Automatic Refinement of Patent Queries using Concept Importance Predictors

Parvaz Mahdabi[†] Linda Andersson[‡] Mostafa Keikha[†] Fabio Crestani[†]
[†]University of Lugano, Faculty of Informatics, Lugano, Switzerland
{parvaz.mahdabi, mostafa.keikha, fabio.crestani}@usi.ch
[‡]Vienna University of Technology, Vienna, Austria
andersson@ifs.tuwien.ac.at

ABSTRACT

Patent prior art queries are full patent applications which are much longer than standard web search topics. Such queries are composed of hundreds of terms and do not represent a focused information need. One way to make the queries more focused is to select a group of key terms as representatives. Existing works show that such a selection to reduce patent queries is a challenging task mainly because of the presence of ambiguous terms. Given this setup, we present a query modeling approach where we utilize patent-specific characteristics to generate more precise queries. We propose to automatically disambiguate query terms by employing noun phrases that are extracted using the global analysis of the patent collection. We further introduce a method for predicting whether expansion using noun phrases would improve the retrieval effectiveness.

Our experiments show that we can obtain almost 20% improvement by performing query expansion using the true importance of the noun phrase queries. Based on this observation, we introduce various features that can be used to estimate the importance of the noun phrase query. We evaluated the effectiveness of the proposed method on the patent prior art search collection CLEF-IP 2010. Our experimental results indicate that the proposed features make good predictors of the noun phrase importance, and selective application of noun phrase queries using the importance predictors outperforms existing query generation methods.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Query formulation, Relevance Feedback

General Terms

Experimentation, Performance, Measurement

Keywords

Patent Search, Query Generation, Relevance Model

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.

1. INTRODUCTION

Patent prior art search is composed of a search over previously filed patents with the aim of retrieving relevant documents, which may invalidate or at least describe prior art work in a patent application, (henceforth referred to as *query patent* in this paper). The challenges of patent prior art search are different from those of standard ad hoc text and web search. The first difference is associated to the query length: patent prior art queries are full patent applications comprising of hundreds of words as opposed to ad hoc search and web search where the queries are very short. The second issue is related to the fact that patent prior art search is a recall oriented task where the goal is to retrieve all relevant documents at early rank positions as opposed to ad hoc and web search, where the goal is to achieve high precision.

We present a solution to the patent prior art search problem allowing the user to submit a full patent document as a query and the retrieval system identifies related patent documents from a corpus accordingly. To provide such functionality, we propose two techniques to process patent documents on demand and extract terms and key phrases in order to form a query to retrieve relevant documents from the patent corpus. These two approaches can be summarized as follows: i) one approach extracts single terms from the query patent using the KL-divergence between the query patent and the collection; ii) the other approach refines the original query by expanding it with selected key concepts (i.e., bigrams or phrases) from the query patent using the global analysis of the patent collection. These two approaches are complementary to each other: the first approach extracts generic terms, favoring recall, while the second aims to find a clear focus for the query by providing more specific phrases, thus increasing precision.

However, after evaluating mean average precision (MAP) over the topic set, we observe that the expanded rank list is not statistically different from the unexpanded rank list. After performing failure analysis on a per query basis, we detected a large variation in performance for different queries; we found that while indeed the expanded rank list improves the quality of results for many queries considerably, the quality of results is poor for some other queries. One of the reasons explaining the hurting retrieval performance for some queries is attributed to the topics for which the main aspect of the query is not considered during expansion. In such cases, only important concepts describing the partial aspects of the query are extracted. This observation shows that expansion using noun phrases was not consistently beneficial for all queries. This suggests that the decision about

query expansion using concepts should be taken in a query dependent way.

In this paper, we propose a method for distinguishing between queries and deciding when to selectively use the result of a refinement technique that is likely to improve the retrieval performance. Our goal is to find queries that have highly positive changes in query performance using refinement. To this end, we use query performance predictors (pre and post-retrieval) [2, 9, 1, 20] and patent-specific features in order to find highly performing queries in the expanded retrieval rank list. To decide when to use the result of the expanded list, we rely on a machine learning approach that tries to predict which one of the two competing approaches will offer the best result for a given query.

This prediction is performed in a selective query refinement framework [23, 1, 3, 8, 10]. It is desirable for us to build a robust patent retrieval system which can be used in an operational setting. We aim at quantifying the performance of the queries in order to build a robust system which can invoke different retrieval strategies in a query dependent way according to the estimated performance of a query. To the best of our knowledge no previous work have used the query performance predictors in the patent domain.

In this paper we explore extracting concepts which explicitly occur in the query patent itself. We also study extracting important concepts associated with the information need underlying the query patent through the process of query expansion by building a relevance model: 1) via pseudo relevance feedback; 2) using sample relevant documents. Our contributions are:

- Investigating different ways of estimating the query model from a query patent utilizing patent-specific characteristics.
- Presenting a method for predicting whether query expansion using concepts would improve the retrieval effectiveness.
- Investigating the impact of different types of expansion in our selective query expansion framework.

We evaluate our model on CLEF-IP 2010 collection and we report significant improvement over the strong CLEF-IP baselines. The results show that by incorporating the predicted noun phrase importance in a selective query expansion framework, we can achieve significant improvement over using query expansion for all queries.

The rest of this paper is organized as follows: Section 2 briefly reviews the related work; Section 3 and 4 define the original and the expanded query models; Section 5 describes the prediction model using query performance measures; Section 6 reports the experimental results and Section 7 reports the conclusion of the work.

2. RELATED WORK

The work performed by patent examiners involves manual query formulation from the query patent in order to find invalidating claims. They consider high term frequency in the document to be a strong indicator of a good query term. Methods to shorten a query patent have been studied for a few years and this research direction has shown to be very challenging mainly due to the presence of ambiguous terms. In the third NTCIR workshop [12], the first patent prior art search track was introduced and several patent test collections were released. Some early works [11, 13] using this collection focused on extracting keywords to form a reduced query.

A recent line of work advocates the use of the full patent application as the query to reduce the burden on patent examiners. This direction has been started by Xue and Croft [21], who conducted a series of experiments in order to examine the effect of different patent fields, and concludes with the observation that the best Mean Average Precision (MAP) is achieved using the text from the *description* section of the query patent with raw term frequencies. Fuji [6] showed that retrieval effectiveness can be improved by combining IR methods with the result of citation extraction.

The current developments in the patent search are driven by the Intellectual Property task within the CLEF[1] initiative. Several teams participated in the prior art search task of the CLEF-IP 2010 and proposed approaches to reduce the query patent by extracting a set of key terms from it. Different participating teams experimented with term distribution analysis in a language modeling setting, and they employed the document structure of the patent documents in various ways [18].

So far, one of the most comprehensive descriptions of the problem and possible solutions for the prior art search is presented by Magdy and Lopez [15]. The authors show that the best performing run of CLEF-IP 2010 uses citations extracted by training a Conditional Random Field (CRF). The second best run uses a list of citations extracted from the patent numbers within the *description* field of some patent queries. They also show that the best run employs sophisticated methods of retrieval using two complementary indices, one constructed by extracting terms from the patent collection and the other built from external resources such as Wikipedia. They compared this two approaches and conclude with an interesting observation that the second best run achieves a statistically indistinguishable performance compared to the best run.

A recent work [7] studies the effect of using Pseudo Relevance Feedback (PRF) for reducing patent queries. Authors decompose a patent application into constituent text segments and compute the language modeling similarities by calculating the probability of generating each segment from the top ranked documents. They showed that although they achieve improvement over PRF, their approach is not able to achieve statistical significance gain over the second best result of CLEF-IP 2010. As a baseline for this paper, we consider an approach which produces comparable results to the second rank participating group of CLEF-IP 2010 and we compare different variations of our proposed method to this baseline. We show that our proposed method significantly outperform the baseline.

In addition to the well known MAP metric we use the Patent Retrieval Evaluation Score (PRES) which is originally proposed by Magdy and Jones [14]. Authors showed that MAP can be a misleading metric for evaluating the performance of patent prior art search because of its inherent characteristic of favoring precision over recall. This metric measures the system recall and the quality of ranking in one score. Our experiments report an improvement in terms of MAP, recall and PRES over the baseline.

[1] http://www.ir-facility.org/clef-ip

3. ESTABLISHING A BASELINE: SINGLE TERM EXTRACTION

Patent prior art queries are full patent applications which are much longer than standard web search topics. These queries are composed of hundreds of terms and do not represent a focused information need. Thus, the success of the patent prior art search relies on the selection of good search queries.

Our goal is to estimate the query model of a query patent in a language modeling framework. This estimation enables us to identify the importance of terms and assign weights to them accordingly. By modeling the term distribution of the query patent we get a detailed representation of the query patent which allows us to expand the query, and to refine the query model by considering relationships between terms. This approach is used to bridge the vocabulary gap between the underlying information need of the query patent and the collection.

In this section, we first describe how we create a language model Θ_Q for the query patent. We use the maximum likelihood estimate smoothed by the background language model, as expressed in Equation 1 to avoid sparseness issues.

$$P(t|\Theta_Q) = (1-\lambda) \cdot P_{ML}(t|D) + \lambda \cdot P_{ML}(t|C) \quad (1)$$

where maximum likelihood estimate P_{ML} is calculated as follows:

$$P_{ML}(t|D) = \frac{n(t,D)}{\sum_{t'} n(t',D)} \quad (2)$$

We introduce a unigram query model by estimating the importance of each term according to a weighted log-likelihood based approach as expressed below:

$$P(t|Q_{orig}) = Z_t \; P(t|\Theta_Q) \; log\left(\frac{P(t|\Theta_Q)}{P(t|\Theta_C)}\right) \quad (3)$$

where $Z_t = 1/\sum_{t \in V} P(t|Q_{orig})$ is the normalization factor and is defined as the Kullback-Leibler divergence between Θ_Q and Θ_C. This approach favors terms that have high similarity to the document language model Θ_Q and low similarity to the collection language model Θ_C. For the rest of the paper Q_{orig} serves as our unigram baseline.

In order to model the query patent more precisely we need a source of additional knowledge about the information need. Patent documents are annotated with International Patent Classifications[2] (IPC). Such classes are language independent keywords assigned as metadata to the patent documents. They are categorizing the content of a patent document and reflecting the field of technology of a patent. These IPC classes resemble tags assigned to documents (henceforth referred to as *conceptual tags* in this paper).

Our goal is to build a relevance model by employing documents that have at least one conceptual tag in common with the query topic. Each relevant document from this sample is assumed to serve as evidence towards the estimation of the relevance model. Note that this relevant samples are not part of the relevance information.

Our approach to construct the relevance model Θ_{IPC} is the following. First, we estimate the level of relevance of a document D with $P(D|\Theta_{IPC})$. Then the top-k terms with the highest probability $P(t|D)$ are picked and used to build Θ_{IPC}. The sample distribution $P(t|\Theta_{IPC})$ is calculated according to Equation 4. This sampling is dependent on the original query patent as it utilizes documents with similar conceptual tags to the query patent.

$$P(t|\Theta_{IPC}) = \sum_{D \in IPC} P(t|D) \cdot P(D|\Theta_{IPC}) \quad (4)$$

Now we explain how the level of relevance of a sample document D is estimated. We can not assume documents in the relevance set to have equal importance. The reason is that documents in the relevance set can be multi-faceted and therefore not entirely relevant to the information need represented by the query patent. So we need to assign importance to the documents according to their level of relevance. We approximate the relevance of a sample document D, denoted by $P(D|\Theta_{IPC})$, based on the divergence between D and Θ_{IPC}. We measure this divergence by calculating the log-likelihood ratio between D and Θ_{IPC}, normalized by the collection C as defined below:

$$\begin{aligned} P(D|\Theta_{IPC}) &\propto H(\Theta_D, \Theta_C) - H(\Theta_D, \Theta_{IPC}) \\ &= Z_D \sum_{t \in V} P(t|\Theta_D) \log \frac{P(t|\Theta_{IPC})}{P(t|\Theta_C)} \end{aligned}$$

where $H(\Theta_D, \Theta_C)$ represents the cross entropy between the sample document D and the collection and $H(\Theta_D, \Theta_{IPC})$ represents the cross entropy between the sample document D and the topical model of relevance Θ_{IPC}. We define $Z_D = 1/\sum_{D \in IPC} P(D|\Theta_{IPC})$ as a document-specific normalization factor. This approach assigns higher scores to documents which contain specific terminology and are more similar to Θ_{IPC} and less similar to the language model of the collection Θ_C. For estimating the term importance $P(t|D)$ in Equation 4, we consider the smoothed maximum likelihood estimate of a term to avoid sparseness issues as shown in Equation 1.

We then mix the estimated relevance model using the conceptual tags and the original query in order to build an expanded query. To do this, we use a linear combination as expressed in the following:

$$P(t|Q_{ex}) = (1-\mu) \cdot P(t|\theta_{IPC}) + \mu \cdot P(t|Q_{orig}) \quad (5)$$

where $P(t|Q_{orig})$ and $P(t|\theta_{IPC})$ show the probability of term t given the original query model and the estimated relevance model, respectively. We refer to this expanded query model as EX-RM.

The performance of different unigram query models presented in this section are compared with each other in the experiment section. For comparison purposes, we also show the performance of Pseudo Relevance Feedback (PRF), as a reference baseline, and we compare this to the query models built in this section. In the experiment section we show that the relevance model constructed based on the conceptual tags (EX-RM) outperforms the result of PRF. To generate a query we pick the top-k terms with higher weights from each query model.

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

4. PHRASE EXTRACTION

In this section, we present our approach for extracting key phrases with similar semantics to patent query. Such phrases will be used to expand and disambiguate the initial unigram query Q_{orig} as estimated in Section 3. We then use the expanded noun phrase query Q_{expand} to retrieve relevant documents from the patent corpus. We use both corpus statistics and linguistic heuristics for finding meaningful phrases. The detail of our solution is as follows: First we identify the set of all candidate key phrases S_p for the query document d, as explained in Section 4.1. We then evaluate the significance of each candidate phrase $p \in S_p$, by assigning a score $s(c)$ between 0 and 1 to each phrase as shown in Section 4.2. Finally we select the top-k phrases to construct an expanded query. In the evaluation section, the quality of Q_{expand} is compared to the unigram query Q_{orig} by reporting the document retrieval results.

4.1 Extracting Candidate Key Phrases

We recognized and extracted candidate noun phrases with length at most 5 from the query patent, with the help of the Stanford *part of speech tagger* [19]. The part-of-speech tagger assigns part-of-speech tags (e.g., noun (NN), verb (VB), adjective (JJ), etc.) to each term $w \in d$. The part-of-speech tagger applies a pre-trained classifier on w and it's surrounding terms in d. We consider all noun phrases as candidate phrases, and compute S_p by extracting all such phrases from d. We are interested to find ordinary phrases rather than extracting named entities. The example noun phrase patterns[3] that we used are listed in Table 1.

Table 1: Example of noun phrase patterns and instances

Pattern	Instance
NN	leukocyte
JJ NN	miniature column
NN NN	blood filtration
JJ JJ NN	hydrophobic polymerizable monomer
NN NN NN	leukocyte removal performance
JJ NN NN	nonwoven polyester fabric
JJ JJ JJ NN	protonic neutral hydrophilic part
NN NN NN NN	blood transfusion side effect
...	...
NN NN NN NN NN	coating leukocyte removal filter material

4.2 Scoring Key Phrases

We used the two methods proposed in [22] for scoring phrases. We briefly revisit the two scoring approaches. The first approach employs the TF/IDF information for evaluating the importance of each phrase, while the second calculates a weight for each phrase using mutual information.

4.2.1 Scoring Phrases based on TF/IDF

The first scoring technique assigns a score $s_t(p)$ to a phrase p which is based on a linear combination of the total TF/IDF score of all terms in p and the degree of *coherence* of p. Coherence quantifies the likelihood of the constituting terms in forming a single concept and is a measure of stability of a phrase in the corpus. Formally, let $|p|$ denote the number of terms in phrase p; we use $w_1, w_2, ..., w_{|p|}$ to refer to the actual terms. $s_t(p)$ is formally defined as:

$$s_t(p) = \sum_{i=1}^{|p|} tf.idf(w_i) + \alpha \cdot coherence(p) \quad (6)$$

where $idf(w_i)$ is the inverse document frequency of w_i and α is a tunable parameter. The first component in $s_t(p)$ captures the importance of each term in p by using the TF/IDF value. A rare term that occurs frequently in d is more important than a common term frequently appearing in d (with low idf). This component will reward rare phrases. The second component in $s_t(p)$ represents how coherent the phrase p is. The coherence of p is defined as:

$$coherence(p) = \frac{tf(p) \cdot (1 + \log tf(p))}{\frac{1}{|p|} \cdot \sum_{i=1}^{|p|} tf(w_i)} \quad (7)$$

where $tf(p)$ is the number of times the phrase p appears in the document d. Equation 7 compares the frequency of p with the average tf of its terms. The additional logarithmic component give importance to phrases appearing frequently in the input document.

4.2.2 Scoring Phrases based on Mutual Information

The second scoring technique assigns $s_m(p)$ to a phrase p which is based on the mutual information (MI) between the terms of phrase p and the idf values from the background corpus. $s_m(p)$ is a linear combination of idf values of terms in p, frequency of p, and the point-wise mutual information among them. $s_m(p)$ is formally defined as:

$$s_m(p) = \sum_{i=1}^{|p|} idf(w_i) + \log \frac{tf(p)}{tf(POS_p)} + PMI(p) \quad (8)$$

where $tf(p)$ and $tf(POS_p)$ are the number of times p and its part-of-speech tag sequence POS_p appear in d and its part-of-speech tag sequence POS_d, respectively. The first part represents how descriptive each term in phrase p is. The second part identifies how frequent the phrase p is at the corresponding POS tag sequence in the document. The third part captures how likely are the terms to from a phrase together. Mutual information compares the probability of observing the constituting terms in phrase p together (the joint probability) with the probabilities of observing those terms independently. The *PMI(p)* for a phrase p is defined as:

$$PMI(p) = \log(\frac{P(p)}{\prod_{i=1}^{|p|} P(w_i)}) \quad (9)$$

where $P(w_i)$ and $P(p)$ denote the probability of occurrence of w_i and phrase p respectively at the appropriate part-of-speech tag sequence. They are formally defined as:

$$P(p) = \frac{tf(p)}{tf(POS_p)},\ P(w_i) = \frac{tf(w_i)}{tf(POS_{w_i})} \quad (10)$$

In order to emphasize on the importance of how frequent the phrase p occurs in the document d we weight Equation 8 by $\frac{tf(p)}{tf(POS_p)}$ as shown below:

[3]Presented instances belong to query 433 in the topic set.

$$s'_m(p) = \frac{tf(p)}{tf(POS_p)} \times \left(\sum_{i=1}^{|p|} idf(w_i) + \log \frac{tf(p)}{tf(POS_p)} + PMI(p) \right)$$

We only keep the top few highest scoring phrases to eliminate redundancy. We will do a deeper analysis on the number of selected phrases in the experimental section.

5. PREDICTING NOUN PHRASE EFFECTIVENESS

Our goal is to predict whether an expanded query using noun phrases will be more effective for retrieval than an unexpanded query. We evaluate the effectiveness of the expanded query by estimating the change in the average precision (AP) for each query. Let $AP(Q_{orig})$ and $AP(Q_{expand})$ be the AP of original unigram query and of the expanded query using noun phrases, respectively. We measure the performance change due to the Q_{expand} as expressed below,

$$chg(AP, Q) = \frac{AP(Q_{expand}) - AP(Q_{orig})}{AP(Q_{orig})} \quad (11)$$

We set a threshold at 10% for this change in AP to distinguish a good expanded query from a bad one and this indicates an effective noun phrase expansion. After identifying a good expanded query according to Equation 11, we use this estimate to decide whether the original query should be expanded or not. We then perform a selective query expansion (SQE) where we only expand effective noun phrase queries.

Before trying to estimate the effectiveness of the noun phrase query, it is interesting to know how good the SQE will perform if the true effectiveness value is used. To this end, we use the average precision of Q_{orig} and Q_{expand} to decide whether to expand a query or not. We refer to this approach as $oracle_{np}$ showing the potential upper bound of what can be achieved by combining the two rank lists based on true effectiveness. Table 2 shows the MAP for the top 1000 results with $oracle_{np}$ in comparison to the original unigram query model (baseline) and the expanded query model using noun phrases (QM-NP2). The † and ‡ symbols indicate that the improvement over the unigram baseline and QM-NP2 is statistically significant at $p<0.01$.

Table 2: Performance results using the true noun phrase effectiveness

model	MAP
baseline	0.1366
QM-NP2	0.1380
$oracle_{np}$	0.1649 † ‡

As the result of Table 2 suggests, we can achieve a 20% improvement over both unigram baseline and QM-NP2, by employing the true effectiveness of the noun phrases. We seek to reach this upper bound by a reasonable estimation of the correct AP values and the change of AP for each query according to Equation 11.

5.1 Features

In order to predict the effectiveness of an expanded query we use a set of features related to the query to estimate AP of both rank lists of Q_{orig} and Q_{expand}. These features will be explained in this Section.

The Query Clarity (QC) measure [2] quantifies the level of effectiveness of a query at retrieving a specific topic. The clarity measure is the Kullback-Leibler (KL) divergence between the query language model $P(w|Q)$ and the collection language model $P(w|C)$. Formally, the clarity score is defined as:

$$D_{KL}(Q||C) = \sum_{w \in V} P(w|Q) log \frac{P(w|Q)}{P(w|C)} \quad (12)$$

A higher clarity score indicates a clearer query with specialized vocabulary and a lower clarity score indicates a more ambiguous query with a very generic language. To calculate a clarity score in a given collection, a *relevance model* is constructed. This model captures the language usage of documents related to the query and therefore it is a collection-dependent query model.

We propose two measures inspired by the clarity measure using patent-specific characteristics. Let IPC_Q be the set of documents with similar topics to Q represented by conceptual tags. The first measure, called *Topical Clarity*, is defined as the KL-divergence between the language model of Q and the language model of IPC_Q. Formally, the Topical Clarity (TC) measure is defined as:

$$D_{KL}(Q||IPC_Q) = \sum_{w \in V} P(w|Q) log \frac{P(w|Q)}{P(w|IPC_Q)} \quad (13)$$

where $P(w|IPC_Q)$ is the relative frequency of term w in documents with similar conceptual tags to Q. We refer to this as the topical clarity. In this case, a larger KL-divergence indicates a query with fewer topics and therefore a more focused query, while a smaller KL-divergence indicates a query with a broader language use.

The second measure called *IPC-based Clarity* captures the similarity between the language usage of IPC_Q and the collection language model. This measure is defined as:

$$D_{KL}(IPC_Q||C) = \sum_{w \in V} P(w|IPC_Q) log \frac{P(w|IPC_Q)}{P(w|C)} \quad (14)$$

An alternative indication of the specificity of a query is to consider the distribution of the informative amount in the query terms [9]. This measure is defined by:

$$\gamma_1 = \sigma_{idf} \quad (15)$$

where σ represents the standard deviation of the *idf* of the terms in Q. Each query term can be associated with an inverse document frequency ($idf(w)$) describing the informative amount that a query term q carries. The $idf(w)$ is defined by:

$$idf(w) = log \frac{N - N_w + 0.5}{N_w + 0.5} \quad (16)$$

where N_w is the number of documents in which the query term w appears and N is the number of documents in the whole collection.

Another measure to predict query performance is called Query Scope (QS) [9]. This measure uses the size of the document set containing at least one of the query terms to

Table 3: Features used in the regression model for query Q

Features	
QC	Query Clarity
TC	Topical Clarity
tag-clarity	IPC-based Clarity
γ_1	Informative Amount in the Query
QS	Query Scope

infer the query performance. Formally, the query scope is defined as:

$$QS = -\log(n_Q/N) \tag{17}$$

where n_Q is the number of documents containing at least one of the query terms, and N is the number of documents in the whole collection.

These features are summarized in Table 3. Note that the length of generated queries are similar so there is no gain in considering this as a feature.

To learn a performance prediction model using these features we define the following regression problem.

$$\underset{\Phi}{argmin} \sum_{Q \in T} ||\Phi(F(Q)) - AP(Q)||^2 \tag{18}$$

where T is a set of training topics and F is a mapping from query to the feature space. F also defines a mapping from the respective rank list of the query to the feature space.

5.2 Evaluating the dependence between the predictors and AP

In this section, we will examine the correlations of the predictors with the query performance. We use AP as the focus measure indicating the query performance in our experiments. To investigate the effectiveness of the predictors, we check the Spearman rank correlation and linear regression because of their power in showing correlation between predictors and AP as suggested by previous studies [4, 9].

The linear regression assumes a linear distribution of the involved variables, which is not necessarily valid in our case. As the distribution of the involved variables is unknown, a non-parametric measure such as Spearman rank correlation which does not assume any particular structure for the relationship can find stronger relationships. However, Spearman rank correlation can not find relationships between the combinations of predictors and AP.

Table 4 summarizes the results of the linear correlations of the predictors (in isolation) with AP on the training data. We know that the relationship between predictors and AP may be nonlinear, but this allows us to compare the importance of the features by examining their coefficients. We also examine the importance of the features by examining the significance of their correlation with AP. Bold values denote statistically significant correlations with AP at the reported level of p-value using paired t-test.

In order to model the complex nonlinear relationships between combinations of predictor variables, we use Stochastic Gradient Boosting Tree (SGBT) [5]. This model produces an ensemble of weak prediction learners, i.e., decision trees. It builds additive regression models in a stage-wise manner

Table 4: Linear Regression and Spearman rank correlation coefficient of the query performance predictors with Average Precision

	LR		Spearman	
Features	r	p-value	rs	p-value
QC	**0.2180**	**0.05**	**0.3645**	**0.01**
TC	**0.2466**	**0.05**	**0.3170**	**0.01**
tag-clarity	0.0943	0.28	**0.1812**	**0.05**
γ_1	0.0491	0.61	**0.1100**	**0.05**
QS	**0.1956**	**0.05**	**0.2278**	**0.01**

and it generalizes them by allowing optimization of an arbitrary differentiable loss function. For the SGBT, we used the gbm2 package implemented in R[4]. SGBT can find a subcombination of features that may aid with the prediction of AP. With this model we can get a prediction of AP for any input. Notice that Φ in Equation 18 represents an additive model of multiple decision trees which is learnt by SGBT.

6. EXPERIMENTS

In this section, we present the results for an experimental evaluation of our proposed method of refining patent queries using concept importance predictors.

First, we describe our experimental setup and the three experimental settings used in our study. In the first setting, we compare different unigram query models built from the query patent and we show their retrieval effectiveness on CLEF-IP 2010 dataset. In the second setting, in order to find out whether we can find a clearer focus of the query patent, we expand the unigram query with extracted important key concepts (e.g., bigrams or phrases). We determine the optimal parameter settings for each query model using training data and we compare the effectiveness of expansion using noun phrases with the baseline unigram queries. In the third setting, in order to find out whether query performance predictors can indicate a successful application of phrases, we conduct an experiment where we estimate the effectiveness of using noun phrases based on the set of features proposed in Section 5. We then combine the result of the unigram query and the expanded query using the outcome of the prediction model. We show that the best performance is achieved by expansion using noun phrases in a query dependent manner.

6.1 Experimental Setup

The retrieval experiments described in this paper are implemented using Terrier[5]. We used CLEF-IP 2010 collection which consists of 2.6 million patent documents. The relevance assessment is provided for the topic set which are patent applications and have *title*, *abstract*, *description*, and *claims*. As mentioned earlier, using *description* text for query generation has been shown to achieve the best MAP in contrast to other patent fields [16]. Therefore, we use the *description* text for building the query model. Patent applications in the topic set are annotated with the metadata tags such as IPC classes. We worked with the english topic set which corresponds to 1348 topics. We note that we did

[4]http://cran.r-project.org/web/packages/gbm/
[5]http://terrier.org/

not use the citation information of the patent applications in our experiments.

During indexing and retrieval, both documents and queries are stemmed using the Porter stemmer. Stop-word removal is performed on both documents and queries using the standard Terrier stop-word list. In addition to that, we also removed all the formulas and numeric references. We used BM25 for retrieving and scoring the documents. This is because we observed that BM25 scores are slightly more effective in practice.

6.2 Experimental Evaluation

6.2.1 Unigram Query Models

In this section the performance comparison of different unigram query models explained in Section 3 is presented. The result of our baseline method, Q_{orig}, is comparable to the second best result of the CLEF-IP 2010 [15]. The second best participating group of CLEF-IP 2010 showed that their approach achieves a statistically indistinguishable performance compared to the best result in CLEF-IP 2010. This ensures our choice for a competitive baseline method. Therefore, Q_{orig} serves as our baseline. In our experiments, we set the smoothing parameter λ in Equation 1 to be 0.5. Table 5 reports a comparison of two query expansion models EX-RM and PRF against our baseline. The expanded query model EX-RM is constructed by building a relevance model from the sample documents with at least one conceptual tag in common with the query. The expanded query model PRF is formed based on Pseudo Relevance Feedback. We combined the original query with the expanded query, where the parameter μ controls the weight of the unigram query as shown in Equation 5. We used the training data for tuning this parameter and the optimal value for μ is set to 0.6. The result of Table 5 are obtained using 10 expansion terms extracted from the top 10 documents and the number of terms used for building the original query is set to 30. Results marked with † achieved statistically significant improvement over the baseline at p-value of 0.01 using t-test.

Table 5: Performance comparison of the unigram query models, the baseline run, relevance models using pseudo feedback documents and sample relevant documents

model	MAP	Recall	PRES
baseline	0.136	0.619	0.535
PRF	0.103	0.590	0.481
EX-RM	0.150 †	0.643	0.553

As follows from Table 5, the PRF method is not able to select the best terms for query generation and all three reported performance measurements decrease compared to the baseline. This is due to the poor quality of search results. However, the relevance model using the sample documents, EX-RM, significantly outperforms the baseline run. This suggests that using the sample documents was beneficial for building the expanded query model and EX-RM on average achieved 13% improvement over the baseline in terms of MAP.

We explore the sensitivity of each of the unigram query models, baseline run, EX-RM and PRF, to the number of query terms that need to be taken into account. We also look into the number of feedback documents that need to be taken into account for both of the expanded unigram query models EX-RM and PRF. Figure 1 presents the MAP of our techniques, for varying values of number of feedback terms, and number of feedback documents. We can see that the number of terms is not highly influential and any value higher than 30 produces the same results. However, the system is more sensitive to the number of feedback documents. It can be seen that values higher than 10 hurts the performance.

6.2.2 Combining Unigram and Phrase Query

We wish to examine the quality of the phrases obtained by the two different techniques explained in Section 4 in the task of prior art search. Our goal is to utilize such phrases to identify relevant documents to the query patent. We first combine the unigram query model from the query patent with the top-k concepts selected by the two scoring methods: a) TF/IDF scoring method, denoted by QM-NP1; b) mutual information-based scoring method, denoted by QM-NP2. We further examine expanded queries in which we select the top-k concepts from the pseudo feedback documents using the two scoring methods, denoted by PRF-NP1 and PRF-NP2. Finally, in order to use the evidence from the relevance set (documents with similar conceptual tags), we selected the top-k scoring noun phrases from the relevance set using the two scoring methods and we refer to this as EX-RM-NP1 and EX-RM-NP2.

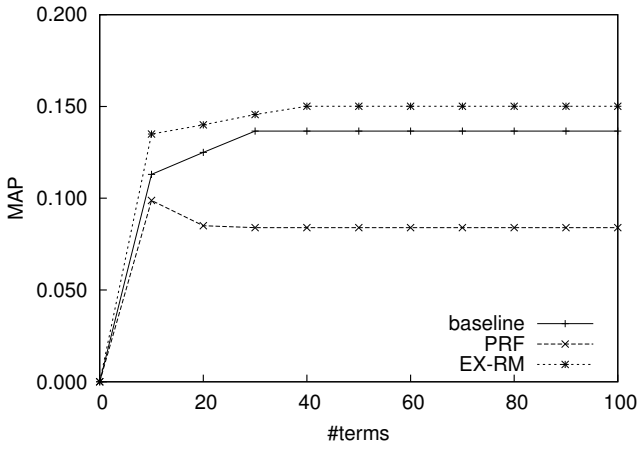
The retrieval results of various combinations of unigram queries with phrases are reported in Table 6. Results marked with † are significantly better than the baseline and ‡ represents the significant improvement achieved by EX-RM-NP2 against EX-RM-NP1.

Table 6: Performance of the expanded query models using phrases

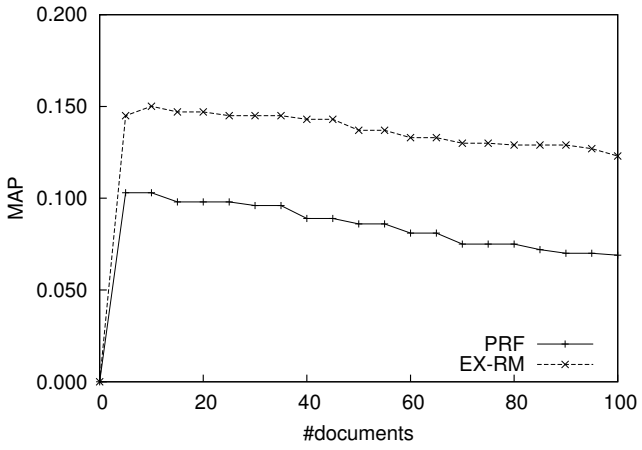
model	MAP	Recall	PRES
baseline	0.136	0.619	0.535
QM-NP1	0.131	0.600	0.521
QM-NP2	0.138	0.621	0.539
PRF-NP1	0.115	0.592	0.494
PRF-NP2	0.112	0.603	0.493
EX-RM-NP1	0.149 †	0.646 †	0.552
EX-RM-NP2	0.156 †‡	0.650 †	0.567

Our experiments indicate that expansion based on phrases extracted by the mutual information-based scoring technique most of the time outperforms TF/IDF based scoring. This suggests that using co-occurrence information is more helpful in identifying key concepts of a query patent compared to using frequency information alone.

As follows from Table 6, extracting concepts from the query patent, as done for QM-NP1 and QM-NP2, does not improve the results over the unigram baseline. As we expected, the PRF based expansion decreases the result in terms of MAP, recall, and PRES. It is clear that both relevance models built using similar conceptual tags, EX-RM-



a) Number of feedback terms (MAP)



b) Number of feedback documents (MAP)

Figure 1: Sensitivity of unigram query models against (a) the number of terms and (b) the number of feedback documents used for query model construction

NP1 and EX-RM-NP2, outperform our unigram baseline significantly. This result demonstrates a positive effect of expansion using both scoring methods. In both cases these improvements hold for MAP, recall and PRES.

A very interesting conclusion which can be made by comparing the results of Table 5 and Table 6 is that despite the significant improvement of EX-RM-NP1 and EX-RM-NP2 over the baseline, the improvement over EX-RM is not significant. We performed an analysis on the query set and we found that almost 600 queries out of 1348 queries were hurt by the expansion using phrases compared to using unigrams. We therefore decided to estimate an upper bound of performance by combining these two approaches in a query dependent manner. As we already saw in Section 5, we found that by using the true effectiveness of the noun phrase queries we can achieve an increase in performance of 20% in terms of MAP. In the next section, we show how we can estimate the importance of a noun phrase query in order to decide whether to expand a given query using noun phrases or not.

In our experiments we considered proximity matches rather than exact phrases. This is due to the fact that using proximity matches gave us consistent gain in retrieval effectiveness in comparison to using exact phrases. We use a window of size 8, as suggested by previous work on proximity matching [17]. We perform a sweep (grid search) on μ to determine the optimal mixture of the original query and the expanded query according to Equation 7. The optimal value is set to 0.6 for all expansion methods.

We selected top-10 phrases and we added them to our unigram queries. We studied the sensitivity of each approach against the number of feedback documents and the number of feedback phrases that need to be taken into account. The results are shown in Figure 2. An important observation to be made from Figure 1 and Figure 2 is that using 10 pseudo relevant documents and around 40 feedback terms resulted in the best performance for all expansion methods.

6.2.3 Selective Query Expansion Using Key Concepts

So far we built the unigram and expanded query models using three different sources. 1) the query patent; 2) the pseudo relevant documents retrieved from PRF; 3) the relevance set which is composed of documents with similar conceptual tags to the query.

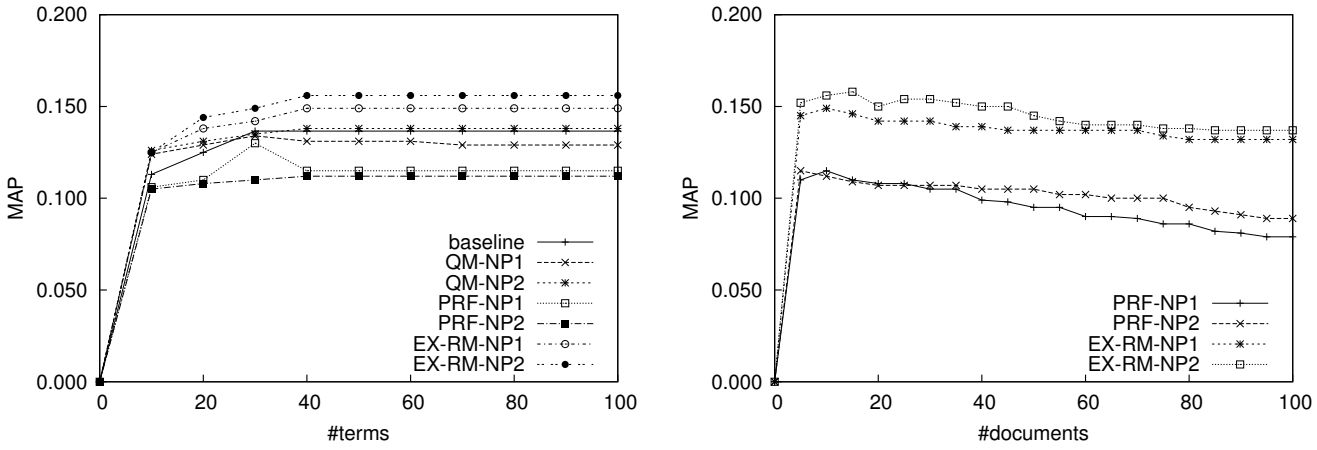
In this section, our goal is to predict whether query expansion using phrases is effective. We first predict the AP of each query in both ranked list of the expanded and unexpanded query using the features described in Section 5. We then calculate the change in AP after expansion based on the predicted values. A positive change in AP after expansion indicates an effective expansion. In the experiments, we considered a change bigger than 10% to be an effective expansion. We use this prediction value to decide which of the two competing methods will offer the best result for a given query.

We used a five-fold cross validation for our experiments. We divided the query topics into five equal parts. We trained the estimator using four out of five parts, and we applied the training model to estimate the AP of the remaining queries. We repeated the same test process on each of the five parts and we report the results on average over all five parts. The same procedure was performed for the expanded and unexpanded lists.

Table 7: Retrieval results on CLEF-IP 2010 using selective query expansion

model	MAP	Recall	PRES
baseline	0.136	0.619	0.535
QM-NP2	0.138	0.621	0.539
SQE_Q	0.152 $*$ $\star$	0.617	0.543
PRF	0.103	0.590	0.481
PRF-NP2	0.112	0.603	0.493
SQE_{PRF}	0.122 $\uparrow$ $\Uparrow$	0.609	0.509
EX-RM	0.150	0.643	0.553
EX-RM-NP2	0.156	0.650	0.567
SQE_{EX-RM}	0.168 † ‡	0.668	0.580

Table 7 shows the result of our method for selective query



a) Number of feedback terms (MAP)

b) Number of feedback documents (MAP)

Figure 2: Sensitivity of the expanded query models using noun phrases against (a) the number of terms and (b) the number of feedback documents used for expanded query model construction

expansion (SQE). The $*$ and $\star$ symbols indicate that the achieved improvement of SQE_Q over the expanded and unexpanded lists, QM-NP2 and baseline, is statistically significant at $p<0.01$. The $\uparrow$ and $\Uparrow$ symbols indicate that the achieved improvement of SQE_{PRF} over the expanded and unexpanded lists, PRF-NP2 and PRF, is statistically significant at $p<0.01$. The $\dagger$ and $\ddagger$ symbols indicate that the achieved improvement of SQE_{EX-RM} over the expanded and unexpanded lists, EX-RM-NP2 and EX-RM, is statistically significant at $p<0.01$.

As follows from Table 7, for all the three settings of our experiments, selective query expansion achieved statistically significant improvement in terms of MAP over automatic query expansion (using expansion on all queries). This indicates that the chosen features were able to accurately predict the AP for the expanded and unexpanded lists of each query. This also suggests that the predicted change in AP was a good indicator of an effective expansion. A per query analysis showed that the result of SQE method was able to detect more than half of the queries which performed well using the expansion and therefore SQE was able to effectively improve the retrieval effectiveness of those queries. The SQE method did not achieve the upper bound performance shown in Table 2, which is due to the error made by the prediction model. Despite the achieved increase in terms of MAP, there is still room for improvement which requires the choice of better features for almost all methods.

We calculated the influential features from the learnt SGBT model [5]. Query clarity, Topical clarity and IPC-based clarity are the most influential features.

7. CONCLUSION AND FUTURE WORK

In this work, we presented several versions of the unigram and the noun phrase queries for prior art search. By evaluating these query models we found that more advance IR techniques will increase performance of specific queries but the aggregated result may degrade against the baseline. To achieve consistent improvement in all queries we worked in a selective query expansion framework. The main contribution of this paper is devising a method for predicting whether expansion using noun phrases will improve the retrieval effectiveness of a query.

We experimentally determined the upper bound of what can be achieved by looking into the true effectiveness using a noun phrase query. We used a few often used features for predicting AP and we proposed some features using patent-specific characteristics. Our selective query expansion method using noun phrases obtained a statistically significant improvement over the expanded and unexpanded queries. Better features still need to be extracted which can capture the quality of the results better.

We experimented with two different scoring methods for selecting noun phrases. The scoring based on mutual information achieved better results over TF/IDF scoring. Another interesting conclusion can be made by comparing the two relevance models which were used in this study. The first relevance model was built based on PRF and the second was built by employing documents with similar conceptual tags to the query. The result of PRF method failed over initial retrieval because of the poor results, but the relevance model built based on conceptual tags outperformed the baseline.

Our work can be extended in future in several ways. First, we can improve the noun phrase extraction by using external resources related to the patent genre. Second, we can define better features so that average precision can be more accurately predicted. Third, instead of relying on the original score of the relevant documents to determine which documents should contribute in building a relevance model in a PRF scenario, we can estimate the true effectiveness of each document using a set of features employing patent-specific characteristics.

8. ACKNOWLEDGMENTS

The work of Parvaz Mahdabi was funded by the Information Retrieval Facility, through the research project "Interactive Patent Search (IPS)".

Fabio Crestani's work was partially supported by a Cátedra de Excelencia appointment at the Universidad Carlos III of Madrid (Spain), program 2011-2012, sponsored by the Banco de Santander.

9. REFERENCES

[1] G. Amati, C. Carpineto, and G. Romano. Query difficulty, robustness and selective application of query expansion. In *ECIR*, pages 127–137, 2004.

[2] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Predicting query performance. In *SIGIR*, pages 299–306, 2002.

[3] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. A framework for selective query expansion. In *CIKM*, pages 236 – 237, 2004.

[4] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In *SIGIR*, pages 18–24, 2004.

[5] J. H. Friedman. Stochastic gradient boosting. In *Computational Statistics and Data Analysis*, volume 38, pages 367–378, 1999.

[6] A. Fujii. Enhancing patent retrieval by citation analysis. In *SIGIR*, pages 793–794, 2007.

[7] D. Ganguly, J. Leveling, W. Magdy, and G. J. F. Jones. Patent query reduction based on pseudo-relevant documents. In *CIKM*, pages 1953–1956, 2011.

[8] C. Hauff, L. Azzopardi, D. Hiemstra, and F. de Jong. Query performance prediction: Evaluation contrasted with effectiveness. In *ECIR*, pages 204–216, 2010.

[9] B. He and I. Ounis. Inferring query performance using pre-retrieval predictors. In *SPIRE*, pages 43–54, 2004.

[10] B. He and I. Ounis. Combining fields for query expansion and adaptive query expansion. *Information Processing Management*, 43(5):1294–1307, 2007.

[11] H. Itoh, H. Mano, and Y. Ogawa. Term distillation in patent retrieval. In *Proceedings of the ACL-2003 Workshop on Patent corpus processing*, volume 20, pages 41–45, 2003.

[12] M. Iwayama, A. Fujii, N. Kando, and A. Takano. Overview of the third NTCIR workshop. In *Proceedings of the ACL-2003 Workshop on Patent corpus processing*, pages 24–32, 2003.

[13] K. Konishi. Query terms extraction from patent document for invalidity search. In *NTCIR-5*, 2005.

[14] W. Magdy and G. J. F. Jones. PRES: A score metric for evaluating recall-oriented information retrieval applications. In *SIGIR*, pages 611–618, 2010.

[15] W. Magdy, P. Lopez, and G. J. F. Jones. Simple vs. sophisticated approaches for patent prior-art search. In *ECIR*, pages 725–728, 2010.

[16] P. Mahdabi, M. Keikha, S. Gerani, M. Landoni, and F. Crestani. Building queries for prior-art search. In *Proceedings of Information Retrieval Facility Conference (IRFC)*, pages 3–15, 2011.

[17] J. Peng, C. Macdonald, B. He, V. Plachouras, and I. Ounis. Incorporating term dependency in the DFR framework. In *SIGIR*, pages 843–844, 2007.

[18] F. Piroi and J. Tait. CLEF-IP 2010: Retrieval experiments in the intellectual property domain. In *CLEF-2010 (Notebook Papers/LABs/Workshops)*, 2010.

[19] K. Toutanova, D. Klein, C. Manning, , and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of HLT-NAACL*, pages 252–259, 2003.

[20] M. Winaver, O. Kurland, and C. Domshlak. Towards robust query expansion: Model selection in the language modeling framework. In *SIGIR*, pages 729–730, 2007.

[21] X. Xue and W. B. Croft. Transforming patents into prior-art queries. In *SIGIR*, pages 808–809, 2009.

[22] Y. Yang, N. Bansal, W. Dakka, P. Ipeirotis, N. Koudas, and D. Papadias. Query by document. In *WSDM*, pages 34–43, 2009.

[23] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow. Learning to estimate query difficulty. In *SIGIR*, pages 512–519, 2005.