (TODS)*, **32**, 29 (2007).
- [38] Matthew Hoffman, David M Blei, and Francis Bach. Online learning for latent dirichlet allocation. *Advances in Neural Information Processing Systems*, **23**, 856–864 (2010).
- [39] Radim Rehurek and Petr Sojka. Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, ELRApages, May, 45–50 (2010).



Zhibo Xiao is a Ph.D candidate in Dalian Maritime University. His research interest is machine learning, mainly he focuses on research on topic models, also he extends his research on automatic summarization and recommender system. He has published research articles in international journals of machine learning. He is referee and editor of machine learning journals.



Feng Che is a master degree candidate in Dalian Maritime University. His research interest are machine learning, recommender system, and automatic summarization.

Enuo Miao is an assistant teacher in Dalian Naval Academy. He received the master degree at Liaoning Normal University (China). Her main research interests are: machine learning, digital water marking.



Mingyu Lu is a Professor in Dalian Maritime University. He was the Founder and Director of Intelligent Technology Research Center(ITREC) and vice head of Information Science Technology department. He received the PhD degree at Tsinghua



University (China). He is referee and Editor of several international journals in the frame of data mining and machine learning. His main research interests are: data minging, machine learning, business intelligence. He is the senior member of China computer society, member of China Machine Learning committee.