Improving Patent Search by Search Result Diversification

Youngho Kim
yhkim@cs.umass.edu

W. Bruce Croft
croft@cs.umass.edu

Center for Intelligent Information Retrieval, College of Information and Computer Sciences
University of Massachusetts Amherst, MA 01003, USA

ABSTRACT

Patent retrieval has some unique features relative to web search. One major task in this domain is finding existing patents that may invalidate new patents, known as prior-art or invalidity search, where search queries can be formulated from query patents (i.e., new patents). Since a patent document generally contains long and complex descriptions, generating effective search queries can be complex and difficult. Typically, these queries must cover diverse aspects of the new patent application in order to retrieve relevant documents that cover the full scope of the patent. Given this context, search diversification techniques can potentially improve the retrieval performance of patent search by introducing diversity into the document ranking. In this paper, we examine the effectiveness for patent search of a recent term-based diversification framework. Using this framework involves developing methods to identify effective phrases related to the topics mentioned in the query patent. In our experiments, we evaluate our diversification approach using standard measures of retrieval effectiveness and diversity, and show significant improvements relative to state-of-the-art baselines.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – retrieval models

General Terms

Algorithms, Experimentation.

Keywords

Search result diversification; prior-art search; patent retrieval;

1. INTRODUCTION

Patent Information Retrieval (Patent IR) is unique and very different from general web search. One major task in this domain is finding existing patents that may invalidate new patents, known as prior-art search or invalidity search [15][30][35]. In this task, users typically input a query document (the new patent), and the search system returns the set of "relevant" patents. Since the whole document is input as an initial query, query processing techniques (e.g., automatic query generation) need to be employed. Henceforth, we refer to this type of prior-art search as "patent search" or "patent retrieval".

Since a patent document generally contains long and complex descriptions of its invention, formulating effective queries is a very difficult task [2][22]. To reduce the burden for users, an automatic query generation has been researched (e.g., [16][33][42]). Since the generation is based on the whole query patent, generated queries could potentially contain hundreds of terms. Note that in this paper, we assume that users only provide a query patent, and search queries are automatically generated based on the input query patents. Typically, search queries must cover diverse aspects of the new patent application in order to retrieve relevant documents that cover the full scope of the patent. Figure 1 shows an example query patent. In this example, the patent application includes several components such as "usage profile", "BIOS", and "operating system". Accordingly, the queries automatically generated for this patent contain the terms to describe each component (e.g., {"reboot", "time", "boot"} for "usage profile information"). In other words, multiple aspects (or topics) are covered in a patent query, and it is important to diversify any retrieval result by covering as many of these aspects as possible. We discuss this example again later in this section.

Query Patent

Title: Method and apparatus for providing content on a computer system based on usage profile.

Abstract

A method and apparatus for determining a *computer system usage profile* … A *basic input output system (BIOS) module* and/or an *operating system module* obtain computer system usage profile information by tracking events such as the *frequency of reboots*, the *time* required to *boot up* and *shutdown* the operating system … data is collected and communicated to a *profile server*…

Search Query Automatically Generated by [42]

usage, *profile*, *reboot*, *time*, *os*, bootstrap, *boot*, memory, manage, processor, video, firmware, *network*, database, ~

List of Relevant Documents

No.	Title	Topic
R1	Extended *BIOS* adapted to establish remote communication for diagnostics and repair	*BIOS*
R2	*BIOS* emulation of a hard file image as a diskette	
R3	*Operating system* architecture with reserved memory space resident program code identified in file system name space	*operating system*
R4	Method for loading an *operating system* through a network	
R5	Method and apparatus for *controlling network* and workstation access	*profile server*
…	…	…

Figure 1: Query Patent Example

In the literature, one important task for patent search is automatically generating effective queries. Given a query patent, several methods (e.g., [33][42]) focus on ranking the terms from

ICTIR'15, September 27–30, 2015, Northampton, MA, USA.

DOI: http://dx.doi.org/10.1145/2808194.2809455

the query patent, and selecting the top *n* terms to form a query. Similarly, sentence ranking, i.e., selecting the top ranked sentences from the patent to use as a query, has also been proposed [16]. In addition, query expansion techniques have been applied to this task of generating effective queries (e.g., [17][32]). Other research on patent retrieval has developed effective retrieval frameworks (e.g., [29][34]). None of this research has studied the problem of search result diversification for patent search.

Search result diversification is the process of re-ordering an initial retrieval result so that the final ranked list can include more diverse topics associated with the query [10][39] (henceforth, "query aspect" is referred to as "query topic" [8][12]). In web search, this technique is adopted for clarifying vague information needs, e.g., a web query "slate" can represent one of a broad range of topics. However, in this paper, we exploit diversification techniques for improving the retrieval performance of patent search by covering more of the topics described in a query patent.

In general, a patent document contains approximately 3,900 words on average [20] and includes complex structure [2][14] (e.g., title, abstract, claim, background and summary sections). In that structure, diverse claims are specified, and background patents related to the application are described. In addition, patent applications can describe multiple components. Thus, we can find a range of topics in a query patent, and the relevant documents can relate to some or all of these topics. Returning to the example in Figure 1, the patent application describes several important topics such as *BIOS*, *operating system,* etc. We can group similar relevant documents pertaining to each topic. For example, R1 and R2 are related to a topic *BIOS*, whereas R3 and R4 refers to *operating system*. In addition, R5 describes a method for controlling network, which relate to another query topic (i.e., *profile server*). Based on these topics, the retrieval result can be diversified, meaning that the ranked documents can be optimized to cover the range of topics. Although a long search query can contain diverse query terms (as shown in Figure 1), the diversity of its retrieval result is not optimized unless a result-level diversification technique is applied. Since relevant documents are related to diverse query topics, diversification techniques (e.g., [1][39]) could also improve overall retrieval effectiveness. For example, the diversification framework proposed in [11] has been shown to improve the ranking of relevant documents in TREC collections, not only in terms of diversity but also in terms of general effectiveness measures (see [11]).

Given this motivation, we explore the problem of *patent search result diversification*. In this diversification process, query topics are first identified, and then re-ranking algorithms (e.g., [10][39]) are applied with the identified topics. Most early research on diversification for web search relied on manually identified topics for a query. In recent research, automatic identification of the topics related to a query has been shown to be effective (e.g., [12][13][18]), but the topics are derived from other resources (e.g., web corpora, web anchor text, and query logs [12][13][36]) that are limited to provide clear information about query topics. On the other hand, in patent search we can potentially identify query topics via the detailed descriptions contained in query patents. Thus, we propose a method to automatically identify query topics based on query documents as follows. Note that we use the terms "query document" and "query patent" interchangeably to refer to a new patent document being validated by prior-art search.

Given a query patent, we extract phrase-level topic vocabulary as the basis for query topics. However, we do not adopt any additional topic structures or grouping of phrases because recognizing such structure is expensive and typically does not lead to significant improvements on diversification performance [11]. Instead, we rank candidate phrases (extracted from a query patent) by combining information such as topicality [27], predictiveness [28], query performance predictors (e.g., query clarity [9]), relevance to query patents, cohesiveness, etc. Then, we consider the top k phrases as topic phrases used for diversification. Following [11], we also use topicality and predictiveness for finding topic phrases. In addition to this, we assume that query performance predictors (e.g., query scope [19]), relevance to query patents (e.g., relevance model [26]) and cohesiveness of phrase terms will also help to identify topic phrases. In order to combine these features, we adopt a learning-to-rank framework that places phrases important to more relevant topics (i.e., topics of relevant documents) at higher ranks. After generating topic phrases, we apply a state-of-the-art diversification algorithm and diverse ranked results are produced.

To summarize, the main contribution of our work is a study of *patent search result diversification* with the aim of improving patent search performance. Our work is the first attempt to exploit result-level diversification techniques for patent search. In addition, through experiments, we investigate the effectiveness of diversification techniques in patent search environments. The primary difference of our work to existing diversification research (e.g., [11][13]) is that we directly derive phrase-level topics from query documents that provide more information than web queries. In addition, we investigate multiple features (e.g., relevance and cohesiveness) to identify query topics. These are reflected in our topic phrase identification method (in Section 3.3). We evaluate our method by comparing it with DSPApprox, a state-of-the-art topic term identification method [11].

The rest of this paper is organized as follows. Section 2 describes related work. In Section 3, we formulate the problem of patent search result diversification, and present our diversification framework. Section 4 describes experimental settings, and Section 5 provides results. Finally, Section 6 concludes this paper.

2. RELATED WORK

2.1 Patent Search

The research related to improving patent search can be classified into two categories: 1) automatic query generation (e.g., [16][33][42]), and 2) developing retrieval models (e.g., [34][43]). There is other research related to patent retrieval (e.g., patent query translation), but this is less relevant for our work.

Automatic query generation: To generate effective queries, previous studies have used the full text of patent applications. They rank the terms in query patents, and select the top n terms for the query. Xue and Croft [42] extracted query terms from the "brief summary" section of query patents by TFIDF scoring. Mahdabi et al. [33] used Kullback-Leibler divergence between query models and collection models for term ranking. To improve single term queries, they extracted key phrases by TFIDF and Mutual Information-based scoring, and expand the initial term queries by the key phrases. Similar to this approach, Ganguly et al. [16] selected the top sentences ranked by similarity to pseudo-relevant documents for query patents. Another approach to generating effective queries exploits query expansion techniques (e.g., [17][32]). Some of this research used external resources for the expansion, e.g., Wikipedia [29] and WordNet [32]. Also, [23] used decision trees to generate effective terms from pseudo-relevant documents. Among these approaches, we applied several methods (i.e., [16][33][42]) in our experiments and used the best one to generate the initial retrieval results for diversification. The most recent and related approach for generating queries is proposed in

[25]. Similar to our work, they also propose to retrieve diverse relevant documents. However, given a query patent, they generate *n* diverse queries, and accordingly *n* different retrieval results are returned (i.e., query-side diversification), which may delay completing a search task. On the other hand, we propose a result-diversification framework, and a single retrieval result is generated for each query patent.

Retrieval models: In early research, existing retrieval models for adhoc retrieval were studied in the context of patent search (e.g., [20]). Through several evaluation competitions for patent retrieval (e.g., NTCIR [15], TREC [30], and CLEF-IP [35]), various refinements to the retrieval methods have been proposed. In experiments, we run an initial retrieval for diversification by using one of the best approaches in these competitions (i.e., [29]).

2.2 Search Result Diversification

Search result diversification is the task of generating a ranked list of documents that covers a range of query topics (or aspects). Previous work on this task can be categorized as: 1) implicit or 2) explicit [39]. We provide a brief summary for each category.

Implicit diversification: The implicit approach does not assume any explicit representation of query topics. MMR [3] and its probabilistic variants [45] can be included in this approach. For diversification, these methods assume that each document in the initial retrieval results represents its own topic and iteratively selects the documents that are dissimilar to previously chosen documents. To measure the dissimilarity, MMR used content-based similarity functions, but probabilistic distance in the language modeling framework has also been used in [45]. In addition, the correlation between documents is adopted as a similarity measure [37][41], and the diversification problem is viewed as minimizing the correlation.

Explicit diversification: In contrast to the implicit method, this approach requires some representation of query topics (e.g., [1][10][39]). There are two different approaches to implementing explicit diversification: *redundancy* and *proportionality*. The redundancy approach is used in many existing methods (e.g., IA-Select [1], xQuAD [39]). These aim to provide less redundant information in the diversified results, i.e., documents are promoted if they include *novel* content that has not appeared in early ranks. On the other hand, the proportionality-based algorithms (e.g., PM-2 [10]) choose the documents with respect to the "popularity" of their topics in the initial ranking, i.e., ranking the documents is proportional to the popularity of each query topic. Both of these approaches have been successful with test collections that contain manually created query topics (e.g., from TREC descriptions [10][39] and taxonomies [1]).

To provide a more realistic context, methods for automatically generating query topics have been studied (e.g., [11][12][36]). As an example, query topics have been generated by clustering similar queries from query logs [36] or anchor texts from the web [12]. More recently, term-level diversification [11] has showed the effectiveness of automatic topic generation based on identifying important vocabulary terms. In this approach, query topics are described by some set of terms, and instead of generating the topics directly, only the important words and phrases associated with the topics are automatically identified, e.g., the words "pain", "joint", "woodwork", and "type" are identified for the latent topics of "joint pain" and "woodwork joint type". After identifying the important vocabulary, the diversification framework (e.g., xQuAD or PM-2) is applied using the identified topic terms (the frameworks consider each term as a topic). The effectiveness of these automatically-found topic terms has been shown to be similar to the manually generated topics, and significantly better than other approaches to automatic topic identification. Our diversification framework for patents uses this approach, and we focus on identifying topic phrases (e.g., "file system" and "system service") and diversifying with respect to these phrases.

2.3 Automatic Topic Term Identification

As reported in [11], automatically identifying topic terms helps to improve diversification. In [11], a set of terms to represent initial retrieval results is generated for an initial ranked list of documents. This is similar to the goal of multi-document summarization (e.g., [27][28][38]). Thus, DSPApprox, a hierarchical summarization algorithm proposed in [28], has been used for identifying topic terms in [11]. This algorithm iteratively selects the terms which maximize *predictiveness* and *topicality*. We describe this algorithm in Section 3.3. In addition to predictiveness and topicality, in this paper we explore additional features to identify topic phrases, i.e., relevance, cohesiveness, and query performance predictors (see Section 3.3.2). Moreover, we examine the effectiveness of these features in the context of diversification.

3. PATENT SEARCH DIVERSIFICATION

3.1 Problem Formulation

Diversification in patent search is designed to improve the retrieval effectiveness of initial ranked results. As discussed in Section 1, we assume that diverse topics are involved in a query patent, and that diversification of initial search results based on those topics will improve retrieval performance.

Given a query patent Q, let $T = \{t_1, t_2, \dots, t_n\}$ be a topic set for Q and for each topic t_i, some weight $w(t_i)$ is defined. Note that this weight is used as the importance [39] or popularity [10] by the diversification algorithm applied. In addition, an initial document list for Q is given, $D = \{d_1, d_2, \dots, d_m\}$, and each d_i's relevance to t_i can be estimated, $\Pr(d_i|t_i)$. Using $\langle T, w(t_i), \Pr(d_i|t_i)\rangle$, diversification algorithms (e.g., [10][39]) typically generate a subset of D which forms a diverse rank result S where S contains a target number of documents. However, recent work [11] found that explicitly specified topic structures (i.e., T) (e.g., grouping topic terms to represent topics such as $t_1 = \{$"user", "data"$\}$ and $t_2 = \{$"share", "security"$\}$) are less beneficial for improving search performance. Instead, only identifying topic terms (e.g., "data", "security", "share", "user") and directly using such terms without the more complex step of topic identification can be effective. Based on this observation, we apply a term-level diversification method to patent search. Instead of using unigram terms, we use phrases to express patent topics because patent documents frequently contain longer technical terms (e.g., "file system"). In addition, phrasal concepts can be effective to retrieve more relevant documents [24][33]. Thus, we identify a set of topic phrases for T, and apply diversification frameworks using these phrases. The formal definition of this diversification method is given as follows.

Let us assume that a topic $t \in T$ can be represented by an arbitrary set of phrases, i.e., $t = \{p_1, p_2, \dots, p_{\#(t)}\}$ where p_i is a topic phrase for Q and $\#(t)$ is the number of phrases to form t. Then, T can be rephrased as:

$$T' = \left\{\left\{p_1^{t_1}, p_2^{t_1}, \dots, p_{\#(t_1)}^{t_1}\right\}, \dots, \left\{p_1^{t_n}, p_2^{t_n}, \dots, p_{\#(t_n)}^{t_n}\right\}\right\}$$

We define a set of phrases that can contain all phrases in T', i.e., $P' = \{p | \forall p \in T'\}$, and the phrase-level diversification is defined as generating a diverse ranked list $S \subset D$ using P'. In effect, each phrase is treated as a topic in the diversification model (see Section

3.2). As a result of diversification, S covers more topic phrases and contains more diverse relevant documents. Next, we describe the diversification framework we use.

3.2 Diversification Framework

Explicit diversification methods (e.g., PM-2 [10] and xQuAD [39]) assume that some set of query topics (or aspects) is specified, and generate diverse ranked results based on these topics. Among many algorithms, we select to use the proportionality-based approach (PM-2) [10] for our diversification task, which is the most recently proposed state-of-the-art technique. The proportionality-based approach focuses on generating a proportional representation of query topics in the final retrieval result, i.e., the documents related to an "important" (i.e., large-portion) query topic are promoted. So, more relevant documents for such an important topic would be found in the diversified result. On the other hand, redundancy-based approaches (e.g., xQuAD) attempt to have as many diverse topics as possible in the final result by demoting the documents related to the query topics already considered in the diversification process. Thus, relevant documents related to some important query topics could be missed by diversification.

The proportionality-based approach exploits the Sainte-Laguë method, allocating seats in proportional representation, for assigning the portions of topics in S such that the number of each topic's documents in S is *proportional* to the weight of the topic, i.e., w_i. Specifically, PM-2 requires a set of topics T, an initial document retrieval list D, and an empty list S. In each iteration, the quotient qt_i of each topic t_i is computed as:

$$qt_i = \frac{w_i}{2s_i + 1} \quad (1)$$

where s_i is the current portion of t_i in S.

Using this, PM-2 selects the most proportional topic t_i^* with the largest qt_i, and places the document $d^* \in D$ into S such that d^* is mostly relevant to t_i^* as well as other topics:

$$d^* \leftarrow \underset{d \in D}{\operatorname{argmax}} \ \lambda \cdot qt_i^* \cdot \Pr(d|t_i^*) + (1-\lambda) \sum_{i \neq i^*} qt_i \cdot \Pr(d|t_i) \quad (2)$$

where $\mathrm{p}(d|t_i)$ is an estimated relevance of d to t_i.

Although Eq. (2) is effective for diversifying web search results, this looks somewhat limited to work for patent search. In PM-2, Eq. (2) only considers the relevance of a document to each topic, not directly to the whole query patent. This setting could work for web search results because the diversification aims to clarify ambiguous web queries. On the other hand, patent search is recall-oriented, i.e., not missing relevant documents in a relatively long retrieval result is more important than placing them at top ranks. So, keeping the documents "relevant" to Q (by some estimation) in S is important. To do this, we combine Eq. (2) with the relevance score of d for Q.

$$d^* \leftarrow \underset{d \in D}{\operatorname{argmax}} \ \mu \cdot \mathrm{relevance}(d) + (1-\mu) \cdot \mathrm{diversity}(d) \quad (3)$$

where $\mathrm{relevance}(d)$ is an estimated relevance score of d for Q and $\mathrm{diversity}(d)$ is the diversity score calculated by Eq. (2).

By Eq. (3), we can choose the document not only related to the appropriate topic but also highly relevant to the query patent. In other words, we basically keep as many relevant documents as possible in S, and would promote the relevant documents related to important topics by the diversity term (i.e., Eq. (2)). In experiments, we use the retrieval score obtained by the baseline retrieval model as the estimation of $\mathrm{relevance}(d)$. After selecting d^*, the algorithm updates the portion of each topic in S (i.e., s_i) by its normalized relevance to d^*:

$$s_i = s_i + \frac{\Pr(d^*|t_i)}{\sum_j \Pr(d^*|t_j)} \quad (4)$$

Then, this process is repeated with the updated s_i, and stops after S contains a target number of documents (e.g., top 100 documents in a retrieval result). Besides, the final ranking of a document is determined by the order in which the document is included in S.

As described in Section 3.1., we use phrase-level diversification for patent search, and thus the set of topic phrases (interpreted as topics) is the input to this diversification model. In the next section, we present our method to generate topic phrases, which is important for diversification performance.

3.3 Automatic Topic Phrase Identification

The goal of identifying topic phrases is generating a list of effective phrases for diversification. As discussed in Section 3.1, we need to generate P' which contains all possible phrases to represent query topics. This is an important task because the diversification model (in Section 3.2) assigns the documents in S primarily based on the input phrases. To identify topic phrases, we assume that each query patent includes sufficient terms to represent its topics.

Given a query patent Q, we extract a set of candidate noun phrases, $P = \{p_1, p_2, \ldots\}$ syntactically recognized[1] in Q, and some subset of P would be an effective set of topic phrases, i.e., P'. To obtain P', previous work [11] has used DSPApprox, the multi-document summarization technique proposed in [27][28]. Since this algorithm can generate a set of terms to efficiently summarize target documents, it is also useful to find a diverse set of topic terms (i.e., phrases) for Q (as shown in [11]). Thus, we can also use DSPApprox for generating P' as follows.

Given an initial document list for Q, we define a set of vocabulary as the terms that appear in at least two documents and are not numbers. For each candidate phrase, we define *topicality* as the extent of how informative the phrase is to describe Q, and *predictiveness* as its ability to predict the occurrences of other vocabulary terms. Note that these terms are also defined in [27][28]. To measure topicality, a relevance model [26] for Q is generated and the clarity score [9] of each phrase is computed using the relevance model. To estimate predictiveness, the conditional probability of a phrase for a vocabulary term can be used (as done in [27]). Specifically, the affinity of a phrase to each vocabulary term is estimated by the conditional probability, and highly predictive phrases are more likely to appear with more vocabulary terms. Based on these definitions, DSPApprox iteratively selects topic phrases by maximizing their topicality and predictiveness. The details of this approach are described in [11].

Since DSPApprox is a simple greedy algorithm only considering topicality and predictiveness, to improve the identification process we propose a learning-to-rank framework that combines these two features with other features. In experiments, we compare the diversification results obtained by both DSPApprox and the learning-to-rank method.

3.3.1 Learning-to-rank Topic Identification

In order to identify effective topic phrases, we rank the candidate phrases extracted from the query patent, i.e., P, and use the top k phrases as topic phrases. For this, our ranking model produces a

[1] The open NLP tool (http://opennlp.apache.org/) is used.

ranked list of the phrases in descending order of their (predicted) effectiveness to derive more query topics. This is formally defined as follows.

Given a query patent Q, let $P = \{p_1, p_2, \dots, p_l\}$ be a set of candidate phrases extracted from Q where l denotes the number of extracted phrases. Suppose that $Y = \{y_1, y_2, \dots, y_l\}$ is a set of ranks, and the order of the ranks is given as: $y_1 \succ y_2 \succ \cdots \succ y_l$ where $\succ$ indicates the preference between two ranks. For each phrase $p_j \in P$, some corresponding rank, y_{p_j}, is assigned. To learn a ranking function, we need to generate training examples, i.e., a ground-truth rank list of P. However, labeling the ground-truth rank of each phrase is too complex because determining its effectiveness for diversification is very difficult (running diversification with every possible ranking of the phrases is intractable). To alleviate this, we exploit DSPApprox to obtain the ground-truth rank list.

Let $R = \{r_1, r_2, \dots\}$ be the set of relevant patents for Q and we generate the rank of each phrase by DSPApprox using R. We first define a vocabulary set as the terms appeared in R. Then, we calculate the topicality and predictiveness of each phrase based on R, and run DSPApprox using these, which gives a list of ranked phrases as: $\hat{Y} = \{\widehat{y_{p_1}}, \widehat{y_{p_2}}, \widehat{y_{p_3}}, \dots\}$ where p_1 is the first selection of DSPApprox, p_2 is the second, p_3 is the third, and so on. The topicality (i.e., query clarity [9]) based R is calculated as:

$$\text{Topic}_R(p) = \Pr(p|R) \cdot \log_2\{\Pr(p|R)/\Pr_c(p)\} \qquad (5)$$

where $\Pr(p|R)$ is the probability of a phrase p by the smoothed language model [44] derived from R and $\Pr_c(p)$ is the collection probability.

In addition, the predictiveness [27] using R is also computed as:

$$\text{Predict}_R(p) = \frac{1}{Z} \cdot \sum\nolimits_{v \in CX_p^R} \Pr_w(p|v) \qquad (6)$$

where CX_p^R is the set of vocabulary terms that co-occur with p within the windows recognized in R, w is the size of each window, which is empirically set as 20, $\Pr_w(p|v)$ indicates such co-occurrence probability using w, and Z is the normalization factor, which is set as the size of vocabulary (see [27] for more detail).

Based on these, the phrases more topically represent R and can cover more terms in R are highly ranked in the ground-truth list, and we assume that such phrases are effective to find diverse "relevant" documents in the diversification process. Note that the topic term identification described in [11] uses DSPApprox without relevance judgments (i.e., unsupervised approach for ranking topic terms), but in our work, we use the supervised learning-to-rank framework with relevant documents and DSPApprox is used for generating training examples by R (thus, the relevant documents are not necessary in testing). In the training phase, a set of query patents, $\mathbb{Q} = \{Q_1, Q_2, \dots\}$, are given, and a feature vector $x_{ij} = f(Q_i, p_{ij}) \in X_i$ is generated for the pair of a query patent and its candidate phrase. Then, $\langle X_i, \hat{Y}_i \rangle$ is used for learning a ranking function. In experiments, we use Ranking SVM [21] as a learning algorithm, and select various numbers of phrases as topic phrases.

3.3.2 Features

To compose a feature vector in our ranking model, we use four types of features: 1) relevance, 2) importance, 3) predictiveness, and 4) cohesiveness. We describe each type as follows.

Relevance: *Relevance* estimates some probabilistic relevance of each phrase to the query patent. Typically a patent document consists of four sections (i.e., title, abstract, claim, and description), and previous work (e.g., [43]) has used section information for retrieving relevant documents by finding a weight on each section. Similar to this, we generate three different section language models: (1) title and abstract, (2) claim, and (3) description, and define a relevance feature based on each language model. For example, given a phrase p, the relevance based on the claims in the query document Q is estimated as:

$$\text{Rel}_{Q(\text{claim})}(p) = \prod_{t \in p} \frac{tf_{t,Q(\text{claim})} + \mu \cdot \Pr_c(t)}{|Q| + \mu} \qquad (7)$$

where t is a unigram in a phrase p, $Q(\text{claim})$ is the claim text of Q, $tf_{t,Q(\text{claim})}$ is the frequency of t in $Q(\text{claim})$, μ is the Dirichlet smoothing parameter [44].

In addition, we also generate another three section language models by using pseudo-relevant documents (i.e., the top N patents ranked in the initial retrieval result), and define three more relevance features based on these models. Overall, the six relevance features would help to identify the phrases more likely to be associated with the section language models, and our learning algorithm could find an optimal weight for each section model.

Importance: *Importance* indicates retrieval effectiveness related to finding relevant documents. To measure this, we leverage the features for predicting query performance (e.g., [9][19]). Given a phrase, we calculate its query clarity score [9] based on the query model directly derived from the query patent or the relevance model of the query patent. Note that the contribution of the topicality feature used in DSPApprox is the same as that of the query clarity feature we use. In addition, we use query scope [19], inverse document frequency, inverse collection term frequency, and word count, which are generally used for measuring pre-retrieval effectiveness. Since the diversification algorithm (described in Section 3.2) mainly uses the topic phrases for diversification, identifying highly effective phrases for retrieving relevant documents is important to increase the retrieval effectiveness of the final retrieval result.

Predictiveness: *Predictiveness* [28] measures the extent to which a term predicts the occurrences of other terms in a query vocabulary. We use two different types of query vocabulary: 1) all terms in the query patent and not numbers, and 2) the terms that appeared in at least two pseudo-relevant documents and not numbers. Note that stop-words and section terms (e.g., background and summary) are ignored. For each query vocabulary, we measure the predictiveness of an instance (i.e., phrase): $\text{Predict}_Q(p)$ and $\text{Predict}_{PR}(p)$ where PR is the pseudo-relevant documents of Q. As used in DSPApprox, these predictiveness features are effective for extracting diverse phrases that can represent the terms in each topic vocabulary.

Cohesiveness: *Cohesiveness* quantifies the coherence of the terms in a phrase. We assume that the terms more co-occurring in query contexts can be keywords. As an example, for the patent of "Method and apparatus for proving content on a computer system based on usage profile" the terms "usage" and "profile" would frequently co-occur, and it is probable that these also appear in relevant documents. Note that predictiveness measures the conditional probability of a phrase to a query vocabulary, but cohesiveness estimates the lexical affinity of each word to the other in a phrase. To capture this, we generate every possible pair of words in a phrase, and calculate the average of Point-wise Mutual Information (PMI) values for all pairs by using Q as follows.

$$\text{CHSV}_Q(p) = \frac{1}{\binom{n}{2}} \cdot \sum\nolimits_{w_i, w_j \in p, i \neq j} \text{PMI}_Q(w_i, w_j) \qquad (8)$$

Table 1. Four Types of Ranking Features

Type	Features
Relevance	Query Relevance (QR) by Title and Abstract, QR by Claim, QR by Description, Pseudo Relevance (PR) by Title and Abstract, PR by Claim, PR by Description
Importance	Inverse Collection Term Frequency, Inverse Document Frequency, Word Count, Query Clarity [9], Query Scope [19]
Predictiveness	Query Document-based (Predict_Q), Pseudo-Relevant-based (Predict_{PR})
Cohesiveness	Query Document-based (CHSV_Q), Pseudo-Relevant-based (CHSV_{PR})

where $\text{PMI}_Q(w_i, w_j) = \left(\text{f}(w_i, w_j) \times n\right) / \left(\text{f}(w_i) \cdot \text{f}(w_j)\right)$, $\text{f}(w)$ is the number of windows containing w in Q, n is the number of all windows in Q and the size of each window is set as 20.

If p is a unigram ($n = 1$), instead of using Eq. (8), we define cohesiveness as the portion of the windows containing p to all windows, i.e., $\text{f}(w)/n$. Besides, we also define this cohesiveness feature based the pseudo-relevant documents, i.e., $\text{CHSV}_{PR}(p)$.

Table 1 summarizes these feature types, and we analyze the effectiveness of each feature type in experiments.

4. EXPERIMENTAL SETUP

To evaluate our approach, we conduct the experiments as follows. For each query document, we generate a baseline query and employ a baseline retrieval model to produce an initial retrieval result. Then, we apply the diversification framework (see Section 3.2) with topic phrases. To generate the topic phrases, we use either DSPApprox or the learning-to-rank method (see Section 3.3).

4.1 Test Collections

We use two different document collections. The first one contains USPTO (United States Patent and Trademark Office) patents provided by NTCIR-6 [15]. This collection contains 981,948 patents published from 1993 to 2000. To develop query patents (new patents), we randomly select 150 patents published in 2000, ensuring that their citations list more than 20 patents, and at least 90% of them are included in the collection. As done in [15], we consider patents cited in each query patent as "relevant", and 22.64 relevant documents are found on average. We call this collection USPTO. The other collection we use is the CLEF-IP 2010 [35] corpus which contains 2.6 million EPO (European Patent Office) patents. We randomly select 300 query patents from the query patent pool they provide. Although the query documents are described in the three official EPO languages (English, German, French), we only work with English documents. Relevance assessments are provided, which also use the citations listed in each query patent (see [35] for more details). The average number of relevant documents is 28.87, and we call this collection EPO. Queries and documents are stemmed using the Krovetz stemmer and standard stop-words are removed.

4.2 Evaluation Metrics

Since we attempt to diversify patent search results, we use conventional IR evaluation metrics to measure retrieval effectiveness as well as diversity metrics which measure "diversity" on retrieval results. For measuring relevance, we utilize MAP, NDCG, and Recall, which are typically used for adhoc retrieval tasks. In addition, PRES [31] is adopted, which is particularly designed for recall-oriented search tasks. This metric reflects the normalized recall incorporated with the quality of ranks of relevant documents observed within the maximum numbers of documents that the user examines (see [31] for details).

As diversity metrics, NRBP [7], α-NDCG [8], ERR-IA (a variant of ERR [5]), MAP-IA [1], and subtopic recall (S-Recall) are used. These metrics penalize redundancy in retrieval results, i.e., how much of the information in each retrieved relevant document the user has already obtained in earlier ranks. Note that these have been used as standard metrics for diversity tasks in TREC [6]. Since patent examiners (i.e., the search users) typically examine 100 patents on average in the invalidity search processes [22], we assume that the top 100 ranked documents are used to calculate the value of each metric.

4.3 Topic Relevance Judgment

Although we develop the list of relevant documents for each query patent, the diversity metrics require the identification of query aspects for the relevant documents. In other words, for each query patent, we need to group relevant documents if they belong to the same topic. The manual judgments required for this would be too laborious, and patent experts are essential because they can fully understand patent topics. To alleviate this, we devise a semi-automatic method. Each patent document contains a list of IPC (International Patent Classification) [2] codes that classify the document into a hierarchical taxonomy. As an example, the IPC code "H01S 3/14" indicates the patents related to "lasers characterized by the material used as the active medium". So, we exploit these codes to generate the topics of each query patent as follows.

Given a query patent, we first extract all IPC codes from its relevant documents. We sort the codes in descending order of the number of corresponding relevant documents, i.e., $c_a \succ c_b$ if $\#\text{rel}(c_a) > \#\text{rel}(c_b)$ where $\#\text{rel}(c)$ indicates the number of relevant documents containing the code c. Then, we scan from the top and remove the code if it covers all relevant documents (i.e., $\#\text{rel}(c) = |R|$) because such a code is too general and does not help to measure true diversity. After this, we assume that each remaining code can represent a topic for the query patent, and map relevant documents to their corresponding topics. In our experiments, the queries in USPTO and EPO include 4.94 and 8.66 topics, respectively.

Since patent documents generally contain IPC codes, it could be argued that diversification can be performed using IPC codes that appear in initial retrieval results. That is, the topic set for each query patent is directly estimated by the IPC codes, i.e., $T = \{t_1 = c_a, t_2 = c_b, \ldots\}$. However, the topics of IPC codes are very abstract and general, e.g., "H01F 1/01" means "magnets or magnetic bodies of inorganic materials". Thus, many documents in the initial retrieval result are related to the same IPC topics, and the diversification algorithm may not perform effectively. Instead, we assume that true topics in a query patent are more specific and concrete. So, we generate sufficient topic phrases for representing detailed topics (as described in Section 3.3). In addition, we consider IPC codes as a crude estimation for true topics, and use them for only evaluating diversity in retrieval results. In the future, we are able to evaluate the IPC-based diversification approach if manually judged query aspects are provided.

[2] http://www.wipo.int/classifications/en/

Table 2: Retrieval Results using Relevance Metrics. The baseline retrieval results for USPTO are generated by the query generation method described in [33], and the baseline for EPO uses the retrieval model proposed in [29]. DSP and LTR denote DSPApprox [11] and our learning-to-rank topic identification method (Section 3.3), respectively. In each column, a significant improvement over each method is marked by the first letter of the method, e.g., B indicates improvement over Baseline, and the paired *t*-test is performed with $p < 0.05$. The best performance is marked by bold. Each retrieval result is truncated at rank 100.

Collection	Method	MAP	PRES	NDCG	Recall@20	Recall
USPTO	Baseline	0.1221 (0.0%)	0.3441 (0.0%)	0.3112 (0.0%)	0.1849 (0.0%)	0.4261 (0.0%)
	DSP	0.1473 [B] (+20.64%)	0.3643 [B] (+5.86%)	0.3483 [B] (+11.92%)	0.2109 [B] (+14.06%)	0.4282 (+0.49%)
	LTR	**0.1568** [BD] (+28.42%)	**0.3789** [BD] (+10.10%)	**0.3596** [BD] (+15.55%)	**0.2216** [BD] (+19.85%)	**0.4441** [BD] (+4.22%)
EPO	Baseline	0.2414 (0.0%)	0.5030 (0.0%)	0.4328 (0.0%)	0.2369 (0.0%)	0.5159 (0.0%)
	DSP	0.2481 [B] (+2.78%)	0.5070 (+0.79%)	0.4416 [B] (+2.03%)	0.2447 [B] (+3.29%)	0.5166 (+0.14%)
	LTR	**0.2585** [BD] (+7.08 %)	**0.5109** (+1.57%)	**0.4546** [BD] (+5.04%)	**0.2543** [BD] (+7.34%)	**0.5189** (+0.58%)

Table 3: Diversification Results. Base indicates the method to generate initial retrieval results for each collection ([33] is used for USPTO, and [29] is employed for EPO). DSP and LTR denote DSPApprox [27] and our learning-to-rank topic identification method (Section 3.3), respectively. In each column, a significant improvement over each method is marked by the first letter of the method, e.g., B indicates improvement over Baseline, and the paired *t*-test is performed with $p < 0.05$. Also, the best performance is marked by bold. Each retrieval result is truncated at rank 100.

Collection	Method	NRBP	α-NDCG	ERR-IA	MAP-IA	S-Recall
USPTO	Baseline	0.1662 (0.0%)	0.4158 (0.0%)	0.2015 (0.0%)	0.0832 (0.0%)	0.7074 (0.0%)
	DSP	0.2299 [B] (+38.35%)	0.4850 [B] (+16.64%)	0.2607 [B] (+3.55%)	0.1007 [B] (+7.24%)	0.7088 (+0.19%)
	LTR	**0.2370** [BD] (+42.63%)	**0.4914** [B] (+18.18%)	**0.2686** [BD] (+13. 10%)	**0.1057** [B] (+22.10%)	**0.7186** [B] (+1.58%)
EPO	Baseline	0.1312 (0.0%)	0.4345 (0.0%)	0.1650 (0.0%)	0.1289 (0.0%)	0.6256 (0.0%)
	DSP	0.1433 [B] (+9.22%)	0.4493 [B] (+3.41%)	0.1766 [B] (+7.03%)	0.1314 [B] (+1.94%)	0.6257 (+0.02%)
	LTR	**0.1446** [B] (+10.21%)	**0.4522** [B] (+4.07%)	**0.1778** [B] (+7.76%)	**0.1325** [B] (+2.79%)	**0.6296** (+0.64%)

4.4 Baseline Retrieval Generation

To generate initial retrieval results, automatic patent query generation methods were employed. Three query generation methods (i.e., [16][33][42]) were tested, and EX-RM [33] was chosen as a baseline method because it significantly outperformed the others in our initial experiment using the USPTO collection. Following previous work [33], we first generate unigram queries by ranking the single terms in query documents; we derive unigram language models based on query documents, and use Kullback-Leibler divergence between query models and collection models for the ranking. Then, the original queries are expanded by relevance models derived from the same IPC documents (i.e., the documents containing at least one common IPC code of the query patent). This expanded query is called EX-RM. In addition to this, noun phrases from query documents are selectively appended to the EX-RM query (EX-RM-NP) because in the experiments of [33] some queries are degraded by added phrases. In our work, we only use EX-RM as a baseline query since selection of effective noun phrases requires more complex statistical learning, and EX-RM-NP could not significantly outperform EX-RM over all queries (see [33]). For retrieval, we use the Indri language model framework [40].

To develop baseline retrieval results for EPO, we use the method described in [29], which performed effectively on the same corpus in CLEF-IP 2010 [35]. Briefly, each query patent was processed by lemmatization and key-phrase extraction, and lemmas and extracted phrases were indexed separately. Then, Okapi BM25 and Indri were used for producing multiple retrieval results, and a SVM regression was employed to merge the different retrieval results (see [29] for more details).

4.5 Parameter Settings

The diversification algorithm described in Section 3.2 is applied to the top 200 documents in initial retrieval results. For web search tasks, the PM-2 performed better with top 50 documents [10], but prior-art search requires the examination of more documents (e.g., top 100 documents [22]). Thus, we empirically use top 200 documents, and consequently, the topic phrase identification techniques (i.e., DSPApprox and the learning-to-rank method) are also performed with these top 200 documents. In addition, we need

Table 4: Feature Analysis using USPTO. In each column, a * indicates a significant difference from {All}, and the paired *t*-test is performed with $p < 0.05$. Each metric is measured by the top 100 documents of retrieval results.

Features	MAP	NRBP
{All}	0.1568 (0.0%)	0.2370 (0.0%)
{All} − {Cohesiveness}	0.1534 (−2.17%)	0.2291 (−3.33%)
{All} − {Relevance}	0.1465* (−6.57%)	0.2223* (−6.20%)
{All} − {Importance}	0.1407* (−10.27%)	0.2171* (−8.40%)
{All} − {Predictiveness}	0.1355* (−13.58%)	0.2084* (−12.07%)

to tune two free parameters for this algorithm, i.e., λ and μ (see Eq. (2) and Eq. (3)). For this, we consider each value in the range of [0.1,1.0] with an increment of 0.1. Also, the topic phrase identification techniques require the free parameter, i.e., k, which indicates the number of topic phrases to be extracted from the candidate pool. We consider $k = \{5, 10, 20, 40, 60, 80, 100\}$. For fair comparison, tuning for these parameters is performed under 10-fold cross-validation with random partitioning; we randomly divide all queries into 10 partitions, and conduct 10 different tests in which each case tests with 1-partition queries by training with the other 9-partition queries. The learning-to-rank topic identification is also performed using the same 10-fold cross-validation. Note that the average number of phrases in the pool is 487.17 and 313.89 over USPTO and EPO query patents, respectively.

5. RESULTS

5.1 Retrieval Effectiveness

We first verify the retrieval effectiveness of the ranked results obtained by each method. Table 2 shows the evaluation results using both USPTO and EPO. In that, DSP and LTR denote diversification using DSPApprox and the learning-to-rank topic identification, respectively. Each retrieval result is truncated at rank 100. For each retrieval result, its retrieval effectiveness is measured by the relevance metrics (i.e., MAP, NDCG, PRES, and recall), and we report an average value of each metric over the query patents in each corpus.

First, our diversification framework can significantly improve baseline retrieval results on most metrics, while recall (at 100) is only improved by LTR using the USPTO collection. That is, the diversification keeps the relevant documents appearing in the initial rank results, and effectively promotes their ranks. This is what we intended by Eq. (3) (in Section 3.2) and the promoted relevant documents would be related to more topic phrases since the diversification algorithm can place the documents for salient topics at higher ranks. In addition, this result is important because, using the diversification, patent examiners are more likely to find relevant patents in early ranks. To highlight this, we measure early recall (recall@20) that can identify the extent of the relevant documents promoted from relatively lower ranks (i.e., below the top 20). From this, we observe that the diversification can retrieve significantly more relevant documents at rank 20. Moreover, the MAP and NDCG scores increase if we use either LTR or DSP for the topic phrase identification, which also supports the same result. Second, LTR looks more effective than DSP in terms of relevance metrics. In USPTO, LTR significantly outperforms DSP in all cases. In particular, recall (at 100) is significantly improved by LTR, which means that the topics identified by LTR can help to promote the relevant documents initially ranked below the top 100 and patent examiners can find more relevant documents by LTR. However, in EPO LTR is significantly better in terms of MAP, NDCG, and early recall while PRES and recall scores are not significantly improved. This is because in EPO LTR is not helpful to promote the relevant documents initially ranked below the top 100 and PRES is significantly affected by recall performance (Note that PRES reflects the normalized recall). Comparing to EPO, the baseline retrieval of USPTO poorly performs (e.g., 0.4261 vs 0.5159 in recall), and LTR may have more chances to promote the relevant documents initially ranked below the top 100 in USPTO.

5.2 Diversification Performance

Next, we evaluate the "diversity" of retrieval results obtained by each method. Specifically, we measure the values of NRBP, α-NDCG, ERR-IA, MAP-IA, S-Recall at overall ranks. Table 3 presents the diversity-based evaluation results. First, for both collections, our diversification approach is effective for generating significantly more diversified results. The diversity performance in USPTO is especially improved, e.g., +42.63% is achieved in terms of NRBP. This result indicates that the diversification can increase the ranks of relevant documents related to diverse topics, and enabling the user to recognize the diverse aspects of query patents. Second, the sub-topic recall is less improved by the diversification. We believe the cause of this result is that within rank 100, the baseline has already found sufficient amounts of each topic from retrieved relevant documents. Thus, the diversification may not find new topics not covered by the initial retrieval results. Third, the diversification performance in USPTO looks better than that in EPO whereas the retrieval effectiveness measured in EPO is much better than that measured in USPTO (see Table 2). This is because the relevant documents in EPO includes more topics, i.e., the (average) number of topics in relevant documents of USPTO and EPO is 4.94 and 8.66, respectively. Thus, the retrieval results for USPTO easily contain relatively more topics, i.e., the ratio of found topics to the whole topics. Lastly, different from the relevance results (Table 2), LTR is significantly better than DSP in terms of only NRBP and ERR-IA when using the USPTO collection. As discussed in the relevance results, the ranks of the overall relevant documents in USPTO are largely promoted by LTR (see Table 2), and such improvements may influence on the NRBP and ERR-IA measures.

Although we use the diversity measures for the evaluation, prior-art search primarily focuses on retrieving more relevant documents and improving retrieval effectiveness is more significant. However, more diversified results can be useful as the users can recognize diverse aspects of the query patent. Furthermore, our diversification approach does not miss the relevant documents in initial retrieval results, and improved retrieval effectiveness is promising because more relevant documents are found at early ranks.

5.3 Feature Analysis

We now provide an analysis of features used in the learning-to-rank topic identification (LTR) described in Section 3.3. As summarized in Table 1, we use four different types of features for LTR, and conduct another experiment to examine the influence of each feature type for diversification. Since calculating the effects of some features on the topic phrase identification is very difficult, we indirectly measure their effectiveness by performing diversification using the topic phrases generated by the target features. We first extract topic phrases by LTR using all features with 10-fold cross-validation, and diversify initial retrieval results. Then, following

Table 5: Examples of the top 5 topic phrases for a sample query patent. AvePrec denotes Average Precision, and each metric is measured by the top 100 documents in the retrieval result of each method. The baseline retrieval result is generated by [33]. DSP and LTR indicate DSPApprox [27] and our learning-to-rank method (Section 3.3) to generate topic phrases, respectively.

Query Patent		
Title: Method and apparatus for providing content on a computer system based on usage profile. **Abstract** A method and apparatus for determining a *computer system usage profile* … A *basic input output system (BIOS) module* and/or an *operating system module* obtain computer system usage profile information by tracking events such as the frequency of re-boots, the time required to boot-up and shut-down the operating system … data is collected and communicated to a *profile server*…		
Baseline		
AvePrec NRBP	0.1288 0.2366	
Diversification		
	DSP	**LTR**
AvePrec NRBP	0.1544 0.2948	0.2183 0.3594
Top 5 Identified Topic Phrases	1. computer device 2. event 3. execution 4. OS 5. microprocessor	1. user profile data 2. BIOS module 3. boot process 4. disk drive 5. boot time

the same partitions, we identify topic phrases by all features except for one feature type, and run the diversification with the identified phrases. After this, we observe the final performance change by the feature drop, i.e., how much the topic phrase identification depends on the dropped feature type. Note that the parameters for this experiment are the same as used previously.

Table 4 shows the feature analysis using the USPTO collection where LTR is notably effective. In that, we use MAP and NRBP (by top 100 documents) for the analysis. First, all the features we used seem to have positive effects on diversification. Whenever a feature is dropped, the value of every metric decreases. Second, the cohesiveness features look less influential than the others since these features may not cause a significant decrease in both MAP and NRBP metrics. One possible reason for this is that the PMI values to represent "cohesiveness" of topic phrases (see Section 3.3.2) might be less effective to find "relevant" phrases (i.e., useful for retrieving relevant documents). However, we additionally identify other significant features, i.e., relevance and importance that represent the relevance of phrases to each section of query patents and their predicted effectiveness to retrieve relevant documents (i.e., query performance predictors).

5.4 Qualitative Analysis

In this section, we provide a qualitative analysis of our topic phrase identification using an example. Table 5 shows the top 5 topic phrases generated for an example query patent (which is in the same as Figure 1 in Section 1). The application in this patent provides profiled information about computer system usage, and several modules such as Basic Input Output System (BIOS), Operating System (OS), and Profile Server make up the whole system. For this query patent, the baseline performs reasonably well (its average precision score is slightly higher than MAP over all queries (see Table 2)), and diversification is effective for improving the initial retrieval result.

One observation is that DSPApprox can identify phrases that describe other query terms, i.e., phrases with high predictiveness. For example, "computer device" appears to be highly representative for the peripheral devices used for BIOS, e.g., printer and keyboard, and "event" stands for the actions recorded in the usage profile, e.g., re-boot and shut-down. On the other hand, our learning-to-rank method (LTR) can recognize key phrases that describe significant topics in the query patent and that are more effective for retrieving relevant documents. As an example, "user profile data" and "BIOS module" are important components for the application, and as discussed in Section 1 (using Figure 1), we assume that such components may form query topics. In addition, these phrases are related to several relevant documents for this query patent (see Figure 1). Moreover, the other phrases, e.g., "boot process" and "boot time" are also effective for retrieving relevant documents such as "Reducing operating system start-up/boot time through disk block relocation" (the title of a relevant document for this query patent).

Another interesting observation is that DSPApprox favors unigram phrases. Although we use the same phrase pool for both methods, unigram phrases are more highly ranked by DSPApprox. This bias can be caused by the high predictiveness scores of one-word phrases since they tend to co-occur with more terms than multi-word phrases. The LTR method uses a supervised learning framework, and the weight on the predictiveness feature can be effectively controlled.

6. CONCLUSION

In this paper, we addressed the problem of diversifying patent search results based on query patents. To solve this, we propose a phrase-level diversification approach. Given an initial retrieval result of each query patent, we identify topic phrases to represent underlying query topics, and diversify based on the identified phrases. Through experiments, we showed that this phrase-level diversification can improve patent search results in terms of retrieval effectiveness and diversity. In addition, we devise a learning-to-rank method to identify topic phrases, and verify its effectiveness in comparison to the state-of-the-art topic term identification algorithm. One advantage of our approach is that laborious human effort to generate training examples or relevance judgments is not required, and this can help to reproduce the proposed work. However, evaluation with manually-judged relevance and diversity topics could help to verify the practical effectiveness of our method. This is left for future work. In addition, we plan to apply our approach to other domains (e.g., legal search), and verify its generalizability.

7. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] Agrawal, R., Gollapudi, S., Halverson, A., and Leong, S. (2009). Diversifying search results. *WSDM*, 5-14.

[2] Bashir, S. and Rauber, A. (2010). Improving retrievability of patents in prior-art search. *ECIR*, 457-470.

[3] Carbonell, J., and Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. *SIGIR*, 335-336.

[4] Carterette, B. and Chandar, P. (2009). Probabilistic models of ranking novel documents for faceted topic retrieval. *CIKM*, 1287-1296.

[5] Chapelle, O., Metlzer, D., Zhang, Y., and Grinspan, P. (2009). Expected reciprocal rank for graded relevance. *CIKM*, 621-630.

[6] Clarke, C. L. A., Craswell, N., Soboroff, I., and Cormack, G. V. (2010). Overview of the TREC 2010 web track. *TREC*.

[7] Clarke, C. L. A., Kolla, M., and Vechtomova, O. (2009). An effectiveness measure for ambiguous and underspecified queries. *ICTIR*, 188-199.

[8] Clarke, C. L. A., Kolla, M., Cormack, G. V., Vechtomova, O., Ashkan, A., Buttcher, S., and MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. *SIGIR*, 659-666.

[9] Cronen-Townsend, S., Zhou, Y., and Croft, W. B. (2002). Predicting query performance. *SIGIR*, 299-306.

[10] Dang, V., and Croft, W. B. (2012). Diversity by proportionality: an election-based approach to search result diversification. *SIGIR*, 65-74.

[11] Dang, V. and Croft, W. B. (2013). Term level search result diversification. *SIGIR*, 603-612.

[12] Dang, V., Xue, X., and Croft, W. B. (2011). Inferring query aspects from reformulations using clustering. *CIKM*, 2117-2120.

[13] Dou, Z., Hu, S., Chen, K., Song, R., and Wen, J.-R. (2011). Multi-dimensional search result diversification. *WSDM*, 475-484.

[14] Fall, C. J., Torcsvari, A., Benzineb, K., and Karetka, G. (2003). Automated categorization in the international patent classification. *ACM SIGIR Forum*, 37(1): 10–25.

[15] Fujii, A., Iwayama, M., and Kando, N. (2007). Overview of the patent retrieval task at the NTCIR-6 workshop. *NTCIR-6*.

[16] Ganguly, D., Leveling, J., Magdy, W., and Jones, G. J. F. (2011). Patent query reduction using pseudo-relevance feedback. *CIKM*, 1953-1956.

[17] Ganguly, D., Leveling, J., Magdy, W., and Jones, G. (2011). United we fall, divided we stand: a study of query segmentation and PRF for patent prior art search. *PaIR*.

[18] He, J., Hollink, V., and de Vries, A. (2012). Combining implicit and explicit topic representations for result diversification. *SIGIR*, 851-860.

[19] He, B., and Ounis, I. (2006). Query performance prediction. *Information System,* 31(7): 585-594.

[20] Iwayama, M., Fujii, A., Kando, N., and Marukawa, Y. (2003). An empirical study on retrieval models for different document genres: patents and newspaper articles. *SIGIR*, 251-258.

[21] Joachims, T. (2006). Training linear SVMs in linear time. *KDD*, 217-226.

[22] Joho, H., Azzopardi, L., and Vanderbauwhede, W. (2010). A survey of patent users: an analysis of tasks, behavior, search functionality and system requirement. *IIiX*, 13-22.

[23] Kim, Y., Seo, J., and Croft, W. B. (2011). Automatic Boolean query suggestion for Professional Search. *SIGIR*, 825-834.

[24] Kim, Y., Seo, J., Croft, W. B., and Smith, D. A. (2014). Automatic suggestion of phrasal-concept queries for literature search. *Information Processing & Management*, 50(4): 568-583.

[25] Kim, Y. and Croft, W. B. (2014). Diversifying query suggestions based on query documents. *SIGIR*, 891-894.

[26] Lavrenko, V. and Croft, W. B. (2001). Relevance-based language model. *SIGIR*, 120-127.

[27] Lawrie, D. (2003). Language models for hierarchical summarization. *PhD Thesis*, University of Massachusetts.

[28] Lawrie, D., Croft, W. B., and Rosenberg, A. (2001). Finding topic words for hierarchical summarization. *SIGIR*, 349-357.

[29] Lopez, P., and Romary, L. (2010). Experiments with citation mining and key-term extraction for prior-art search. *CLEF*.

[30] Lupu, M., Piroi, F., Huang, X., Zhu,, J., and Tait, J. (2009). Overview of the TREC 2009 chemical IR track. *TREC-18*.

[31] Magdy, W. and Jones, G. J. F. (2010). PRES: a score metric for evaluating recall-oriented information retrieval applications. *SIGIR*, 611-618.

[32] Magdy, W. and Jones, G. J. F. (2011). A study on query expansion methods for patent retrieval. *PaIR*, 19-24.

[33] Mahdabi, P., Andersson, L., Keikha, M., and Crestani, F. (2012). Automatic refinement of patent queries using concept importance predictors. *SIGIR*, 505-514.

[34] Mase, H., Matsubayashi, T., Ogawa, Y., Iwayama, M., and Oshio, T. (2005). Proposal of two-stage patent retrieval method considering the claim structure. *ACM Transactions on Asian Language Information Processing*, 4: 190-206.

[35] Piroi, F., and Tait, J. (2010). CLEF-IP 2010: Retrieval experiments in the intellectual property domain. *IRF Technical Report.*

[36] Radlinski, F., Szummer, M., and Craswell, N. (2010). Inferring query intent from reformulations and clicks. *WWW*, 1171-1172.

[37] Rafiei, D., Bharat, K., and Shukla, A. (2010). Diversifying web search results. *WWW*, 781-790.

[38] Sanderson, M. and Croft, W. B. (1999). Deriving concept hierarchies from text. *SIGIR*, 206-213

[39] Santos, R. L. T., Macdonald, C., and Ounis, I. (2010). Exploiting query reformulations for web search result diversification. *WWW*, 881-890.

[40] Strohman, T., Metzler, D., Turtle, H., and Croft, W. B. (2005). Indri: a language-model based search engine for complex queries (extended version). *UMASS CIIR Technical Report*.

[41] Wang, J. and Zhu, J. (2009). Portfolio theory of information retrieval. *SIGIR*, 115-122.

[42] Xue, X. and Croft, W. B. (2009). Transforming patents into prior-art queries. *SIGIR*, 808-809.

[43] Xue, X. and Croft, W. B. (2009). Automatic query generation for patent search. *Proc. SIGIR*, 2037-2040.

[44] Zhai, C. and Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. *SIGIR*, 334-342.

[45] Zhai, C., Cohen, W. W., and Lafferty, J. (2003). Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. *SIGIR*, 10-17.