Classifying and Filtering Blind Feedback Terms to Improve Information Retrieval Effectiveness

Johannes Leveling
Centre for Next Generation Localisation (CNGL)
Dublin City University
Dublin 9, Ireland
jleveling@computing.dcu.ie

Gareth J. F. Jones
Centre for Next Generation Localisation (CNGL)
Dublin City University
Dublin 9, Ireland
gjones@computing.dcu.ie

ABSTRACT

The classification of blind relevance feedback (BRF) terms described in this paper aims at increasing precision or recall by determining which terms decrease, increase or do not change the corresponding information retrieval (IR) performance metric. Classification and IR experiments are performed on the German and English GIRT data, using the BM25 retrieval model. Several basic memory-based classifiers are trained on different feature sets, grouping together features from different query expansion (QE) approaches. Combined classifiers employ the results of the basic classifiers and correctness predictions as features. The best combined classifiers for German (English) yield 22.9% (26.4%) and 5.8% (1.9%) improvement for term classification wrt. precision and recall compared to the best basic classifiers. IR experiments based on this term classification have also been performed. Filtering out different types of BRF terms shows that selecting feedback terms predicted to increase precision improves the average precision significantly compared to experiments without BRF. MAP is improved by +19.8% compared to the best standard BRF experiment (+11% for German). BRF term classification also increases the number of relevant and retrieved documents, geometric MAP, and P@10 in comparison to standard BRF. Experiments based on an optimal classification show that there is potential for improving IR effectiveness even more.

Categories and Subject Descriptors

H.3.3 [**INFORMATION STORAGE AND RETRIEVAL**]: Information Search and Retrieval—*Query formulation, Relevance feedback, Search process*

General Terms

Performance, Measurement, Experimentation

Keywords

Query Reformulation, Query Expansion, Blind Relevance Feedback, Machine Learning

1. INTRODUCTION

Blind relevance feedback (BRF, also called pseudo-relevance feedback) for information retrieval (IR) is regarded as a means to improve recall and precision. Some previous studies have shown that BRF may decrease precision [11], and the general usefulness of BRF to improve IR effectiveness has been questioned [1]. A possible explanation might be that QE with synonyms or other related terms found in documents presumed to be relevant also adds noise to the results. Thus, more relevant documents may be retrieved but precision for certain topics may degrade.

The main objective of this paper is to answer the questions: *"Are there BRF terms which have a negative effect on IR performance metrics (i.e. are there good and bad BRF terms)?"*, *"Can good and bad feedback terms be differentiated automatically?"*, and *"What is the impact of selecting only good feedback terms on precision and recall?"*

In this paper, experiments are performed to calculate the change in performance resulting from adding a single term to a query. From these results, training data for a machine learning approach (ML) is derived. Memory-based learning is applied to automatically classify which individual terms have positive, negative, or zero effect on IR metrics. Filtering out feedback terms with a negative effect is expected to increase the corresponding IR metric. Different BRF strategies using the term classification results are evaluated, filtering out candidate feedback terms which are predicted to be not useful.

The rest of this paper is organized as follows: Section 2 introduces related work. Section 3 describes the experimental IR setup. Section 4 proposes features for feedback term classification and presents results for the basic and combined classifiers. Retrieval experiments using different blind feedback strategies and their results are discussed in Section 5, before ending with an outlook on future work in Section 6.

2. RELATED WORK

Blind relevance feedback (BRF) is a well-known approach to improve IR performance by extracting terms from top-ranked documents retrieved in an initial retrieval step, expanding the query with these terms, and retrieving a final result set with the modified query [16]. BRF in IR has been extensively researched, investigating the decision whether to use selective or massive feedback [2] or how to best rank candidate feedback terms [16], but still some questions remain unanswered, e.g. how to dynamically adapt feedback parameters [12].

Robertson [13] names several alternatives to deal with query expansion terms, including using all terms from known relevant documents (leaving processing to the term weighting scheme), including a term selection stage to omit poor terms instead of assigning them low weights, and including user interaction to select useful expansion terms. Robertson expresses the question *"How useful would a candidate term be?"* as *"How much effect would adding it to the search formulation have on retrieval performance?"* and suggests that the decision for including a term in an expanded query should be based on its increase in effectiveness.

Cao, Nie and Robertson [3] differentiate between good and bad feedback terms. They train an SVM classifier based on seven features, including BM25 document scores and achieve an accuracy of about 69% for the classification of terms into good and bad expansion terms on three TREC collections. An Oracle (optimal) classification results in 18-30% improvement in MAP. They conclude that there is much potential improvement to be gained if the classification accuracy can be further improved and that BRF may not have to be recall-oriented: *"adding the expansion terms does not hurt, but improves precision"*. In contrast, the classification described in this paper employs many more features and aims at increasing precision and recall. IR experiments are conducted for both German and English, with combined classifiers employed to improve classification accuracy.

Ogilvie, Vorhees, and Callan [12] aim at dynamically adapting the number of feedback terms, e.g. optimizing it for each topic individually. They compare results from eight retrieval systems and observe that systems are able to obtain almost all possible benefit with a fixed number of feedback terms. Results show that the best static number of feedback terms and the resulting improvement over no expansion vary with the systems (10-100 terms resulting in 5.7-31.5% improvement in MAP). The findings suggest that current term weighting and a static number of feedback terms leave little room for further improvement. They also find that 15-55% improvement in MAP over no expansion is possible when the number of feedback terms is determined dynamically for each topic (4-30% over feedback with a static number of terms). Their results also seem to suggest that performance improvements stem from a combination of terms rather than from a single term. In the term classification described in this paper, a set of terms (including the empty set) is determined for QE implicitly, based on performance changes calculated for adding a single term. The best number of feedback terms for each topic is also implicitly determined by the cardinality of the set of terms selected for feedback, i.e. if all BRF terms are considered as bad, the query will not be expanded. Furthermore, the experiments described here are motivated by a different rationale. Selecting a cutoff value builds on the assumption that good terms are likely to appear consecutively in the top term ranking. Instead, it is presumed here that the ranking by traditional term selection value is not good enough to support finding a single cutoff value which divides the terms into good and bad terms.

Lavrenko and Croft [8] investigate a formal probabilistic approach to estimate a model of relevance with no additional training data. The proposed method can be viewed as a form of massive query expansion, where the original query is replaced with a distribution over the entire vocabulary. The method does not require careful tuning of parameters (e.g. the number of feedback documents). On TREC ad-hoc data, the relevance model outperforms the language modeling baseline approach with query expansion on one query set, but performs worse on the second query set.

Gey and Petras [6] observe that BRF improves performance only for some topics. They state that for geographic information retrieval (GIR), the most useful feedback terms seem to be proper nouns (location names), but do not investigate this suggestion. Leveling [9] explores the idea that location names are most useful for BRF in GIR in more detail. Experiments on the GeoCLEF newspaper data showed that restricting feedback terms to a certain type (i.e. location names) and filtering out other feedback terms can significantly improve MAP, but not for all topics.

These approaches are similar to the work presented in this paper as they identify a type of useful feedback terms and filter out less useful feedback terms. However, the approaches use only a single property of a feedback term (its part-of-speech) and are applied to a specific domain only (GIR), whereas classification experiments in this paper are based on different feature sets and intended for a more general application domain (ad hoc IR).

Van Halteren, Zavrel, and Daelemans [17] combine different classification systems and approaches to improve part-of-speech tagging. They report a reduction of 24.3% error rate for tagging the LOB corpus when combining different classification results. In contrast, the experiments described in this paper are based on a single ML system, combining basic classifiers trained on different feature sets by using their results and predicted correctness as classification features.

Other related research includes optimizing document selection for BRF and learning to rank, which aims at reordering a document set to obtain higher precision (e.g. [4]). However, these topics are outside of the scope of the experiments described in this paper.

3. SYSTEM DESCRIPTION AND EXPERIMENTAL SETUP

3.1 Documents, Topics, and Relevance Assessments

The retrieval experiments described in this paper are based on data from the German Indexing and Retrieval Test database (GIRT), which has been used in the domain-specific track at the Cross Language Retrieval Forum (CLEF) for several years (see, for example [7]). The document collections are available in German and English and consist of 151,319 documents each.[1] GIRT documents contain metadata on publications from the social sciences, represented as structured XML documents. The metadata scheme defines 14 fields, including abstract, authors, classification terms, controlled terms, date of publication, and title.

A GIRT topic resembles topics from other retrieval campaigns such as TREC. It contains a brief summary of the information need (topic title), a longer description (topic description), and a part with information on how documents are to be assessed for relevance (topic narrative). Retrieval queries are typically generated from the title (T) and description (D) fields of topics. For example, the description of GIRT topic 177 is *Find publications focusing on jobless adolescents who have not completed any vocational training.* GIRT includes 150 German and English topics from the

[1] In 2006, 20,000 abstracts from Cambridge Scientific Abstracts were added to the English GIRT document collection. Since no relevance assessments are available for topics from before 2006, these documents were discarded for the experiments described in this paper.

domain-specific track at CLEF from 2003 to 2008 (25 topics each year), together with official relevance assessments.

In addition to providing a number of topics and the corresponding relevance assessments large enough to generate data for ML techniques. GIRT data includes additional resources, most notably a bi-lingual thesaurus, which serves as an external domain-specific knowledge resource for the term classification. Furthermore, the document collection is available in German and in English, which makes a comparison of the importance of term features between two different languages possible.

3.2 The Information Retrieval System

The Lucene toolkit[2] was employed to preprocess the topics and documents, and to index and search the document collection. Support for the BM25 retrieval model [16, 15] and for the corresponding BRF approach (see Equation 1 and 2) was implemented into Lucene by one of the authors.

The BM25 score for a document and a query Q is defined as:

$$\sum_{t \in Q} w^{(1)} \frac{(k_1+1)\mathit{tf}}{K+\mathit{tf}} \frac{(k_3+1)\mathit{qtf}}{k_3+\mathit{qtf}} \quad (1)$$

where Q is the query, containing terms t and $w^{(1)}$ is the RSJ (Robertson / Sparck-Jones) weight of T in Q [14]:

$$w^{(1)} = \frac{(d+0.5)/(D-d+0.5)}{(n-d+0.5)/(N-n-D+d+0.5)} \quad (2)$$

where

- k_1, k_3, and b are model parameters. The default parameters for the BM25 model used are $b = 0.75$, $k_1 = 1.2$, and $k_3 = 7$.
- N is the number of documents in the collection and D is the number of documents known or presumed to be relevant for a topic.
- n is the document frequency for the term and d is the number of relevant documents containing the term.
- tf is the frequency of the term within a document; qtf is the frequency of the term in the topic.
- $K = k_1((1-b) + b \cdot \mathit{doclen}/\mathit{avg\_doclen})$
- *doclen* and *avg_doclen* are the document length and average document length, respectively.

BRF terms are ranked by a term selection value (TSV). The method to compute the TSV for a term for the IR experiments used in this paper was first introduced by [13] and is shown in Equation 3.

$$TSV = (d/D) \cdot w^{(1)} \quad (3)$$

For all retrieval experiments, the topic title and description were used to create IR queries for Lucene (TD). The document structure was flattened into a single index by collecting the abstract, title, controlled terms and classification text as content and discarding the remaining fields (e.g. author, publication-year, and language-code). Standard Lucene modules were employed to tokenize the text and to fold upper case characters to lower case. Stopword lists from Jacques Savoy's web page on multilingual IR resources[3] for German (603 words) and English (571 words) were used to identify stopwords. Stemming of topics and documents was performed using the German or English Snowball stemmer provided in Lucene.

4. FEEDBACK TERM CLASSIFICATION

4.1 Classification Features

One strategy to improve on existing BRF is to form a distinction between *good* terms and *bad* terms for QE, where good terms are terms which increase retrieval effectiveness and bad terms are those which do not. In this paper, the basic term classifiers for precision (P-classification) are denoted by Pc_i, and term classifiers for recall (R-classification) are denoted by Rc_i. Similarly, classifiers using classification results are denoted by Pp_i and Rp_i, and combined classifiers are denoted by Pcc_i and Rcc_i. The class resulting from a classifier is denoted by C(classifier). Three classes representing the effect of a term on recall or precision were defined, corresponding to a decrease in performance metric, no change, or an increase. Class labels represent the abbreviated form of the change: positive (p), zero (z), or negative (n) effect. Each feedback term in the top-50 feedback terms ranked by TSV was added to the original query. The term classes were obtained by computing the difference between average precision and the number of relevant documents at 1000 documents for the original query and the query expanded with this feedback term, using the sign of the change to derive the class.

The training data contains 7500 training instances for each language, corresponding to 50 feedback terms for 150 topics, which consist of of term features and class labels. For the classification according to precision (P-classification), all three classes occurred (German: 2564 terms in p, 1080 z, and 3856 n; English: 2642 p, 582 z, and 4276 n).

One of the most important evaluation metrics for classification is accuracy, which is defined as the number of correctly classified instances divided by the number of all instances. Always selecting class n (the majority baseline) achieves 51.4% accuracy, i.e. classifiers should outperform this trivial baseline. For the R-classification, only two classes occurred (German: 885 p and 6615 z, English: 281 p and 7219 z). The majority baseline achieves 88.2% accuracy. Note that the number of relevant and retrieved documents can (in theory) decrease when a query is expanded by adding terms. However, in the training data used for the experiments described in this paper, this case did not occur.

4.2 Feature Sets

Features for basic term classifiers are grouped in seven feature sets (FSj, $j \in \{1, \dots, 7\}$), which roughly correspond to different approaches to QE (e.g. co-occurrence based QE or thesaurus-based QE). All features are numeric, except for one string feature, i.e. the suffix removed from a word during stemming. Note that representing the string feature as a set of binary features (one for each suffix) might cause problems when multiple languages are examined because the number and type of removed suffixes is not known in advance for both training and test data.

Basic classifiers were initially trained on a single feature set. In some cases, this approach had to be extended to cover

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

[3] http://members.unine.ch/jacques.savoy/clef/index.html

two or three feature sets to obtain a reasonable accuracy. Combined classifiers aggregate results from basic classifiers. For the basic classifiers, 80 features f_i ($i \in \{1, \ldots, 80\}$) were considered, which are briefly described in Table 1. Seven features corresponding to classifications of the basic classifiers and seven features corresponding to the predicted correctness of classifiers based on the same feature sets were used for the combined classifiers.

FS1 includes term-specific features which have been used in traditional BRF (e.g. d). FS1 also contains a non-numeric feature corresponding to the suffix which has been removed from the term during stemming (f_{12}). Query-specific features and properties of the result set size are grouped in FS2. For example, if a query contains only few and rare words, only few documents can be retrieved at all (e.g. less than 1000 documents), BRF may be found to improve effectiveness in general for this query. FS3 contains features determining the relation between a feedback term and terms in the original query, including thesaurus relations and the Levenstein distance which can indicate if a term is a morphological variant or a spelling error. These features are motivated by knowledge-based approaches to query expansion. BM25 document scores (FS4) have also been used as term classification features by Cao, Nie and Robertson [3]. For the classification experiments described here, document score differences are also considered (FS5). Statistical thesauri are based on co-occurrence measures for terms. Features in FS6 try to model co-occurrences of feedback terms and terms in the original query in the document collection by well-known co-occurrence measures (mutual information, phi-square-coefficient, and log-likelihood ratio). Finally, features in FS7 represent positional information of the feedback term. The flat index generated from different document fields may indicate the importance of individual document fields. The (normalized) absolute position of a feedback term may help to associate document structure with importance of document fields for feedback (e.g. title and abstract occur at the beginning of the document, subject terms are given at the end).

Two additional feature sets (FS8 and FS9) are derived from results of the basic classifiers. FS8 includes all term classification results from basic classifiers (denoted as $C(Pc_i)$ or $C(Rc_i)$). For example, a corresponding training instance is (1:p, 2:p, 3:z, 4:n, 5:p, 6:p, 7:p, class: p). FS9 contains predictions on the correctness of the basic classifications ($C(Pp_i)$ or $C(Rp_i)$). Instead of training the predictions on the classes p, z, and n, the classes determine if the classification by a basic classifier is correct (c) or incorrect (i). For example, a corresponding training instance for FS9 is (1:c, 2:c, 3:i, 4:i, 5:c, 6:c, 7:c, class: p).

4.3 Training Data

For the 150 English (German) topics, 15574 (16200) documents have been assessed as relevant, i.e. there are on average 103.8 (108) relevant documents per topic. BRF typically uses 20-30 terms and 10-20 documents. For the ML experiments described here, T was set to 50 to obtain a pool of term candidates for BRF. D was set to 20 to allow generating a large number of training instances while avoiding a bias towards bad terms. For aggregated feature values, different aggregating functions were employed (e.g. minimum, maximum, and average). Feature values are normalized when necessary, e.g. the term position in a document was normalized by document length. Multi-class features (e.g. thesaurus relations) were represented as a set of binary features. Some features are included both in terms of absolute values and value changes. The GIRT thesaurus was employed to determine semantic relations between query terms and candidate feedback terms. For phrase lookup, Wikipedia article titles in German and English were used as an external resource (versions from June and May 2008, respectively).

Two sets of training data were generated. The first set corresponds to term classification for precision (P-classification), the second set corresponds to term classification for recall (R-classification). Instances in both training have the same feature values, but differ in their class labels. The training data for the R-classification is imbalanced, showing few instances of the interesting class p. However, no downsampling or upsampling of the training data was applied. Liu, Chawla et al. [10] report that imbalanced data sets are still a problem for ML and there is no best solution yet. For the experiments described in this paper, explicit feature selection was avoided by training classifiers on different feature sets. Negative effects resulting from imbalanced data seem to be alleviated by combining the results of basic classifiers.

4.4 Training with TiMBL

TiMBL [5] implements a memory-based learning approach and supports different algorithms for supervised ML. For the experiments described in this paper, TiMBL's IB1 algorithm was employed (k-NN or k-nearest distances classification). TiMBL was used to train classifiers for both the classification of terms with respect to precision (P-classification) and the classification of terms with respect to recall (R-classification). All classifiers were cross-validated with the *leave-one-out* method, i.e. trained on the set of training instances except for the test instance.

4.5 Results for Basic Classifiers

Table 2 shows the classification results, including F-Score, area under curve (AUC), and accuracy as reported by TiMBL. In addition, the number of true positives for the most interesting class ($TP(p)$) for 2564 terms in p for German, and 2642 terms for English is given. Table 3 shows the results for classifying terms by change in recall. Adding a single term to the original query may decrease precision. In contrast, no term was identified which adversely affected the number of relevant and retrieved documents when added to the original query. BRF does not decrease recall in the cases observed (although it is theoretically possible). Only a few terms increase the number of retrieved relevant documents; the majority of feedback terms do not affect this performance metric at all. This may be one explanation why BRF could be considered to mainly enhance recall: few terms would actually decrease the number of relevant and retrieved documents, and additional terms provide more context to differentiate relevant documents. However, adding terms which do not affect recall may still result in lower average precision, because more noise is added to the query.

The basic classifiers yield a relatively low accuracy and the feature sets vary in the number of features they contain. Thus, some basic classifiers for P-classification and R-classification are trained on features from more than one feature set. However, a simple combination of features does not always yield a higher classification performance (cf. Pc_7 vs. Pc_1). This observation was confirmed by results from the combined classifier (Pcc_1), which achieved a higher F-score, but identified less true positives for the class p than

Table 1: Classification features.

Feature	Feature set	Description
f_1-f_6	FS1	inverse TSV rank, TSV score, qtf, n, d, term length
f_7-f_{11}	FS1	does the term contain lowercase characters, uppercase characters, digits, punctuation, or other non-alphabetic characters?
f_{12}	FS1	suffix removed from the term during stemming
f_{13}-f_{16}	FS2	query length in tokens; result set size for the original query, for the query expanded with the feedback term, and percentual change between both
f_{17}-f_{21}	FS3	is the term part of the original query or an expansion term?, are the term and any query term part of a phrase (using Wikipedia article names as a phrase lexicon)?, is the term a compound constituent of an original term or vice versa?
f_{22}-f_{27}	FS3	is the term is a prefix, suffix, or infix string of any query term, or vice versa?
f_{28}	FS3	minimum edit-distance (Levenstein distance) between term and query terms
f_{29}-f_{32}	FS3	thesaurus relations (synonymy, broader term, narrower term, and related term) between term and original query terms
f_{33}-f_{47}	FS4	BM25 document score of the document at kth position ($k \in \{1, 10, 20, 30, \ldots, 100, 200, 300, 400, 500\}$)
f_{48}-f_{62}	FS5	differences between BM25 document score for the original query and for the query expanded with the feedback term
f_{63}-f_{71}	FS6	minimum, maximum, and average values for mutual information, phi-square-coefficient, and log-likelihood ratio between term and query terms
f_{72}-f_{80}	FS7	minimum, maximum, and average values for absolute position of the term in a document; relative position of the term to query terms and the offset of the term
	FS8	classification results from basic classifiers
	FS9	correctness predictions for basic classifiers

classifier Pc_2. The basic classifiers seem to perform similarly for both P-classification and R-classification.

The low classification accuracy for FS1 may indicate that there is no obvious relation between the TSV or inverse TSV rank and the feedback terms to be used. Specifically, results seem to suggest that other features for ranking terms might be better and that there is no obvious cutoff for the best number of terms. The BM25 scores used in FS4 show that there may be a relation between BM25 document scores and the quality of the extracted feedback terms.

4.6 Results for Combined Classifiers

The simple combination of classification results does not achieve a higher accuracy than the best basic classifiers (see Pcc_1 in Table 2 and Rcc_1 in Table 3). However, combining classification results from the basic classifiers with results from classifiers predicting the correctness of the basic classifiers improved accuracy significantly. For comparison, classifiers using all 80 features were also evaluated. These classifiers yield a performance comparable to or higher than that of the basic classifiers, but are outperformed for both P-classification and R-classification for English and German by the basic classifiers Pc_2 and Rc_2 and by the combining classifiers Pcc_2 and Rcc_2.

Classifiers including the predicted correctness of the classification as features show a higher accuracy compared to the basic classifiers. The best P-classification (Pcc_2) is achieved by a combined classifier and yields an accuracy of 82.5% and an F_1-score of 82.4% for both German and English. These results are considerably higher than the 69% accuracy reported by Cao, Nie and Robertson [3], although their experiments are based on different data. The best combined classifier (Rcc_2) achieved an accuracy of 96.5% and an F_1-score of 96.4% for German (98.6% accuracy and F-Score for English).

4.7 Estimating the Impact of Classification

A preliminary estimate of the impact of classifying feedback terms can be obtained by computing the ratio of *good* terms used to all feedback terms used. For the optimal classification, the ratio is 1 (if only terms of class p are used). For standard BRF in German, the ratio is 0.342 (2565 out of 7500) for the P-classification and 0.118 for the R-classification (885 out of 7500). The classifier Pcc_1 (Pcc_2) increases the ratio to 0.621 (0.790, respectively). The classifier Rcc_1 (Rcc_2) increases the ratio to 0.701 (0.865). For English, term classification improves the ratio similarly. In summary, standard blind relevance feedback employs many *bad* feedback terms. The combined classifiers using predicted correctness of basic classifiers show that *good* and *bad* feedback terms can be automatically identified with high accuracy. An improved BRF approach is proposed which filters out *bad* terms and employs only *good* feedback terms.

5. IR EXPERIMENTS

The effect of feedback term classification was evaluated in IR experiments on the German and English GIRT data, filtering out different types of feedback terms. The goal of the classification was to use only *good* and to discard *bad* feedback terms (select/reject decision). The definition of *good* varies with the experiment:

- All terms obtained from the standard BRF approach are considered as good (using feedback terms from the classes p, z, and n). This corresponds to standard BRF, where all terms are selected.
- Good BRF terms are those which do not decrease the metric (classes p and z).
- Good BRF terms are those which increase the metric (class p).

Table 2: Classification results for P-classification on German and English data.

Name	Feature set	German Results				English Results			
		F-Score	AUC	ACC	TP(p)	F-Score	AUC	ACC	TP(p)
all	FS1-FS7	0.651	0.697	0.651	1417	0.639	0.656	0.639	1415
Pc$_1$	FS1	0.494	0.571	0.494	980	0.504	0.538	0.504	1018
Pc$_2$	FS2, FS4	**0.671**	**0.720**	**0.671**	**1526**	**0.651**	**0.671**	**0.651**	**1436**
Pc$_3$	FS1, FS3	0.493	0.571	0.492	976	0.507	0.541	0.508	1029
Pc$_4$	FS4	0.657	0.708	0.657	1500	0.636	0.657	0.637	1395
Pc$_5$	FS5	0.611	0.676	0.612	1316	0.576	0.611	0.576	1177
Pc$_6$	FS6	0.617	0.670	0.618	1270	0.529	0.559	0.528	1094
Pc$_7$	FS1, FS7	0.480	0.561	0.480	959	0.508	0.542	0.508	1039
Pp$_1$	FS1	0.781	0.781	0.781	-	0.798	0.798	0.798	-
Pp$_2$	FS2, FS4	0.806	0.766	0.810	-	0.788	0.760	0.790	-
Pp$_3$	FS1, FS3	0.755	0.755	0.755	-	0.764	0.764	0.764	-
Pp$_4$	FS4	0.794	0.758	0.800	-	0.777	0.751	0.780	-
Pp$_5$	FS5	0.737	0.719	0.739	-	0.689	0.680	0.690	-
Pp$_6$	FS6	0.819	0.800	0.821	-	0.797	0.795	0.797	-
Pp$_7$	FS1, FS7	0.687	0.686	0.687	-	0.715	0.715	0.715	-
Pcc$_1$	FS8	0.679	0.714	0.685	1265	0.636	0.648	0.685	1161
Pcc$_2$	FS8, FS9	**0.824**	**0.848**	**0.825**	**1966**	**0.837**	**0.848**	**0.825**	**2032**

Table 3: Classification results for R-classification on German and English data.

Name	Feature set	German Results				English Results			
		F-Score	AUC	ACC	TP(p)	F-Score	AUC	ACC	TP(p)
all	FS1-FS7	0.896	0.738	0.896	470	0.961	0.703	0.962	119
Rc$_1$	FS1	0.846	0.631	0.845	312	0.935	0.549	0.935	37
Rc$_2$	FS2, FS4	**0.910**	**0.771**	**0.912**	**520**	**0.967**	**0.747**	**0.968**	**143**
Rc$_3$	FS1, FS3	0.849	0.639	0.849	323	0.932	0.539	0.931	32
Rc$_4$	FS4	0.879	0.693	0.883	393	0.962	0.711	0.964	123
Rc$_5$	FS5	0.873	0.662	0.880	333	0.949	0.626	0.951	77
Rc$_6$	FS6	0.836	0.607	0.836	271	0.937	0.560	0.938	43
Rc$_7$	FS1, FS7	0.849	0.642	0.849	329	0.935	0.544	0.936	34
Rp$_1$	FS1	0.926	0.862	0.926	-	0.965	0.862	0.862	-
Rp$_2$	FS2, FS4	0.941	0.793	0.943	-	0.976	0.768	0.977	-
Rp$_3$	FS1, FS3	0.918	0.837	0.919	-	0.968	0.873	0.873	-
Rp$_4$	FS4	0.910	0.750	0.914	-	0.972	0.759	0.759	-
Rp$_5$	FS5	0.900	0.728	0.910	-	0.956	0.731	0.731	-
Rp$_6$	FS6	0.926	0.869	0.926	-	0.967	0.859	0.859	-
Rp$_7$	FS1, FS7	0.913	0.822	0.913	-	0.958	0.823	0.823	-
Rcc$_1$	FS8	0.907	0.722	0.914	417	0.962	0.658	0.967	91
Rcc$_2$	FS8, FS9	**0.964**	**0.906**	**0.965**	**734**	**0.986**	**0.871**	**0.986**	**210**

Retrieval experiments corresponding to each of these definitions were conducted, filtering out candidate feedback terms which were predicted to be bad terms. For comparison, baseline experiments without feedback and with feedback and experiments with a perfect classification of terms (Oracle) have been performed, using the classification in the training data as a gold standard. Results for the retrieval experiments are given in Table 4, showing the number of relevant documents at 1000 retrieved documents (rel_ret), MAP, geometric mean average precision, and precision at 10 documents (P@10). For the baseline experiments with standard BRF, the number of documents assumed to be relevant was set to 20 (R=20). The number of feedback terms was varied from 5 to 50. Only the best performing baseline runs are shown (i.e. $T = 10$ or $T = 15$). Results which are significantly better than the IR baseline are indicated by a *, results significantly better than the best BRF baseline by a $^+$. Significance testing was performed using the Wilcoxon test with $p < 0.05$.

Experiments based on the simple combination of classification results showed a slightly worse performance in comparison to standard BRF. A possible explanation is that the accuracy of the classification results was not high enough.

5.1 IR Experiments for R-classification

The R-classification did not outperform standard BRF. There are only a few instances affecting the number of relevant and retrieved documents at all, and none was found which decreased this number for 1000 retrieved documents. While the classification accuracy might be high enough to show a significant change in the number of retrieved relevant documents, this effect has not even been observed when the perfect classification was used in IR experiments. An interesting observation can be made from the number of relevant documents retrieved for the English experiments, where the IR experiment with the optimal classifier (R_{opt}) returns less relevant documents than for experiments using the R-classification. The explanation for this effect is that

Table 4: Selected results for monolingual retrieval experiments on German and English GIRT documents. For comparison, results from a perfect classification (opt) are included.

Run	BRF terms	German Results				English Results			
		rel_ret	MAP	GMAP	P@10	rel_ret	MAP	GMAP	P@10
baseline	N/A	11257	0.3331	0.2174	0.6273	10829	0.3330	0.2050	0.5720
baseline	T=10, p,z,n	13241	0.3972*	0.2547	0.6367	10951	0.3410	0.1723	0.5807
baseline	T=15, p,z,n	13326	0.3979*	0.2564	0.6340	10944	0.3387	0.1717	0.5813
Pcc$_1$	p	12999	0.3898*	0.2734	0.6567	11214	0.3629^{*+}	0.2076	0.5967
Pcc$_1$	p,z	12998	0.3902*	0.2781	0.6573	11199	0.3633^{*+}	0.2073	0.5947
Pcc$_2$	p	13651	**0.4416**$^{*+}$	**0.3264**	**0.6973**	**11702**	**0.4084**$^{*+}$	**0.2605**	**0.6487**
Pcc$_2$	p,z	**13656**	0.4402^{*+}	0.3245	0.6927	11692	0.4079^{*+}	0.2579	**0.6487**
P$_{opt}$	p	14115	0.4922^{*+}	0.3905	0.7547	12034	0.4515^{*+}	0.3127	0.6987
P$_{opt}$	p,z	14100	0.4865^{*+}	0.3762	0.7500	12034	0.4484^{*+}	0.3057	0.6980
Rcc$_1$	p	12632	0.3664*	0.2391	0.6260	10814	0.3378	0.1911	0.5833
Rcc$_2$	p	**13149**	**0.3876***	**0.2665**	**0.6393**	**10940**	**0.3417**	**0.1930**	**0.5853**
R$_{opt}$	p	13297	0.3981*	0.2791	0.6600	10657	0.3367	0.1876	0.5720

while a single term might not contribute to the number of relevant documents, a set of feedback terms adds more context information to the query, which may result in higher ranked relevant documents. Thus, while terms may not individually be selective of relevant context, combinations of such terms may be highly effective. In summary, all term classification experiments aiming at increasing this number returned a lower number of relevant documents than most of the standard BRF approaches.

5.2 IR Experiments for P-classification

The best P-classification is based on the combined classifier (Pcc$_2$) and returns significantly better results than standard BRF. For the P-classification, even other performance metrics were slightly higher than for standard BRF experiments and for experiments with the R-classification.

There are only small differences in the results for experiments using terms of class p and experiments using both terms with p and z. Experiments using terms predicted not to affect precision (class z) did not affect any retrieval metric significantly compared to experiments using only class p. However, in some use cases e.g. for tasks such as machine translation in cross-lingual IR or in word sense disambiguation, these additional terms might provide useful context information.

Selecting only terms of class p for the perfect P-classification yields significantly better results than in any other experiment (0.4922 MAP). Interestingly, experiments with the perfect P-classification also return a higher number of relevant and retrieved documents than the perfect R-classification, which was meant to maximize this metric.

Improvements with BRF seem to be more difficult for the English GIRT data. The main reasons for this are that in BRF for German, more compounds, compound constituents, and (understemmed) morphologic variants are added as (good) feedback terms. English has a less rich morphology and thus, less improvement using BRF may be possible, because a so-called topic shift may be caused by unrelated terms.

Finally, the optimal P-classification shows that there is still room for performance gain if the classification can be further improved. For German, only 89.7% of the optimal MAP was achieved (90.4% for English).

6. CONCLUSION AND OUTLOOK

As a conclusion, attempts to answer the three research questions given in the introduction shall be presented. Few BRF terms (on their own) seem to affect IR performance at all. While no terms have been identified which decrease the number of relevant and retrieved documents, precision may degrade if a single term is added to the query. Hence, the distinction between terms which increase, decrease or do not change a performance metric is meaningful.

The classification of feedback terms into *good* and *bad* terms proves to be difficult, even when combining classifications on different feature sets. Using classifications from basic classifiers together with predictions on the correctness of the classifications as features increases accuracy considerably. Thus, an automatic classification of *good* and *bad* terms with reasonable accuracy is possible, but classification must be highly accurate to show significant improvements for IR.

The notion of BRF primarily being a recall-enhancing device should be carefully reconsidered. While the general usefulness of BRF may be conversely dicussed [1], traditional BRF has been found to improve IR effectiveness only slightly or only for certain topics. In contrast, the experiments in this paper support the findings reported by Cao, Nie and Robertson [3]: selecting feedback terms by their predicted impact on average precision is found to improve MAP significantly. It was also found that other performance metrics are improved as well.

The approach presented in this paper is not limited to BRF, but can also be used for relevance feedback, where user judgments are typically made on a per-document basis, but terms are extracted using similar heuristics as in BRF. Future work will include selecting feedback terms by taking the confidence score of classification into account and investigating the behaviour of this approach on a larger corpus (i.e. TREC ad hoc retrieval).

7. ACKNOWLEDGMENTS

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (CNGL) project.

8. REFERENCES

[1] B. Billerbeck and J. Zobel. Questioning query expansion: An examination of behaviour and parameters. In K.-D. Schewe and H. E. Williams, editors, *Proceedings of the Fifteenth Australasian Database Conference (ADC 2004)*, volume 27, pages 69–76, Dunedin, New Zealand, 2004. Australian Computer Society.

[2] C. Buckley, G. Salton, and J. Allan. The effect of adding relevance information in a relevance feedback environment. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on research and development in information retrieval, Dublin, Ireland*, pages 292–300, New York, NY, USA, 1994. Springer.

[3] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval, Singapore, Singapore*, pages 243–250, New York, NY, USA, 2008. ACM.

[4] Y. Cao, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon. Adapting ranking SVM to doument retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, Seattle, Washington, USA*, pages 186–193, New York, NY, USA, 2006. ACM.

[5] W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. TiMBL: Tilburg memory based learner, version 6.2, Reference Guide. Technical Report 09-01, ILK, 2004.

[6] F. Gey and V. Petras. Berkeley2 at GeoCLEF: Cross-language geographic information retrieval of German and English documents. In F. Gey, R. Larson, M. Sanderson, H. Joho, and P. Clough, editors, *Working Notes for the CLEF 2005 Workshop, 21–23 September, Vienna, Austria*, 2005.

[7] M. Kluck. The domain-specific track in CLEF 2004: Overview of the results and remarks on the assessment process. In C. Peters, P. Clough, J. Gonzalo, G. J. F. Jones, M. Kluck, and B. Magnini, editors, *Multilingual Information Access for Text, Speech and Images, 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004, Bath, UK, September 15-17, 2004, Revised Selected Papers*, volume 3491 of *Lecture Notes in Computer Science (LNCS)*, pages 260–270. Springer, Berlin, 2005.

[8] V. Lavrenko and B. W. Croft. Relevance-based language models. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval, New Orleans, Louisiana, United States*, pages 120–127, New York, NY, USA, 2001. ACM.

[9] J. Leveling. Exploring term selection for geographic blind feedback. In *Proceedings of GIR-2007, the 4th Workshop on Geographical Information Retrieval (hosted by CIKM 2007)*, Lisbon, Portugal, 2007.

[10] Y. Liu, N. V. Chawla, M. P. Harper, E. Shriberg, and A. Stolcke. A study in machine learning from imbalanced data for sentence boundary detection. *Computer Speech and Language*, 20(4):468–494, 2005.

[11] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[12] P. Ogilvie, E. Vorhees, and J. Callan. On the number of terms used in automatic query expansion. *Information Retrieval*, 12(6):666–679, 2009.

[13] S. E. Robertson. On term selection for query expansion. *Journal of Documentation*, 46(4):359–364, 1990.

[14] S. E. Robertson and K. Sparck-Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146, 1976.

[15] S. E. Robertson, S. Walker, and M. Beaulieu. Okapi at TREC-7: Automatic ad hoc, filtering, VLC and interactive track. In D. K. Harman, editor, *The Seventh Text REtrieval Conference (TREC-7)*, NIST Special Publication 500-242, pages 253–264, Gaithersburg, MD, USA, 1998. National Institute of Standards and Technology (NIST).

[16] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In D. K. Harman, editor, *Overview of the Third Text Retrieval Conference (TREC-3)*, pages 109–126, Gaithersburg, MD, USA, 1995. National Institute of Standards and Technology (NIST).

[17] H. van Halteren, J. Zavrel, and W. Daelemans. Improving accuracy in word class tagging through the combination of machine learning systems. *Computational Linguistics*, 27(2):199–229, 2001.