Research Paper Search Using a Topic-Based Boolean Query Search and a General Query-Based Ranking Model

Fukuda, Satoshi
Kyushu University

Tomiura, Yoichi
Kyushu University

Ishita, Emi
Kyushu University

https://hdl.handle.net/2324/2551020

出版情報：Database and Expert Systems Applications (Lecture Notes in Computer Science). 11707, pp.65-75, 2019-08-06. Springer Nature
バージョン：
権利関係：

Research Paper Search Using a Topic-based Boolean Query Search and a General Query-based Ranking Model

Satoshi Fukuda, Yoichi Tomiura, and Emi Ishita

Kyushu University, Fukuoka, Japan 819-0395
{s.fukuda, tom}@inf.kyushu-u.ac.jp
ishita.emi.982@m.kyushu-u.ac.jp

Abstract. When conducting a search for research papers, the search should return comprehensive results related to the user's query. In general, a user inputs a Boolean query that reflects the information need, and the search engine ranks the research papers based on the query. However, it is difficult to anticipate all possible terms that authors of relevant papers might have used. Moreover, general query-based ranking methods emphasize how to rank the relevant documents at the top of the results, but require some means of guaranteeing the comprehensiveness of the results. Therefore, two ranking methods that consider the comprehensiveness of relevant papers are proposed. The first uses a topic-based Boolean query search. This search converts every word in the abstract set and query into a topic via topic analysis by Latent Dirichlet Allocation (LDA) and conducts a search at the topic level. The topic assigned to synonyms of a search term is expected to be the same as that assigned to the search term. Each paper is ranked based on the number of times it is matched with each topic-based Boolean query search executed for various LDA parameter settings. The second is a hybrid method that emphasizes better results from our topic-based ranking result and a general query-based ranking result. This method is based on the observation that the paper sets retrieved by our method and by a general ranking method will be different. Through experiments using the NTCIR-1 and -2 datasets, the effectiveness of our topic-based and hybrid methods are demonstrated.

Keywords: Latent Dirichlet Allocation, Research paper search, Search recall

1 Introduction

When searching for research papers, it is important that the search returns comprehensive results related to a user's information needs. In many cases, a user inputs a Boolean query that reflects his/her information needs to an academic search engine, and acquires the research papers that are most closely related to the query. However, there are two main problems when searching for papers in such a search engine.

The first problem is a construction of query. When a user searches research papers for a method of extracting a hierarchical relationship between words, for example, the user first defines a Boolean query that expresses his/her information needs as follows: "hierarchical relationship" AND extract. If this query produces any hits, then the results are likely to satisfy the information needs. However, there are many synonyms and similar expressions for “hierarchical relationship” and “extract.” Therefore, the user might extend the query to (“hierarchical relationship” OR “hierarchical structure”) AND (extract OR acquire). Unfortunately, this query does not return relevant papers in which, for example, “hierarchical relationship” or “hierarchical structure” appears but neither “extract” nor “acquire” appears. This query also misses relevant papers if the author uses terms such as “obtain” or “get” instead of “extract” and “acquire.” Similarly, the above query also misses relevant papers that use terms such as “superordinate-subordinate relation” and “part-whole relationship,” which are specific words for “hierarchical relationship” in the field. Thus, it is difficult for a user to fully predict how representations of search terms might have been expressed by other authors.

The second problem concerns the ranking model to find the relevant papers. In daily search, it is important that several documents ranked highly satisfy the information needs, because we need only check some highly ranked documents to find those that satisfy the information needs. However, in an academic search, it is not sufficient to check only highly ranked documents: the comprehensiveness of relevant documents is also important. More specifically, completeness when confirming across a certain amount, that is, the cumulative recall up to rank *r*, is important, and a high recall is desirable for realistic values of *r* (e.g., *r* = 1,000 for 90% recall).

Therefore, we propose two ranking methods and model a Boolean query-based search by considering the comprehensiveness of relevant papers.

(1) Search method based on a topic-based Boolean query

We propose a search method in which every word in each abstract in the research paper set (determining of the research paper set is described in Section 3) and each search term in the Boolean query is converted to a topic using Latent Dirichlet Allocation (LDA) [4], and then topic-level matching is conducted. By converting words to topics, abstracts that do not include the search terms but include synonyms or similar expressions are matched. In the topic analysis by LDA, the same topic is assigned to words that tend to appear together in many abstracts. For example, if words w, w_1, w_2, and w_3 appear together in many abstracts and words w', w_1, w_2, and w_3 appear together in many other abstracts, then the same topic tends to be assigned to words w, w', w_1, w_2, and w_3 in such abstracts. In this case, w and w' have the potential to be semantically similar; that is, when word w is specified in the query, a research paper that matches a Boolean query in which w is replaced by w' could be a relevant search result. Using this characteristic of LDA, we construct a query search system that conducts a search based on the topics assigned to the search terms and collects papers that exactly match the topic-based Boolean query from the topics assigned to each word in the abstracts.

Using the topic-based Boolean query search described above, we propose a new ranking method. First, we set the range of parameters assigned to LDA, and execute a topic-based Boolean query search for each parameter setting. The system then sorts research papers in descending order of the number of times they matched the query search for each parameter setting. This approach is based on the following observation. In LDA, it is necessary to provide the hyper-parameters $\alpha, \beta,$ and number of topics K in advance. In many empirical studies using LDA, a symmetric Dirichlet distribution with $\alpha = 50/K, \beta = 0.1$ is used [4]. However, there may not be truly optimal parameters for a given document set. On the other hand, we know empirically that a group of words generated from the same topic with a certain high probability and a group of documents containing common topics at a certain ratio or more are retained even if the number of topics and hyper-parameters assigned to LDA changes slightly, and such word groups and document groups represent the stable relationships between words and between documents that do not depend on the slight differences in parameters. The stable relationships found through integrating multiple topic analysis results achieve a comprehensive search without specifying all possible search terms related to the query.

(2) Hybrid method to integrate the ranking results of our topic-based Boolean query search and general query-based ranking model

Our topic-based search (ranking) method focuses on collecting a comprehensive set of relevant papers. Therefore, if a user searches for research papers using our method and a general ranking method (as represented by a query likelihood model), the set of highly ranked papers may be significantly different, because the latter model emphasizes how several documents satisfying the user's information needs should be ranked at the top of the results. We examined the precision and overlap ratio in two types of research paper set ranked by our topic-based ranking method and Wei and Croft's ranking method, which is a query likelihood model using LDA [20]. The results using a search task of the NTCIR-1 dataset [7] are shown in Fig. 1. The horizontal axis represents the rank of papers sorted by each method. The left vertical axis represents the ratio of relevant papers contained in the paper set within a certain interval on the horizontal axis, i.e., the precision, and the right vertical axis represents the ratio of overlapping papers within two types of paper sets given by each method within certain intervals on the horizontal axis, i.e., the overlap ratio. Fig. 1 indicates that many relevant papers are included in the papers ranked highly by each method; however, the overlap ratio between each high-ranked paper sets is relatively low at 0.349, and the overlap rates in the subsequent intervals are also low. From these results, it is highly likely that the research papers ranked highly by each method will be relevant for the user's information needs. Therefore, we expect that more effective paper search results can be obtained by integrating the paper sets output by two different ranking approaches.

2 Related Work

2.1 Query-based Academic Search

The challenge for an academic search is to comprehensively collect research papers related to a user's information needs (i.e., recall-oriented) [9, 19]. Many academic searches engines require multiple queries when a user comprehensively collect relevant papers, however, constructing queries manually is a heavy burden for users. Therefore, not only systems using word-based queries but also specific systems using elements other than words have been developed. For example, some systems use the body of a paper [2, 12, 23], the URL of a web page [16], and user profiles [1, 5, 6]. However, general academic search engines such as Google Scholar, Web of Science, and Scopus require word-based queries, and we also consider a word-based search engine. For systems using a word-based search query, there have been many studies on query term

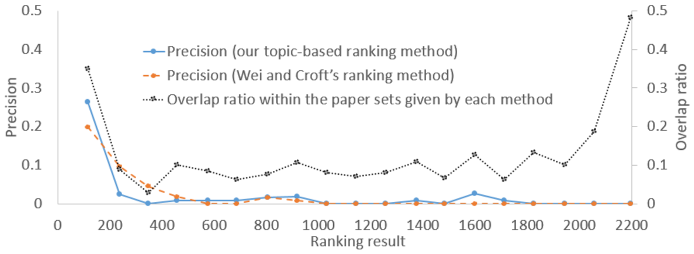


Fig. 1. Precision and overlap ratio given by our topic-based ranking method and by Wei and Croft's method using a search task ("0006") of the NTCIR-1 dataset. This search task datum has 2,196 relevance judgment data. We ranked using all of these data, and set the intervals at 5% (i.e., top 1–121, 122–243, …). In Wei and Croft's method we set $\alpha = 0.1, \beta = 0.1, K = 10, \lambda = 0.7,$ and $\mu = 50$. The overlap ratio was calculated by Dice coefficient.

suggestion for estimating alternative queries using the initial query [9, 19] and query expansion for automatically expanding the initial query [17, 21]. To estimate or extend the query, collections ranked highly by the initial query [9, 19] or thesaurus [17] are used, but relevant papers with words and phrases that are not included in these resources may be missed. Our topic-based search method does not require such resources because of the expansion to words that are potentially similar to the search word using an LDA topic model.

Recently, along with the development of an academic database, an academic search system using research paper-specific meta-information such as the title, author, proceedings, and venue has been actively studied [12, 21]. Our task uses only the abstract, so we can retrieve research papers that do not include meta-information.

2.2 Query-based Ranking Model

A general search model based on a query forms a ranking by calculating the degree of association with the query based on the document model in each document. For the document modeling, there have been many attempts to develop a mathematically descriptive framework by introducing a stochastic language model. A stochastic language model using a query is called a query likelihood model [14], and is often adopted in the ranking module for academic search tasks [9, 19].

In query likelihood models, a multinomial distribution model is widely used when expressing a document by a language model. This can be expressed as:

$$P(Q|D) = \prod_{q \in Q} P(q|D) \tag{1}$$

where D is a document, q is a query term in query set Q, and $P(Q|D)$ is the likelihood of the document model generating the query terms. For the estimation of $P(q|D)$, Zhai and Lafferty [22] proposed a query likelihood model using Dirichlet smoothing to assign a probability value to words that do not appear in the document, as shown below:

$$P(w|D) = \frac{N_D}{N_D + \mu} P'(w|D) + \left(1 - \frac{N_D}{N_D + \mu}\right) P'(w|coll) \tag{2}$$

where N_D is the number of word tokens in D, $P'(w|D)$ is the maximum likelihood estimate of w in D, and $P'(w|coll)$ is the maximum likelihood estimate of w in the entire collection. μ is a smoothing parameter. Hereafter, we call this method LM (Language Model) Search. The LM search uses only superficial language information appearing in the document. Several query likelihood models using the potential relationships among words have been studied. These use document clusters obtained by analyzing a document collection [11, 20] and word embedding [3, 13]. One study on a query likelihood model using LDA, similar to our topic-based search method, is that of Wei and Croft [20], who proposed a ranking model incorporating topic analysis results into a language model, as shown below:

$$P(w|D) = \lambda\left(\frac{N_D}{N_D+\mu}P'(w|D) + \left(1-\frac{N_D}{N_D+\mu}\right)P'(w|coll)\right) + (1-\lambda)\left(\sum\nolimits_{t=1}^{K}\frac{n_{-i,j}^{(w_i)}+\beta_{w_i}}{\sum_{v=1}^{V}n_{-i,j}^{(v)}+\beta_v}\times\frac{n_{-i,j}^{(D_i)}+\alpha_{z_i}}{\sum_{t=1}^{T}n_{-i,t}^{(D_i)}+\alpha_t}\right). \quad (3)$$

where λ is a smoothing parameter. $n_{-i,j}^{(w_i)}$ is the number of instances of word w_i assigned to topic j, not including the current token, and $n_{-i,j}^{(D_i)}$ is the number of words in D_i assigned to topic j, not including the current token. $\sum_{v=1}^{V}n_{-i,j}^{(v)}$ is the total number of words assigned to topic j and $\sum_{t=1}^{T}n_{-i,t}^{(D_i)}$ is the total number of words in D_i, not including the current word. Hereafter, we call this search method LDA+LM Search. We integrate our topic-based search method with the LM (LDA+LM) Search, and attempt to improve the ranking performance by emphasizing better results from individually ranked results.

3 Search Method Based on a Topic-based Boolean Query using Multiple Topic Analysis Results

Our topic-based search method consists of five steps: (1) a user defines the Boolean query and collects research papers; (2) the system conducts the preprocessing of the abstract set; (3) the system performs topic analysis of the abstract set using LDA with various parameter settings; (4) the system performs the topic-based Boolean query search using a topic analysis result for each parameter setting; (5) the system ranks papers based on the number of times it matched each topic-based Boolean query search. We call this method “Topic Search.” In the following, we describe each step in detail.

Query Definition and Collection of Research Paper Set. In step (1), our system requires the user to define a Boolean query in the following form:

(w_1 OR w_2 OR … w_m) AND (w_1’ OR w_2’ OR … w_n’) AND …

where *w* is a search term, the words in parentheses connected by OR comprise a concept unit, and each word inside the parentheses expresses a synonym of the same concept.

Simultaneously with the definition of the query, the system requires the user to comprehensively collect research papers that avoid missing relevant papers. At this time, various approaches are conceivable for collecting research papers more exhaustively, such as the system asks the user to specify a research field or an academic journal name and the system requires the user to construct another more comprehensive query. The research paper set collection used in our experiment is described in Section 5.1.

Preprocessing. Step (2) comprises two modules, one for the conversion of search terms and one for the removal of unnecessary words. In the conversion of search terms module, all search terms in the concept unit are converted to the same special symbol not appearing in the abstract set, and each search term appearing in each abstract in the database is converted to the corresponding symbol, as shown in Fig. 2. This process ensures that the same topics are assigned to all terms in each concept unit in the topic analysis. In the removal of unnecessary words module, we use TreeTagger [18] to tokenize and convert the original form, and retain only nouns, verbs, and adjectives that occur two or more times in the abstract set to be analyzed as features.

Topic Analysis using LDA. The LDA used in step (3) supposes that an abstract d is a sequence of words and each word is generated from a topic. Let $\mathbf{w}^{(d)} = (w_1^{(d)}, \cdots, w_{l_d}^{(d)})$ be the sequence of words for the d-th abstract, $\mathbf{z}^{(d)} = (z_1^{(d)}, \cdots, z_{l_d}^{(d)})$ be the sequence of topics, $\mathbf{w} = (\mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(D)})$ be the abstract set, and $\mathbf{z} = (\mathbf{z}^{(1)}, \cdots, \mathbf{z}^{(D)})$ be the sequence of topics for the whole collection. Following [4], $\mathbf{z}$ is generated using Gibbs Sampling according to:

$$P(\mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{P(\boldsymbol{w}, \boldsymbol{z}|\alpha, \beta)}{\sum_{\boldsymbol{z}} P(\boldsymbol{w}, \boldsymbol{z}|\alpha, \beta)}. \tag{4}$$

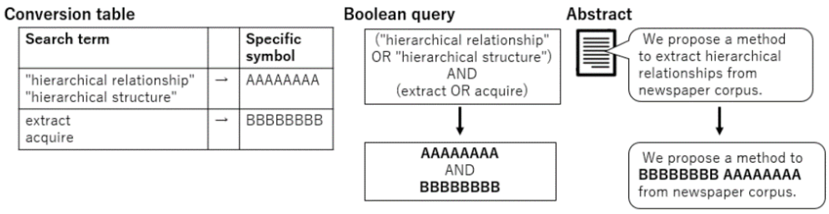
What is needed for our method is to assign a topic to every word in the abstract set. Therefore, using Gibbs Sampling, $\mathbf{z}$ with a relatively high probability of eq. (4) are intensively generated, and $\mathbf{z}$ that maximizes eq. (4) among them is determined. Finally, $\mathbf{z}$ that maximizes eq. (4) that can be reached from this $\mathbf{z}$ is found using gradient method.

Topic-based Boolean Query Search. In step (4), the system first constructs a topic-based Boolean query from a symbol-based query using the topic analysis result. The format of the topic-based Boolean query is as follows:

$$\mathrm{AND}_{i=1}^{I}\left(t_{i,1} \text{ OR } t_{i,2} \text{ OR } \cdots \text{OR } t_{i,J_i}\right)$$

where I is the number of concept units constituting the query, and $\{t_{i,1}, t_{i,2}, \cdots, t_{i,J_i}\}$ is the topic set assigned to the special symbol with which the i-th concept unit is replaced. The procedure for constructing the topic-based Boolean query is shown in Fig. 3. The system seeks the topics assigned to each special symbol in the abstract set that exactly match the symbol-based Boolean query, and then converts those symbols into their corresponding topics. Note that if different topics are assigned in different abstracts, they are joined by OR according to the above format. Research papers that exactly match the Boolean query constructed in step (1) are likely to satisfy the information needs, and so a topic-based Boolean query constructed using these research papers will properly represent the information needs.

After constructing a topic-based Boolean query, the system examines the topic types for all words in each abstract given by LDA. Finally, abstracts that exactly match a topic-based Boolean query are identified and these research papers are returned. For example, a topic-based Boolean query “Topic 0 AND (Topic 1 OR Topic 3)” for which research paper A contains topics 0 and 1, research paper B contains topics 0, 1 and 3, research paper C contains topics 0 and 3, research paper D contains topic 2, and research paper E contains topics 1, 2 and 3 will return research papers A, B, and C.


Conversion table
Boolean query
Abstract
("hierarchical relationship" OR "hierarchical structure") AND (extract OR acquire)
AAAAAAAA AND BBBBBBBB
We propose a method to extract hierarchical relationships from newspaper corpus.
We propose a method to BBBBBBBB AAAAAAAA from newspaper corpus.

Search term		Specific symbol
"hierarchical relationship" "hierarchical structure"	→	AAAAAAAA
extract acquire	→	BBBBBBBB

Fig. 2. Example of the conversion of a Boolean query and abstract.

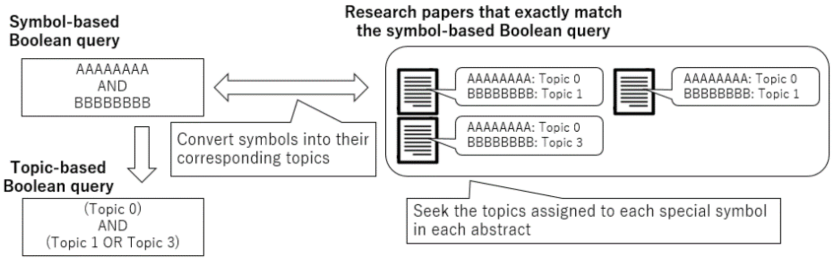

Symbol-based Boolean query
AAAAAAAA AND BBBBBBBB
Research papers that exactly match the symbol-based Boolean query
AAAAAAAA: Topic 0 BBBBBBBB: Topic 1
AAAAAAAA: Topic 0 BBBBBBBB: Topic 1
AAAAAAAA: Topic 0 BBBBBBBB: Topic 3
Convert symbols into their corresponding topics
Topic-based Boolean query
(Topic 0) AND (Topic 1 OR Topic 3)
Seek the topics assigned to each special symbol in each abstract

Fig. 3. Example of query conversion.

Topic-based Ranking Method. In step (5), after executing the topic-based Boolean query search for each of the parameter settings assigned to LDA, the system counts the number of times each paper is included in each search result. Based on the number of matches for all papers, the system ranks each paper in descending order. For example, we suppose that research papers A, B, and C are included in the result of a topic-based Boolean query search with parameters (α, β, K) = (0.1, 0.1, 10), research papers A and C are included in the result of a topic-based search with parameters (α, β, K) = (0.1, 0.01, 10), and research papers A, B, C, and D are included in the result of a topic-based search with parameters (α, β, K) = (0.01, 0.1, 10). The system counts 3, 2, 3, 1, and 0 as the number of matches for research papers A, B, C, D, and E, respectively, and ranks the papers in the order A, C, B, D, and E. Even if relevant papers have not been retrieved by a topic-based Boolean query search because of the inappropriate parameter setting, this influence can be reduced by integrating multiple topic analysis results.

4 Hybrid Search

In this section, we describe a ranking method that integrates the ranking results of the Topic Search with topics from the LM (or LDA + LM) Search. Hereafter, we call this method “Hybrid Search.” In the design of this method, we use two simple approaches to re-rank the papers based on the ranking results given by two type of search methods.

$$r_3 = \min(r_1, r_2) \tag{5}$$

$$r_3 = r_1 + r_2 \tag{6}$$

where r_1 and r_2 are the ranks of a research paper determined by Topic Search and LM (LDA+LM) Search, respectively. After the rankings for all research papers have been determined by eq. (5) or (6), the system sorts papers in ascending order based on r_3.

5 Experiment

5.1 Experimental Settings

We used the test collections of the NTCIR-1 and -2 datasets [7, 8]. These sets contain 132 search tasks that describe the conditions of research papers satisfying the information needs. Each search task datum has approximately 1,000–4,000 relevance judgment data that determine whether it satisfies the information needs for its task from the research papers collected using the pooling method [10], and each paper is manually judged according to the following criteria: highly-relevant, relevant, partially relevant, and non-relevant. In this experiment, we regarded "highly relevant," "relevant," and "partially relevant" to be equivalent to "relevant," and used 40 search tasks that have approximately 10–100 relevant papers. The Boolean queries in each search task were manually constructed by one subject by reading the contents of the search tasks.

We compared the following two baseline methods and five versions of our method:

- **LM Search (baseline)**: rank papers calculated by eqs. (1) and (2).
- **LDA+LM Search (baseline)**: rank papers calculated by eqs. (1) and (3).
- **Topic Search**: rank papers by multiple topic analysis results.
- **Hybrid (Topic & LM) Search (min)**: integrate the ranking results of Topic Search and LM Search using eq. (5).
- **Hybrid (Topic & LM) Search (sum)**: integrate the ranking results of Topic Search and LM Search using eq. (6).
- **Hybrid (Topic & LDA+LM) Search (min)**: integrate the ranking results of Topic Search and LDA+LM Search using eq. (5).
- **Hybrid (Topic & LDA+LM) Search (sum)**: integrate the ranking results of Topic Search and LDA+LM Search using eq. (6).

For the evaluation, we used the cumulative recall, where the recall is calculated as (number of relevant papers in the top n% research paper sets of the ranking result) / (number of relevant papers in the paper set). We set n to 1–100% with increments of 1%. The macro averages of the cumulative recall for the top n% of research papers in the ranking results of 40 search tasks are calculated. The method that has the highest rank that achieves a specific cumulative recall is best at its recall point. We evaluated Topic Search with parameter $\alpha \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$, $\beta \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$, and $K \in \{6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$. We also evaluated LM Search, LDA+LM Search and each Hybrid Search using 40-fold cross validation. At this time, we performed a grid search using $\alpha \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$, $\beta \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$, $K \in \{6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$, $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, and $\mu \in \{10, 20, 30, 40, 50\}$ to determine the optimal setting of parameters in each cross-validation fold. For the Gibbs Sampling, we set the number of iterations to 10,000.

5.2 Experimental Results and Discussion

The ranks when achieving a specific cumulative recall are shown in Table 1. First, we compare Topic Search and LM Search. From Table 1, when the target recall is 0.75 or more, Topic Search significantly improves the rank that achieves the desired recall in the ranking results. In particular, Topic Search shows an improvement in the ranking result of 15% or more compared with LM Search when the target recall is 0.90 and

Table 1. Comparison of the top *n*% rank in the ranking result when achieving a specific cumulative recall.

	Target recall							
	0.650	0.700	0.750	0.800	0.850	0.900	0.950	1.000
Topic Search	20%	25%	29%	33%	39%	46%	60%	95%
LM Search	20%	24%	30%	36%	46%	61%	77%	99%
LDA+LM Search	18%	22%	27%	33%	41%	51%	62%	95%
Hybrid (Topic & LM) Search (min)	17%	21%	28%	31%	36%	47%	57%	97%
Hybrid (Topic & LM) Search (sum)	14%	18%	22%	27%	32%	42%	57%	95%
Hybrid (Topic & LDA+LM) Search (min)	17%	21%	26%	30%	37%	46%	57%	97%
Hybrid (Topic & LDA+LM) Search (sum)	13%	17%	21%	27%	33%	40%	55%	97%

0.95. This means that the number of search results including the same number of relevant papers can be reduced by 15% or more by using Topic Search instead of LM Search. We next compare the results of Topic Search and LDA+LM Search. When the target recall is set to 0.85 or more, Topic Search returns a higher rank that achieves the desired recall than LDA+LM Search. From these results, when ranking research papers based on queries in a research paper search that requires comprehensive results for relevant papers, we can confirm that our topic-based search method outperforms the query likelihood methods with the same rank in the ranking results.

Next, we examine the effectiveness of our hybrid search. From Table 1, when the target recall ranges from 0.65–0.95, Hybrid (Topic & LDA+LM) Search (sum) achieves the recall with the highest rank. In particular, when the target recall is from 0.65–0.95, Hybrid (Topic & LDA+LM) Search (sum) shows an improvement in the ranking result of 5% or more compared with LDA+LM Search. Also, Hybrid (Topic & LDA+LM) Search (sum) improves the ranking result by 5–8% compared with Topic Search when the target recall is from 0.65–0.95. From these results, we can confirm the effectiveness of integrating two types of query-based ranking methods that have a different purpose and analytical approach for the collection of relevant papers.

6 Conclusion

We have proposed two kinds of research paper search methods using a word-based Boolean query constructed by a user. The first method uses a topic-based Boolean query search. This search converts every word in the abstract set and query into a topic via topic analysis and conducts a search at the topic level. Using this search method, we ranked research papers by combining the results of multiple topic-based searches for each parameter setting. The second method integrates the ranking results using our topic-based ranking method and a query likelihood ranking method. In future work, we will integrate our topic-based method with other query-based ranking methods such as word embedding [3, 13] in the Hybrid Search.

Acknowledgements. This work was supported by JSPS KAKENHI Grant Number JP15H01721. We thank Stuart Jenkinson, PhD, from Edanz Group (www.edanzediting.com/ac) for editing a draft of this manuscript.

References

1. Amami, M., Pasi, G., Stella, F., Faiz, R.: An LDA-based approach to scientific paper recommendation. In: NLDB, pp. 200–210 (2016).
2. Dhanda, M., Verma, V.: Recommender system for academic literature with incremental dataset. Procedia Computer Science 89, 483–491 (2016).
3. Ganguly, D., Roy, D., Mitra, M., Jones, G.J.F.: A Word embedding based generalized language model for information retrieval. In: SIGIR, pp. 795–798 (2015).
4. Griffiths, T.L. and Steyvers, M.: Finding scientific topics. In: National Academy of Sciences, pp. 5228–5253 (2004).
5. Hassan, H.A.M.: Personalized research paper recommendation using deep learning. In: UMAP, pp. 327–330 (2017).
6. Hong, K., Jeon, H., Jeon, C.: Personalized research paper recommendation system using keyword extraction based on userprofile. Convergence Information Technology 8(16), 106–116 (2013).
7. Kando, N., Kuriyama, K., Nozue, T., Eguchi, K., Kato, H., Hidaka, S., Adachi, J.: The NTCIR workshop: the first evaluation workshop on Japanese text retrieval and cross-lingual information retrieval. In: Information Retrieval with Asian Languages Workshop (1999).
8. Kando, N.: Overview of the second NTCIR workshop. In: NTCIR Workshop, pp. 35–43 (2001).
9. Kim, Y., Seo, J., Croft, W.B.: Automatic Boolean query suggestion for professional search. In: SIGIR, pp. 825–834 (2011).
10. Kuriyama, K., Kando, N., Nozue, T., Eguchi, K.: Pooling for a large-scale test collection: An analysis of the search results from the first NTCIR workshop. Information Retrieval 5(1), 41–59 (2002).
11. Liu, X., Croft, W.B.: Cluster-based retrieval using language models. In: SIGIR, pp. 186–193 (2004).
12. Mai, G., Janowicz, K., Yan, B.: Combining text embedding and knowledge graph embedding techniques for academic search engines, In: SemDeep–4 at ISWC (2018).
13. Masumura, R., Asami, T., Masataki, H., Sadamitsu, K., Nishida, K., Higashinaka, R.: Hyperspherical query likelihood models with word embeddings. In: IJCNLP, pp. 210–216 (2017).
14. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: SIGIR, pp. 275–281 (1998).
15. Sugiyama, K., Kan, M.-Y.: Scholarly paper recommendation via user's recent research interests. In: JCDL, pp. 29–38 (2010).
16. Takaku, M., Egusa, Y.: Simple document-by-document search tool "Fuwatto Search" using web API. In: ICADL, pp. 312–319 (2014).
17. Tannebaum, W., Rauber, A.: Using query logs of USPTO patent examiners for automatic query expansion in patent searching. Information Retrieval 17(5–6), 452–470 (2014).
18. TreeTagger Homepage, http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/.
19. Verberne, S., Sappelli, M., Kraaij, W.: Query term suggestion in academic search. In: ECIR, pp. 560–566 (2014).
20. Wei, X., Croft, W.B.: LDA-based document models for ad-hoc retrieval. In: SIGIR, pp. 178–185 (2006).
21. Xion, C., Power, R., Callan, J.: Explicit semantic ranking for academic search via knowledge graph embedding. In: WWW, pp. 1271–1279 (2017).
22. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems 22(2), 179–214 (2004).
23. Zhao, W., Wu, R., Liu, H.: Paper recommendation based on the knowledge gap between a researcher's background knowledge and research target. Information Processing & Management, 52(5), 976–988 (2016).