Identification of Low/High Retrievable Patents using Content-Based Features

Shariq Bashir
Department of Software Technology and Interactive Systems
Vienna University of Technology
Favoritenstrasse 9-11/118, A-1040, Vienna, Austria
bashir@ifs.tuwien.ac.at

Andreas Rauber
Department of Software Technology and Interactive Systems
Vienna University of Technology
Favoritenstrasse 9-11/118, A-1040, Vienna, Austria
rauber@ifs.tuwien.ac.at

ABSTRACT

Document retrievability is a measurement used in information retrieval for identifying the bias of retrieval systems. In order to measure system bias for a specific document collection, an exhaustive set of queries is processed, measuring the frequency with which each document is retrieved. For better understanding and handling system bias, we need to understand the characteristics of documents that influence retrievability, and ideally be able to identify documents with high and low retrievability in advance. For this purpose, we identify a number of content-based features, which can be used effectively to classify a corpus into documents with low and high retrievability w.r.t a specific retrieval system. Our experiments on patent collections show that these features can achieve more than 80% classification accuracy on different systems, and hint at the need to combine different retrieval systems for optimizing recall.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models

General Terms

Measurement, Performance, Experimentation

1. INTRODUCTION

The way how information is described and disclosed in patent applications is quite different from other information retrieval (IR) domains [5]. The vocabulary of patent applications is quite diverse, which leads to an extremely large dictionary. Many vague or general terms are often used in order to avoid narrowing the scope of the invention. Combinations of general terms often have a special meaning that also has to be captured. Patent applications further contain many acronyms and new terminology. These characteristics of patent documents create nontrivial consequences on the findability of patents in retrieval systems, making some patents easier to find within top rank results of queries, while others are never listed within the top-c ranked documents for any reasonable query, leaving them virtually inconsistent in a document collection [1, 2].

High findability *(also called retrievability)* of each and every document in the collection within top rank results of queries is considered an important factor in recall-oriented application domains, like patents or legal documents retrieval [3, 11]. In these domains, unlike web retrieval, it is essential for users (patent examiners, attorneys, or researchers), to have access to all relevant documents. Non-accessibility of single relevant patent document may, for instance, have non-trivial consequences on the outcome of approving patent applications.

Each retrieval system ranks documents differently for a given query. Thus individual documents have different retrievability scores in different retrieval systems [1]. In order to precisely understand the bias of different systems, we analyze different characteristics of patents that make individual patents low or high retrievable in a particular retrieval system. This allows users to determine, which system is useful for which type of retrieval tasks. For instance, some systems are useful for finding all those patents, which frequently use *rare terms* for narrowing down the scope of their invention, although these systems show large bias toward some patents. Similarly, some systems make *strong clusters of patents* more retrievable than *weaker clusters*. Furthermore, we want to identify patents with low and high retrievability in advance (using classification systems). This will allow us to treat these two types of patents specifically to enhance retrieval performance.

Automatic classification of a document corpus into low and high retrievable documents has several fruitful research directions. For example, it may assist in devising strategies towards combining multiple ranking lists of different systems, or with the help of automatic classification, patent examiners can give special emphasis in analyzing the contents of those patents which show strong correlation with low retrievability on most of systems. We evaluate these aspects for a range of different retrieval systems, specifically with standard TFIDF, OKAPI BM25 [13], Jelined-Mercer smoothing language model [18], Dirichlet (Bayesian) smoothing [16], Absolute discounting smoothing [12], and Two-Stage smoothing [16].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PaIR'09, November 6, 2009, Hong Kong, China.
Copyright 2009 ACM 978-1-60558-809-4/09/11 ...$10.00.

The remainder of this paper is organized as follows. In Section 2, we provide an overview about related work on retrievability analysis. Section 3 introduces the retrievability measurement framework, the strategies applied for controlled query generation, as well as the content-based features extracted for automatically identifying low/high retrievability patents using a classification approach. Section 4 describes the experimental setup, detailing the data set used as well as the retrieval systems evaluated. A detailed evaluation of the discriminative power of the features is provided in Section 5, together with classifier performance analysis. Some conclusions as well as an outlook on utilizing the proposed classification approach for improving retrieval performance is provided in Section 6.

2. RELATED WORK

The evaluation of retrieval systems has always received much attention in the IR research community. Conventionally, retrieval systems are evaluated based on *A*verage Precision and *R*ecall, *Q*-measure, *N*ormalized Discounted Cumulative Gain, *R*ank-Based Precision, *B*inary Preference (bref) and other metrics [14]. However, these metrics do not evaluate, what we can find and cannot find globally throughout the whole collection. Still, for some specific retrieval applications like *patents* or *legal* domains, recall is considered more important than precision.

In addition to using traditional IR metrics for evaluation, Azzopardi et al. [1] measure retrieval systems on the basis of retrievability scores of individual documents, measuring how likely a document can be found at all with a specific system. In their framework, a system is called better than others, if its retrievability inequality between documents is less than other systems. For analyzing retrievability inequality among different systems, they use Lorenz Curve and Gini Coefficient. Their experiments with AQUAINT and .GOV datasets yield that with a TREC-style evaluation, a proportion of the least retrievability documents *(sometimes more than 80% documents)* can be removed without significantly degrading performance. This is because the retrieval systems are unlikely to ever retrieve these documents due to the bias they exhibits over the collection of documents.

Bashir et al. in [2] further analyze retrievability of documents specifically with respect to relevant and irrelevant queries to identify, whether highly retrievable documents are really highly retrievable, or whether they are simply more accessible from many irrelevant queries rather than from relevant queries. Evaluation is based on a model of controlled query generation as described further below. Experiments revealed, that 90% of patents which are highly retrievable across all types of queries, are not highly retrievable on their relevant query sets. Furthermore, they analyzed that retrievability remained constant across all documents when considering only relevant queries, as opposed to the rather large differences encountered when considering all potential queries.

Custis et al. [3], evaluate query expansion methods for legal domain applications retrieval on the basis of query document term mismatch. For this purpose, they systematically introduce query document term mismatch into a corpus in a controlled manner and then measure the performance of IR systems as the degree of term mismatch changes. Jordan et al. [8] consider controlled query generation method useful for evaluating the impact of retrieval systems performance rather than relying on only using predefined set of queries of cranfield evaluation paradigm. The main purpose of their study was to expose the performance of different algorithms, how they react to queries of varying length and term quality (in case of noisy terms). Their approach also serves as the basis of the evaluations presented in Section 4.

3. RETRIEVABILITY ANALYSIS

To evaluate the reasons for the retrievability performance of specific systems and documents (patents), we first identify low/high retrievable patents using standard retrievability measurement. To this end, we use controlled query generation to achieve comparable and repeatable results. We then extract features from the patents for training a classifier to detect potentially low and high retrievable patents in a corpus. The following sections describe each of these steps in detail.

3.1 Retrievability Measurment

For identifying patents retrievability in a retrieval system, we use the retrievability measurement framework proposed by Azzopardi et al. [1]. Given a retrieval system RS with a collection of documents D, the concept of retrievability is to measure how much each and every document $d \in D$ is retrievable in top-c rank results of all queries, if RS is presented with a large set of queries $q \in Q$. Defined in this way, the retrievability of a document is essentially a cumulative score that is proportional to the number of times the document can be retrieved within that cut-off c over the set Q. A retrieval system is called best retrievable, if each document $d \in D$ has nearly the same retrievability score, i.e. is equally likely to be found. More formally, retrievability $r(d)$ of $d \in D$ can be defined as follows.

$$r(d) = \sum_{q \in Q} f(k_{dq}, c) \tag{1}$$

Here, $f(k_{dq}, c)$ is a generalized utility/cost function, where k_{dq} is the rank of d in the result set of query q, c denotes the maximum rank that a user is willing to proceed down the ranked list. The function $f(k_{dq}, c)$ returns a value of 1 if $k_{dq} \leq c$, and 0 otherwise.

Retrievability inequality can further be analyzed using the *Lorenz Curve*. Documents are sorted according to their retrievability score in ascending order, plotting a cumulative score distribution. If the retrievability of documents is distributed equally, then the Lorenz Curve will be linear. The more skewed the curve, the greater the amount of inequality or bias within the retrieval system.

The **Gini coefficient** G is used to summarize the amount of bias in the Lorenz Curve, and is computed as follows.

$$G = \frac{\sum_{i=1}^{n} (2 \cdot i - n - 1) \cdot r(d_i)}{n \sum_{j=1}^{n} r(d_j)} \tag{2}$$

where $n = |D|$ is the number of documents in the collection. If $G = 0$, then no bias is present because all documents are equally retrievable. If $G = 1$, then only one document is retrievable and all other documents have $r(d) = 0$. By comparing the Gini coefficients of different retrieval methods, we can analyze the retrievability bias imposed by the underlying retrieval system on the given document collection.

3.2 Controlled Query Generation

Clearly, it is impractical to calculate the absolute retrievability scores because the set Q *(all queries)* would be extremely large and require a significant amount of computation time as each query would have to be issued against the index for any given retrieval system. In order to perform the measurements in a practical way a subset of all possible queries is commonly used that is sufficiently large and contains relatively probable queries. For generating reproducible and theoretically consistent queries, we use the method of controlled query generation (CQG) [8]. CQG is an efficient mechanism for evaluating the impact of specific query characteristics on retrieval systems performance using automatically generated queries. We use two different variations, based on how patent experts generate queries for searching their relevant information in patent corpus.

Query Generation combining Frequent Terms *(QG-FT)*: In this CQG approach, we try to reflect the way how patent examiners generate query sets in *patent invalidity search* problems [9]. In invalidity search, the examiners have to find all existing patent specifications that describe the same invention for collecting claims to make a particular patent invalid. In this search process, the examiners extract relevant query terms from a new patent application, particularly from the *Claim* section for creating query sets [7, 10]. We first extract all those frequent terms that are present in the Claim section of each patent document and have a support *(frequency)* greater than a certain threshold. Then, QG-FT combines the single frequent terms of each individual patent document into `two`, `three` and `four` terms combinations.

Query Generation with Document Relatedness *(QG-DR)*: QG-FT creats queries based on those frequent single terms which are present in only one single document. However, patent applications may contain many vague or technical terms to hide the relation to other documents from patent examiners [5]. In such situations patent examiners extract relevant terms from other patent documents that are similar to the new patent application using the concept of document relatedness [4, 6].

With QG-DR, we adopt this strategy. We first define a set of related documents for each document in the corpus based on k-nearest neighbor with cosine similarity. QG-DR then generates a set of queries based on each of these sets of related documents. Using the relative entropy of individual terms in language modeling [8], the most discriminating terms are identified for constructing `two`, `three` and `four` terms combinations queries. The steps for QG-DR are as follows.

1. Construct a related document set R for each patent document in the collection using k-nearest neighbor approach.
2. Defined language model for each related document set, and extract most discriminative terms for automatically creating queries.
3. For each related document set R, sort the terms in the vocabulary using language model of Equation 3. where $P(t|R)$ represents the probability of term t in set R and $P(t|D)$ represents the probability of term t in the whole collection. After this, identify the top-n terms that contribute most to the relative entropy
4. Combine the single identified terms into `two`, `three` and `four` terms combinations for constructing longer queries.

$$score(t) = P(t|R)log\frac{P(t|R)}{P(t|D)} \tag{3}$$

3.3 Content-based Feature Extraction

Once, we have identified patents with low and high retrievability, we need to extract features from the patents that may allow a-priori identification via a classification system. We compute a number of statistical and information-theoretic features from these patents. The feature set consists of 6 diverse features, whereas features RTR *(Rare Terms Ratio)*, ATP^{rd} *(Average Terms Probabilities in Related Patents)* and ATP *(Average Terms Probabilities in Whole Set)* are further computed on `one`, `two` and `three` term combinations. For all other features we consider only `one` gram histograms. This brings the total dimensions of the feature set to 13[1].

3.3.1 Rare Terms Ratio *(RTR)*:

In this feature we analyze the retrievability in different retrieval systems on the basis of presence of rare terms ratio in individual patents. We consider a term "*rare*", if it's f_t (the number of all patents that contain term t) is $\leq$ 200. If a patent contains a large ratio of rare terms then this indicates that the patent either describes a very new invention or does not disclose its invention properly. This feature is useful for analyzing, in how far different retrieval systems can discover such type of patents. Formally this feature can be calculated as

$$RTR = \frac{|T^r|}{|T^a|}, \{where\ T^r \subseteq T^a,\ f_t \leq 200,\ t \in T^r\} \tag{4}$$

Where T^a represents the set of all unique terms in patent, while T^r represents the set of only those terms which have $f_t \leq$ 200. RTR of `2-grams` and `3-grams` are calculated analogously.

3.3.2 Average Terms Frequencies *(ATF)*:

This feature analyzes the effect of term frequencies on patents retrievability. We consider both rare terms and all terms in patents. This feature is suitable in analyzing; whether a system gives preference to patents with larger frequencies or smaller frequencies. Experiments show that this feature is independent of patent length. Histograms *(rare and all terms)* of this feature can be calculated as

$$ATF^a = \frac{\sum_{t\in}^{T^a} \frac{f_{(d,t)}}{|d|}}{|T^a|} \tag{5}$$

$$ATF^r = \frac{\sum_{t\in}^{T^r} \frac{f_{(d,t)}}{|d|}}{|T^r|} \tag{6}$$

Where ATF^a represents the average terms frequencies with all terms, and ATF^r represents the average terms frequencies with rare terms. $f_{(d,t)}$ is the frequency of term t in patent d, and $|d|$ represents the length of patent. In Equations 5 and 6, $f_{(d,t)}$ is divided with $|d|$ for removing the effect of varying patents length.

[1]Feature ATF is further computed with rare terms and all terms.

3.3.3 Frequent Terms Count (FTC):

In ATF, we consider all terms of individual patents for analysis. FTC counts the number of terms in patents that have a document frequency ($f_{(d,t)}$) larger than a certain threshold. We use $f_{(d,t)} \geq 6$ for this purpose.

3.3.4 Average Terms Probabilities in Related Patents (ATP^{rd}):

With this feature, we analyze the retrievability effect of individual patents on the basis of average terms probabilities ($P(t|D^r)$) in their top-35 most similar patents. Higher retrievability score for larger values of this feature indicates that the retrieval system makes "*stronger clusters*" more retrievable than "*weaker clusters*". This feature can be calculated as follows:

$$ATP^{rd} = \frac{\sum_{t\in}^{T^a} P(t|D^r)}{|T^a|} \tag{7}$$

Where D^r represents the set of top-35 most similar patents of $d \in D$. We use k-nearest neighbor with cosine similarity for finding similar patents. $P(t|D^r)$ represents the probability of term t in set D^r. ATP^{rd} of `2-grams` and `3-grams` are calculated accordingly.

3.3.5 Average Terms Probabilities in Whole Set (ATP):

ATP^{rd} analyzes the effect of patent retrievability on the basis of average terms probabilities in only top-35 most similar patents, while in this feature we consider the whole collection. Higher findability for larger values of this feature indicates that a retrieval system is useful for finding all those patents which frequently use general terms. The histogram of this feature can be calculated as.

$$ATP = \frac{\sum_{t\in}^{T^a} P(t|D)}{|T^a|} \tag{8}$$

Where $P(t|D)$ represents the probability of term t in whole collection. ATP of `2-grams` and `3-grams` are calculated accordingly.

3.3.6 Patent Length (PL):

This feature analyzes the effect of patent length on retrievability. Higher retrievability with higher values of this feature indicates that a retrieval system makes longer patents more retrievable than shorter ones.

3.3.7 Other Featuers

Some other features that we considered interesting and tried during our experiments but could not find to be discriminative for identifying low/high retrievability patents, are: (a) Absolute and relative (w.r.t. patent length) number of individual terms in a patent, (b) average distance of patents in a k-nearest neighbor cluster of size 5, 10, and 30, (c) number of patents that are within a certain distance k of a given patent, (d) number of stop words in a patent, ratio of stop words to non-stop words, (e) ratio of terms in a patent that appear also in its claim section, (f) maximum term frequency ($f_{(d,t)}$) value in a patent's vector, and (g) minimum and maximum f_t values of the terms in a patent, i.e. does a patent consist predominantly of terms that appear in many other patent or in few other documents (similar to (f), but explicitly only on f_t).

Based on the features identified above, we train classifiers to identify potentially low/high retrievable patents in different systems. For the experiments reported further below, we select random sets of 800 low and 800 high retrievable documents for each retrieval system for classifiers training. We consider a patent of the class low retrievable *(based on the experiments of Section 4)*, if it has a retrievability score $r(d)$ <300, whereas patents with a $r(d) \geq 700$ are classified as high retrievability. We use C4.5 decision tree (J48) implement in `Waikato Environment for Knowledge Acquisition (WEKA)` [15] for training of classifiers. 1,600 random patents are then used for evaluating the classifiers.

4. EXPERIMENTS SETUP

4.1 Data Set

For our experiments, we use a collection of freely available patents from the US patent and trademark office, downloaded from (`http://www.uspto.gov/`). We collect all patents that are listed under `United State Patent Classification` (USPC) classes *422 (Chemical apparatus and process disinfecting, deodorizing, preserving, or sterilizing)*, and *423 (Chemistry of inorganic compounds)*. There are a total of 54,353 patents in our collection, with an average patent size of 3,317.41 words (without stop words removing). Since, our main interest of this paper is to precisely understand the cause behind low retrievability and automatic retrievability classification. Therefore, rather than using very large collection for experiments, we concentrate only on specific USPC classes. Due to lack of space, we could not show our experimental results with other USPC classes. However, the results are almost similar as report with USPC classes 422 and 423.

In controlled query generation (CQG) with both methods we consider only the `Claim section` of every document as this is the section that most professional patent searchers use as their basis for query formulation [4, 7, 10]. However, for retrieval we index the full text of all documents (`Title, Abstract, Claim, Description`). This reflects the default setting in a standard full text retrieval engine. Some basic statistical properties of the data collection used for the experiments are listed in Table 1, with average lengths being given in number of words without stopword removal. Before indexing, we remove stop words and stem the words. For indexing and querying we use *Apache LUCENE*[2] IR toolkit.

4.2 Retrieval Models

Two standard IR models and four different variations of language models with term smoothing are used for retrievability analysis. These are **TFIDF**, the OKAPI retrieval function **(BM25)**, Jelinek-Mercer language model **(JM)**, Dirichlet (Bayesian) language model **(DirS)**, Absolute Discounting language model **(AbsDis)**, and Two-Stage language model **(Two-Stage)**.

TFIDF & BM25:
`TFIDF` with cosine similarity and OKAPI `BM25` [13] as defacto standard retrieval systems are used as a baseline that the other retrieval models are compared to.

Jelinek-Mercer (JM):
Jelinek-Mercer smoothing language model [18] combines the relative frequency of a query term $w \in q$ in the document d with the relative frequency of the term in the collection

[2]http://lucene.apache.org/java/docs/

(D) as a whole. The maximum likelihood estimate is moved uniformly toward the collection model probability $P(w|D)$:

$$P(w|M_d) = (1-\lambda)\frac{f_{(d,w)}}{|d|} + \lambda P(w|D) \quad (9)$$

$f_{(d,w)}$ represents the frequency of term w in document d. The value of λ is normally suggested ($\lambda = 0.7$).

Dirichlet (Bayesian) Smoothing (DirS):
As longer documents allow us to estimate the language model more accurately, therefore Dirichlet smoothing [16] makes them to smooth less. If we use the multinomial distribution to represent a language model, the conjugate prior of this distribution is the Dirichlet distribution. This gives:

$$P(w|M_d) = \frac{f_{(d,w)} + \mu P(w|D)}{|d| + \mu} \quad (10)$$

As μ gets smaller, the contribution from the collection model becomes smaller also, and more emphasis is given to the relative term weighting. According to Zhai et al. [16], the optimal value of μ is around $2,000$.

Absolute Discounting (AbsDis):
In this smoothing model [12] all non-zero counts are discounted by subtracting a constant δ from the counts of each term. Using this principle, the probability mass acquired from the present terms is distributed over unseen events uniformly. Absolute discounting formulates as follows [18]:

$$P(w|M_d) = \frac{max(f_{(d,w)} - \delta, 0) + \delta|T^a|p(w|D)}{|d|} \quad (11)$$

Where $|T^a|$ is the number of unique terms in the document d, and $0 < \delta < 1$.

Two-Stage Smoothing (Two-Stage):
In this model [16, 17] , the system first smoothes the document language model using the Dirichlet prior. Then, the system mixes the document language model with a 'query background' model using Jelinek-Mercer smoothing. The smoothing function is therefore:

$$P(w|M_d) = (1-\lambda)\frac{f_{(d,w)} + \mu P(w|D)}{|d| + \mu} + \lambda P(w|D) \quad (12)$$

Where μ is the Dirichlet prior parameter and λ is the Jelinek-Mercer parameter. In our experimentation setting, we set the parameters $\mu = 2000$ and $\lambda = 0.7$ respectively.

4.3 Controlled Query Sets Generation

In QG-FT, we consider all the frequent single terms, which have a *minimum support*$\geq$3 in the `Claim section`. There are a total of 47,621 single frequent terms present in collection USPC classes 422, 423. For generating larger length queries for every patent, we expand the single frequent terms into `two`, `three` and `four` term combinations. For patents which contain a large number of single frequent terms, the different co-occurring term combinations of size `two`, `three` and `four` can become very large. Therefore, for generating a similar number of queries for every patent, we put an upper bound of 90 queries generated for every patent document. After removing duplicate queries, Table 2 shows the distribution of different queries sets.

In QG-DR mechanism, we construct the related document set for every patent in the collection considering 35 neighbors. After applying language modeling on related documents sets, we select the top 70 terms that contribute most to the relative entropy with the language model. `Two`, `three` and `four` term queries are constructed with the same approach and maximum number of queries per document boundary as above.

	USPC (422,423)
# Documents	54,353
# Unique Terms	229,788
Avg. Doc. Length	3,317.41
Avg. Title Length	9.54
Avg. Abstract Length	217.96
Avg Claim Length	1627.56
Avg. Descr. Length	2517.93

Table 1: Properties of patent collections

Query Size	CQG Appr.	#Queries	ARS
2 terms	*QG-FT*	548,390	335.9
	QG-DR	436,273	549.6
3 terms	*QG-FT*	753,682	303.5
	QG-DR	590,820	480.3
4 terms	*QG-FT*	855,215	225.6
	QG-DR	587,782	360.7
Total Queries	*QG-FT*	2,157,287	
	QG-DR	1,614,875	

Patent Collection with USPC Classes (422,423)
ARS - Average Retrievability Score/Query

Table 2: Queries set sizes and average retrievability scores (ARS)

5. RETRIEVABILITY ANALYSIS

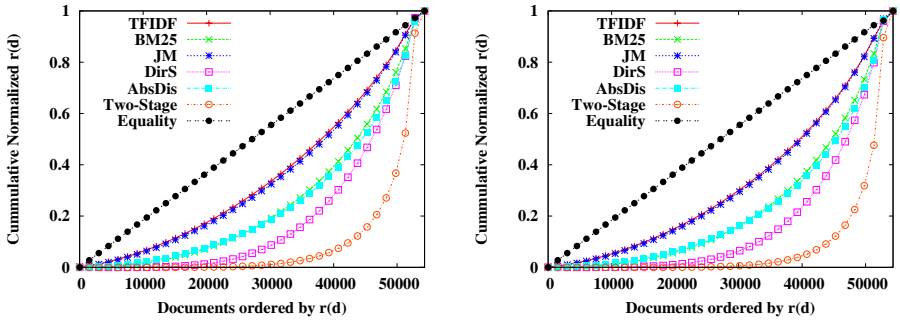
We show the retrievability inequality of different retrieval systems using Lorenz Curves with a `rank cut-off` factor of `c = 30` (Figure 1). Table 3 shows the retrievability inequality with other rank cut-off factors using Gini coefficient. As indicates by the Lorenz Curves of Figure 1 and Gini Coefficient of Table 3, TFIDF and JM show less retrievability inequality compared to other systems, since their Lorenz curves are less skewed and both systems have lower Gini coefficient values on almost all rank cut-off factors. Therefore, both systems make all patents more findable than other systems. On the other side, Two-Stage shows the highest retrievability inequality. In content-based feature analysis section (Section 5.1), we will show that on retrievability measurement Two-Stage and BM25 although do not seem to be good systems for patents retrieval. However both systems have still some unique features which make them useful for some special retrievability tasks. For instance, finding those patents which used large number of rare terms, or used terms with lower average term probabilities occurred in whole collection.

5.1 Feature Analysis

In Figure 2 and Figure 3, we first individually analyze the feature set of Section 3.3, w.r.t their discrimination power for identifying patents with low and high retrievability. (For better visual clarity, all the values of features shown in Figures 2 and 3 are smoothed across 35 patents.)

5.1.1 *Rare Terms Ratio (RTR):*

This feature is useful for analyzing, in how far different retrieval systems can discover those patents in top rank re-



with QG-FT **with QG-DR**

Figure 1: Lorenz Curve of retrievability scores for USPC (422,423), with *rank cut-off*=30. *Equality* refers to a optimal system which has no bias.

Ret. Sys.	CGQ Appr.	rank cut-off factors				
		30	40	50	70	90
TFIDF	*QG-FT*	0.33	0.34	0.35	0.34	0.32
	QG-DR	0.37	0.36	0.36	0.35	0.35
BM25	*QG-FT*	0.58	0.53	0.50	0.48	0.48
	QG-DR	0.62	0.57	0.53	0.50	0.49
JM	*QG-FT*	0.33	0.33	0.34	0.37	0.35
	QG-DR	0.36	0.35	0.35	0.35	0.34
DirS	*QG-FT*	0.59	0.54	0.50	0.43	0.40
	QG-DR	0.63	0.58	0.54	0.47	0.42
AbsDis	*QG-FT*	0.51	0.48	0.46	0.44	0.44
	QG-DR	0.54	0.51	0.49	0.46	0.44
Two-Stage	*QG-FT*	0.82	0.78	0.75	0.70	0.65
	QG-DR	0.85	0.82	0.79	0.72	0.70

Table 3: Gini coefficient values with different retrieval models on USPC (422, 423), for different *rank cut-off factors (c)*. As c increases, G steadily decreases indicating that lower bias is experienced when considering longer ranked lists.

sults that frequently use rare terms. On retrievability measurement (Figure 1), TFIDF and JM show less retrievability inequality than other systems. However, under this feature TFIDF and JM perform worst in finding patents with a high ratio of rare terms, i.e. RTR is a good indicator for identifying low findability documents for these two retrieval systems. The results of Two-Stage under this feature shows that it is useful for finding such patents - since in Two-Stage all those patents that contain a large ratio of rare terms have somehow higher retrievability score.

5.1.2 *Average Terms Frequency (ATF):*

Due to the effect of length normalization, BM25 on this feature makes all those patents more findable which have *smaller* average term frequencies. However on the other side same patents have *higher document length* than other patents (see Figures 2 and 3). Similar to BM25, JM and Two-Stage also make smaller average term frequencies patents more findable, while DirS and AbsDis make *larger* average term frequency patents more retrievable. The effect of TFIDF is linear with this feature values against different low and high retrievability scores.

5.1.3 *Frequent Terms Count (FTC):*

Contrary to ATF analysis, BM25, Two-Stage and JM on this feature make patents with higher values of this feature more retrievable. DirS and AbsDis make smaller values of this feature more retrievable. The performance of TFIDF is again linear with different values of this feature.

5.1.4 *Average Terms Probability in Related Patents (ATP^{rd}):*

TFIDF, JM, AbsDis, and DirS all make strong clusters more retrievable; however on BM25 and Two-Stage weaker clusters have high retrievability. This indicates that BM25 and Two-Stage are both suitable for finding all those patents which frequently use alternative terms as compared to those terms which appear frequently in their related patents.

5.1.5 *Average Terms Probability in Whole Set (ATP):*

TFIDF, DirS, AbsDis and JM with this feature make larger average terms probabilities patents more retrievable. Therefore, on these systems all those patents are absent from top rank results, which frequently used lower $P(t|D)$ terms. The performance of BM25 and Two-Stage is almost linear with different ATP values. This indicates, that these system are also suitable in finding those patents which frequently used new terminology in describing their invention or used terms with lower $P(t|D)$ values.

5.1.6 *Patent Length (PL):*

The results indicate that BM25 and Two-Stage make longer patents more findable, while AbsDis and DirS are suitable for finding short patents. However, if we interpret the results of PL and ATF together, then it becomes clear that BM25 and Two-Stage make only those longer length patents more retrievable which have smaller ATF values *(due to effect of length normalization these type of patents appear on top results of queries)*, whereas with AbsDis and DirS some shorter length patents with higher ATF can increase their appearance in top rank results of queries. Retrievability effect of TFIDF and JM is linear with the values of this feature.

5.2 Retrievability Classification

In order to automatically identify low/high retrievable patents for each system, we train J48 decision tree classifiers using 1,600 randomly selected low/high retrievable patents. Table 4 shows the classification accuracy of J48 with another testing set of 1,600 random patents. On most systems with both CQG approaches, J48 achieves greater than 80% classification accuracy. This validates the hypothesis, that content-based features can separate low/high retriev-

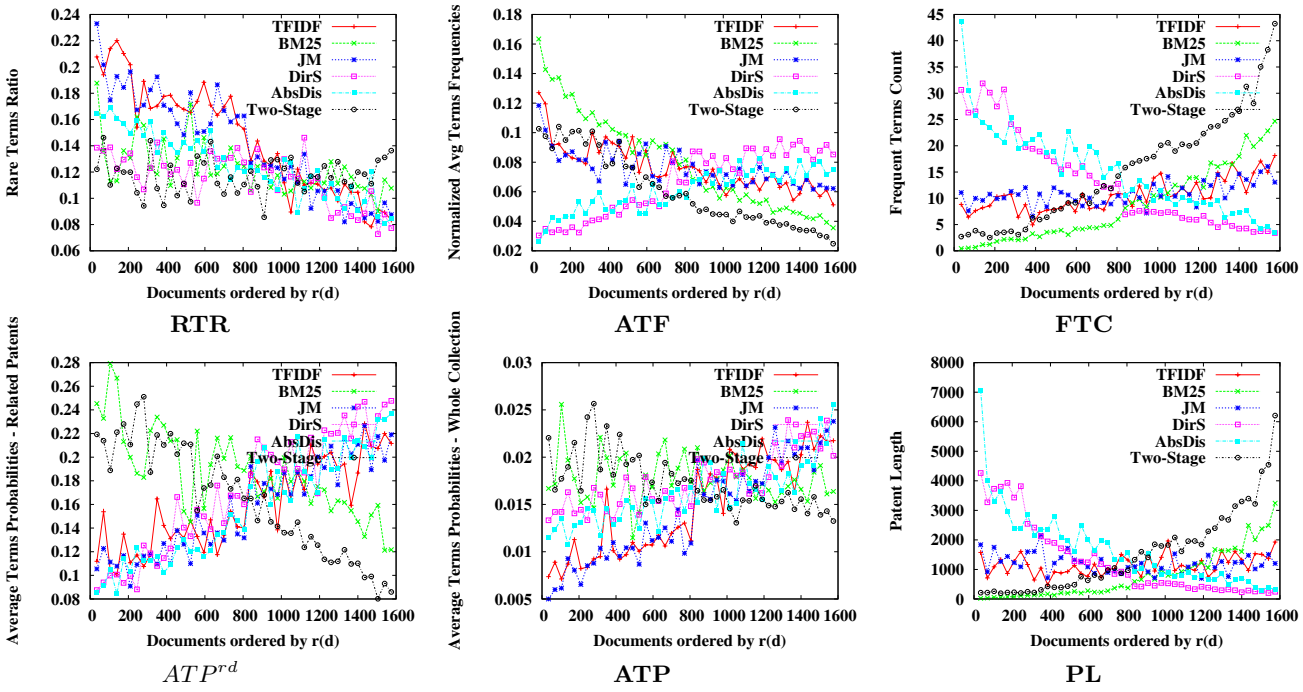


Figure 2: Effect of content-based features for USPC (422,423) with *rank cut-off*=30 using QG-FT.

Ret. Sys.	CGQ Appr.	rank cut-off factors				
		30	40	50	70	90
TFIDF	*QG-FT*	0.81	0.81	0.79	-	-
	QG-DR	0.82	0.86	0.86	0.89	-
BM25	*QG-FT*	0.92	0.95	0.94	0.91	0.91
	QG-DR	0.91	0.92	0.92	0.93	0.92
JM	*QG-FT*	0.85	0.88	0.89	0.90	-
	QG-DR	0.82	0.84	0.87	0.87	-
DirS	*QG-FT*	0.88	0.90	0.88	0.86	0.87
	QG-DR	0.87	0.87	0.84	0.83	0.86
AbsDis	*QG-FT*	0.77	0.79	0.80	0.86	0.86
	QG-DR	0.77	0.79	0.80	0.79	0.84
Two-Stage	*QG-FT*	0.94	0.94	0.95	0.95	0.95
	QG-DR	0.94	0.95	0.94	0.94	0.95

Table 4: Classification accuracy of J48 with different retrieval systems on USPC (422, 423). '-' indicates no accuracy is achieved - since all patents become high retrievable.

able patents with reasonable accuracy without doing extensive retrievability measurements. On high rank cut-off factors, particular with $c \geq 70$, we could not evaluate reasonable classification accuracy with TFIDF and JM, because on high rank cut-off factors both retrieval systems make almost all patents highly retrievable.

.

6. CONCLUSIONS

Document Retrievability is a measurement used in IR for identifying the bias of retrieval systems. Retrievability of documents is commonly analyzed using a *single retrievability curve*, which is not sufficient for analyze the complex aspects behind the low retrievability, such as why some documents show low retrievability in one system while they are highly retrievable in other systems. In general, it is assumed that *document length* is the main factor behind *low retrievability*. However, our experiments show that, there are also a number of other factors which make documents low retrievable in particular system. Our analysis with *content-based features* reveal that, despite to low retrievability, some systems are still valuable for specific retrieval tasks of patents. For instance, Two-Stage Smoothing which has large retrievability inequality as compared to other systems is useful for finding all those patents which frequently used rare terms or terms with lower average term probabilities occurred in whole collection. Finally, we can show that patents with low or high retrievability under a given retrieval system can be identified automatically using content based features. Our experiments on patent collections show that, we can achieve more than 80% classification accuracy, for identifying patents with low retrievability. This offers the possibility to address this sub-collection in a more focused manner to optimize overall retrieval performance, or to strategically combine different retrieval systems.

7. ACKNOWLEDGMENTS

We thank three anonymous reviewers for providing helpful comments on this work. The first author's PhD scholarship is supported by Higher Education Commission, Pakistan under Foreign Scholarships program.

8. REFERENCES

[1] L. Azzopardi, V. Vinay. Retrievability: an evaluation measure for higher order information access tasks. *In Proc. of CIKM '08*, pages 561–570, October 26-30, 2008, Napa Valley, California, USA.

[2] S. Bashir, A. Rauber. Analyzing document retrievability in patent retrieval settings. *In Proc. of*

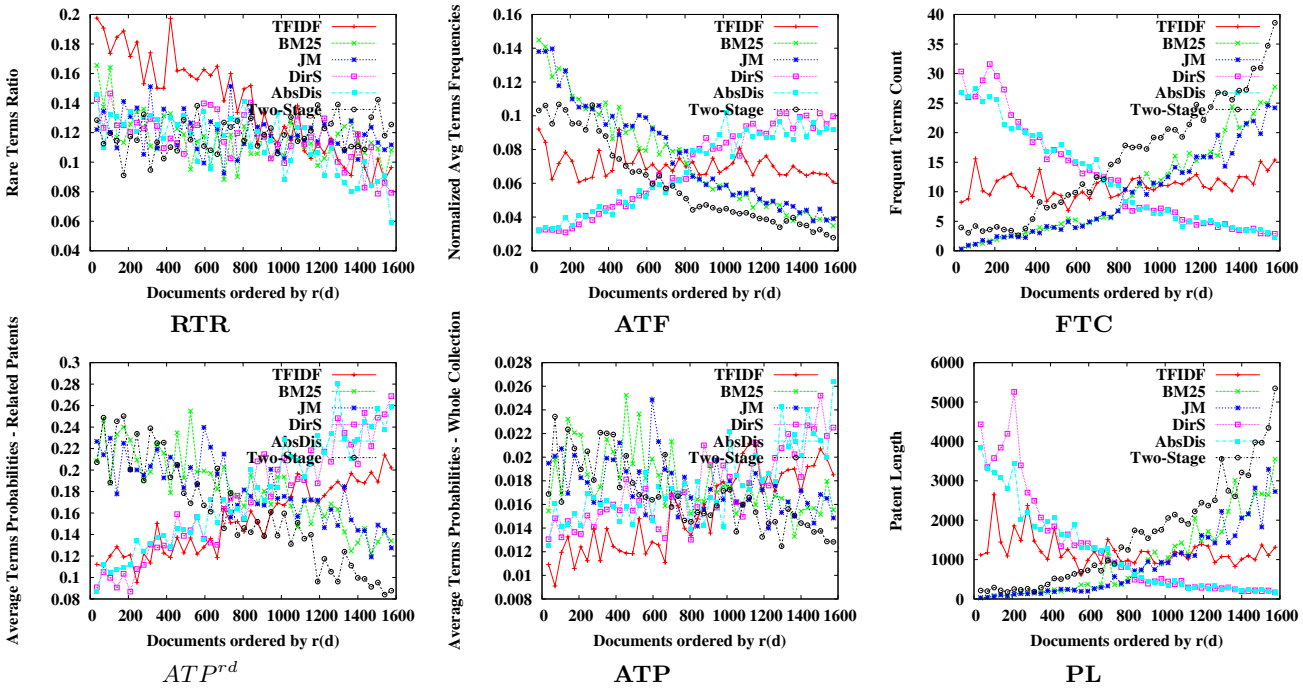


Figure 3: Effect of content-based features for USPC (422,423) with *rank cut-off*=30 using QG-DR.

DEXA '09, pages 753–760, August 31-September 4, 2009, Linz, Austria.

[3] T. Custis, K. Al-Kofahi. A new approach for evaluating query expansion: query-document term mismatch. *In Proc. of SIGIR '07*, pages 575–582, July 23-27, 2007, Amsterdam, The Netherlands.

[4] H. Doi, Y. Seki, M. Aono. A patent retrieval method using a hierarchy of clusters at TUT. *In Proc. of NTCIR-5 Workshop*, 2005, Tokyo, Japan.

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

[6] A. Fujii, M. Iwayama, N, Kando. Introduction to the special issue on patent processing. *Information Processing and Management: an International Journal*, Volume 43, Issue 5, 2007, pp. 1149–1153.

[7] H. Itoh, H. Mano, Y, Ogawa. Term distillation in patent retrieval. *ACL '03: Proceedings of the ACL-2003 workshop on Patent corpus processing*, 2003, pp. 41–45, Sapporo, Japan.

[8] C. Jordan, C. Watters, Q. Gao. Using controlled query generation to evaluate blind relevance feedback algorithms. *In Proc. of JCDL '06*, 2006, Pages 286–295, Chapel Hill, NC, USA.

[9] K. Konishi. Query terms extraction from patent document for invalidity search. *In Proc. of NTCIR '05: NTCIR-5 Workshop Meeting*, 2005, Tokyo, Japan.

[10] K. Konishi, A. Kitauchi, T. Takaki. Invalidity patent search system at NTT data. *In Proc. of NTCIR-4 Workshop Meeting*, 2004, Tokyo, Japan.

[11] A. Kontostathis, S. Kulp. The Effect of normalization when recall really matters. *In Proc. of IKE '08*, 2008, Pages 96–101, Las Vegas, Nevada, USA.

[12] H. Ney, U. Essen, R. Kneser. On structuring probabilistic dependences in stochastic language modeling. *Computer Speech and Language*, 1994, Pages 8:1–38.

[13] S. Robertson, S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. *In Proc. of SIGIR '94*, 1994, Pages 345–354, Dublin, Ireland.

[14] T. Sakai. Comparing metrics across TREC and NTCIR: the robustness to system bias. *In Proc. of CIKM '08*, Pages 581–590, October 26-30, 2008, Napa Valley, California, USA.

[15] I. H. Witten, E. Frank. Data mining: practical machine learning tools and techniques. *Morgan Kaufmann, 2nd edition*, 2005, USA.

[16] C. Zhai. Risk minimization and language modeling in text retrieval. *PhD thesis, Carnegie Mellon University*, 2006.

[17] C. Zhai, J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. *In Proc. of SIGIR '01*, 2001, Pages 334–342, New Orleans, Louisiana, United States.

[18] C. Zhai. Statistical language models for information retrieval. *Tutorial Presentation at the 29th ACM SIGIR*, 2006.