Informatik. Forschung und Entwicklung manuscript No.
(will be inserted by the editor)

Query Expansion Methods for Collaborative Information Retrieval

Armin Hust

e-mail: armin.hust@a-z-technology.de

Received: date / Revised version: date

Zusammenfassung Information Retrieval Systeme haben in den letzten Jahren nur geringe Verbesserungen in der Retrieval Performance erzielt. Wir arbeiten an neuen Ansätzen, dem sogenannten Collaborativen Information Retrieval (CIR), die das Potential haben, starke Verbesserungen zu erreichen. CIR ist die Methode, mit der durch Ausnutzen von Informationen aus früheren Anfragen die Retrieval Peformance für die aktuelle Anfrage verbessert wird. Wir haben ein eingeschränktes Szenario, in dem nur alte Anfragen und dazu relevante Antwortdokumente zur Verfügung stehen. Neue Ansätze für Methoden der Query Expansion führen unter diesen Bedingungen zu Verbesserungen der Retrieval Performance.
Schlüsselworte information retrieval – text mining – query expansion – collaborative information retrieval

Abstract The accuracy of ad-hoc document retrieval systems has reached a stable plateau in the last few years. We are working on so-called collaborative information retrieval (CIR) systems which have the potential to overcome the current limits. We define CIR as a task, where an information retrieval (IR) system uses information gathered from previous search processes from one or several users to improve retrieval performance for the current user searching for information. We focus on a restricted setting in CIR in which only old queries and correct answer documents to these queries are available for improving a new query. For this restricted setting we propose new approaches for query expansion procedures. We show how CIR methods can improve overall IR performance.
Key words information retrieval – text mining – query expansion – collaborative information retrieval

---

Subject Classification H.3.3

1 Introduction

We introduce the research area of Collaborative Information Retrieval (CIR), motivate and characterize the primary goals of this work, query expansion procedures for CIR, and outline the structure and contents of this work.

1.1 Information Retrieval

Although Information Retrieval has now been studied for decades there is no clear and comprehensive definition for Information Retrieval.

A newer definition, according to the IR-group of the German Informatics Society [9], states that "IR considers information systems according to their role in the knowledge transfer process from a human knowledge producer to an information seeker. The problems arising from vague queries and uncertain knowledge are the main focus of the IR-group. Vague queries are characterized by the fact that answers to these queries are a priori not uniquely defined (...). The uncertainty and/or the incompleteness of the knowledge often results from a restricted representation of its semantics, since the representation of the knowledge is not limited to some special forms (e.g. text documents, multimedia documents, facts, rules, semantic nets). Additionally IR considers applications where the stored knowledge itself may be uncertain or incomplete (e.g. technical or scientific data sets)" and states that "From these problems the necessity for an evaluation of the quality of the answers of an information system arises, where the utility of the system according to the support for the users with respect to solving their problems has to be considered."

This definition is very general. It stresses the vagueness and uncertainty of stored knowledge and queries. It also stresses the utility of the retrieved information for the users, helping them to solve their problems. Utility is an idea introduced by the von Neumann-Morgenstern utility theory [33] and is closely connected with their

idea of preference relations, both of which come from the field of economics.

An example illustrates the different preference relations users may have. A physician, a chemist and a lawyer may query an IR system for information about the medicament "Lipobay" or its American name "Baycol". While the physician may be interested in medication, indication and contra-indication, the chemist may by interested in chemical structure and undergoing reactions of the active ingredient; the lawyer may be interested in legal cases, lawsuits, court decisions and compensations. It is clear that each of these users has his or her own personal preferences as to which documents an IR system presents in response to the query.

Another research area overlapping with the IR area is the usage of context knowledge for a more detailed specification of the information need. Because queries can be vague, it might be possible to use knowledge about the context the user is working in to influence the query processing and achieve better retrieval results. Some of the aspects of the user's context (according to [11]) are: which tasks the user is busy with at the time of the query, which documents have been viewed within the last few minutes, which document is currently being processed by the user. Research in this area integrates modelling and representation of the context information, and integrates this information into the IR processes.

1.2 Collaborative Information Retrieval

The ultimate goal in IR is finding the documents that are useful to the information need expressed as a query. Much work has been done on improving IR systems, in particular in the Text Retrieval Conference series (TREC) [30]. In 2000, it was decided at TREC-8 that this task should no longer be pursued within TREC, in particular because the accuracy has plateaued in the last few years; then in TREC-2003, the HARD-Track and the Robust Retrieval Track have been included again with similar goals as the prior ad-hoc track. We are working on new approaches which learn to improve retrieval effectiveness from the interaction of different users with the retrieval engine. Such systems may have the potential to overcome the current plateau in ad-hoc retrieval.

We call our approach Collaborative Information Retrieval (CIR). CIR on top of an IR system uses all the methodologies that have been developed in this research field. Moreover, CIR is a methodology where an IR system makes full use of all the additional information available in the system, especially

- the information from previous search processes, i.e., individual queries and complete search processes
- the relevance information gathered during previous search processes, independent of the method used to obtain this relevance information, i.e., explicitly by user relevance feedback or implicitly by unobtrusively detected relevance information.

The collaborative aspect here differs from other collaborative processes. We do not assume that different users from a working team or a specific community collaborate loosely or tightly through some information exchange or workflow processes. Instead, we assume that users can benefit from search processes carried out at former times by other users (although those users may not know about the other users and their search processes) as long as the relevance information gathered from these previous users has some significant meaning.

Subject to these assumptions we expect that collaborative searches will improve overall retrieval quality for all users.

However, the methodology and methods described in this paper can not stand alone; a practical application should also consider these fields of "personalization" and "context" (as described in section 8.2.1).

1.3 Delimitation of Collaborative Information Filtering

The objective of information filtering (IF) is to classify/categorize documents as they arrive in the system. IF makes decisions about relevance or non-relevance rather than providing a ranked output list. In IF, the document collection can be seen as a stream of documents trying to reach the user, and unwanted documents are removed from the stream. The collaborative approach, called Collaborative Information Filtering (CIF) takes into account user preferences of other "like-minded" users.

While in CIR as described above, the user query is the central focus point, in CIF the documents are central. CIF can be described as a "push" technology, where documents are pushed against the user query (or user profile), while CIR is a "pull" technology, drawing the relevant documents from the collection.

1.4 Outline of this work

We limit ourselves to the text retrieval field, sometimes also called text mining, which is only a part of the information retrieval research area. As a first approach to CIR we also limit ourselves to developing, analyzing and evaluating algorithms which can be used for IR effectiveness improvements, based on individual queries which may be stated by different users. Because our evaluation data is static (queries, documents and relevance judgements), we can not consider complete search processes of users and especially ignore such vague queries, which can only be answered in dialogue by iterative reformulations of the queries.

This paper is organized as follows:

- section 2 describes related work in the field of query expansion.
- section 3 introduces the vector space model and query expansion procedures that have been developed for use in the vector space model.

- section 4 describes the document collections we use for evaluating our new algorithms and includes the description of some properties of the collections.
- section 5 introduces the environment of Collaborative Information Retrieval and describes the methodology used in the experiments and in the evaluation.
- section 6 shortly describes the algorithms that have been developed to be used in CIR.
- section 7 summarizes the improvements that we have achieved by our different algorithms.
- section 8 summarizes this paper, draws some conclusions, and describes the essential factors for improving retrieval performance in CIR.

2 Related Work

Usage of short queries in IR produces a shortcoming in the number of documents ranked according to their similarity to the query. Users issuing short queries retrieve only a few relevant documents, since the number of ranked documents is related to the number of appropriate query terms. The more query terms, the more documents are retrieved and ranked according to their similarity to the query [22]. In cases where a high recall is critical, users seldom have many ways to restate their query to retrieve more relevant documents.

Thus, IR systems try to reformulate the queries in a semi-automatic or automatic way. Several methods, called query expansion methods, have been proposed to cope with this problem [2], [19]. These methods fall into three categories: usage of feedback information from the user, usage of information derived locally from the set of initially retrieved documents, and usage of information derived globally from the document collection. The goal of all query expansion methods is to finally find the optimal query which selects all the relevant documents.

Query Expansion. The first publications describing query expansion procedures are Sparck-Jones [29], Minker et al. [20] and Rijsbergen [31]. Older procedures are described by Donna Harman in [10], experiments in the SMART systems have been described by Buckley et al. [3]. A comprehensive overview of newer procedures is available from Efthimiadis in [7]. Another newer technique, called local context analysis (LCA), was introduced by Xu and Croft in [36]. While pseudo relevance feedback assumes that all of the highly ranked documents are relevant, LCA assumes that only some of the top ranked documents initially retrieved for a query are relevant and analyzes these documents for term co-occurrences.

Query Clustering. Newest procedures in the field of query expansion are dealing with query bases, a set of persistent past optimal queries, for investigating similarity measures between queries. The query base can be used either to answer user queries or to formulate optimal queries (refer to Raghavan [23] and Sever [27]). Wen et al. [34] are using query clustering techniques for discovering frequently asked questions or most popular topics on a search engine. This query clustering method makes use of user logs which allows to identify the documents the users have selected for a query. The similarity between two queries may be deduced from the common documents the users selected for them. Cui et al. [5] take into account the specific characteristics of web searching, where a large amount of user interaction information is recorded in the web query logs, which may be used for query expansion. Agichtein et al. [1] are learning search engine specific query transformations for question answering in the web.

Relevance Feedback. Gathering relevance feedback is another field of research in this area. Automatic acquisition of relevance information is necessary for improving IR performance, since users are not willing or do not intend to give feedback about the relevance of retrieved documents. IR effectiveness does not improve unlimited after a few iterations of relevance feedback (refer to Salton et. al. [25]), and a mathematical structure of relevance effectiveness 'converging' to a limit, which behaves like a 'stable' point, is described by Dominich [6]. White et al. [35] compare two systems, where one is using explicit relevance feedback (where searchers explicitly have to mark documents relevant) and one is using implicit relevance feedback. They focus on the degree to which implicit evidence of document relevance can be substituted for explicit evidence. Joachims [16] acquires relevance information by merely using the clickthrough data while the documents presented to the user have been ranked by two different IR systems.

Term Weighting. Work in the field of term weighting procedures has been done ever since IR research. The dynamics of term weights in different IR models have been discussed in [4], going back to the work of [32]. The different models analyze the transfer of probabilities in the term space, mainly for, but not limited to, the probabilistic IR models.

These methods are model-independent; they apply to the (Extended) Boolean models as well as the Probabilistic models (refer to [8]) and to the Vector Space model applied in this work.

3 Basics and Terminology

This section introduces the basic vector space model (VSM) which is employed in our work. Additionally, we introduce a specific pseudo-relevance feedback (PRF) method, which is known to be successful for improving retrieval performance.

3.1 Vector Space Model

The vector space model, introduced by Salton [24], assigns weights to index terms in queries and in documents.

These term weights are ultimately used to compute the degree of similarity between each document stored in the system and the user query. By sorting the retrieved documents in decreasing order of this degree of similarity, the vector space model takes into consideration documents which match the query terms only partially.

Definition 1 (Vector Space Model) *Documents as well as queries are represented by vectors in a vector space. The set of N documents and L queries are denoted by*

$$D = \{d_j | 1 \leq j \leq N\} \tag{1}$$
$$Q = \{q_k | 1 \leq k \leq L\}. \tag{2}$$

Each individual document d_j and query q_k is represented by its vector

$$d_j = (d_{1j}, d_{2j}, ..., d_{Mj})^T \tag{3}$$
$$q_k = (q_{1k}, q_{2k}, ..., q_{Mk})^T, \tag{4}$$

where M is the number of terms in the collection and T denotes the transpose of the vector.

Each position i in the vectors corresponds to a specific term t_i in the collection. The values d_{ij} or q_{ik} respectively indicate the weighted presence or absence of the respective term in the document d_j or query q_k. The weights d_{ij} and q_{ik} are all greater than or equal to 0.

Term weights d_{ij} and q_{ik} in Equations 3 and 4 can be computed in many different ways. Different weighting schemes, so called tf-idf weighting schemes, have been developed by Salton and Buckley [25], the older work by Salton and McGill [26] reviews various term-weighting techniques. A newer work by Kolda [18] evaluates different weighting schemes and compares the results achieved by each of the weighting methods. The main idea behind the most effective term weighting schemes is related to the basic principles of clustering techniques [26]. Moreover it allows the usage of different weighting schemes for the document representation and the query representation.

Despite its simplicity, the vector space model is a resilient ranking strategy with general collections. It yields ranked answer sets which are difficult to improve upon without query expansion or relevance feedback within the framework of the vector space model. A large variety of alternative ranking methods have been compared to the vector space model but the consensus seems to be that, in general, the vector space model is either superior or almost as good as the known alternatives. Furthermore, it is simple and fast. For these reasons, the vector space model is a popular retrieval model nowadays.

Definition 2 (Cosine Similarity) *The ranking function normally used in the vector space model is the so called cosine-similarity. The vector space model proposes to evaluate the degree of similarity of the document d_j with regard to the query q_k as the correlation between these vectors. This correlation can be quantified, for instance, by the cosine of the angle between these two vectors. That is, the similarity* sim *between a document d_j and a given query q_k is measured by the cosine of the angle between these two M dimensional vectors:*

$$\text{sim}(d_j, q_k) = \frac{d_j^T \cdot q_k}{\|d_j\| \cdot \|q_k\|} = \frac{\sum_{i=1}^{M} d_{ij} \cdot q_{ik}}{\|d_j\| \cdot \|q_k\|} \tag{5}$$

where $\|\cdot\|$ is the Euclidean norm of a vector. In the case that the vectors are already normalized (and hence have a unit length) the similarity is simply the scalar product between the two vectors.

Figure 1 illustrates (in the 2-dimensional space) the document and query vectors of unit length lying on the surface of the unit-hypersphere. The cosine of angle α measures the similarity between a document and a query.

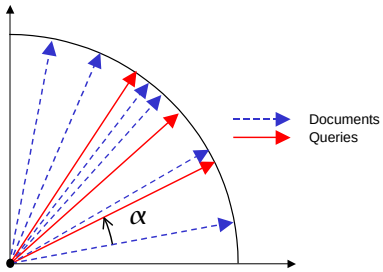

Fig. 1 Similarity of documents and queries

For our purposes we also need to measure the similarity between (sets of) documents and between queries.

Definition 3 (Inter-Document, Inter-Query Similarity) *The similarity* dsim *between two documents d_k and d_l and* qsim *between two queries q_k and q_l is measured by the cosine of the angle between these M dimensional vectors*

$$\text{dsim}(d_k, d_l) = \text{sim}(d_k, d_l) \tag{6}$$
$$\text{qsim}(q_k, q_l) = \text{sim}(q_k, q_l) \tag{7}$$

according to Equation 5.

3.2 Pseudo Relevance Feedback

Pseudo relevance feedback (PRF) avoids the interaction of the IR system with the user after the list of the retrieved documents is created in the first stage. PRF works in three stages: First documents are ranked according to their similarity to the original query. Then highly ranked documents are all assumed to be relevant (refer to [36]) and their terms (all of them or some highly weighted terms) are used for expanding the original query. Then documents are ranked again according to their similarity to the expanded query.

We employ a variant of pseudo relevance feedback described by Kise et al. [17]. In our comparisons with the newly developed methods, we will use the PRF method.

Let E be the set of document vectors given by

$$E = \left\{ d_j \middle| \frac{sim(d_j, q_k)}{\max_{1 \leq i \leq N}\{sim(d_i, q_k)\}} \geq \theta \right\} \tag{8}$$

(i.e., the $(1-\theta)$-percentage of highest ranked documents), where q_k is the original query and θ is a threshold parameter of the similarity. Then the sum D_k of the document vectors in E

$$D_k = \sum_{d_j \in E} d_j \tag{9}$$

is used as expansion terms for the original query. The expanded query vector q'_k is obtained by

$$q'_k = q_k + \alpha \frac{D_k}{\parallel D_k \parallel} \tag{10}$$

where α is a parameter for weighting the expansion terms. Then the documents are ranked again according to their similarity $\mathrm{sim}(d_j, q'_k)$. Parameters θ in Equation 8 and α in Equation 10 are tuning parameters.

4 The Text Collections

This section describes the contents of the text collections used in the evaluation. We mention some properties of the collections which are limiting factors for the retrieval performance of an IR system.

4.1 Contents of the Text Collections

We use standard document collections and standard queries (also called questions or topics in some specific TREC collections) provided by the SMART project [28] and the TREC conferences series [30]. For the TREC collections, relevance judgements are not available for each combination of query and collection; thus, we are limited to those queries and collections, where relevance judgements are available. Additionally, we have generated special collections from the TREC collections to show special effects of our algorithms, and we use two real world collections that have been gathered especially for these experiments. In our experiments we used the following 16 collections:

- The SMART collections ADI (articles about information sciences), CACM (articles from 'Communications of the ACM' journal), CISI (articles about information sciences), CRAN (abstracts from aeronautics articles), MED (medical articles) and NPL (articles about electrical engineering).
- The TREC collections CR with 34 queries out of topics 251 - 300 using the "title", "description" and "narrative" topics to investigate the influence of query length, FR with 112 queries out of topics 51 - 300.
- The TREC QA (question answering) question collection prepared for the Question Answering track held at the TREC-9 conference. From the question set 201-893, questions 201-700 were created without reference to the documents. Then in a separate pass, equivalent but re-worded questions (701-893) were created from a subset of these 500 questions. The QA-AP90 collection contains only those questions having a relevant answer document in the AP90 (Associated Press articles) document collection, the QA-AP90S collection (extracted from the QA-AP90 collection) having questions with inter-question similarity (refer to definition 3) of 0.65 or above to any other question.
- The QA-2001 collection prepared for the Question Answering track held at the TREC-10 conference.
- The PHIBOT collections PHYSICS (articles about physics) and SCIENCE (articles about sciences except physics) are real world collections gathered by a web search engine [21]. Ground truth data has been gathered from documents the user has clicked on from the list which is presented to the user after the query has been executed.

On the one hand by utilizing these collections, we take advantage of the relevance judgements (also called ground truth data) for performance evaluation. On the other hand, we do not expect to have queries having highly correlated similarities as we would expect in a real world application, except for the QA-AP90 and QA-AP90S collections. So it is a challenging task to achieve performance improvements for our methods.

4.2 Preparation of the Text Collections

Terms used for document and query representation were obtained by stemming and eliminating stopwords. Then document and query vectors were created according to the tf-idf weighting scheme (see section 3.1). The document weights d_{ij} are computed as

$$d_{ij} = \frac{1}{n_j} \cdot tf_{ij} \cdot idf_i, \tag{11}$$

n_j is the normalization factor $n_j = \sqrt{\sum_{i=1}^{M} (tf_{ij} \cdot idf_i)^2}$, tf_{ij} is a weight computed from the raw frequency f_{ij} of a term t_i (the number of occurrences of term t_i in document d_j)

$$tf_{ij} = \sqrt{f_{ij}} \tag{12}$$

and idf_i is the inverse document frequency of term t_i given by

$$idf_i = \log \frac{N}{n_i}, \tag{13}$$

where n_i is the number of documents containing term t_i and N is the number of documents in the collection. The query weights q_{ik} are computed as

$$q_{ik} = \frac{1}{n_k} \cdot \sqrt{f_{ik}}, \tag{14}$$

where n_k is the normalization factor $n_k = \sqrt{\sum_{i=1}^{M} f_{ik}}$ and f_{ik} is the raw frequency of a term t_i in a query q_k (the number of occurrences of term t_i in query q_k).

4.3 Properties of the Text Collections

Table 1 lists statistics about the collections after stemming and stopword elimination has been carried out, statistics about some of these collections before stemming and stopword elimination can be found in Baeza-Yates [2] and Kise et al. [17].

4.3.1 Similarities of Queries to Documents Some of the current limitations in IR are described here. Figures 2 and 3 present the similarity between each query and documents. Figure 2 displays the similarity of each query to its relevant documents, figure 3 displays the similarity of each query to its non-relevant documents. The dots indicate the similarity of an individual document to a query. The thin connecting line indicates the average similarity of all relevant (or non-relevant) documents for each query. The thick line averages these similarities over all queries.

We can see that average similarity of a query to its relevant documents is higher than average similarity of a query to its non-relevant documents. But very often it occurs that there are non-relevant documents having a higher similarity to a query than relevant documents. From this observation it follows that retrieval precision is decreasing if similarity between a query and non-relevant documents is high.

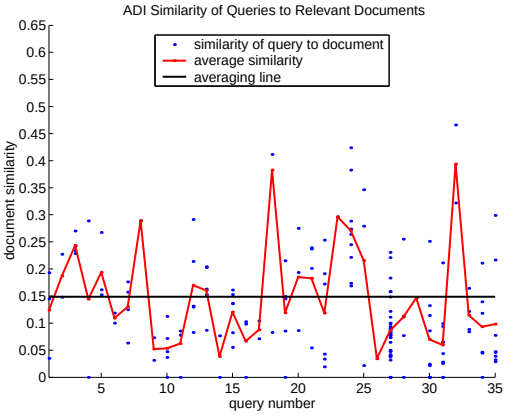

Fig. 2 ADI: similarities of relevant documents

4.3.2 Inter-Query Similarities In our preliminary considerations for usage of similarities between different queries for retrieval performance improvements, we decided to analyze the inter-query similarities (refer to definition 3). As we already stated, we did not expect to have queries having highly correlated similarities as we would expect in real world applications.

Indeed, the following histograms in figures 4 and 5 have very low inter-query similarity (as for most of the text collections). Figure 4 displays the distribution of the inter-query similarity, including those similarities which are 0. Since this is the dominating factor in each text collection, we also produced the same histogram leaving out inter-query similarities of 0 (figure 5). Also the

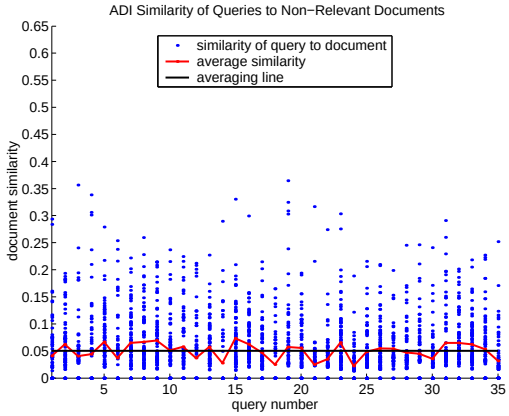

Fig. 3 ADI: similarities of non-relevant documents

mean and the median value as well as the variance and the standard deviation are indicated in each graph. The vertical lines are: the mean similarity (solid line), and the values of the mean similarity ± the standard deviation (dotted lines).

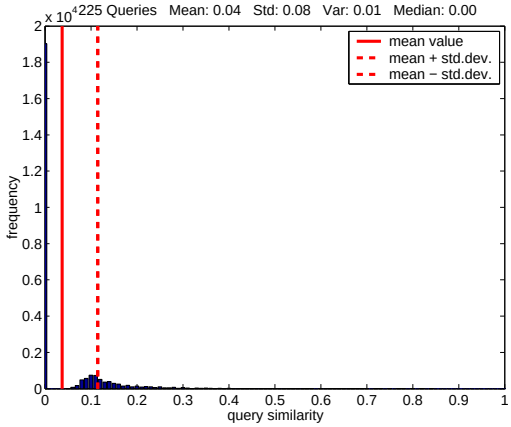

Fig. 4 CRAN: distribution of query similarities

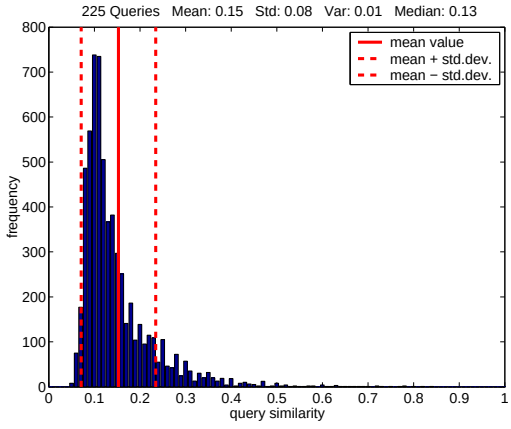

Fig. 5 CRAN: distribution of query similarities

4.3.3 Correlation between Query Similarities and Document Similarities We analyzed the inter-query similarities as opposed to the inter-document similarity of

	ADI	CACM	CISI	CRAN	MED	NPL	PHY-SICS	SCI-ENCE
size(MB)	0.1	1.2	1.4	1.4	1.1	3.8	4.9	20.6
number of documents	82	3204	1460	1400	1033	11429	375	2175
number of terms	340	3029	5755	2882	4315	4415	35312	104891
mean number of terms per document	17.9 (short)	18.4 (short)	38.2 (med)	49.8 (med)	46.6 (med)	17.9 (short)	308.2 (long)	322.1 (long)
number of queries	35	52	112	225	30	93	230	1108
mean number of terms per query	5.7 (med)	9.3 (med)	23.3 (long)	8.5 (med)	9.5 (med)	6.5 (med)	1.9 (short)	2.0 (short)
mean number of relev. documents per query	4.9 (low)	15.3 (med)	27.8 (high)	8.2 (med)	23.2 (high)	22.4 (high)	1.7 (low)	2.0 (low)

	CR-desc	CR-narr	CR-title	FR	QA	QA-AP90	QA-AP90S	QA-2001
size(MB)	93	93	93	69	28.2	3.7	3.7	20.1
number of documents	27922	27922	27922	19860	6025	723	723	4274
number of terms	45717	45717	45717	50866	48381	17502	17502	40626
mean number of terms per document	188.2 (long)	188.2 (long)	188.2 (long)	189.7 (long)	230.7 (long)	201.8 (long)	201.8 (long)	220.5 (long)
number of queries	34	34	34	112	693	353	161	500
mean number of terms per query	7.2 (med)	22.8 (long)	2.9 (short)	9.2 (med)	3.1 (short)	3.2 (short)	3.5 (short)	2.7 (short)
mean number of relev. documents per query	24.8 (high)	24.8 (high)	24.8 (high)	8.4 (med)	16.4 (med)	2.8 (low)	3.2 (low)	8.9 (med)

Table 1 Statistics about the test collections

the relevant documents. If there were a direct correlation between inter-query similarities to inter-document similarities of the relevant documents, it would be easy to derive the relevant documents for a given new query from the documents being relevant to the existing old queries.

From these considerations, we derived the creation of the following graphs (see figure 6): each graph indicates the inter-query similarities for each two pairwise different queries on the x-axis and the inter-document similarity of the relevant documents on the y-axis as a dot. For example, a dot at coordinates $(0.5, 0.9)$ shows that there are two queries having an inter-query similarity of 0.5 and their relevant documents have an inter-document similarity of 0.9. The line in each graph is the least-squares estimator for the polynomial of degree 1 fitting best to the clouds of dots.

We see that there is no simple correlation between inter-query similarity and inter-document similarity. There are low inter-query similarity and their relevant documents have a high inter-document similarity and vice versa. This holds for all text collections.

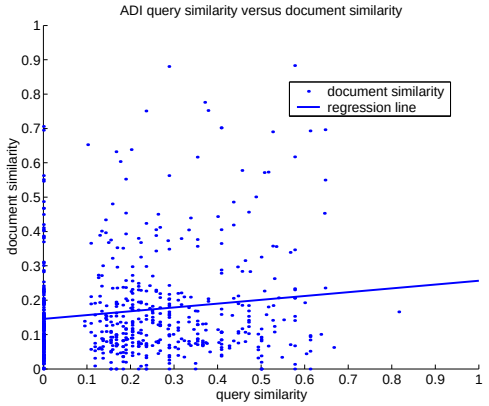

Fig. 6 ADI: query similarity vs. document similarity

4.3.4 Overlap of Relevant Documents It is essential for achieving retrieval performance improvements to have some "overlapping" relevant documents for pairs of queries.

Definition 4 (Overlap of Relevant Documents) *Let $q_k, q_l \in Q,\ k \neq l$ be two different queries. Let RD_k, RD_l be the sets of documents being relevant to the queries q_k and q_l respectively. Then the overlap of relevant documents for these two queries is the number of documents in the set $O_{kl} = RD_k \cap RD_l = \{d_j |\ d_j \in RD_k \wedge d_j \in RD_l\}$.*

For all our new query expansion procedures we expect retrieval performance improvements if the overlap of relevant documents is high. Table 2 gives some statistics about the overlap of relevant documents, figure 7 presents the individual overlap for each pair of queries for one collection. The x-axis and y-axis are labelled by the query number, the z-axis indicates the overlap for each pair of queries. Since the overlap between each pair of queries is symmetric ($|O_{kl}| = |O_{lk}|$), we left out the symmetric part for clarity.

5 Collaborative Information Retrieval

We first explain the motivation for our new approaches to Collaborative Information Retrieval. We introduce the general methodology used in our algorithms and the evaluation methodology.

5.1 Motivation for Collaborative Information Retrieval

In our approach we use global relevance feedback which has been learned from previous queries instead of local relevance feedback which is produced during execution of an individual query. The motivation for our query

	ADI	CACM	CISI	CRAN	MED	NPL	PHY-SICS	SCI-ENCE
pairs of queries	595	1326	6216	25200	435	4278	26335	613278
max overlap	7	17	70	18	0	36	1	4
query pairs with overlap	90	134	1154	686	0	181	25	75
percentage of query pairs with overlap	15.1%	10.1%	18.6%	2.7%	0.0%	4.2%	0.1%	0.01%

	CR-desc	CR-narr	CR-title	FR	QA	QA-AP90S	QA-AP90	QA-2001
pairs of queries	561	561	561	6216	239778	12880	62128	124750
max overlap	27	27	27	10	140	16	16	11
query pairs with overlap	35	35	35	385	760	195	237	259
percentage of query pairs with overlap	6.2%	6.2%	6.2%	6.2%	0.3%	1.5%	0.4%	0.2%

Table 2 Statistics about overlap of relevant documents

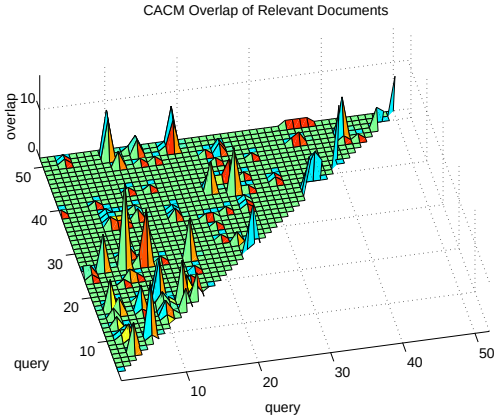

Fig. 7 CACM: overlap of relevant documents

expansion method is straightforward, especially in an environment where document collections are static, and personal preferences and context knowledge are ignored:

- If documents are relevant to a query which has been issued previously by a user, then the same documents are relevant to the same query at a later time when that query is re-issued by the same or by a different user. This is the trivial case, where similarities between the two different queries is the highest.
- In the non-trivial case a new query is similar to a previously issued query only to a certain degree. Then our assumption is that documents which are relevant to the previously issued query will be relevant to the new query only to a certain degree.

It does not necessarily follow that, if a new query is dissimilar to a previously issued query, the documents which are relevant to the previously issued query are not relevant to the new query.

As described in the introduction (section 1.1), we are aware of the problems of "personal preferences" and "context", but in our first steps towards techniques we avoid further complexity by ignoring these challenges.

Our approach is to find the exact degree of similarity between queries (which of course includes finding the exact degree of dissimilarity) that maximizes the improvements in retrieval performance. We do this by expanding the newly issued query to include terms from previously issued queries and/or documents known as being relevant to the previously issued queries.

5.2 Methodology of Collaborative Information Retrieval Methods

All our new query expansion procedures work as follows:

- for each new query to be issued, compute the similarities between the new query and each of the existing old queries and select those old queries having a similarity to the new query which is greater than or equal to a given threshold
- from these selected old queries get the sets of relevant documents from the ground truth data (the relevance judgements)
- from these sets of relevant documents select some or all terms (depending on the method) for expansion of the new query
- use these terms to expand the new query and issue the new expanded query

The algorithmic description is given here:

```
for each new query q do
 compute the set S = {q_k | sim(q_k, q) ≥ σ, 1 ≤ k ≤ L}
 compute the sets RD_k = {d_j | q_k ∈ S and
         d_j is relevant to q_k}
 compute the expanded query q' = cirf(q, S, RD_k)
         by some function cirf
end
```

where S is the set of existing old queries q_k with a similarity of σ or higher to the new query q, RD_k are the sets of the documents being relevant to the queries q_k and $cirf$ is a function for query expansion.

The goal is to find suitable functions $cirf$ which can be efficiently computed and which maximize the effectiveness of the new query q' in terms of recall and precision. Our approach is searching for neighbors of the new query (illustrated in figure 8). If suitable neighbors of a query q within a given distance are found, we try to derive information about the documents which are relevant to q from its nearest neighbors.

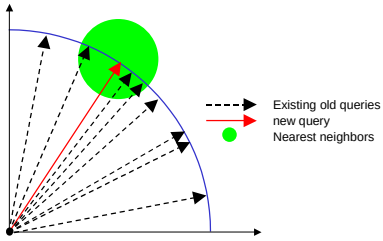

Fig. 8 Motivation for CIR methods: usage of the nearest neighbors

These functions introduce a new level of quality in the IR research area: while the term weighting functions such as tf-idf only work on documents and document collections, and relevance feedback works on a single query and uses information from their assumed relevant and non-relevant documents only, CIR now works on a single query, and uses the information of all other queries and their known relevant documents.

5.3 Evaluation Methodology

The evaluation follows the "leave one out" technique used in several areas such as document classification, machine learning etc.

From the set of L queries contained in each text collection, we select each query one after the other and treat it as a new query $q_l, 1 \leq l \leq L$. Then for each fixed query q_l we use the algorithm as described in section 5.2. Of course the now fixed query q_l itself does not take part in the computation of the query expansion. We vary parameters of the algorithms according to suitable values, and select those parameters where highest performance improvements (in terms of average precision over all queries) have been achieved.

Recall and precision are the standard measures for evaluation of IR systems ([2], [19]). For each query, recall is the fraction of relevant documents that have been retrieved, precision is the fraction of retrieved documents which are relevant. Precision values are computed at the standard 11-point-recall-levels 0.0, 0.1,...,1.0. Average precision averages these precision values over all queries. Recall/precision graphs are produced according to the standard method of interpolated average precision computation.

Statistical tests provide information about whether observed differences in different methods are really significant or just by chance. We employ the "paired t-test" described in [12]. Table 5 gives the significance indicators from statistical testing of the experimental results. Each row contains the results of two tests, i.e., test method X against method Y and vice versa.

- An entry of $++$ $(--)$ in a table cell indicates that method X (Y) is almost guaranteed to perform better than method Y (X) at significance level $\alpha = 0.01$.
- An entry of $+$ $(-)$ indicates that method X (Y) is likely to perform better than method Y (X) at significance level $\alpha = 0.05$.
- An entry of o indicates that there is low probability that one of the methods is performing better than the other method.

6 Query Expansion Methods for CIR

In this section, we shortly describe the query expansion methods which we have developed for CIR. For detailed information refer to [14], [15] and [13].

6.1 Methods Description

6.1.1 Query Similarity and Relevant Documents, QSD. Method QSD (refer to [14]) uses the relevant documents of the most similar queries for query expansion of a new query. The new query is rewritten as a sum of selected relevant documents of existing old queries, which have a high similarity to the new query, i.e.,

$$q' = q + \sum_{k=1}^{|S|} \sigma_k \frac{RD_k}{\parallel RD_k \parallel}, \tag{15}$$

where $|S|$ is the number of selected queries, σ_k are the similarities $\text{sim}(q_k, q) \geq \sigma$ (σ is the threshold value) and RD_k are the sets of relevant documents.

6.1.2 Query Linear Combination and Relev. Documents, QLD. Method QLD (refer to [15]) uses the relevant documents of the most similar queries, which are used in re-writing the new query as a linear combination of the most similar queries. This query expansion method reconstructs the new query as a linear combination of existing old queries. Then the terms of the relevant documents of these existing old queries are used for query expansion, i.e.,

$$q' = q + \sum_{k=1}^{|S|} \tilde{\lambda_k} \frac{RD_k}{\parallel RD_k \parallel}, \tag{16}$$

where the $\tilde{\lambda_k}$ are parameter for weighting the expansion terms.

The $\tilde{\lambda_k}$ are computed as follows: in most cases we cannot represent the new query q exactly as a linear combination of the old queries q_k, i.e.,

$$q = \sum_{k=1}^{|S|} \lambda_k q_k \tag{17}$$

will not have a solution for the coefficients λ_k. Equation 17 is equivalent to a system of linear equations

$$Q\lambda = q \tag{18}$$

where $Q = (q_1, q_2, \ldots, q_{|S|})$ is a matrix of M rows and $|S|$ columns and $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_{|S|})^T$ is a column vector consisting of $|S|$ elements. Because Q is normally singular ($M \gg |S|$) and there is no solution to the system, we find a vector $\tilde{\lambda}$ so that it provides a closest fit to the equation in some sense. Our approach is to minimize the Euclidean norm of the vector $Q\lambda - q$, i.e we solve

$$\tilde{\lambda} = argmin_\lambda \|Q\lambda - q\| \tag{19}$$

where $\tilde{\lambda} = (\tilde{\lambda_1}, \tilde{\lambda_2}, \ldots, \tilde{\lambda_{|S|}})^T$ is called the least squares solution for the system $Q\lambda = q$.

6.1.3 Document Term Reweighting, Query Term Reweighting. Methods DTW and QTW [13] use the relevant documents of the most similar queries for giving more weight to those ambiguous terms in the documents or in queries, that match the semantics of the same terms in queries or documents. If, for example, the queries use the term 'bank' in conjunction with other terms related to financial topics, then the term 'bank' meaning 'financial institution' will be weighted higher than the term 'bank' meaning 'dike' or 'wall'.

The set of N documents is written as a matrix $D = (d_j)_{1 \le j \le N}$, the set of L queries is written as a matrix $Q = (q_k)_{1 \le k \le L}$. We also write the relevance judgements as a matrix $R = (r_{jk})_{1 \le j \le N, 1 \le k \le L}$, where $r_{jk} = 1$ if d_j is relevant to q_k and $r_{jk} = 0$ if d_j is not relevant to q_k.

The similarity matrix SIM is computed according to Equation 5

$$SIM = sim(D, Q) = D^T \cdot Q. \tag{20}$$

The SIM matrix and the matrix R are of same size. Both have N rows and L columns. In the best case the SIM matrix computed according to equation (20) would be identical to the given R matrix. In this case the precision for each query would be 100% at every recall level, because every relevant document has a similarity of 1 to the query and every non-relevant document has a similarity of 0 to the query.

Because this will almost never be the case, we try to find transformation matrices to achieve this goal. Method DTW computes a matrix W_D satisfying

$$W_D \cdot SIM \approx R, \tag{21}$$

method QTW computes a matrix W_Q satisfying

$$SIM \cdot W_Q \approx R. \tag{22}$$

The existence- and uniqueness-proof of these matrices follows from theorems for pseudo-inverse matrices.

Then, for DTW we reweight the document terms by $W_D \cdot D$ before we expand a new query by the document terms of relevant documents of most similar queries, and for QTW we reweight query terms by $Q \cdot W_Q$ before we compute the set of most similar queries.

6.2 Experiments Description

The newly developed methods were evaluated using different settings for parameters σ (see section 5.2). Best parameter value settings have been obtained by experiment and those which give the highest average precision were used for reporting here.

We analyzed the deviations in average precision for small changes in the optimum values of σ. Small changes in σ only led to small changes in average precision (from -12% up to +4%). This indicates that the algorithms are robust, and that there are no "jump discontinuities" in the neighborhood of the optimum σ values.

Then we combined two methods for query expansion in this ways: First, after having expanded the new query using the PRF method, we applied one of the methods QSD, QLD, DTW and QTW against the expanded query. These methods are reported as the PRFxxx methods. Second, after having expanded the new query using the QSD, QLD and QTW methods, we applied the PRF method against the expanded query. These methods are reported as the xxxPRF methods.

Recall/precision graphs for methods QLD and QSD are presented in figures 9 and 10.

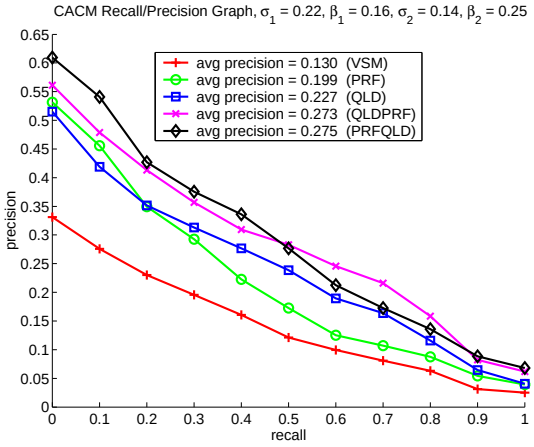

Fig. 9 CACM: recall/precision graphs for QLD method

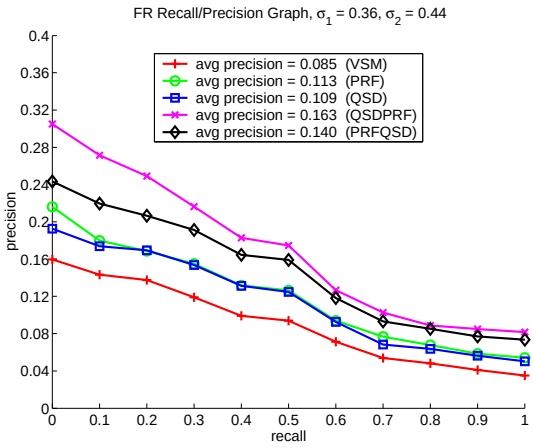

Fig. 10 FR: recall/precision graphs for QSD method

7 Improvements

Our new methods are compared to the PRF method, since this is widely used in IR systems and outperforms the VSM in any case. Tables 4 and 5 summarize the results of the newly developed methods. In table 4 the best value of average precision is indicated by bold font, the second best value is indicated by italic font. In those cases, where our new methods outperform the PRF method, the value is underlined. Table 5 summarizes the results from significance testing.

Results are non-uniform. In some cases, the basic methods perform better than PRF. This is the case for the text collections having queries with high inter-query similarities. In most cases, the combined methods outperform the PRF method.

Highest relative performance improvements are reported in table 3. The ratio of improvement is computed as follows: Let X be the average precision obtained by one of the methods and let Y be the average precision obtained by the PRF method. Then the ratio is calculated by $ratio = \frac{X-Y}{Y}$. A positive value for the ratio indicates an improvement of method X over method PRF.

	relative improvement	achieved by method
ADI	+1.2%	PRFQLD
CACM	+37.9%	PRFQLD
CISI	+34.0%	QLDPRF
CRAN	+8.0%	PRFQLD
MED	-1.3%	PRFQLD
NPL	+0.4%	PRFQLD
PHYSICS	+0.0%	PRFQLD
SCIENCE	+0.0%	PRFQLD
CR-desc	+8.0%	PRFQTW
CR-narr	+0.7%	PRFQLD
CR-title	+12.0%	PRFQLD
FR	+44.1%	QSDPRF
QA	+10.5%	PRFQLD
QA-AP90	+7.6%	QLDPRF
QA-AP90S	+19.3%	QLDPRF
QA-2001	+0.2%	PRFQLD

Table 3 Best relative performance improvements

For a quick overview figures 11 and 12 show the average precision achieved by each method in bar graphs.

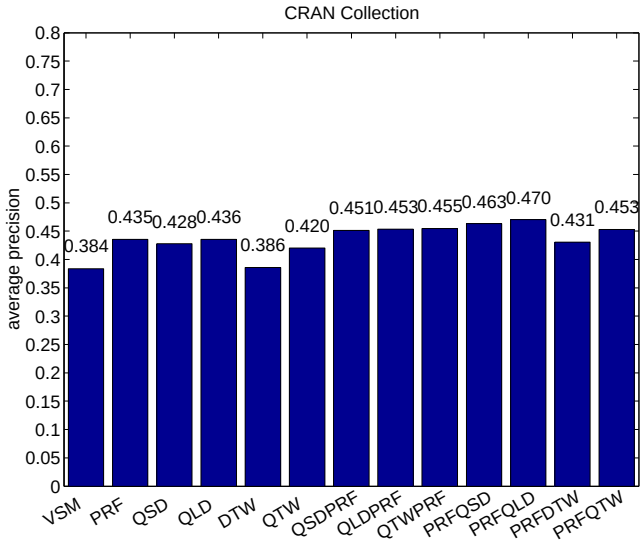

Fig. 11 CRAN: average precision comparison

8 Conclusions and Application

8.1 Conclusions

We have studied methods for improving retrieval performance in a restricted Collaborative Information Retrieval environment where information about relevant documents from previous search processes carried out by several users is available for the current query.

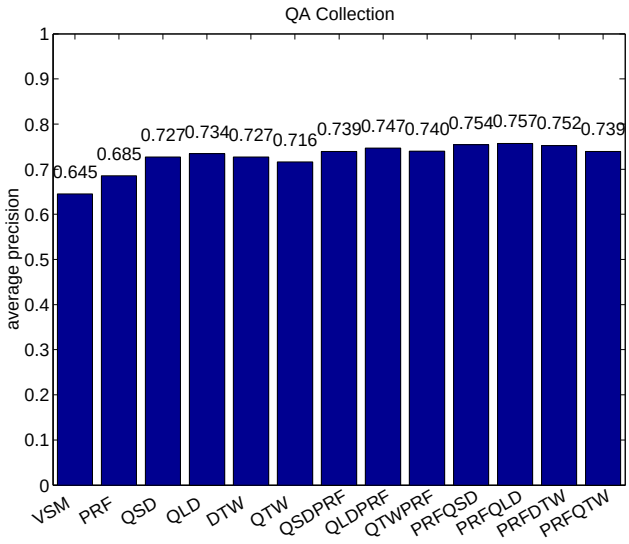

Fig. 12 QA: average precision comparison

Specifically, we developed, evaluated and analyzed new algorithms for query expansion, since query expansion methods are known to be successful in improving retrieval performance.

Results of the newly developed methods are encouraging. Retrieval performance improvements were achieved in most cases. From the basic methods QSD, QLD, DTW and QTW best results were achieved in the combination with the Pseudo Relevance Feedback (PRF) method.

For some text collections, no significant retrieval performance improvements could be achieved, neither in the basic methods nor in applying the combined methods.

We identified three essential factors for retrieval performance improvements:

1. similarity between queries, also called inter-query similarity (refer to section 4.3.2)
2. similarity of queries to their relevant documents and similarity of queries to their non-relevant documents (refer to section 4.3.1)
3. the overlap of relevant documents for pairs of queries (refer to section 4.3.4)

We think that the first two factors are more important for achieving improvements than the last factor. Best performance improvements have been achieved in text collections, where the inter-query similarity is high, although the overlap in relevant documents is not high.

Low or no retrieval performance improvements were achieved in those cases where the inter-query similarity in the average is low. Also for text collections where similarity of queries to their non-relevant documents is high in the average, we could not achieve high performance improvements.

No performance improvements were achieved in the cases where the overlap of relevant documents is low (MED, PHYSICS, SCIENCE).

	ADI	CACM	CISI	CRAN	MED	NPL	PHY-SICS	SCI-ENCE
VSM	0.375	0.130	0.120	0.384	0.525	0.185	0.616	0.569
PRF	0.390	0.199	0.129	0.435	**0.639**	*0.224*	**0.638**	**0.587**
QSD	0.374	0.237	0.142	0.428	0.503	0.184	0.612	0.567
QLD	0.369	0.227	*0.171*	0.436	0.507	0.185	0.614	0.569
DTW	0.356	0.142	0.122	0.386	0.494	0.182	0.599	0.561
QTW	0.364	0.154	0.131	0.420	0.500	0.183	0.611	0.565
PRFQSD	*0.391*	0.256	0.151	*0.463*	0.611	0.223	0.634	*0.584*
PRFQLD	**0.394**	**0.275**	0.169	**0.470**	*0.631*	**0.225**	**0.638**	**0.587**
PRFDTW	0.372	0.208	0.133	0.431	0.602	0.221	0.627	0.575
PRFQTW	0.388	0.231	0.133	0.453	0.606	0.222	0.635	0.583
QSDPRF	0.388	0.257	0.145	0.451	0.609	**0.225**	*0.636*	0.582
QLDPRF	0.385	*0.273*	**0.173**	0.453	0.613	0.207	0.611	**0.587**
QTWPRF	0.380	0.206	0.137	0.455	0.609	*0.224*	0.635	0.582

	CR-desc	CR-narr	CR-title	FR	QA	QA-AP90	QA-AP90S	QA-2001
VSM	0.175	0.173	0.135	0.085	0.645	0.745	0.643	0.603
PRF	0.204	*0.192*	0.169	0.113	0.685	0.757	0.661	*0.614*
QSD	0.172	0.173	0.152	0.109	0.727	0.810	0.786	0.603
QLD	0.175	0.175	0.164	0.108	0.734	0.812	0.789	0.603
DTW	0.150	0.173	0.132	0.098	0.727	0.785	0.732	0.601
QTW	0.150	0.173	0.144	0.106	0.716	0.808	0.762	0.601
PRFQSD	0.196	*0.192*	0.180	0.140	*0.754*	0.813	0.781	0.613
PRFQLD	*0.208*	**0.193**	**0.190**	0.144	**0.757**	0.814	0.782	**0.615**
PRFDTW	0.200	0.191	0.154	0.123	0.752	0.791	0.733	0.611
PRFQTW	**0.221**	0.191	0.180	0.127	0.739	0.809	0.755	0.612
QSDPRF	0.195	0.191	0.177	**0.163**	0.739	0.813	*0.786*	*0.614*
QLDPRF	0.204	*0.192*	0.184	*0.161*	0.747	**0.815**	**0.789**	0.613
QTWPRF	0.179	*0.192*	*0.189*	0.157	0.740	*0.815*	0.764	0.613

Table 4 Average precision obtained in different methods

methods X	methods Y	ADI	CACM	CISI	CRAN	MED	NPL	PHY-SICS	SCI-ENCE
PRF	VSM	+	++	++	++	++	++	+	++
QSD	PRF	o	o	o	o	−−	−−	−	−−
QLD	PRF	o	o	++	o	−−	−−	−	−−
DTW	PRF	−	−	o	−−	−−	−−	−−	−−
QTW	PRF	o	−	o	o	−−	−−	−	−−
PRFQSD	PRF	o	++	+	++	o	o	o	o
PRFQLD	PRF	o	++	++	++	o	o	o	o
PRFDTW	PRF	o	o	o	o	o	o	−	−−
PRFQTW	PRF	o	+	o	+	o	o	o	−
QSDPRF	PRF	o	o	o	o	o	o	o	o
QLDPRF	PRF	o	+	++	o	o	o	−−	o
QTWPRF	PRF	o	o	o	o	o	o	o	−−

methods X	methods Y	CR-desc	CR-narr	CR-title	FR	QA	QA-AP90	QA-AP90S	QA-2001
PRF	VSM	++	+	+	+	++	+	o	++
QSD	PRF	−−	−	o	o	++	++	++	−−
QLD	PRF	−−	o	o	o	++	++	++	−−
DTW	PRF	−	−	−−	o	++	++	++	−−
QTW	PRF	−	−	o	o	++	++	++	−−
PRFQSD	PRF	o	o	o	+	++	++	++	o
PRFQLD	PRF	o	o	o	+	++	++	++	o
PRFDTW	PRF	o	o	o	o	++	++	++	o
PRFQTW	PRF	o	o	o	o	++	++	++	o
QSDPRF	PRF	o	−	o	+	++	++	++	o
QLDPRF	PRF	o	o	o	+	++	++	++	o
QTWPRF	PRF	o	o	o	+	++	++	++	o

Table 5 Paired t-test results for significance levels $\alpha = 0.05$ and $\alpha = 0.01$ in different methods

8.2 Application

Creation of a real-world Information Retrieval System was not the goal of this work. However, we can imagine that such a system, similar to well known search engines, could be implemented in a future step.

8.2.1 Implementation For the implementation of our CIR methods in a real-world search engine, we have to consider some additional topics that were ignored in this work to avoid further complexity:

1. We have to gather relevance judgements for retrieved documents from users' actions and store these relevance judgements in an appropriate way. These relevance judgements could be assigned explicitly by the users to the results of their individual queries or complete search processes, or taken implicitly by observing users' interaction with the retrieved documents (refer to section 2).
2. We have to incorporate some "personalization" functions. Our assumptions (refer to section 5.1) will not hold in a real-world scenario. Different users querying a system may have different information needs, for example, a query "jaguar speed" may refer to the

speed of an animal, the speed of a car, or the speed of an operating system.
3. We have to incorporate some "context" functions. For a query "jaguar speed", the same user may be interested in the speed of the car at some time, while the same user may be interested in the speed of an operating system some time later.

8.2.2 Heuristics for successful Application of CIR Methods The CIR methods described in this work can significantly improve retrieval performance. The following description gives hints for successful integration of CIR methods into a real-world Information Retrieval System.

The first step is to analyze the underlying text collection and the set of queries, the second step is to adjust the set of queries to meet the requirements for successful application of CIR methods.

Analyze the underlying text collection and adjust the set of queries according to:

1. The similarity of queries to their relevant documents and the similarity of queries to their non-relevant documents (refer to section 4.3.1). This is the only factor we can not adjust in text and query collections.
2. The inter-query similarity (refer to section 4.3.2). In the case where we have no pairs of queries with high similarities, the inter-query similarity distribution looks like a normal distribution with positive skewness (see figures 4 and 5). In this case, we should add further queries together with the judgements for their relevant documents to the stored query set until the inter-query similarity distribution looks more like a standard normal distribution or looks like a normal distribution with negative skewness.
3. The overlap of relevant documents for pairs of queries (refer to section 4.3.4). In the case where there is no or low overlap in relevant documents (see figure 7), CIR methods will not be successful. In this case, we should add further queries together with the judgements for their relevant documents to the stored query set until we reach a sufficient overlap in relevant documents. Because we identified this factor being less important, we should first try to adjust the inter-query similarity. While adding new queries and relevance judgements to adjust the inter-query similarity, there may be a high probability that the overlap of relevant documents is also adjusted in the same process.

References

1. Eugene Agichtein, Steve Lawrence, and Luis Gravano. Learning search engine specific query transformations for question answering. *Proceedings of the 10th International World Wide Web Conference*, pages 169–178, 2001.
2. Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Publishing Company, 1999.
3. Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic query expansion using smart. *NIST Special Publication 500-226: Proceedings of the Third Text Retrieval Conference (TREC-3)*, pages 69–80, 1994.
4. Fabio Crestani and Cornelius J. van Rijsbergen. A study of probability kinematics in information retrieval. *ACM Transactions on Information Systems (TOIS)*, 16(3):225–255, 1998.
5. Hang Cui, Ji-Rong Wen, Jian-Yun Nieand, and Wei-Ying Ma. Probabilistic query expansion using query logs. *Eleventh International World Wide Web Conference*, 2002.
6. Sándor Dominich. *Relevance Effectiveness in Information Retrieval*, chapter 5, pages 215–232. Mathematical Foundations of Information Retrieval. Kluwer Academic Publishers, 2001.
7. Efthimis N. Efthimiadis. Query expansion. *Annual Review of Information Science and Technology*, 31:121–187, 1996.
8. Norbert Fuhr and Chris Buckley. A Probabilistic Learning Approach for Document Indexing. *ACM Transactions on Information Systems*, 9(3):223–248, 1991.
9. Norbert Fuhr. Goals and tasks of the IR group. Homepage of the IR group of the German Informatics Society, 1996. `http://ls6-www.cs.uni-dortmund.de/ir/fgir/mitgliedschaft/brochure2.html`.
10. Donna Harman. Relevance feedback revisited. *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1–10, 1992.
11. Andreas Henrich. IR research at university of bayreuth. Homepage of the IR-research group, 2002. `http://ai1.inf.uni-bayreuth.de/forschung/forschungsgebiete/ir_mmdb`.
12. David Hull. Using statistical testing in the evaluation of retrieval experiments. *16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 329–338, 1993.
13. Armin Hust, Markus Junker, and Andreas Dengel. A Mathematical Model for Improving Retrieval Performance in Collaborative Information Retrieval. *Kluwer Information Retrieval Special Issue: Advances in Mathematical/Formal Methods in Information Retrieval*, 2004. to appear.
14. Armin Hust, Stefan Klink, Markus Junker, and Andreas Dengel. Query Expansion for Web Information Retrieval. *Proceedings of Web Information Retrieval Workshop, 32nd Annual Conference of the German Informatics Society*, volume P-19 of *Lecture Notes in Informatics*, pages 176–180, 2002.
15. Armin Hust, Stefan Klink, Markus Junker, and Andreas Dengel. Query Reformulation in Collaborative Information Retrieval. *Proceedings of the International Conference on Information and Knowledge Sharing, IKS 2002*, pages 95–100, 2002.
16. Thorsten Joachims. Unbiased evaluation of retrieval quality using clickthrough data. Technical report, Cornell University, Department of Computer Science, 2002.
17. Koichi Kise, Markus Junker, Andreas Dengel, and Keinosuke Matsumoto. Experimental evaluation of passage-based document retrieval. *Sixth International Conference on Document Analysis and Recognition ICDAR-01*, pages 592–596, 2001.

18. Tamara G. Kolda and Dianne P. O'Leary. A semidiscrete matrix decomposition for latent semantic indexing information retrieval. *ACM Transactions on Information Systems*, 16(4):322–346, 1998.
19. Christopher D. Manning and Hinrich Schütze. *Foundations of Natural Language Processing*. MIT Press, 1999.
20. Jack Minker, Gerald Wilson, and Barbara Zimmerman. An evaluation of query expansion by the addition of clustered terms for a document retrieval system. *Information Storage and Retrieval*, 8:329–348, 1972.
21. Phibot search engine. Homepage, 2002. `http://phibot.org`.
22. Yonggang Qiu and Hans-Peter Frei. Concept-based query expansion. *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 160–169, 1993.
23. Vijay V. Raghavan and Hayri Sever. On the reuse of past optimal queries. *18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 344–350, 1995.
24. Gerard Salton. *The SMART retrieval system — experiments in automatic document processing*. Prentice Hall, Englewood Cliffs, New Jersey, 1971.
25. Gerard Salton and Chris Buckley. Term weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988.
26. Gerard Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Co., New York, 1983.
27. Hayri Sever. *Knowledge Structuring for Database Mining and Text Retrieval Using Past Optimal Queries*. PhD thesis, University of Louisiana, Lafayette, LA, 1995.
28. Ftp directory at cornell university. Homepage, 1968–1988. `ftp://ftp.cs.cornell.edu/pub/smart`.
29. Karen Sparck-Jones and Roger M. Needham. Automatic term classification and retrieval. *Information Storage and Retrieval*, 4:91–100, 1968.
30. Text REtrieval Conference (TREC). Homepage, 1992–2003. `http://trec.nist.gov`.
31. Cornelius J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979.
32. Cornelius J. van Rijsbergen. Towards an information logic. *Proceedings of the 12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 77–86, 1989.
33. John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.
34. Ji-Rong Wen, Jian-Yun Nie, and Hong-Jiang Zhang. Query clustering using user logs. *ACM Transactions on Information Systems*, 20(1):59–81, 2002.
35. Ryen W. White, Ian Ruthven, and Joemon M. Jose. The use of implicit evidence for relevance feedback in web retrieval. *Advances in Information Retrieval, 24th BCS-IRSG European Colloquium on IR Research, ECIR 2002, Proceedings*, volume 2291 of *Lecture Notes in Computer Science*, pages 93–109, 2002.
36. Jinxi Xu and W. Bruce Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, 18(1):79–112, 2000.