Combining and selecting characteristics of information use

Ian Ruthven[*1], Mounia Lalmas[2] and Keith van Rijsbergen[3]

[1]Department of Computer and Information Sciences, University of Strathclyde, Glasgow, G1 IXH
[2]Department of Computer Science, Queen Mary, University of London, London, E1 4NS
[3]Department of Computing Science, University of Glasgow, Glasgow, G12 8QQ

Ian.Ruthven@cis.strath.ac.uk, mounia@dcs.qmul.ac.uk, keith@dcs.gla.ac.uk

Abstract

In this paper we report on a series of experiments designed to investigate the combination of term and document weighting functions in Information Retrieval. We describe a series of weighting functions, each of which is based on how information is used within documents and collections, and use these weighting functions in two types of experiments: one based on combination of evidence for ad-hoc retrieval, the other based on selective combination of evidence within a relevance feedback situation. We discuss the difficulties involved in predicting good combinations of evidence for ad-hoc retrieval, and suggest the factors that may lead to the success or failure of combination. We also demonstrate how, in a relevance feedback situation, the relevance assessments can provide a good indication of how evidence should be selected for query term weighting. The use of relevance information to guide the combination process is shown to reduce the variability inherent in combination of evidence.

1 Introduction

Most relevance feedback (RF) algorithms attempt to bring a query closer to the user's information need by reweighting or modifying the terms in a query. The implicit assumption behind these algorithms is that we can find an optimal combination of weighted terms to describe the user's information need at the current stage in a search. However, relevance as a user judgement is not necessarily dictated only by the presence or absence of terms in a document. Rather relevance is a factor of what concepts the terms represent, the relations between these concepts, how users interpret them and how they relate to the information in the document. From studies such as those conducted by Barry and Schamber, [BS98], it is clear that current models of RF, although successful at improving recall-precision, are not very sophisticated in expressing what makes a document relevant to a user. Denos et al., [DBM97] for example, make the good point that although users can make explicit judgements on why documents are relevant, most systems cannot use this information to improve a search.

Not only are users' judgements affected by a variety of factors but they are based on the document text. RF algorithms, on the other hand, typically are based on a representation of a text and only consider frequency information or the presence or absence of terms in documents. These algorithms do not look deeper to see what it is about terms that indicate relevance; they ignore information on how the term is *used* within documents. For example a document may only be relevant if the terms appear in a certain context, if certain combinations of terms occur or if the main topic of the document is important. Extending feedback algorithms to incorporate the *usage* of a term within documents would not only allow more precise querying by the user but also allows relevance feedback algorithms to adapt more subtly to users' relevance judgements.

In this paper we investigate how incorporating more information on the usage of terms can improve retrieval effectiveness. We examine a series of term and document weighting functions in combination and in selective combination: selecting which characteristics of a term (e.g. frequency, context, distribution within documents) or document (complexity, ratio of useful information) should be used to retrieve documents. This research extends the initial study presented in [RL99] which demonstrated that a subset of the weighting function used in this paper were successful for precision enhancement. In particular we investigate the role of combination of evidence in RF.

[*] Corresponding author. This work was completed while the first author was at the University of Glasgow.

The following sections outline how we describe term and document characteristics (section 2), the data we used in our experiments (section 3), our experimental methodology (section 4), and the results of three sets of experiments. The first set of experiments examines each characteristic as a single retrieval function (section 5). The second set looks at combining evidence of term use in standard retrieval (section 6), and the third set examines selecting evidence in RF (sections 7 and 8). We summarise our findings in section 9.

2 Term and document characteristics

In this section we outline five alternative ways of describing term importance in a document or collection - or *term characteristics*. Three of these are standard term weighting functions, *idf*, *tf* and *noise*, another two are developed for this work.

• *inverse document frequency*, based on how often a term appears within a collection, described in section 2.1
• *noise*, also based on how often a term appears within a collection but based on within-document frequency, section 2.2
• *term frequency*, based on how often a term appears within a document, section 2.3
• *thematic nature*, or *theme*, based on how a term is distributed within a document, section 2.4
• *context*, based on the proximity of one query term to another query term within the same document, section 2.5

In addition, we introduce two *document* characteristics. These describe some aspect of a document that differentiates it from other documents.

• *specificity*, based on how many unique terms appear in a document, section 2.6.
• *information-noise*, based on the proportion of useful to non-useful content within a document, section 2.7

These characteristics were chosen to be representative of general weighting schemes – those that represent information on general term appearance, e.g. *idf*, *tf*, - and specific weighting schemes – those that represent specific features on how terms appear in documents, e.g. *theme*.

2.1 *idf*

Inverse document frequency, or *idf*, [SJ72] is a standard IR term weighting function that measures the infrequency, or rarity, of a term's occurrence within a document collection. The less likely a term is to appear in a collection the better is it likely to be at discriminating relevant from irrelevant documents. In these experiments we measure *idf* by the equation shown in equation 1.

$$idf\,(t) = \log\left(\frac{n}{N} + 1\right)$$

Equation 1: inverse document frequency (*idf*)
where *n* is the number of documents containing the indexing term *t*
and *N* is the number of documents in the collection

2.2 *noise*

The second term characteristic we investigated was the *noise* characteristic discussed in [Sal83, Har86], equation 2. The noise characteristic gives a measure of how important a term is within a collection but unlike *idf*, *noise* is based on within-document frequency.

$$noise_k\,(t) = \sum_{i=1}^{N} \frac{frequency_{ik}}{total\_frequency_k} \log \frac{total\_frequency_k}{frequency_{ik}}$$

Equation 2: *noise*
where *N* = number of documents in the collection,
$frequency_{ik}$ = the frequency of term *k* in document *i*,
$total\_frequency_k$ = total frequency of term *k* in the collection

From equation 2, if a term appears in only one document, it receives a *noise* score of zero. Terms that appear more commonly throughout a collection receive a higher *noise* value. The *noise* value is then inversely proportional to its discrimination power. The *noise* characteristic as defined here therefore requires normalisation, [Har86], to ensure that the *noise* value of a term reflects its discriminatory power. To normalise the *noise* score, we subtracted the *noise* score of a term from the maximum *noise* score.

The normalised *noise* characteristic gives a maximum *noise* score to a term if all its occurrences appear in one document and the lowest *noise* score if all occurrences of the term appear in different documents.

2.3 *tf*

Including information about how often a term occurs in a document - *term frequency* (*tf*) information - has often been shown to increase retrieval performance, e.g. [Har92]. For these experiment we used the following formula,

$$tf_d(t) = \log(occurrences_t(d) + 1) / \log(occurrences_{total}(d))$$

Equation 3: term frequency (*tf*)

where $occurrences_t(d)$ $occurrences_t(d)$ is the number of occurrences of term *t* in document *d*, $occurrences_{total}(d)$ is the total number of term occurrences in document *d*.

2.4 *theme*

Previous work by for example Hearst and Plaunt [HP93] and Paradis and Berrut [PB96], demonstrate that taking into account the topical or thematic nature of documents can improve retrieval effectiveness. Hearst and Plaunt present a method specifically for long documents, whereas Paradis and Berrut's method is based on precise conceptual indexing of documents.

We present a simple term-based alternative based on the distribution of term occurrences within a document. This is based on the assumption that the less evenly distributed the occurrences of a term are within a document, then the more likely the term is to correspond to a localised discussion in the document, e.g. a topic in one section of the document only, Figure 1 left-hand side. Conversely, if the term’s occurrences are more evenly spread throughout the document, then we may assume that the term is somehow related to the main topic of the document, Figure 1 right-hand side. Unlike Hearst and Plaunt we do not split the document into topics and assign a sub- or main-topic classification. Instead we define a *theme* value of a term, which is based on the likelihood of a term to be a main topic.

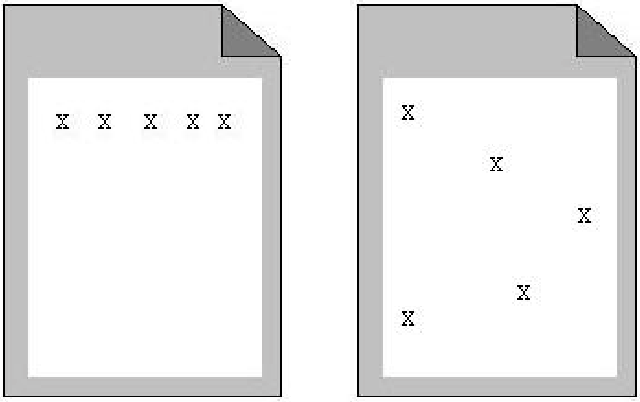

Figure 1: Localised discussion of term X (left-hand side), general discussion of term X (right-hand side)

The algorithm which we developed for this is shown in equation 4. This value is based on the difference between the position of each occurrence of a term and the *expected* positions. Table 1 gives a short example for a document with 1000 words, and five occurrences of term *t*. First, we calculate whether the first occurrence of term *t* occurs further into the document that we would expect, based on the expected distribution ($first_d(t)$ - line two, equation 1; Column 7, Table 1). Next we calculate whether the last occurrence of the term appears further from the end of the document than we would expect ($last_d(t)$ - line two, equation 1; Column 8, Table 1). For the remainder of the terms we calculate the difference between the expected position of a term, based on the actual position of the last occurrence and the expected difference between two occurrences (– line two; Column 4-6, Table 1) ($\sum_{i=2}^{n-1} \left| predicted\_position_i(t) - actual\_position_i(t) \right|$).

$$theme_d(t) = (length_d - difference_d(t)) / length_d$$

where

$$difference_d(t) = first_d(t) + last_d(t) + \sum_{i=2}^{n-1} \left| predicted\_position_i(t) - actual\_position_i(t) \right|$$

$$\begin{aligned} first_d(t) &= 0, \quad if \quad actual\_position_1(t) \le distribution_d(t) \\ &= actual\_position_1(t) - distribution_d(t), \quad ow \end{aligned}$$

$$\begin{aligned} last_d(t) &= 0, \quad if \quad (length_d - actual\_position_n(t) \le distribution_d(t)) \\ &= length_d - (actual\_position_n + distribution_d(t)), \quad ow \\ predicted\_position_i &= actual\_position_{i-1} + distribution_d(t) \\ distribution_d(t) &= length_d / occurrences_d(t) \end{aligned}$$

Equation 4: *theme* characteristic
where *distribution*$_d$(*t*) is the expected distribution of term *t* in document *d*,
assuming all occurrences of *t* are equally distributed,
predicted_position$_i$ is the expected position of the *i*th occurrence of term *t*,
actual_position$_i$ is the actual position of the *i*th occurrence.
occurrences$_d$(*t*) is the number of occurrences of term *t* in document *d*.
n is the number of query terms in the document.

length	occs	distr	epos	pos	diff	first	last	difference	*theme*
1000	5	200	-	100		0			
			300	500	200				
			700	551	349				
			751	553	547				
			753	700	600				
			900				100		
					600	0	100	700	0.3

Table 1: Example calculation of *theme* value for a term

We then sum these values to get a measure of the difference between the expected position of the term occurrences and their actual positions. The greater the difference between where term occurrences appear and where we would expect them to appear, the smaller the *theme* value for the term. The smaller the difference, the larger the *theme* value for the term.

2.5 *context*

There are various ways in which one might incorporate information about the context of a query term. For example, we might rely on coocurrence information [VRHP81], information about phrases [Lew92], or information about the logical structures, e.g. sentences, in which the term appears [TS98]. We defined the importance of context to a query term as being measured by its distance from the nearest query term, relative to the average expected distribution of all query terms in the document.

$$contextd(t) = (distribution_d(q) - \min_d(t)) / distribution_d(q)$$
$$\min_d(t) = \min_{t \neq t'} \left|(position_d(t) - position_d(t'))\right|$$
$$distribution_d(q) = length_d \,/\, occurrences_d(q)$$

Equation 5: *context* characteristic for term *t* in document *d*
where *distribution*$_d$(*q*) is the expected distribution of all query terms in the document,
assuming terms are distributed equally
position$_d$(*t*) is the position of term *t* and *min*$_d$(*t*) is the minimum difference
from any occurrence of term *t* to another, different query term.

2.6 *specificity*

The first document characteristic we propose is the *specificity* characteristic which is related to *idf*. The *idf* characteristic measures the infrequency of a term's occurrence within a document collection; the less likely a term is to appear in a document the better is it likely to be at discriminating relevant from irrelevant documents. However, *idf* does not consider the relative discriminatory power of other terms in the document.

If a document contains a higher proportion of terms with a high *idf*, it may be more difficult to read, e.g. if it contains a lot of technical terms. On the other hand a document containing a lot of terms with very low *idf* values may contain too few information-bearing words. We propose the *specificity* characteristic as a measure of how specialised a document's contents are, relative to the other documents in the collection. This is a very simple measure as we do not take into account the domain of the document or external knowledge sources, which would allow us to represent the complexity of the document based on its semantic content.

The *specificity* characteristic is a document characteristic, giving a score to an entire document rather than individual terms. It is measured by the sum of the *idf* values of each term in the document, divided by the number of terms in the document, giving an average *idf* value for the document, equation 6.

$$specificity(d) = \frac{\sum_{i \in d}^{n} idf(i)}{n}$$

Equation 6: *specificity* document characteristic of document *d*
where *n* = number of terms in document *d*

2.7 *information-to-noise*

The *specificity* characteristic measured the complexity of the document based on *idf* values. An alternative measure is the *information-to-noise* ratio, [ZG00], abbreviated to *info-noise*. This is calculated as the number of tokens after processing (stemming and stopping) of the document divided by the length of the document before stopping and stemming, equation 7.

$$info\_noise(d) = \frac{processed\_length(d)}{length(d)}$$

Equation 7: *info_noise* document characteristic of document *d*
where *processed_length*(*d*) = number of terms in document *d* after stopping and stemming
length(*d*) = number of terms in document *d* before stopping and stemming

info_noise, as described in [ZG00], measures the proportion of useful to non-useful information content within a document.

2.8 Summary

The *idf* and *noise* characteristics give values to a term depending on its importance within a collection, the *tf* and *theme* characteristics give values depending on the term's importance within an individual document and the *specificity* and *info_noise* characteristics give values to individual documents based on their content. The *context* characteristic gives a value to a term based on its proximity to another query term in the same document. Each of the term characteristics can be used to differentiate documents based on how terms are used within the documents and the document characteristics allow differentiation of documents based on their content. The document characteristics also allow RF algorithms to base feedback decisions on the document taken as a whole, rather than only individual components of the document.

Each of the algorithms that calculate the characteristic values give scores in different ranges. In our experiments we scaled all values of the characteristics to fall within the same range, 0 - 50, to ensure that we were working with comparable values for each characteristic. In the next section we outline the data we used in our experiments.

3 Data

For the experiments reported in this paper we used two sets of collections. The first is a set of three small test collections (**CACM**, **CISI** and **MEDLARS** collections[1]), the second is a set of two larger collections (the Associated Press (1988) (**AP**) and the Wall Street Journal (1990-92) (**WSJ**) collection from the TREC initiative [VH96]). Statistics of these collections are given in Table 2.

	CACM	**CISI**	**MEDLARS**	**AP**	**WSJ**
Number of documents	3204	1460	1033	79 919	74 520
Number of queries used[2]	52	76	30	48	45
Average document length[3]	47.36	75.4	89	284	326
Average words per query[4]	11.88	27.27	10.4	3.04	3.04
Average relevant documents per query	15.3	41	23	35	24
Number of unique terms in the collection	7861	7156	9397	129 240	123 852

Table 2: Details of CACM, CISI, MEDLARS, AP and WSJ collections

The AP and WSJ test collections each come with fifty so-called TREC topics. Each topic describes an information need and those criteria that were used in assessing relevance when the test collection was created. A TREC topic has a number of sections, (see Figure 2 for an example topic). In our experiments we only used the short **Title** section from topics 251 – 300 as queries, as using any more of the topic description may be an unrealistic as a user query.

> **Number**: 301
> **Title**: International Organized Crime
> **Description**:
> Identify organisations that participate in international criminal activity, the activity, and, if possible, collaborating organisations and the countries involved.
> **Narrative**:
> A relevant document must as a minimum identify the organisation and the type of illegal activity (e.g., Columbian cartel exporting cocaine). Vague references to international drug trade without identification of the organisation(s) involved would not be relevant.

Figure 2: Example of a TREC topic

Stopwords were removed, using the stopword list in [VR79], and the collections were stemmed using the Porter stemming algorithm [Por80].

4 Outline of experiments

In this paper we describe three sets of experiments:

[1] *http://www.dcs.gla.ac.uk/idom/ir_resources/test_collections/*

[2] Each collection comes with a number of queries. However, for some queries there are no relevant documents in the collection. As these queries cannot be used to calculate recall-precision figures they are not used in these experiments. This row shows the number of queries, for each collection, for which there is at least one relevant document.

[3] After the application of stemming and stopword removal.

[4] This row shows the average length of the queries that were used in the experiments.

i. *retrieval by single characteristic*. In section 5 we present results obtained by running each characteristic as a single retrieval function. In this section we examine the relative performance of each characteristics on the test collections, and discuss why some characteristics perform better than others as retrieval functions.

ii. *retrieval by combination of characteristics*. In section 6 we investigate whether combining characteristics can improve retrieval effectiveness over retrieval by single characteristic. We also discuss factors that affect the success of combination, such as the size of the combination and which characteristics are combined.

iii. *relevance feedback*. In section 7 we investigate how we can use relevance assessments to *select* good combinations of characteristics of terms and documents to use for relevance feedback. We describe several methods of selecting which characteristics are important for a query and compare these methods against methods that do not use selection of characteristics.

5 Retrieval by single characteristic

In this section we examine the performance of running each characteristic (term and document characteristics) as a single retrieval function (retrieval by the sum of the *idf* value of each query term, retrieval by the sum of *tf* values of each query term, etc.). The results are presented in section 5.2 but before this, in section 5.1, we look at how document characteristics should be used to score documents.

5.1 Document characteristics - initial investigations

As the *specificity* and *info-noise* characteristics are document rather than term characteristics, they give a value to each document irrespective of which terms are in the query. However, we can use the document characteristics to produce different rankings based on two criteria:

i. *which documents receive a score*. Although all documents have a value for the *specificity* and *info-noise* characteristics, we may choose to score only those documents that contain at least one query term, as these documents are those that are the most likely to be relevant. To investigate this, we assessed two methods of retrieving documents - the *query dependent* - and the *query independent* strategies.

In the query independent strategy the retrieval score of a document is the characteristic score (*info_noise* or *specificity*). This method gives an identical ranking of documents for all queries. In the query dependent strategy the retrieval score of a document is also the characteristic score but this score is only assigned to those documents that contain at least one query term. If the document contains no query terms then the retrieval score is zero. In this method we, then, retrieve all documents that contain a query term before the documents that contain no query terms, giving a different ranking to each query.

ii. *how to order the documents*. The *specificity* characteristic gives high scores to more complex documents, whereas the *info_noise* characteristic gives high scores to documents that have a high proportion of useful information. This means that we are asserting that relevant documents are more likely to have a higher amount of useful information or a higher complexity. This requires testing. We tested two strategies - *standard* - in which we rank documents in decreasing order of characteristic score and *reverse* - in which we rank documents in increasing order of characteristic score.

These two criteria give us four combinations of strategy - query dependent and standard, query independent and standard, query dependent and reverse, query dependent and reverse. Each of these strategies correspond to a different method of ranking documents.

The results of these ranking strategies are shown in Table 3 for the *specificity* characteristic and Table 4 for the *info_noise* characteristic[5]. Also shown in each table, for comparison, are the results of two random retrieval runs on each collection[6]. These are also based on a query dependent strategy (random order of all documents containing a query term, followed by random order of the remaining documents) and a query independent strategy (a completely random ordering of all documents).

[5] Full recall-precision tables for all experiments are given in an electronic appendix, available at http://www.cs.strath.ac.uk/~ir/papers/AppendixAPart1.pdf. The corresponding tables will be noted as footnotes throughout the paper. Appendix A, Tables A.3 – A.13.

[6] Tables A.1 and A.2.

	standard *specificity*		reverse *specificity*		random	
Collection	**query dependent**	**query independent**	**query dependent**	**query independent**	**query dependent**	**query independent**
CACM	**1.19**	0.98	**1.19**	1.18	1.14	0.36
CISI	**10.55**	2.83	2.75	3.51	4.66	3.86
MEDLARS	4.62	3.33	4.62	4.48	**12.39**	4.82
AP	0.33	0.06	**0.47**	0.05	0.28	0.05
WSJ	0.42	0.10	**0.57**	0.02	0.35	0.04

Table 3: Average precision figures for *specificity* characteristic
Highest average precision figures for each collection are shown in bold

From Table 3, the *specificity* characteristic is best applied using a query dependent strategy. Whether or not it is applied in decreasing order of characteristic value (standard), or increasing order of characteristic score (reverse) is collection dependent. However the overall preference is for the reverse strategy.

From Table 4 the *info_noise* characteristic is best applied using the query-dependent standard strategy: ordering documents containing a query term and with the highest proportion of useful information at the top of the ranking.

	standard *info_noise*		reverse *info_noise*		random	
Collection	**query dependent**	**query independent**	**query dependent**	**query independent**	**query dependent**	**query independent**
CACM	**1.67**	0.50	0.86	1.63	1.14	0.36
CISI	4.08	3.28	3.48	2.78	**4.66**	3.86
MEDLARS	8.67	2.56	8.25	2.98	**12.39**	4.82
AP	**0.44**	0.05	0.29	0.05	0.28	0.05
WSJ	**0.48**	0.03	0.32	0.03	0.35	0.04

Table 4: Average precision figures for *info_noise* characteristic
Highest average precision figures for each collection are shown in bold

On all collections, except the MEDLARS collection, at least one method of applying the characteristics gave better performance than random (query independent), and with the exception of MEDLARS and CISI also performed better than the query dependent random run. One possible reason for the poorer results on these collections is that the range of document characteristic values for these collections is not very wide. Consequently the characteristics do not have enough information to discriminate between documents.

It is better to rank only those documents that contain a query term than all documents. This is not surprising as, using the query dependent strategy, we are in fact re-ranking the basic *idf* ranking for each query. We shall discuss the relative performance of the document characteristics against the term characteristics in the next section. Although the document characteristics do not give better results than the term characteristics (see next section), they do generally give better results than the random retrieval runs and can be used in combination to aid retrieval.

5.2 Single retrieval on all characteristics

The results from running each characteristic as a single retrieval function are summarised in Table 5[7], measured against the query dependent random strategy. This is used as a baseline for this experiment as all the characteristics prioritise retrieval of documents that contain a query term over those documents that contain no query terms. Hence this method of running a random retrieval is more similar in nature to the term characteristics and, as it gives higher average precision, provides a stricter baseline measure for comparison.

The majority of characteristics outperform the query dependent random retrieval baseline. However some characteristics do perform more poorly than a random retrieval of the documents (*info_noise* on CISI, *theme*, *specificity* and *info_noise* on MEDLARS, *context* on WSJ)[8].

[7] Tables A.14 – A. 18.

[8] All characteristics, for all collections except MEDLARS, outperformed a completely random retrieval.

	Characteristic							
Collection	***idf***	***tf***	***theme***	***context***	***specificity***	***noise***	***info-noise***	**random**
CACM	22.00	22.70	4.36	14.80	1.19	**24.15**	1.67	1.14
CISI	11.50	**12.50**	5.10	9.60	10.55	11.00	4.08	4.66
MEDLARS	43.10	43.70	11.10	36.10	4.60	**43.90**	8.80	12.39
AP	**10.10**	9.86	4.63	9.57	0.47	1.00	0.44	0.28
WSJ	**12.19**	7.39	1.00	0.04	0.42	1.05	0.48	0.38

Table 5: Average precision figures for term and document characteristics used as single retrieval functions
Highest average precision figures for each collection are shown in bold

The order in which the characteristics performed is shown in Figure 3 where > indicates statistical significance and >= indicates non-statistical significance.[9]

The document characteristics perform quite poorly as they are insensitive to query terms. That is, although, when using these characteristics we score only documents that contain a query term, the document characteristics do not distinguish between documents that contain good query terms and documents that contain poor query terms.

CACM	*noise*	>=	*tf*	>=	*idf*	>	*context*	>=	*theme*	>	*inf_n*	>	*spec*	>	*random*		
CISI	*tf*	>	*idf*	>	*noise*	>	*spec*	>	*context*	>	*theme*	>	*random*	>=	*inf_n*		
MEDLARS	*noise*	>=	*tf*	>=	*idf*	>	*context*	>	*random*	>	*theme*	>	*inf_n*	>	*spec*		
AP	*idf*	>	*tf*	>=	*context*	>	*theme*	>	*noise*	>	*spec*	>=	*inf_n*	>	*random*		
WSJ	*idf*	>	*tf*	>	*noise*	>=	*theme*	>=	*spec*	>=	*inf_n*	>	*random*	>	*context*		

Figure 3: Statistical and non-statistical differences between characteristics on all collections
where *spec* = specificity, *inf_n* = info_noise

On nearly all collections the standard characteristics (*idf*, *tf*, *noise*[10]) outperformed our new characteristics. One possible reason for this is that, although, the new term characteristics (*theme*, *context*) give a weight to every term in a document, unlike the standard characteristics they do not always give a non-zero weight. The *context* characteristic, for example, will only assign a weight to a term if at least two query terms appear in the same document. In the case of the two larger collections we have relatively smaller queries. Hence the co-occurrence of query terms within a document may be low with the resulting effect that most terms have a zero weight for this characteristic. This, in turn, will lead to a poor retrieval result as the characteristic cannot distinguish well between relevant and non-relevant documents.

Similarly, the *theme* characteristic, as implemented here, will also lead a high proportion of terms being assigned a zero weight compared with the *tf* characteristic. One reason for this is that *theme* assigns a zero weight to a term if it only appears once within a document. A collection such as the MEDLARS collection, which has a high number of terms that only appear in one document may be more susceptible to this, as it contains a large number of unique terms.

The standard characteristics are also less *strict* algorithms: the information they represent, e.g. frequency of a term within a document, is more general than that represented by the new characteristics. This will mean that the standard characteristics will be useful for a wider range of queries. For example, *tf* will be a useful characteristic for most query terms as, generally, the more often a query term appears within a document, the more likely the document is to be relevant. The *theme* characteristic, on the other hand, will only be useful for those queries where the query terms are related to the main topic of the document. For queries where this condition is not met, the *theme* characteristic will not be useful.

Even though the new characteristics do not perform as well as the traditional weighting functions they do improve retrieval effectiveness over random retrieval. These algorithms should not be seen as alternative weighting schemes but as *additional* ones: ones that provide additional methods of discriminating relevant from non-relevant material. In RF these additional characteristics will be used to score query terms if they are useful at indicating relevant documents for individual queries. That is, by providing evidence of different aspects of

[9] Calculated using a paired t-test, $p < 0.05$, holding recall fixed and varying precision

[10] Harman's, [Har86], experimental investigation of the *noise* term weighting function on the Cranfield collection showed superior results for *noise* over *idf*. In these experiments, this held for the shorter CACM and MEDLARS collection. However in the larger collections, the *noise* characteristic performed relatively poorly.

information use, they can be used to help retrieval performance in combination with other characteristics. This combination of evidence is the subject of the next section.

6 Retrieval by combination of characteristics

In the previous section we looked at the performance of each characteristic individually. In this section we look at whether the retrieval effectiveness of characteristics will be improved if we use them in combination.

Belkin et al., [BKFS95], examined the role of multiple query representations in ad-hoc IR. Their argument in favour of different representations of queries is twofold:

i. empirical evidence that different retrieval functions retrieve different documents, e.g. [Lee98]. In our approach combinations of different representations of query terms can retrieve documents that fulfil different criteria for relevance.

ii. theoretical: different query representations can provide different interpretations of a user's underlying information need. This has a strong connection to Ingwersen's work on polyrepresentation - multiple representations of the same object, in our case a query, drawn from different perspectives can provide better insight into what constitutes relevance than a single good representation, [Ing94].

In this experiment we tested all possible combinations of the characteristics, running each possible combination as a retrieval algorithm. For each collection, we effectively run the powerset of combinations, each set comprising a different combination of characteristics. For each combination, the retrieval score of a document was given by sum of the score of each characteristic of each query term that occurred in the document. For example, for the combination of *tf* and *theme*, the score of a document was equal to the sum of the *tf* value of each query term plus the sum of the *theme* value of each query term.

Two versions of this experiment were run, the first used the values of characteristics given at indexing time, the second treated the characteristics as being more or less important than each other. There are several reasons for this. For example, some characteristics may reflect aspects of information use that are more easily measured than another; others are better as retrieval functions and should be treated as being more important. We incorporate this by introducing a set of scaling weights (*idf* 1, *tf* 0.75, *theme* 0.15, *context* 0.5, *noise* 0.1, *specificity* and *information_noise* 0.1[11]) that are used to alter the weight given to a term at indexing time. Each indexing weight of a term characteristic is multiplied by the corresponding scaling weight, e.g. all *tf* values are multiplied by 0.75, all *theme* values by 0.15, etc.

This gives us two conditions - *weighting* and *non-weighting* of characteristics - for each combination of characteristics.

In the following sections we shall summarise our findings regarding three aspects: the effect on retrieval effectiveness of combining characteristics, the effect of weighting characteristics, and the effect of adding individual characteristics to other combinations. Each of these will be discussed in a separate section in sections 6.1 - 6.3. We shall summarise in section 6.4[12].

6.1 Effecting of combining characteristics

Our experimental hypothesis is that combining characteristics can increase retrieval effectiveness over using individual characteristics. In section 6.3 we shall discuss how well the individual characteristics performed in combination. In this section we shall examine the basic hypothesis and discuss general findings.

In Table 6 we outline the effect on individual characteristic performance by the addition of other characteristics. Of the 127 possible combinations of characteristics for each collection, each characteristic appeared in 63 combinations. Each row is a count of how many of these 63 combinations containing each characteristic had higher average precision (*increase*) than the characteristic as a single retrieval function, lower average precision (*decrease*), or no change in average precision (*none*). For example, how many combinations containing *idf* gave an average precision figure that was better, worse or identical to the average precision of *idf* alone?

The first general conclusion from Table 6 is that all characteristics can benefit from combination with another characteristic or set of characteristics. Furthermore, with the exception of the *noise* characteristic on the CACM,

[11] These weights were derived from experiments using a sample of the data from each collection.

[12] Tables A.19 – A.70.

and the *tf* and *idf* characteristics on the CISI, any characteristic was more likely to benefit from combination than be harmed by it. This conclusion held under both the weighing and non-weighting conditions.

The second general conclusion is that the performance of a characteristic as a single retrieval function (section 4.2) is a good indicator of how well the characteristic will perform in combination.

The poorer the characteristic is at retrieving relevant documents the more likely it is to benefit from combination with another characteristic. For each collection, on the whole, the poorer characteristics[13] improve more often in combination with other characteristics. The reverse also holds: if a characteristic is good as a single retrieval function, then there is less chance that it will be improved in combination. For example the best characteristics in the small collections (*tf*, *idf* on CISI, and *noise* on CACM) showed the lowest overall improvement in combination. However the overall tendency is beneficial: combination benefits more characteristics than it harms.

Collection	**Condition**	**Change**	***idf***	***tf***	***theme***	***context***	***spec***	***noise***	***info_ noise***
CACM	NW	increase	**54**	**41**	**63**	**63**	**62**	15	**62**
		decrease	9	22	0	0	0	**48**	0
		none	0	0	0	0	1	0	1
	W	increase	**50**	**42**	**63**	**63**	**62**	11	**62**
		decrease	8	18	0	0	0	**52**	0
		none	5	3	0	0	1	0	1
CISI	NW	increase	27	1	**63**	**63**	**49**	**39**	**63**
		decrease	**35**	**62**	0	0	14	24	0
		none	1	0	0	0	0	0	0
	W	increase	23	7	**63**	**63**	**52**	**40**	**63**
		decrease	**34**	**53**	0	0	0	23	0
		none	6	3	0	0	11	0	0
MEDLARS	NW	increase	**47**	**44**	**63**	**63**	**63**	**43**	**63**
		decrease	16	19	0	0	0	20	0
		none	0	0	0	0	0	0	0
	W	increase	**45**	**55**	**63**	**60**	**63**	**37**	**63**
		decrease	18	8	0	3	0	26	0
		none	0	0	0	0	0	0	0
AP	NW	increase	**47**	**55**	**63**	**59**	**62**	**62**	**62**
		decrease	16	8	0	4	1	1	1
		none	0	0	0	0	0	0	0
	W	increase	**54**	**60**	**62**	**61**	**63**	**60**	**63**
		decrease	4	0	3	0	0	0	0
		none	5	3	0	2	0	3	0
WSJ	NW	increase	**40**	**63**	**63**	**63**	**63**	**63**	**63**
		decrease	22	0	0	0	0	0	0
		none	1	0	0	0	0	0	0
	W	increase	**46**	**63**	**63**	**63**	**63**	**60**	**63**
		decrease	8	0	0	0	0	3	0
		none	9	0	0	0	0	0	0

Table 6: Effect of combination on individual characteristics
where *increase* = increase in average precision when combined, *decrease* = decrease in average precision when in combination, *none* = no difference in average precision when in combination,
NW = non-weighting condition, *W* = weighting condition
Bold figures indicate the predominant effect of the characteristic in combination

In the remainder of this section we look at what affects the success of combination. In particular, we look examine the size of combinations and the components of combinations.

[13] These were the *theme*, *context*, *specificity* and *info_noise* for the CACM, CISI and MEDLARS collections and *theme*, *context*, *noise*, *specificity* and *info_noise* for the AP and WSJ collections.

In Table 7 we analyse the success of combination by *size* of combination, that is how many characteristics were combined. For each condition, weighting and non-weighting, on each collection we ranked all combinations by average precision[14]. We then took the median[15] value and the size of the combinations that appeared above and below this point. In Table 7 bold figures indicate where most combinations, of a given size, appeared (above or below the median point).

In the majority of cases the larger combinations (combinations of 4-7 characteristics) performed better than the median value, and the smaller combinations (combinations of 1-3 characteristics) performed worse than the median. There was little difference between the weighting and non-weighting conditions.

One possible reason for the success of the larger combinations is that poor characteristics have a lower overall effect in a larger combination. That is, if we only combine two characteristics and one of these is a poor characteristic, then there is a greater chance that the combination will perform less well than the better individual characteristic. Conversely, if we combine a number of characteristics, and one is poorer than the rest, then this will not have such a great effect on the performance of the combination.

Collection	**Position**	**Condition**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
CACM	Above	NW	2	5	12	**20**	**17**	**7**	**1**
		W	2	6	13	**21**	**15**	**6**	**1**
	Below	NW	5	16	23	15	4	0	0
		W	5	15	22	14	6	0	0
CISI	Above	NW	2	7	**19**	**21**	**15**	0	**1**
		W	2	9	17	**22**	**11**	**2**	**1**
	Below	NW	5	14	16	14	6	7	0
		W	5	12	18	13	10	5	0
MEDLARS	Above	NW	0	5	15	**24**	**13**	**6**	1
		W	0	7	18	13	**18**	**7**	**1**
	Below	NW	7	16	20	11	8	1	0
		W	7	14	18	22	3	0	0
AP	Above	NW	0	7	11	**20**	**18**	**7**	**1**
		W	0	3	11	**23**	**19**	**7**	**1**
	Below	NW	7	14	24	15	3	0	0
		W	7	18	24	12	2	0	0
WSJ	Above	NW	1	5	13	**21**	**17**	**7**	**1**
		W	0	3	12	**23**	**18**	**7**	**1**
	Below	NW	7	16	22	14	4	0	0
		W	7	18	23	12	3	0	0

Table 7: Distribution of combinations over ranking of average precision where *Above* = combination falls above or at median point of ranking, *Below* = combination falls below median point of ranking, *NW* = non-weighting condition, *W* = weighting condition

A further reason for larger combinations performing more effectively is that they allow for a more *distinct* ranking. That is, the more methods we have of scoring documents, the less chance that documents will receive an equal retrieval score.

Now we look at how the components of the combinations affect the success of combining characteristics. As stated before, each characteristic appeared in a total of 63 combinations. Table 8 presents how many of these combinations appeared above the median combination in the ranking of average precision, i.e. how many times a combination containing a characteristic performed better than average. The better individual characteristics, e.g. *idf* and *tf*, appeared in more combinations above the median than below for all collections. The poorer characteristics, e.g. *info_noise*, tended to appear in more combinations below the median than above.

[14] Tables A.131 – A.141, in http://www.cs.strath.ac.uk/~ir/papers/AppendixAPart2.pdf

[15] For each collection, in each condition, there were 127 possible combinations, the median point was taken to be the 64th combination in the ranking of all combinations.

	CACM	CACM	CISI	CISI	MEDLARS	MEDLARS	AP	AP	WSJ	WSJ
	NW	**W**	**NW**	**W**	**NW**	**W**	**NW**	**W**	**NW**	**W**
idf	**42**	**41**	**38**	**43**	**41**	**40**	**39**	**43**	**41**	**46**
tf	**47**	**52**	**41**	**44**	**42**	**50**	**51**	**47**	**52**	**47**
theme	**33**	**32**	**44**	**38**	**48**	**42**	**30**	**41**	**32**	**41**
context	29	30	20	16	28	28	**41**	**45**	**44**	**42**
spec	30	**32**	30	**32**	31	**33**	**37**	**32**	**32**	**33**
noise	**49**	**50**	27	29	**41**	**37**	**36**	**36**	**32**	**34**
inf	**32**	**32**	**32**	31	28	31	**32**	31	**34**	30

Table 8: Number of appearances of a characteristic in a combination appearing above median combination
Bold figures indicate where the majority of the combinations containing an individual characteristic appeared above the median value.

This is not necessarily to say, however, that poor characteristics always decrease the performance of a combination, see section 6.4 for example. Often a characteristic that performs less well as a single characteristic can improve a combination. What is important is how well a combination of characteristics separates relevant from irrelevant documents for an individual query: a particular combination may work poorly on average but work well for certain queries. This is important for our RF experiments, in which we select which are good characteristics for individual queries, section 7.

To summarise our findings: combinations of characteristics, whether weighted or not, is beneficial for all characteristics on all collections tested. This benefit is greater when the characteristic is poor as a single retrieval function but the overall benefits of combination still holds for good characteristics. The larger combinations (4-7 characteristics) tend to be better than small (1-3 characteristics) as retrieval functions over the collections.

6.2 Effect of weighting characteristics

Our basis behind weighting characteristics was that some characteristics may be better at indicating relevance than others. In Table 9 we summarise the effect of weighting on each collection, indicating the number of combinations that increased/decreased in average precision when using weighting. Overall, 47% of combinations improved using weighting on CACM collection, 61% on CISI, 60% MEDLARS, 69% on AP and 66% on WSJ.

As can be seen for all collections, except CACM, weighting was beneficial in that it improved the average precision of more combinations than it decreased. Generally these improvements were statistically significant.

Collection	**Increase**		**Decrease**	
	Significant	**Non-significant**	**Significant**	**Non-significant**
CACM	24 *20%*	32 *27%*	31 *26%*	**33** *28%*
CISI	**59** *49%*	14 *12%*	37 *31%*	10 *8%*
MEDLARS	**45** *38%*	27 *23%*	23 *19%*	25 *21%*
AP	**51** *43%*	32 *27%*	22 *18%*	15 *13%*
WSJ	**67** *56%*	12 *10%*	26 *22%*	15 *13%*

Table 9: Effect of weighting on combination performance
Significant = statistically significant change, **Non-significant** = non statistically significant change
Bold figures indicate predominant effect of weighting on each collection

Table 10 breaks down these figures by size of combination, the number of characteristics in the combination. The combination that benefited most from weighting were also these tended to be the ones that performed best in combination, i.e. those combination of four or greater characteristics.

Collection	Change	2	3	4	5	6	7
CACM	**Increase**	8	14	17	**12**	**4**	0
	Decrease	13	21	18	9	3	**1**
CISI	**Increase**	9	**22**	**24**	**11**	**7**	**1**
	Decrease	12	13	11	10	0	0
MEDLARS	**Increase**	9	**19**	**23**	**14**	**6**	0
	Decrease	12	16	12	7	1	**1**
AP	**Increase**	8	**21**	**27**	7	1	**1**
	Decrease	13	14	8	19	6	0
WSJ	**Increase**	8	**19**	**25**	**19**	**7**	**1**
	Decrease	13	16	10	2	0	0

Table 10: Effect of weighting by size of combination
Bold figures indicate predominant effect on each size of combination

In Table 11, we analyse which characteristics appeared in the combinations that did better using weighting than no weighting. Generally, combinations containing *idf* and *tf* were helped by weighting across the collections and *theme* and *context* were helped in the larger collection. The only characteristic to be consistently harmed by weighting was the *noise* characteristic.

	idf	***tf***	***theme***	***context***	***spec***	***noise***	***info_noise***
CACM	**36** 64%	**42** 75%	**34** 61%	23 41%	**33** 59%	18 32%	26 46%
CISI	**46** 63%	**49** 67%	27 37%	32 44%	**42** 58%	21 29%	**38** 52%
MEDLARS	**43** 60%	**40** 56%	29 40%	35 49%	**46** 64%	9 13%	**48** 67%
AP	**52** 63%	**46** 55%	**55** 66%	**45** 54%	40 48%	15 18%	**48** 58%
WSJ	**54** 68%	**45** 57%	**49** 62%	**45** 57%	39 49%	20 25%	39 49%

Table 11: Appearance of individual characteristics in combinations that were improved by weighting
Bold figures indicate those characteristics for which weighting was beneficial overall.

Weighting is generally beneficial but it is important to get good values for the characteristics. For example, both *idf* and *tf* were good individual retrieval algorithms and were highly weighted which helped their performance in combination as the combination was more heavily biased towards the ranking given by these characteristics.

noise, on the other hand, was a variable retrieval algorithm in that it performed well on some collections and more poorly on others. As it was weighted lowly the overall effect of *noise* in combination was lessened in the weighting condition. Consequently in cases where *noise* would have been a good individual retrieval algorithm the combination did not perform as well as it might have without weighting.

A final observation is that although weighting did not generally improve the best combination for the collections[16], it did tend to improve the performance of the middle ranking combinations significantly. These were the combinations that appeared in the middle of the ranking of combinations described in section 6.1.

Weighting then was a success in that it improved the performance of most combinations. However it achieved this by decreasing the performance of the poorer combinations and increasing the performance of the average combinations.

[16] Tables A.131 – A.141

6.3 Effect of adding individual characteristics

In section 6.1 we gave general conclusions about the effect of combining characteristics. In this section we look more closely at the effect of combining individual characteristics and the effect of characteristics on the performance of a combination of characteristics. In Table 12 we summarise the effect of adding a characteristic to other combinations, e.g. adding *idf* to the 63 combinations that did not already contain *idf*. We measure whether the new information causes an increase in average precision (adding *idf* improves retrieval), a decrease in average precision (adding *idf* worsens retrieval), or no change in average precision (adding *idf* gives the same retrieval effectiveness).

We look first at the addition of individual characteristics to any combination of other characteristics.

On all collections the addition of *idf* or *tf* information to a combination of characteristics was beneficial. This was more pronounced in the larger AP and WSJ collections, and held under both the weighting and non-weighting conditions.

The addition of *theme* information improves the performance of other combinations in smaller collections using either weighting or non-weighting. In the larger collections, the *theme* characteristic only improved performance under the weighting condition.

		CACM		**CISI**		**MEDLARS**		**AP**		**WSJ**	
		No Wgt	**Wgt**	**No Wgt**	**Wgt**	**No Wgt**	**Wgt**	**No Wgt**	**Wgt**	**No Wgt**	**Wgt**
idf	Inc	**51**	**58**	**54**	**50**	**47**	**48**	**55**	**63**	**62**	**62**
	Same	0	1	0	0	0	0	0	0	0	0
	Dec	12	4	9	13	16	15	8	0	1	1
tf	Inc	**60**	**59**	**57**	**54**	**53**	**56**	**60**	**62**	**62**	**62**
	Same	1	3	1	1	1	1	1	1	1	1
	Dec	2	1	5	8	9	6	2	0	0	0
theme	Inc	**33**	**26**	**48**	**45**	**51**	**49**	22	**38**	26	**54**
	Same	2	6	3	2	1	1	1	2	2	2
	Dec	28	31	12	16	11	13	**40**	23	**35**	7
context	Inc	27	18	8	12	17	14	**56**	**63**	**59**	**48**
	Same	2	4	0	0	0	0	0	0	0	0
	Dec	**34**	**41**	**55**	**51**	**46**	**49**	7	0	4	15
spec	Inc	19	14	16	22	17	13	**46**	4	22	6
	Same	1	**36**	3	17	0	**35**	1	0	2	**54**
	Dec	**43**	13	**44**	**24**	**46**	15	14	**56**	**39**	3
noise	Inc	**60**	**50**	9	29	**51**	**53**	**48**	**57**	**52**	**48**
	Same	1	6	1	0	2	1	2	2	5	15
	Dec	2	7	**53**	**34**	10	9	13	4	6	0
info_ noise	Inc	**37**	18	**46**	18	18	16	**31**	5	**45**	5
	Same	0	**35**	1	16	0	**32**	1	**57**	0	**54**
	Dec	26	10	16	**29**	**45**	15	**31**	1	18	4

Table 12: Effect of the addition of a characteristic to combinations of characteristics
Bold figures indicate predominant effect of each characteristic

The addition of *context* characteristic performed poorly in the smaller collections, performing more poorly when using weighting. In the larger collections the majority of combinations improved after the addition of *context* information.

With exception of the CISI, the addition of the *noise* characteristic improves performance in both weighting and non-weighting conditions.

The two document characteristics – *specificity* and *info_noise* – are very susceptible to how they are treated. The *specificity* characteristic tends to decrease the effectiveness of a combination of characteristics if the characteristics are not weighted. If the characteristics are weighted, then addition of *specificity* information is neutral: the combination performs as well as without the *specificity* information. The WSJ collection is the exception to this general conclusion. For this collection, under no weighting, the addition of *specificity* increases the effectiveness of a combination. Under weighting *specificity* decreases the effectiveness of a combination.

The *info_noise* characteristic tends to improve the effectiveness of a combination when using no weighting and to be neutral with respect to weighting, i.e. it does not change the performance of the combination. The main exception to this is the MEDLARS collection in which *info_noise* tends to harm the performance of a combination when not using weighting.

Having considered which characteristics improved or worsened combinations, we now examine which combinations are affected by the addition of new information. In Tables A.142 – A.151, in the Appendix, we present a summary of how often individual characteristics will improve a combination containing another characteristic, e.g. how many combinations containing *idf* are improved by the addition of *tf*.

Under both the weighting and non-weighting conditions the following generally held:

- *idf* improved combinations containing *context* more than other characteristics and improved combinations containing *noise* least of all
- *tf* improved combinations containing *context* or *noise* more than other characteristics and *theme* least
- *theme* improved combinations containing *context* most and combinations containing *tf* least
- *context* improved combinations containing *noise* least
- *specificity* improved combinations that contained *theme* and *info_noise* more than combinations containing other characteristics
- for the *noise* characteristic there were no general findings except that combinations containing *idf* were usually less likely to be improved by the addition of *noise* information
- *info_noise* improved combinations containing *theme* and *specificity* most often.

Weighting slightly altered which combinations performed well but the basic trends were the same across the conditions. On the larger collections, one effect of weighting was to reduce the effect of individual characteristics in that the effect of adding a characteristic was less likely to be dependent on which characteristics were already in the combination.

One further observation is that term weighting schemes that represent similar features (e.g. *idf* and *noise* which both represent global term statistics, and *tf*/*theme* which both represent within-document statistics) generally combine less well. That is combining these pairs of weights does not generally help retrieval as much as combining complementary weights, e.g. *idf* and *tf*, *idf* and *theme*, etc. Combining the two document characteristics, however, does seem to give better results.

6.4 Summary

Our hypothesis was that combining evidence – combining characteristics of terms – can improve retrieval effectiveness over retrieval by single characteristics. In section 6 we demonstrated that this was generally the case: all characteristics could benefit from combination. Where combinations work well is where the additional characteristics act as additional means of ranking documents. That is separating documents by other sets of features. Other researchers have considered this, e.g. Salton and Buckley examine *idf* as a precision-enhancing funciton and *tf* as a recall-enhancing function, [SB88]. Similarly, Cooper, [Coo73][17], discusses the difficulty of assessing likely utility without considering additional features of document content.

However not all combinations are successful. Two aspects of combination that are likely to predict success are the nature of the characteristics– complementary functions combine better – and the success of the characteristic as a single retrieval function.

Weighting the characteristics to reflect the strength of each characteristic as a single retrieval function is also generally a good idea. However it can be difficult to set optimal weights for two reasons: firstly it is likely that good weights will be collection dependent as the individual characteristics have different levels of effectiveness on different collections.

Secondly the weights should reflect the effectiveness of the characteristics relative to each other. However this becomes difficult to assess when we combine characteristics, as we have to measure the relative strength of each

[17] We are grateful to the anonymous referee for pointing us to this paper.

characteristic against a set of characteristics, e.g. the effectiveness of *idf* in combination with *tf* and *theme*. The performance of the characteristics as individual retrieval functions gives us some guidance on how to set weights but some experimentation is necessary to set useful values.

Smeaton, [Sme98], suggests that retrieval strategies which are conceptually independent should work better in combination, and that retrieval strategies that work to same general level of effectiveness should be suitable for conjunction. In his experiments Smeaton demonstrated that, although this does generally hold, it can be difficult to produce a good combination. We reinforce these findings in this paper and demonstrate how weighting the different retrieval functions – different characteristics – can help the combination process.

In Table 13, we show the best combination of characteristics for each collection. As can be seen which set of characteristics constitutes the best combination differs over the collections. If we use weighting of characteristics, then the best combination for a collection may also change. This is a further difficulty with a straight combination of evidence: it is difficult to derive a good set of characteristics that can be used on all collections. In the next section we propose a method to counter this difficulty: using the relevant documents to select a good set of characteristics for individual queries, irrespective of which collection they are being applied.

Collection and condition	**Best combination**	**Average precision of best combination**
CACM (NW)	***tf + noise***	*30.26*
CACM (W)	***idf + tf + noise***	*25.68*
CISI (NW)	***idf + tf***	*12.87*
CISI (W)	***idf + tf***	*12.84*
MEDLARS (NW)	***theme + noise***	*48.64*
MEDLARS (W)	***theme + noise***	*47.29*
AP (NW)	***idf + tf + context + noise***	*15.31*
AP (W)	***all***	*14.09*
WSJ (NW)	***idf + tf***	*15.65*
WSJ (W)	***all***	*15.73*

Table 13: Best combinations for each collection and condition
(**NW** = non-weighting condition, **W** = weighting condition)

7 Relevance feedback

Our intention behind the set of experiments described in this paper is twofold: first to demonstrate that taking into account how terms are used within documents can improve retrieval effectiveness; secondly that it is possible, for each query, to select an optimal set of characteristics for retrieval based on the relevance assessments from a user.

That is, we are not only asserting that considering how terms are used *can* improve retrieval, but that the characteristics that *will* improve retrieval will vary across queries and collections. For example, for some queries the context in which the query terms appear will be important, whereas for other queries it may be how often the query terms appear. For each query, then, there will be a set of characteristics that will best indicate relevance. In the experiments described in the remainder of this paper we test whether this hypothesis holds by investigating methods of *selecting* characteristics of query terms.

7.1 Methodology

In these experiments we performed a series of relevance feedback experiments, selecting characteristics to represent query terms based on the differences between the relevant and non-relevant documents.

Our methodology was as follows:

- rank all documents in a collection using the combination of all the characteristics
- take the 30 top documents from the initial *all* ranking; the combination of all characteristics
- calculate for each term the average score for each characteristic in the relevant and non-relevant set, e.g. the average *tf* value for term 1 in relevant documents, the average *tf* value for term 1 in non-relevant documents.

- select which characteristics of each query term to use to score documents and how the characteristics should be used. Three strategies were tried, each will be discussed separately in sections 7.3-7.5. Each strategy constructs a modified query containing characteristics of terms.
- re-rank the remaining retrieved documents
- calculate recall-precision values using a full-freezing ranking scheme [CCR71] to ensure that we are only comparing the effect of each technique on the unretrieved, relevant documents.
- compare the results given, over the same set of documents, by doing no relevance feedback, the results obtained from the best combination of criteria (section 6.4, Table 13) and an alternative relevance feedback algorithm, the $F_{4.5}$ method (section 7.2).

This set of experiments was designed to test the hypothesis that some queries or documents will be more suited to certain combinations of characteristics and that we can select these characteristics automatically. For example some queries will do better if we take into account *tf* or *theme* rather than *context*.

7.2 F4.5

We need to compare our technique for relevance feedback against another relevance feedback algorithm. For this we have chosen the $F_{4.5}$ weighting algorithm [RSJ76], equation 8, which assigns a new weight to a term based on relevance information. This technique for reweighting query terms was chosen partly because it has been shown to give good results but also because it does not add any new terms to the query. As our technique also does not add any new terms to the query but only modifies the existing query, we feel this is a fair comparison with which to test our techniques.

$$w_q(t) = \log \frac{(r+0.5)(N-n-R+r+0.5)}{(n-r+0.5)(R-r+0.5)}$$

Equation 8: $F_{4.5}$ function, which assigns a weight to term *t* for a given query.
r = the number of relevant documents containing the term *t*, *n* = the number of documents containing *t*, *R* = the number of relevant documents for query *q*, and *N* = number of documents in the collection

7.3 Feedback strategy one

In this method we select for each query which characteristics to use for each query term based on the average values, described in section 7.1. For example, if the average *context* value for a term was greater in the relevant documents than in the non-relevant documents, then the *context* of that term was taken to be a better indicator of relevance than non-relevance and so was included in the new query. The modified query is a set of characteristics of the query terms.

7.4 Feedback strategy two

The previous strategy selectively combined evidence on a query-to-query basis, ranking all documents based on the same set of query term characteristics. This strategy starts with the set of characteristics produced by Feedback 1, then selects which of these characteristics to use on a document-to-document basis. The result of this is that we first select a set of characteristics based on the set of relevant documents and then decide which of these characteristics to use to score each document. The intuition behind this is: if a characteristic is indicated as a good indicator of relevance then we should not only bias retrieval of documents which demonstrate this characteristic but suppress retrieval of documents which do not. For example, if a term must appear often in a document – high *tf* value – to be relevant, then documents that only contain a few occurrences of the term should not be considered.

We use the same averaging technique as in the previous strategy to construct a modified query. Then, for each document we compare the characteristic score of each query term in the document against the average score. If the characteristic score is greater than the average then we count the score as part of the document score; if not we ignore the evidence. This experiment is, then, a more strict case of Feedback 1. Feedback 1 selected characteristics with which to rank all documents, whereas this experiment selects characteristics for a query and then uses them selectively across documents.

7.5 Feedback strategy three

This final experiment is also a refinement of Feedback 1. In Feedback 1 we included a characteristic of a term in a query if it was better at indicating relevance than non-relevance. In this experiment we also take into account how well a characteristic indicates relevance. We first select a set of characteristics as in Feedback 1, then weight each term by the ratio of the average characteristic value in the relevant to the non-relevant documents. This ratio is taken to be an indication of how well a characteristic indicates relevance and is used to weight characteristics.

The contribution of a characteristic of a term to the retrieval score of a document is the ratio multiplied by the weight of the characteristic of the term in the document. This combined weight is a measure of the discrimination power of a characteristic of a term (the ratio) and its indexing strength (the indexing weight in the document). In the weighting condition (described in section 6) a third weight is given by the characteristic weight. The intuition behind this is that if a characteristic does not discriminate well over the relevant and non-relevant set then we should not prioritise this information.

8 Feedback results

In this section we examine three sets of results, to test different aspects of our feedback techniques.

i. the results from running our feedback strategies as *predictive* strategies. This is the methodology outlined above and is designed to test whether the feedback techniques help retrieve more relevant documents based on an initial sample of relevant documents. Results from this test will be discussed in section 8.1.

ii. the results from running the strategies as *retrospective* strategies. In this case we use the strategies to form modified queries based on knowledge of all the relevant documents. This success of a feedback strategy in retrospective feedback is measured by how well it ranks all the relevant documents, rather than by how well it improves the retrieval of new relevant documents. This technique, then should give the optimal performance of a feedback strategy and is discussed in section 8.2.

iii. the characteristics used in the feedback strategies. In section 8.3 we examine which characteristics were used in the feedback strategies. We do this to see if we can draw any conclusions about the performance of the feedback strategies based on which characteristics were selected to describe query terms.

8.1 Predictive feedback

Collection	Condition	No feedback	Best combination	$F_{4.5}$	Feedback 1	Feedback 2	Feedback 3
CACM	**NW**	25.28	**30.26** **19.70%**	26.58 5.14%	27.38 8.31%	23.28 -7.91%	27.62 9.26%
CACM	**W**	24.34	25.68 5.51%	25.51 4.81%	25.98 6.74%	21.79 -10.48%	**26.44** **8.63%**
CISI	**NW**	11.66	12.87 10.38%	14.05 20.50%	14.1 20.93%	13.73 17.75%	**15.11** **29.59%**
CISI	**W**	12.02	12.84 6.82%	14.2 18.14%	14.55 21.05%	14.21 18.22%	**15.57** **29.53%**
MEDLARS	**NW**	45.92	48.64 5.92%	47.93 4.38%	48.69 6.03%	48.23 5.03%	**49.41** **7.60%**
MEDLARS	**W**	45.29	47.29 4.42%	47.61 5.12%	48.14 6.29%	47.61 5.12%	**48.9** **7.97%**
AP	**NW**	12.04	**15.31** **27.16%**	12.46 3.49%	13.15 9.22%	12.09 0.42%	13.19 9.55%
AP	**W**	14.09	14.09 0.00%	14.58 3.48%	14.88 5.61%	14.51 2.98%	**15.01** **6.53%**
WSJ	**NW**	13.33	**15.65** **17.40%**	13.53 1.50%	14.4 8.03%	13.96 4.73%	14.47 8.55%
WSJ	**W**	15.73	15.73 0.00%	15.89 1.02%	16.37 4.07%	15.86 0.83%	**16.47** **4.70%**

Table 14: Summary of predictive relevance feedback experiments
Bold figures represent the highest increase in average precision for each case
(**NW** = non-weighting condition, **W** = weighting condition)

Table 14 presents the results of the predictive experiments. Each row shows the average precision after four iterations of feedback[18] plus the percentage increase in average precision over no feedback (Table 14, column 3).

There are several conclusions from our predictive feedback experiments.

Firstly, the selective feedback strategies (Feedback 1 – Feedback 3) do perform well. On the weighting condition at least one of the Feedback methods outperformed the No Feedback and Best Combination methods. However, if we did not use weighting then the Best Combination method outperformed the Feedback strategies on the AP, CACM and WSJ collections. Out of the ten tests (five collections, weighting and non-weighting conditions), seven achieved best overall performance with a Feedback strategy. This latter finding demonstrates that selecting a good combination of characteristics for each query is better than using the best combination of characteristics for a set of queries. In addition, on all cases, the Feedback 1 and Feedback 3 strategies outperform the $F_{4.5}$ baseline.

Secondly, comparing the weighting and non-weighting conditions: the better the initial ranking, the better the feedback performance. That is, whichever condition gave the better average precision for the initial ranking (No feedback column) also gave the better average precision after four iterations of feedback. However, the conditions that gave the poorer initial average precision gave the higher improvement after feedback measured as a percentage increase. Thus, good initial rankings give better feedback in the sense that they retrieve relevant documents better but feedback improves a poor ranking more than a good ranking.

This latter conclusion possibly, in part, arises because there is greater improvement to be gained from a poor initial ranking than a good initial ranking. Weighting, however, does not change the relative performance of the feedback algorithms: if one feedback strategy performs better than another under the non-weighting condition, it will also perform better under the weighting condition.

Thirdly, there is a marked preference for the Feedback 3 strategy. This strategy selects term characteristics for each query term and also uses the discrimination power of a characteristic of a term to score documents. The extra information given by the discrimination power between relevant and non-relevant documents is the cause of the better performance of Feedback 3 over the other feedback strategies.

On the larger collections (AP and WSJ), those collections that also have the shorter queries, the highest average precision was given by the Feedback 3 strategy using weighting of characteristics. This method uses the most evidence to score documents: evidence on the quality of the characteristics through the use of weighting, selection of good term characteristics and the weighting given by the discrimination between relevant and non-relevant documents.

On all the collections the Feedback 3 strategy outperformed the Feedback 1 strategy which outperformed the Feedback 2 strategy. The Feedback 2 and 3 strategies are both refinements of the basic Feedback 1 strategy and both use additional evidence to make a retrieval decision. In the case of Feedback 2 this additional information comes in the form of the index scores of the query term characteristics in individual documents and in the Feedback 3 strategy it comes from the discrimination power of a query term characteristic over the set of relevant and non-relevant documents. The consistency of the performance of the Feedback 3 strategy over the Feedback 2 strategy suggests discriminatory power is a better source of additional evidence.

8.2 Retrospective feedback

In Table 15 we present the results of the retrospective feedback experiments[19]. These experiments use all the relevant documents to modify the query and this extra evidence should give better performance in RF. The first observation is that, for all collections and conditions, a Feedback method does give best overall results: selection methods of feedback do give consistent increases in retrieval effectiveness. The selection methods all gives better results than the retrospective $F_{4.5}$ baseline.

For all collections, weighting gives better overall performance than no weighting.

[18] Tables A.71 – A.121.

[19] Tables A.121 – A.130.

The most unusual case is the performance of the Feedback 3 strategy, when using weighting. This test not only performed more poorly than the Feedback 2 and Feedback 3 strategies but also performed more poorly when used retrospectively than predictively.

The Feedback 3 strategy uses three types of weights: index weights attached to terms, relevance feedback weights derived from analysing the relevant documents and weights use to reflect the relative importance of the characteristics. The index weights and characteristics weights are identical in the predictive and retrospective strategies are identical, and the relevance feedback weights do give an increase in the non-weighting condition, so it appears that some interaction of the three are responsible. A deeper analysis is necessary to uncover the underlying problem.

Collection	Condition	No feedback	Best combination	$F_{4.5}$	Feedback 1	Feedback 2	Feedback 3
CACM	**NW**	25.28	30.26 19.70%	27.02 6.88%	**39.9** **57.83%**	39.68 56.96%	37.65 48.93%
CACM	**W**	24.34	25.68 5.51%	25.67 5.46%	**39.28** **61.38%**	39.27 61.34%	38.01 56.16%
CISI	**NW**	11.66	12.87 10.38%	13.21 13.29%	19.48 67.07%	19.68 68.78%	**20.3** **74.10%**
CISI	**W**	12.02	12.84 6.82%	13.56 12.81%	20.06 66.89%	20.52 70.72%	**20.83** **73.29%**
MEDLARS	**NW**	45.92	48.64 5.92%	47.87 4.25%	52.59 14.53%	51.68 12.54%	**56.13** **22.23%**
MEDLARS	**W**	45.29	47.29 4.42%	47.28 4.39%	51.67 14.09%	50.43 11.35%	**56.66** **25.10%**
AP	**NW**	12.04	15.31 27.16%	12.64 4.98%	17 41.20%	16.53 37.29%	**18.61** **54.57%**
AP	**W**	14.09	14.09 0.00%	14.16 0.50%	19.01 34.92%	18.4 30.59%	**19.91** **41.31%**
WSJ	**NW**	13.33	15.65 17.40%	13.73 3.00%	15.13 13.50%	**17.35** **30.16%**	15.57 16.80%
WSJ	**W**	15.73	15.73 0.00%	15.88 0.95%	16.66 5.91%	**17.9** **13.80%**	15.95 1.40%

Table 15: Summary of retrospective relevance feedback experiments
Bold figures represent the highest increase in average precision for each case
(**NW** = non-weighting condition, **W** = weighting condition)

8.3 Characteristics used in feedback

In this section we examine the characteristics that were selected in each of the selection feedback algorithms. In particular we concentrate on the Feedback 1 strategy which selects characteristics for query terms and the Feedback 2 strategy which then selects terms across documents. This is intended to analyse the performances of the feedback algorithms by which characteristics they selected in the feedback runs. Table 16 summarises the characteristics used in the Feedback 1 strategy (in which characteristics are selected for the query) and Table 17 summarises the characteristics used in the Feedback 2 strategy (in which characteristics are also selected for each document).

The predictive cases (Columns 3 and 4) are averaged over four iterations of feedback. As the use of weighting changes the ranking of documents at each iteration, different relevant documents will be used for feedback in the weighting and non-weighting conditions. Consequently the figures for the two conditions are different. The retrospective case is measured over all the relevant documents and so the results of the selection procedures are identical for the non-weighting and weighting conditions (Column 5).

For the Feedback 1 strategy, the selection of characteristics tended to follow the quality of the characteristics as retrieval algorithms: characteristics that performed well as a retrieval function tended to be selected more often in RF. This seems intuitively correct: the characteristics that are better indicators of relevant are more likely to be selected.

There was very little difference between the characteristics selected in the weighting and non-weighting characteristics for the Feedback 1 strategy. The only exception to this was the CACM collection. For this collection the non-weighting condition showed a much higher percentage of characteristics were chosen across the query terms. This high use of characteristics does not, however, appear to have improved retrieval effectiveness as the Feedback strategies performed worse than the Best Combination method for the non-weighting condition on the CACM (Table 14). The use of fewer characteristics in the weighting condition did help the retrieval effectiveness of the Feedback strategy.

Over all the collections there was a greater use of characteristics (more characteristics were selected for each query term) in the retrospective strategy than in the predictive strategy. The retrospective techniques base their selection on the difference between the relevant documents and the rest of the document collection, whereas the predictive strategies base the selection decision on the difference between the relevant and non-relevant on a sample of the top-ranked retrieved documents. As the latter set of documents may be relatively similar, the averaging procedure used to decide which characteristics are selected may not be able to differentiate good characteristics as well in the predictive as in the retrospective case.

Collection	**Characteristics**	**Predictive no weighting**	**Predictive weighting**	**Retrospective weighting**
CACM	*idf*	41	37	**60**
	tf	39	35	**60**
	theme	48	30	46
	context	**69**	24	38
	specificity	45	48	43
	noise	**61**	31	38
	info_noise	**55**	**60**	7
CISI	*idf*	33	33	**54**
	tf	32	31	**53**
	theme	22	22	38
	context	33	33	**57**
	specificity	48	43	32
	noise	34	34	**56**
	info_noise	**54**	**55**	**70**
MEDLARS	*idf*	**53**	**53**	**74**
	tf	**52**	**53**	**73**
	theme	**51**	**53**	**70**
	context	49	49	**72**
	specificity	37	43	43
	noise	**54**	**54**	**73**
	info_noise	40	39	40
AP	*idf*	**61**	**61**	**82**
	tf	**55**	**55**	**82**
	theme	42	42	**73**
	context	**55**	**55**	**75**
	specificity	39	39	**67**
	noise	19	19	16
	info_noise	39	39	25
WSJ	*idf*	**62**	**62**	**85**
	tf	**51**	**51**	**83**
	theme	43	40	**72**
	context	**54**	**53**	**77**
	specificity	42	39	**96**
	noise	12	12	8
	info_noise	21	22	7

Table 16: Characteristics used in Feedback 1 strategy
Bold figures indicate that a characteristic was used for the majority of terms

Table 17 analyses the usage of characteristics in the Feedback 2 strategy. We shall recap this strategy with an example: if the *tf* value of query term *t* is selected to form part of the query – is a good indicator of relevance - we first calculate the average *tf* value of *t* in the relevant documents. This average value is compared with the value of *t* in each remaining document in the collection that contains *t*. If the value of *t* in document *d* is greater than the average then we use the *tf* value of *t* to give a retrieval score to *d*.

Table 17 displays the percentage of documents that received a score using this strategy, e.g. on average, for the CACM collection, only 6% of the documents containing a query term, had a *tf* value for the term that was greater than the average relevant *tf*.

The *idf* and *noise* characteristics were used to score each of the remaining documents. These characteristics are based on global information and give the same value to a term in each document in which the term occurs. Consequently they cannot be used to differentiate between documents. The *idf* or *noise* characteristic of a term will always be greater than or equal to the average *noise* or *idf* value in the relevant documents and so the term will always be chosen to score documents in the Feedback 2 strategy. What differs in this strategy is the use of the document characteristics and the document-dependent term characteristics: *tf*, *theme*, and *context*.

Collection	**Characteristics**	**Predictive no weighting**	**Predictive weighting**	**Retrospective weighting**
CACM	*idf*	**100**	**100**	**100**
	tf	24	29	**83**
	theme	21	20	34
	context	20	18	41
	specificity	45	38	17
	noise	**100**	**100**	**100**
	info_noise	**100**	**100**	**100**
CISI	*idf*	**100**	**100**	**100**
	tf	**65**	**67**	**90**
	theme	34	36	39
	context	**66**	**67**	**85**
	specificity	41	39	30
	noise	**100**	**100**	**100**
	info_noise	**100**	**100**	32
MEDLARS	*idf*	**100**	**100**	**100**
	tf	**55**	**55**	**87**
	theme	**52**	**53**	**64**
	context	**53**	**56**	**52**
	specificity	48	48	15
	noise	**100**	**100**	**100**
	info_noise	46	49	16
AP	*idf*	**100**	**100**	**100**
	tf	18	19	**54**
	theme	26	29	37
	context	5	6	17
	specificity	39	34	7
	noise	**100**	**100**	**100**
	info_noise	27	27	8
WSJ	*idf*	**100**	**100**	**100**
	tf	20	18	**51**
	theme	23	30	38
	context	4	5	18
	specificity	11	17	6
	noise	**100**	**100**	**100**
	info_noise	20	24	0

Table 17: Characteristics used in Feedback 2 strategy

As in the Feedback 1 strategy there was roughly a similar percentage of usage of characteristics in the weighting and non-weighting strategies. Comparing the predictive and retrospective strategies, there was a greater use of the term characteristics and less use of the document characteristics for the same reasons as for the Feedback 1 strategy.

The Feedback 2 strategy works better retrospectively than predictively, usually because it eliminates more poor characteristics and uses a higher proportion of better ones.

The Feedback 2 strategy performed less well than the Feedback 1 strategy overall. This suggests that Feedback 2 method of eliminating weak evidence is not useful for RF.

8.4 Summary

Our main findings from the feedback experiments are that selecting characteristics of query terms can provide better retrieval effectiveness than re-weighting the terms ($F_{4.5}$) or selecting a good combination of terms for all queries. In addition, using some measure of the discrimination power of a term (Feedback 3) can improve the performance over simple selection (Feedback 1) in predictive feedback. In addition, weighting the characteristics at indexing can also improve effectiveness of the query term characteristics.

9 Conclusion

In this paper we investigated three areas:

i. the performance of new term and document characteristics. These characteristics showed variable performance as retrieval functions. Characteristics that only weighted documents, and did not weight terms, performed relatively poorly as they are unable to distinguish potentially relevant from irrelevant documents. Even when only ranking documents that contain a query term, the document characteristics still did not perform as well as term characteristics. The standard IR term weighting functions *idf* and *tf* performed well over all the collections tested.

ii. the performance of characteristics in combination. Combining characteristics to form a joint retrieval function was shown to be a good idea overall. Combination is successful for most characteristics but we have only outlined general indications of what makes a good combination of characteristics. It still remains difficult to predict more precisely how characteristics will perform in combination and how well they will perform for individual queries.

iii. the performance of characteristics in relevance feedback. Although it is difficult to predict how characteristics will perform in combination, the relevance assessments for a query can be used, predictively and retrospectively, to select a good set of characteristics for each query term. This method of feedback, generally, works better than choosing a single good set of characteristics to be used for all query terms.

The work outlined in this paper describes a basic analysis of term and document weighting in combination and in relevance feedback. A much deeper analysis of what factors influence the success of each weighting scheme will require taking into account factors such as length of document, number of unique terms per document, number of relevant documents per query, etc. Even though we have presented only general conclusions here, we believe that the main conclusions demonstrate that taking into account how terms are used can, and should, be considered further in document ranking. In particular the use of relevance feedback techniques for selecting which aspects of a term's use is appropriate for scoring documents seems to be a worthwhile approach for increasing the effectiveness of interactive IR systems.

Acknowledgements

This research is supported by the British Library and Information Commission funded project '*Retrieval through explanation'* (http://www.dcs.gla.ac.uk/ir/projects/explanation/). The members of the Glasgow University Information Retrieval Group provided many helpful comments on this work. We are also very grateful for the constructive and insightful comments provided by the anonymous reviewer.

References

[BL98] C. L. Barry and L. Schamber. *Users' criteria for relevance evaluation: a cross-situational comparison*. Information, Processing and Management. **34**. 2/3. pp 219-237. 1998.

[BKFS95] N. J. Belkin, P. Kantor, E. A. Fox and J. A. Shaw. *Combining the evidence of multiple query representations for information retrieval*. Information Processing and Management. **31**. 3. pp 431-448. 1995.

[CCR71] Y. K. Chang C. Cirillo, and J. Razon. *Evaluation of feedback retrieval using modified freezing, residual collection & test and control groups*. The SMART retrieval system: experiments in automatic document processing.(G. Salton, ed.). Chapter 17. pp 355-370. Prentice-Hall. 1971.

[Coo73] W. S. Cooper. On selecting a measure of retrieval effectiveness. Part 1. Journal of the American Society for Information Science. **24**. pp 87 – 100. Reprinted in *Readings In Information Retrieval*. K. Sparck Jones and P. Willett (eds). Morgan Kaufmann. 1997.

[DBM97] N. Denos, C. Berrut and M. Mechkour. *An image system based on the visualization of system relevance via documents*. Database and Expert Systems Applications (DEXA '97). Toulouse. pp 214-224. 1997.

[Har86] D. Harman. *An experimental study of factors important in document ranking.* Proceedings of the Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 186-193. Pisa. 1986.

[Har92] D. Harman. *Ranking algorithms*. Information retrieval: data structures & algorithms. (W. B. Frakes and R. Baeza-Yates, ed). Chapter 14. pp 363 - 392. 1992.

[HP93] M. A. Hearst and C. Plaunt. *Subtopic structuring for full-length document access*. Proceedings of the Sixteenth ACM SIGIR Conference on Research and Development in Information Retrieval. pp 59-68. Pittsburgh. 1993.

[Ing94] P. Ingwersen. *Polyrepresentation of information needs and semantic entities: elements of a cognitive theory for information retrieval interaction*. Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 101-110. Dublin. 1994.

[Lee98] J. H. Lee. *Combining the evidence of different relevance feedback methods for information retrieval.* Information Processing and Management. **34**. 6. pp 681-691. 1998.

[Lew92] D. D. Lewis *An Evaluation of phrasal and clustered representations on a text categorization task*. Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 37 – 50. Copenhagen. 1992.

[PB96] F. Paradis and C. Berrut. *Experiments with theme extraction in explanatory texts.* Second International Conference on Conceptions of Library and Information Science (CoLIS 2). pp 433-437. Copenhagen. 1996.

[Por80] M. F. Porter. *An algorithm for suffix stripping*. Program. **14**. pp 130 - 137. 1980.

[RSJ76] S. E. Robertson and K. Sparck Jones. *Relevance weighting of search terms*. Journal of the American Society of Information Science. **27**. pp 129-146. 1976.

[RL99] I. Ruthven and M. Lalmas. *Selective relevance feedback using term characteristics*. CoLIS 3, Proceedings of the Third International Conference on Conceptions of Library and Information Science. Dubrovnik. 1999.

[SB88] G. Salton and C. Buckley. *Term-Weighting Approaches in Automatic Text Retrieval.* Information Processing and Management. **24**. 5. pp 513-523. 1988.

[Sal83] G. Salton and M. McGill. *Introduction to modern information retrieval*. McGraw-Hill Book Company. New York. 1983.

[Sme98] A. Smeaton. *Independence of contributing retrieval strategies in data fusion for effective information retrieval.* Proceedings of the 20th BCS-IRSG Colloquium. Springer-Verlag Workshops in Computing. Grenoble. 1998.

[SJ72] K. Sparck Jones. *A statistical interpretation of term specificity and its application in retrieval*. Journal of Documentation. **28**. 11-20. 1972.

[TS98] A. Tombros and M. Sanderson. *The advantages of query-biased summaries in Information Retrieval* Proceedings of the Twenty-First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2-10. Melbourne. 1998.

[VR79] C. J. van Rijsbergen. *Information Retrieval*. Butterworths. 1979.

[VRHP81] C J van Rijsbergen, D. Harper and M. Porter. *The selection of good search terms*. Information Processing and Management. **17**. pp 77-91. 1981.

[VH96] E. M. Voorhees and D. Harman. *Overview of the Fifth Text REtrieval Conference (TREC-5).* Proceedings of the 5th Text Retrieval Conference. pp 1-29. Nist Special Publication 500-238. Gaitherburg.1996.

[ZG00]. X. Zhu and S. Gauch*. Incorporating quality metrics in centralized/distributed information retrieval on the WWW*. Proceedings of the Twenty-Third Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 288 – 295. Athens. 2000.