The effect of accessing non-matching documents on relevance feedback

MARK D DUNLOP
University of Glasgow

Traditional information retrieval (IR) systems only allow users access to documents that match their current query and, therefore, users can only give relevance feedback on matching documents (or those with a matching strength greater than a set threshold). This paper shows that, in systems that allow access to non-matching documents (e.g. hybrid hypertext and information retrieval systems), the strength of the effect of giving relevance feedback varies between matching and non-matching documents. For positive feedback the results shown here are encouraging as they can be justified by an intuitive view of the process. However, for negative feedback the results show behaviour that cannot easily be justified and that varies greatly depending on the model of feedback used.

---

1 INTRODUCTION

Free text information retrieval [van Rijsbergen 1979, Frakes and Baeza-Yates 1992] is based around the process of retrieving natural language documents from a document collection. A user typically expresses his/her information need through a free text, natural language query. The system then uses this query and information about the content of documents to match documents to the query (usually based around statistical similarities in word usage). In an ideal implementation, only documents that are relevant to the topic the user is interested in would be retrieved. In practice, systems are far from ideal for

---

Author's Address: Computing Science, University of Glasgow, Glasgow G12 8QQ, Scotland; E-mail: mark@dcs.gla.ac.uk; URL: http://www.dcs.gla.ac.uk/~mark/

two main reasons: the information provided in the user's query is insufficient to specify the information need and the information retrieval system is not sophisticated enough to process the query perfectly (often not even adequately). While much of the work in information retrieval concentrates on improving the matching process, from conceptually small but challenging improvements such as indexing phrases [Smeaton 1995] through to complex attempts to develop models of information [van Rijsbergen and Lalmas 1996], this paper looks in depth at relevance feedback: a method for gaining more information from users in order to improve the query specification.

Relevance feedback [Harman 1992a,b] is a well-established technique that allows a user to comment on how well an information retrieval (IR) system has performed and to improve that performance by providing additional information. Once an initial query has been entered and the retrieval system has provided a list of matched documents, the user is able to tell the system which of these matched documents are actually relevant to his/her information need and which are not. This feedback information can then be used, in conjunction with the initial query, to formulate a revised list of matched documents that is, hopefully, closer to the ideal list. Although a conceptually simple technique, the implementation varies greatly depending on the underlying IR model – this paper considers an intuitive view of what should happen and compares this with the three most widespread approaches.

The main thrust of the paper is a critique of the relationship between matching strength, true relevance, and the effect feedback has on a query. The matching strength is a measure of how useful the *system* considers a given document is to the user. Relevance is a statement by the *user* on whether or not a given document is relevant (this is actually a continuum but for simplicity we shall split the continuum into three sections: relevant, irrelevant and unknown/uncertain). The effect a feedback action has on a query can be measured by giving feedback on a single document, issuing a new query (based on the initial query plus this single feedback instance) and comparing the new and old queries – the bigger the difference the larger the effect the feedback action had. Although users do not, typically, use relevance feedback in this way, the approach does provide a useful method for assessing how much effect a feedback action has on a query.

In traditional IR user interfaces the user was only presented with documents that matched the query (system's view of matching). This is not the case with many modern IR systems that allow users access to non-matching documents. There are two main ways in which users can access non-matching documents: by following links in a hybrid information retrieval and hypermedia environment [e.g. Dunlop and van Rijsbergen 1993] or by accessing the results of old queries [e.g. Sanderson 1991]. Both these approaches introduce the question of how systems will react when feedback is given on a document that the system did not consider as a match and, more generally, what is the relationship between matching, relevance and strength of feedback.

While much work has shown that relevance feedback works in general [Salton 1971; Haines and Croft 1993] and has looked at the relationship between strength of feedback and the number of terms used [Harman 1992; Haines

and Croft 1993] and at the relationship between strength of feedback and the independence assumption [Harper and van Rijsbergen 1978], this paper looks at the relationship between feedback strength and how well the fed-back documents matched the initial query.

2 INTUITIVE VIEW

Although considerable work is required to establish the true expectations users of free text retrieval systems have of feedback, the following is presented as a working model of users' intuition.

For positive feedback, when the user states "this document is indeed relevant", we would expect the effect of feedback on a query to be inversely related to how well the document matched the query. Consider a query Q and two documents M, which matches well with query Q, and M', which does not. It would seem obvious that users would expect that giving positive feedback on document M would have little effect on the query – (s)he is simply confirming the retrieval decision. However, giving positive feedback on document M' should have a considerable effect on the query – the system should take account of the new information which is attempting to correct an incorrectly assigned document.

For negative feedback, when the user states "this document is not relevant", the opposite behaviour could be expected. Giving negative feedback on document M should have a large effect (as it represents a significant difference of opinion between the system and the user), while giving negative feedback on M' less (as this is only confirming the system's view).

3 VECTOR SPACE FEEDBACK

Most approaches to IR model a document as a vector of clues to relevance. These clues are most often terms derived directly from the words used in documents and are often weighted. The vector space model of information retrieval takes the vector representation further and considers retrieval as performing geometric operations in Euclidean space [Salton 1971]. Under this model of information retrieval the angle between the two vector representations, of the document and of the query, provides the measure of matching strength. The most widely used measure is the cosine coefficient [van Rijsbergen 1979 pp. 39] which measures the cosine of this angle. In the language of simple set notation the cosine coefficient (cc) can be described as:

$$cc(\mathbf{D},\mathbf{Q}) \quad = \quad \frac{\mathbf{D} \bullet \mathbf{Q}}{\|\mathbf{D}\| \, \|\mathbf{Q}\|}$$

where

$$\mathbf{D},\mathbf{Q} \quad = \quad \text{the vectors being compared}$$

$$\mathbf{D} \bullet \mathbf{Q} \quad = \quad \sum_{i=1}^{N} \mathbf{D}_i \mathbf{Q}_i \qquad \text{the dot product of D \& Q}$$

$$\|\mathbf{A}\| \quad = \quad \sqrt{\sum_{i=1}^{N} \mathbf{A}_i^2} \qquad \text{the size of A}$$

$$N \quad = \quad \text{dimensionality of the space (i.e. number of discrete terms)}$$

Full vector space relevance feedback is performed by adding a fraction of the feedback document vector to the query document vector (scaling the vector takes into account the difference in desired strength between one feedback action and the initial query). The scaling factor is a real number in the range -1 to 1, with negative values representing negative feedback. The feedback process can be described as:

$$\mathbf{Q}' \quad = \quad \text{norm}(\mathbf{Q} \oplus k\mathbf{D})$$

where

$$\mathbf{Q} \quad = \quad \text{the query vector normalised to unit length}$$
$$\mathbf{D} \quad = \quad \text{the normalised feedback document vector}$$
$$\mathbf{Q}' \quad = \quad \text{the post feedback query}$$
$$k \quad = \quad \text{the feedback strength constant } (-1 \le k \le 1)$$
$$\mathbf{A} \oplus \mathbf{B} \quad = \quad \text{vector addition of } \mathbf{A} \text{ and } \mathbf{B}$$
$$\text{norm}(\mathbf{A}) \quad = \quad \mathbf{A}/\|\mathbf{A}\| \quad \text{(ensures vectors are maintained at unit length)}$$

This process is a variation of the feedback model developed by Rocchio [1965] and can be visualised as the addition of two 2-D vectors. Although in a retrieval engine the space would be of very high dimensionality, the 2-D drawings serve as useful illustrations of the process. Figure 1 shows positive feedback on two documents, *M* and *M´*, after an initial query *Q*.
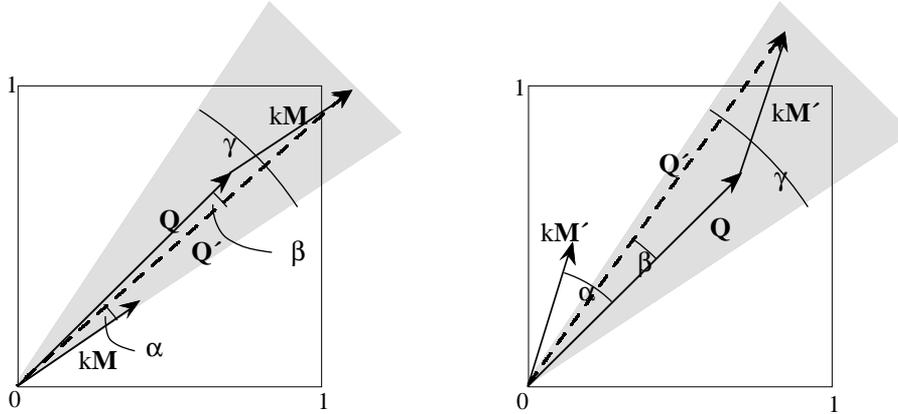
1a: Feedback on matched document   1b: Feedback on non-matched document

Figure 1: Diagrammatic addition of query vector to feedback vector

In both diagrams the light grey region depicts the area in which any document vectors will be considered as matches to **Q**. The area is defined as an angle of γ with the vector **Q** in the centre. When using the cosine coefficient for retrieval the light grey region is equivalent to the area in which documents are retrieved for a specific cut off, i.e. *cc(**Q**,**D**)≥cos(γ / 2)*. The dashed vector, **Q**´, represents the sum of the query vector with the document vector before normalisation. Figure 1, and later figure 4, is based on a feedback strength constant of 0.5 which is rather larger than normal but makes the diagrams clearer. The matching angle, γ, is also rather larger than would typically be found – a smaller angle would, however, have no effect on the differences in feedback strength since these differences are only dependent on the pre- and post-query vectors, and not on any cut offs to the matched documents list

As can be seen from figure 1, the angle β is greater when the angle α is greater. In terms of relevance feedback[1], this implies that the angular difference between the original and new query vectors is greater when the angular difference between the document and the original query is greater. This is not surprising and is in line with the intuitive argument for positive feedback presented in section 2.

If the space is rotated such that **Q** lies along the x-axis then the angle β can be calculated as follows:

$$\beta = \tan^{-1}\frac{\mathbf{Q}´_y}{\mathbf{Q}´_x}$$
$$= \tan^{-1}\frac{k\mathbf{D}_y}{\|\mathbf{Q}\| + k\mathbf{D}_x}$$
$$= \tan^{-1}\frac{\|k\mathbf{D}\|\sin\alpha}{\|\mathbf{Q}\| + \|k\mathbf{D}\|\cos\alpha}$$

---

[1] when the lengths of **M**, **M**´, and **Q** are fixed at k, k, and 1 respectively.

where

$$\mathbf{Q} \quad = \quad \text{query vector rotated by } \sigma \text{ such that it lies along the x-axis}$$
$$\mathbf{Q}' \quad = \quad \text{new query vector (rotated by } \sigma)$$
$$\mathbf{D} \quad = \quad \text{relevant document vector (rotated by } \sigma)$$
$$\mathbf{V}_i \quad = \quad \text{the i component of vector } \mathbf{V}$$

Since the final definition does not decompose the vectors into constituent parts, it does not require the vectors to be rotated to align with the x-axis and is valid for any dimensionality. Since $\mathbf{Q}$ and $\mathbf{D}$ are of unit length this can be further simplified to:

$$\beta \quad = \quad \tan^{-1}\!\left(\frac{k \sin \alpha}{1 + k \cos \alpha}\right)$$

The expression within brackets is guaranteed to be positive since both sine and cosine give positive results when α lies between 0° and 90°. The maximum angle β is thus achievable when α=90° and can be defined as $\tan^{-1}(k)$. This yields a minimum cosine coefficient between the old and new queries of $\cos(\tan^{-1}(k))$. The minimum value of β is achievable when the query and document are parallel, yielding a maximum cosine coefficient of $\cos(0°)=1$ (i.e. positive feedback on a perfect match has no effect on the query). Figure 2 shows the result of this calculation for various levels of positive feedback strength.
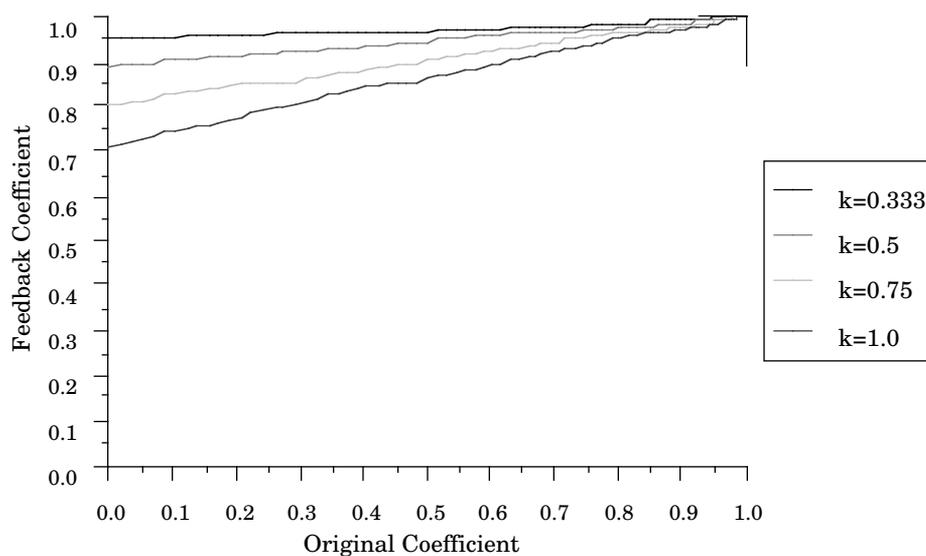
Figure 2: Effect of positive feedback for various values of k – *cos* α *against cos* β

Figure 2 clearly shows that positive feedback under the vector space model is in line with the intuitive argument – the closer the match between a document and a query the smaller the difference between the pre- and post-feedback queries. This is further supported by an exploded view of the line k=0.5, which is shown in Figure 3 together with experimental results as discrete points. These points represent the values achieved in an experimental setting to confirm the mathematical model and are based on a

vector space retrieval system using the CACM test collection. The experiment logged the actual results of giving full vector feedback on each document in the collection after issuing each of the standard queries. The points clearly confirm that the mathematical model does support experimental settings.
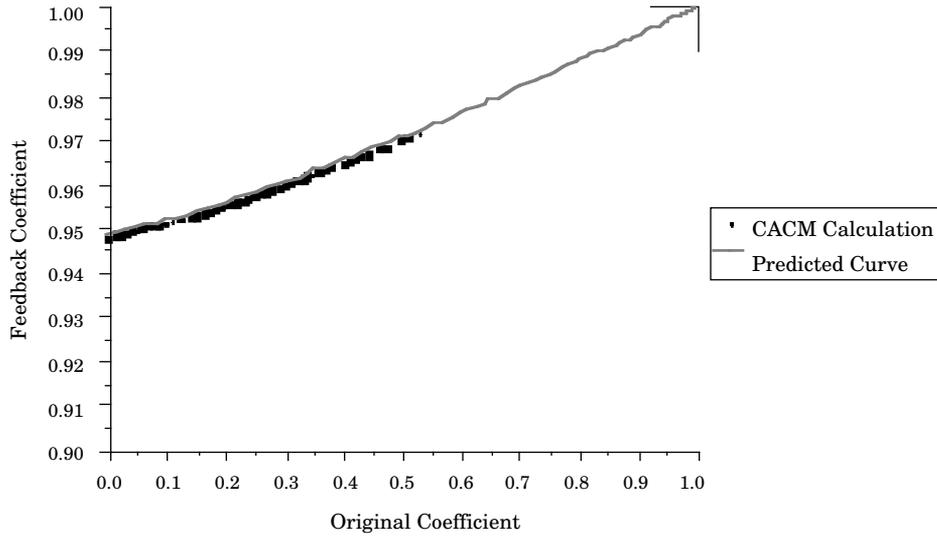
Figure 3: Exploded view of k=*0.5 line*

For negative feedback the process is very similar except that *k* is negative. This leads to the vector additions diagrams shown in figure 4 and matching coefficient graphs shown in figures 5 and 6 (for all these diagrams *k* is negative to represent negative feedback).

4a: Feedback on matched document   *4b: Feedback on non-matched document*
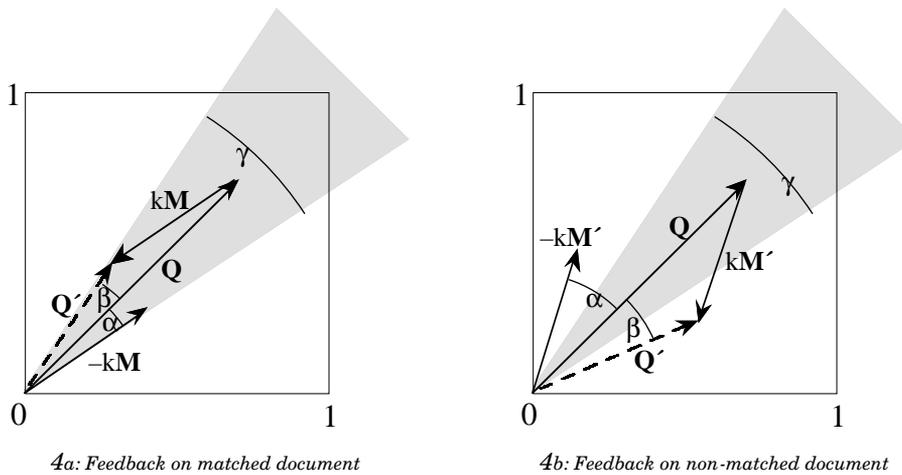
Figure 4: Diagrammatic subtraction of query vector from feedback vector

Figure 4 shows an example of negative feedback on a non-matching document being stronger than that on a matching document. This is not in line with the intuitive argument set out in section 2 but is confirmed by figures 5 and 6.

These figures show a curve for the effect of negative feedback that is far from the inverse of that given for positive feedback. Figure 6 shows a function which is overall increasing, to the extent that when negative feedback is given on a document that is a perfect match no change occurs. This is clearly in contradiction to the intuitive argument
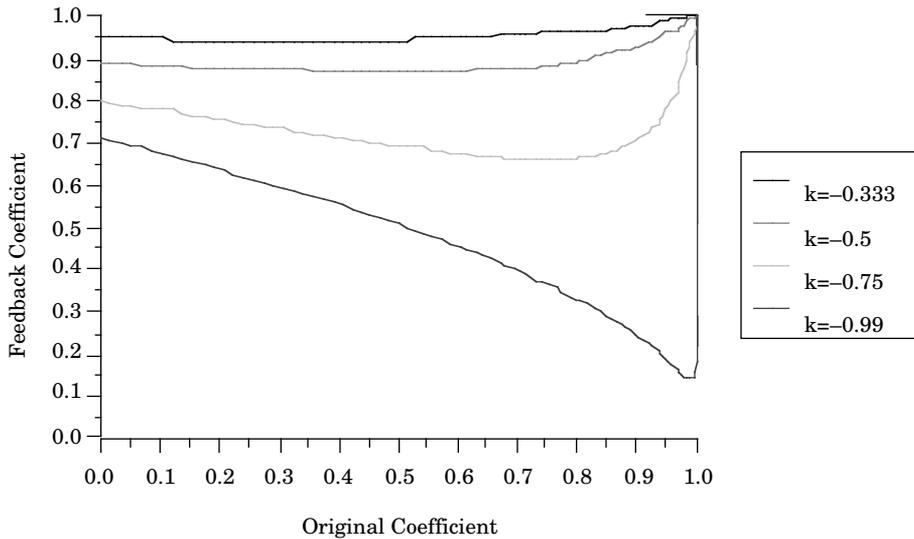
Figure 5: Effect of negative feedback for various values of k
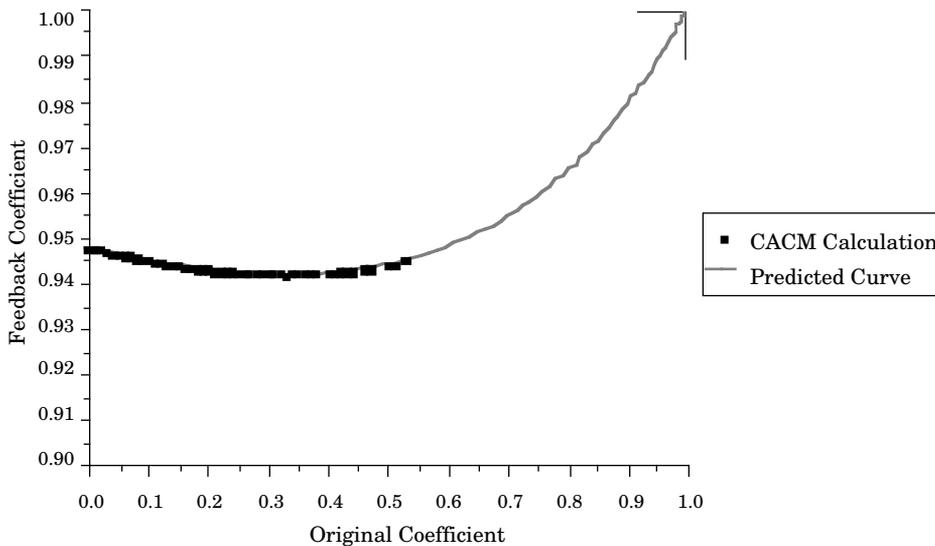
Figure 6: Exploded view of k≈–0.333 line

Towards the low end of the matching range (in the case of $k \approx -0.333$, matchings under 0.4), negative feedback does act as expected. Since this is the main range of matching values experienced in practical implementations, the imbalance in shape of curve may not be too serious an issue in practice. The effect of negative feedback on zero matching documents and the strength of negative feedback overall are more significant.

When a user gives negative feedback on a document that was a perfect non--match to the query, a change to the query occurs under the vector space model (i.e. the system predicts a document as not matching, the user agrees, but the query is altered). This is counter to the intuitive argument and no similar effect is observed for positive feedback, when a user states a perfect match is relevant.

When considering the curves for $k=0.75$ and $k=-0.75$, the largest effect of positive feedback yields a feedback coefficient of approximately 0.80 whereas a feedback coefficient of approximately 0.65 can be achieved with negative feedback. Although this minimum coefficient occurs outwith the range of normal matching strengths and 0.75 is strong feedback, all graphs show a stronger effect for negative than positive feedback throughout.

The non-intuitive effects of negative feedback could be partly overcome through various massages (e.g. using a smaller k for negative feedback than positive). However, the full vector approach of feedback can only be said to be intuitive for positive feedback – with negative feedback being counter-intuitive under the model. Furthermore, negative feedback is far from an inverse of positive feedback: if the user gives positive feedback on a document D, issues a feedback-query then gives negative feedback, the original status is not re-established.

Ide and Salton [1971] showed that using both negative and positive feedback information tended to retrieve "more relevant documents within the top 10% of the document collection than positive feedback [alone], but that the relevant documents remaining in the lower 70% of the collection are assigned much lower ranks by the [combined positive and] negative strategy than by the positive [only] strategy". In their experiments the user was only presented with matching documents; the results here would indicate that use of these documents for negative feedback would result in extreme changes to the query. This may be an explanation for the behaviour of early negative feedback experiments on the SMART system. Furthermore, Ide and Salton developed a *selective negative feedback* approach to attempt to reduce the problem by using negative feedback information to, essentially, create a list of banned terms while not affecting the weight of terms in general.

Aalbersberg [1992] experimented with a technique, called incremental relevance feedback, which was based around the user giving feedback on individual documents and the system automatically issuing a new query. This is very similar to the evaluation method used here and the results are consistent. The Aalbersberg experiments showed that, in general, increasing the length of the feedback vector decreased the performance for negative feedback. This is in contrast to positive feedback where the performance increased to a peak and then tailed off as the length of the feedback vector increased. This can be explained by the results shown here since negative feedback has such an extreme effect under this model of IR.

4 TERM ADDITION FEEDBACK

The discussion above concentrated on a vector addition and subtraction approach to relevance feedback. Many systems, however, adopt a simpler approach of adding the most descriptive terms from a document on positive feedback [Harman 1992b] and reducing term weights on negative feedback.

For positive feedback, in systems that add descriptive terms from the document to the query, there are two basic approaches: weighted, where query terms have a weight that is increased when feedback terms are added, or binary, where terms are simply added to the set of terms representing the query.

Giving positive feedback in a weighted system has the predictable effect of always increasing the weight of terms that are in the relevant document. If fixed increments are used to increase term weights then the better the match the smaller these fixed increments will be as a fraction of the term weight (since the terms' weights must be high in order for the document to match well). Regardless of the increment method, in practice positive feedback will strengthen the query in one direction if the feedback is on a matched document, but will tend to generalise the query when positive feedback is given on a non-matched document. This behaviour supports the intuitive view and is demonstrated in the Table 1.

This section makes use of four tables of the same format that give examples of different feedback approaches. The tables show three main columns – an initial query vector, a strongly matching document and a weakly matching document. The query column is further split into a list of terms and their weights, while the other two columns are split into a list of terms, a list of weights for those terms in that document and a list of weights showing the effect that feedback on that document has on the query. As an example, in Table 1 giving positive feedback on the good match, M, results in the weight of term *test* is increased from 1.0 in the initial query to a new weight of 1.2 while the weight of *very* is left unchanged. Finally, the tables show the cosine value between then old and new query descriptions, which is a measure of the similarity between the old and new query vectors with 1.0 being a perfect match. In Table 1, the query after feedback on the good match, Q'_1, gives a cosine weight of 0.998 with the original query, Q, which is closer to a perfect match than 0.983 for matching Q against Q'_2.

Query		Good match			Bad match		
	Q		M	Q'₁		M´	Q'₂
test	1	test	1	1.2			1
document	1	document	1	1.2	document	1	1.2
				0	sample	1	0.2
				0	bad	1	0.2
very	1			1			1
good	1	good	1	1.2			1
match	1	match	1	1.2	match	1	1.2
				0	many	1	0.2
				0	new	1	0.2
			Cosine	*0.998*		*Cosine*	*0.983*

Table 1: positive weighted feedback examples

When giving positive feedback in a binary system, the feedback strength is inversely proportional to how well the document matched, supporting the intuitive view. When a matched document is marked as relevant, few new terms will be added to the set of terms representing the query, thus giving a small change to the query. However, when a non-matching document is marked as relevant a large change will occur since many new terms will be added. Table 2 shows two extreme cases of boolean positive feedback – the good match does not affect the query description (cosine=1.0) while the poorly matching document results in a description where all terms have value 1 which is a large change to the query.

Query	Q	Good match	M	Q_1'		M′	Q_2'
test	1	test	1	1			1
document	1	document	1	1	document	1	1
				0	sample	1	1
				0	bad	1	1
very	1			1			1
good	1	good	1	1			1
match	1	match	1	1	match	1	1
				0	many	1	1
				0	new	1	1
		Cosine		*1.000*		*Cosine*	*0.745*

Table 2: positive boolean feedback examples

For negative feedback the effects are less clear but also differ between weighted and binary approaches. In a weighted retrieval system negative feedback can be implemented as a reduction in the weights of terms that occur in the marked document. There are two variants of negative feedback: models that only permit positive weights to be recorded for terms and models that permit negative weights (essentially terms that the user *does not* want to see).

If negative weights are permitted then negative feedback will behave in a sensible manner and will be the inverse operation of giving positive feedback (provided no vector normalisation takes place). Matched documents marked as non-relevant will generalise the query whereas non-matched documents marked as non-relevant will strengthen topics that are considered as not relevant – in essence introduce low pass filters to filter out documents that match specific topics. Since well matched documents contain many highly weighted terms, giving negative feedback on these will not necessarily have a large effect on the query while giving negative feedback on a poorly matched document may have a very large proportional effect by introducing many newly weighted terms (albeit with negative weights). Although this is counter to the intuitive argument, this approach to negative feedback does ensure that negative feedback is the inverse operation to positive. Table 3 shows a larger effect (0.976) when feedback is given on a bad match than when it is given on a good match (0.995) – contrary to the intuitive argument.

Query	Q	Good match	M	Q'_1	Bad match	M'	Q'_2
test	1	test	1	0.8			1
document	1	document	1	0.8	document	1	0.8
				0	sample	1	-0.2
				0	bad	1	-0.2
very	1			1			1
good	1	good	1	0.8			1
match	1	match	1	0.8	match	1	0.8
				0	many	1	-0.2
				0	new	1	-0.2
		Cosine		*0.995*	*Cosine*		*0.976*

Table 3: negative weighted feedback examples (with negative weights permitted)

When weights are restricted to positive values, negative feedback will act in line with the intuitive argument by tending to have the strongest effect on documents that match the query (since the algorithm will be able reduce the weight of more non-zero weighted terms). However, this approach does not provide the inverse operation of positive feedback. As the cosine measure normalises the vectors it can give a higher similarity for documents that are very similar to the query since negative feedback essentially scales the query description (Table 4 shows this for the good match which is very close to an 80% scaled version of the original query description). It could, however, be argued that the effect is in line with the intuitive argument since more weights are adjusted for the bad match than the good match.

Query	Q	Good match	M	Q'_1	Bad match	M'	Q'_2
test	1	test	1	0.8			1
document	1	document	1	0.8	document	1	0.8
				0	sample	1	0
				0	bad	1	0
very	1			1			1
good	1	good	1	0.8			1
match	1	match	1	0.8	match	1	0.8
				0	many	1	0
				0	new	1	0
		Cosine		*0.995*	*Cosine*		*0.994*

Table 4: negative weighted feedback examples (with positive only weights)

Negative feedback is rarely used in binary weighted systems as the strength of the feedback can be extreme – essentially terms from marked non-relevant documents would be removed from the query. This does have a strong effect when giving negative feedback on a matching document and little effect on a non-matching one; however the effects are too strong to be of practical use.

Although the behaviour of weighted negative feedback can differ from the direct relationship presented in the intuitive argument, the model can treat positive and negative feedback as inverse operations and can be interpreted as introducing useful low band filters to filter out non-relevant documents on certain topics. The binary term addition approach is also intuitive for positive feedback (even though the feedback strength can be extreme), but is not suitable for negative feedback.

5 PROBABILISTIC FEEDBACK

As the probabilistic model of information is based on extrapolating from known relevant information to unknown documents, positive feedback should work as predicted by the intuitive argument. When a user marks a document as relevant it will be added to the underlying base of known relevant documents. This addition will have a larger effect on future queries for documents that were not predicted as relevant than if the user added a document that was predicted as relevant.

To test the intuitive argument an experiment was run using an artificially created test collection. A set of document description vectors D was created and a subset R of these documents was then considered as relevant – both these sets were randomly created from a 10-dimensional space. Taken with a randomly chosen examination document vector E this provided enough information to calculate weights under the common probabilistic model. Since the pure probabilistic model is based on extrapolating information about known relevant documents to the unknown set there is no query required and most models do not include a query (the system is 'kick-started' using a specialist algorithm [e.g. Croft and Harper 1979]). For the experiments in this section the effect that a feedback has on the state of the IR engine is measured with respect to the examination document instead of a query document. Although not as direct as using a query it will still give an impression of how the IR engine's predictions of relevance changed after a feedback action, measured relative to E.

The full model of weighting recommended by Robertson and Sparck Jones [1976] (sometimes known as F4) was used to calculate the initial weights for the examination vector E based on the relevant set R. A second weighted vector E' was calculated based on the set $R+D_i$, for each document D_i in the collection. The two vectors E and E' represent respectively the assigned term weights based on the pre-defined set of marked relevant documents and this set plus the newly marked relevant document. These two vectors were compared to measure the strength of the effect of marking this document as one of the relevant ones, measured relative to the examination document (i.e. how weights in E are affected by additionally marking D_i as relevant). Figure 7 shows the strength of the original matching against the correlation between the two weighted vectors (the cosine of the angle between the two vectors). Figure 8 shows similar graphs for different queries. In line with standard probabilistic retrieval engines, the constant part of the matching algorithm is removed. This is usually ignored in IR systems, which are usually only concerned with ranking documents, as it is constant for all documents for a given query. For figures 7 and 8, however, the removal of this constant introduces a floating zero point which is of no great significance.
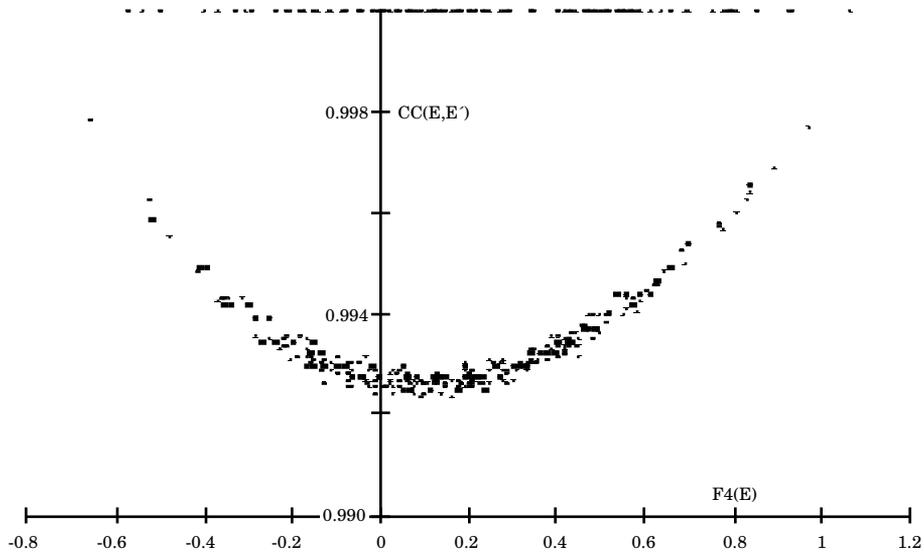
Figure 7: Effect of probabilistic feedback against matching strength
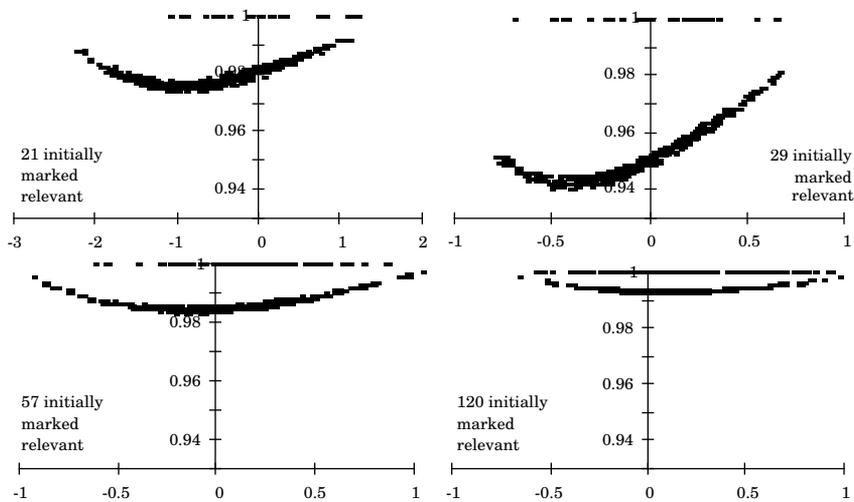

Figure 8: Effect of feedback against matching strength & number of relevant documents

Figures 7 and 8 support the argument that, in general, the probabilistic model performs positive feedback in line with the intuitive argument. They do, however, show that in practice the probabilistic model differs from the intuitive argument on very poorly matching documents. As the original matching strength becomes very low, the effect of giving feedback starts to reduce rather than continues to increase. While counter to the intuitive argument, it could be argued that this is favourable as caution should be taken when the retrieval engine has categorised a document so poorly. Figure 8 shows that, in general, the smaller the set of relevant documents the greater the effect of adding a new document to that set – while the intuitive argument says nothing concerning this, it is a reasonable effect. Figure 8 also shows that some relationship exists between the spread of marked relevant documents and the shape of the curve (the points with comparison coefficient 1 were already marked relevant and thus show the spread of relevant documents).

The argument is less clear for negative feedback since only information about known relevant documents is used for retrieval. Harper and van Rijsbergen [1978] argued that, for relevance feedback, the set of non-relevant documents can be estimated by the set of documents not known to be relevant. This argument is based on the observation that a very small fraction of the documents not known to be relevant are relevant and hence the unknown set is approximately equal to the non-relevant set. Positive relevance feedback adds considerable extra information by adding one extra document to the very small set of documents that are relevant. Negative feedback, however, provides very little information since it adds another document to the set of irrelevant documents – this set being the vast majority of the document collection. Taken with the approximation result from above, there is no additional information provided by marking a document as irrelevant. This appears to contradict the intuitive argument from section 2 which is based on error correction: the user marking a matched document as irrelevant *is* providing a considerable amount of information.

This problem may be caused by interaction between two intuitive arguments: that used by probabilistic retrieval on expanding known information and that, presented earlier, based on error correction. The probabilistic ranking principle [Robertson 1977] states that, for a given amount of information about the documents and the query, no improvement can be made over the ranked list produced by probability models. As such, the only way to correct for the clash with the intuitive argument presented here is to view the correction information as additional information. This could be achieved by viewing the wrongly assigned documents (those classed as relevant but that are actually irrelevant) as a set of documents, essentially a set of banned topics. Standard techniques could then be used to predict how likely a document is to be in this set, i.e. how similar is it to other wrongly predicted documents, and this could then be used together with the probability of a document being relevant to produce the final list of documents to the user. Further work needs to be carried out to investigate this approach.

6 CONCLUSIONS

The possibility of accessing non-matching documents, either through history mechanisms or through hypertext links, has serious implications for the performance of relevance feedback. These effects must be taken into account in systems that provide feedback facilities (especially negative feedback) and browsing facilities to access non-matching documents. Some of the effects are also observable with systems that only allow access to matching documents.

For positive feedback the results can be easily predicted and, for most models of feedback, result in an inverse relationship between matching strength of the document and how strongly the query is affected by giving feedback on the document.

In general, the three methods behave similarly for positive feedback but differ for negative feedback. One significant difference between the full vector space

and term addition models is the use of normalisation in the vector space model which results in positive and negative feedback not being inverse operations (i.e. negative feedback cannot be considered to undo positive feedback), whereas they are inverse operations in the term addition models that permit negative weights. Overall the effects of negative feedback are not directly proportional to matching strength and thus contradict the intuitive argument presented in this paper. Under the full vector space model, negative feedback as well as not being the inverse operation to positive feedback is, typically, much stronger in its effect than positive feedback. The traditional probabilistic model cannot easily take into account knowledge that some documents have been marked as irrelevant (instead of being a member of the massive set of probably irrelevant documents). Although various alterations could be made to the retrieval models to make them behave more in line with the intuitive argument, these results show that different models of feedback perform differently for negative feedback and often contradict the intuitive view proposed in this paper.

If the presented intuitive argument is in line with actual users' expectations, the gulf of execution [Norman 1986] between users' understanding of the system and its actual behaviour must be bridged by those implementing negative feedback. Although further work is required to establish how important the issues raised in this paper are in practice with operational systems, real users and real information needs, the behaviour of negative feedback described here must be taken into account when implementing feedback facilities.

7 ACKNOWLEDGEMENTS

8 REFERENCES

AALBERSBERG, I.J. 1992. Incremental relevance feedback. *Proceedings of ACM-SIGIR 92 - Copenhagen*, pp. 11-22.

CROFT, W.B., AND HARPER, D.J. 1979. Using probabilistic models of document retrieval without relevance information, *Journal of Documentation*, vol. 35, pp. 285-295.

DUNLOP, M.D. 1991. *Multimedia Information Retrieval*. Ph.D. Thesis, Computing Science Department, University of Glasgow. Report 1991/R21.

DUNLOP, M.D., AND VAN RIJSBERGEN, C.J. May 1993. Hypermedia and free text retrieval. *Information Processing and Management*, vol. 29(3), pp. 287-298.

FRAKES, W.B., AND BAEZA-YATES, R. (Editors). 1992. *Information Retrieval: Data Structures and Algorithms*. Prentice-Hall.

HAINES, D., AND CROFT, W.B. 1993. Relevance feedback and inference networks. *Proceedings of ACM-SIGIR 93 Conference – Pittsburgh*, pp. 2-11.

HARMAN, D. 1992a. Relevance feedback and other query modification techniques. In Frakes and Baeza-Yates [1992] (Chapter 11, pp. 241-263).

HARMAN, D. 1992b. Relevance feedback revisited. *Proceedings of ACM-SIGIR 92 Conference – Denmark*, pp. 1-10.

HARPER, D.J., AND VAN RIJSBERGEN, C.J. Sept. 1978. An evaluation of feedback in document retrieval using co-occurrence data. *Journal of Documentation*, vol. 34(3), pp.189-216.

IDE, E. 1969. *Relevance Feedback in an Automatic Document Retrieval System*. M.Sc. thesis, Report ISR-15, National Science Foundation, Department of Computer Science, Cornell University, N.Y., U.S.A.

IDE, E., AND SALTON, G. 1969. *Interactive Search Strategies and Dynamic File Organisation in Information Retrieval*. Report ISR-16, National Science Foundation, Department of Computer Science, Cornell University, N.Y., U.S.A. (reproduced in Salton [1971]).

NORMAN, D.A. 1986. Cognitive Engineering. In NORMAN, D.A., AND DRAPER, S.W. 1986. *User Centred System Design*. Lawrence Eribaum Associates, New Jersey.

ROBERTSON, S.E. 1979. The probability ranking principle in IR. *Journal of Documentation*, vol. 34, 294-304.

ROBERTSON, S.E., AND SPARCK JONES, K. 1976. Relevance weighting of search terms. *Journal of American Society for Information Science*, vol. 27, pp. 129-146.

ROCCHIO, J.J. 1965. *Relevance Feedback in Information Retrieval*. Report ISR-9, National Science Foundation, Department of Computer Science, Cornell University, N.Y., U.S.A. (reproduced in Salton [1971]).

SALTON, G. (Ed.). 1971. *The SMART Retrieval System*. Prentice Hall, New Jersey.

SANDERSON, M., AND VAN RIJSBERGEN, C.J. 1991. NRT (News Retrieval Tool). *Electronic Publishing Origination, Dissemination & Design*, vol. 4(4), pp. 205-217.

SEBASTIANI, F. 1994. A possible terminological logic for modelling information retrieval. *Proceedings of ACM SIGIR 94 Conference – Dublin*, pp. 122-130.

SMEATON, A.F. 1995. Linguistic Approaches to Text Management: An Appraisal of Progress. *Journal of Document and Text Management*, vol. 2(2), pp. 67-80.

VAN RIJSBERGEN, C.J. 1979. *Information Retrieval (second edition)*. Butterworths, London.

VAN RIJSBERGEN, C.J., AND LALMAS, M. 1996. An information calculus for information retrieval. *Journal of American Society for Information Science*, vol. 47(5), to appear.