From Uncertain Inference to Probability of Relevance for Advanced IR Applications

Henrik Nottelmann and Norbert Fuhr

Institute of Informatics and Interactive Systems, University of Duisburg-Essen,
47048 Duisburg, Germany, {nottelmann,fuhr}@uni-duisburg.de

Abstract. Uncertain inference is a probabilistic generalisation of the logical view on databases, ranking documents according to their probabilities that they logically imply the query. For tasks other than ad-hoc retrieval, estimates of the actual probability of relevance are required. In this paper, we investigate mapping functions between these two types of probability. For this purpose, we consider linear and logistic functions. The former have been proposed before, whereas we give a new theoretic justification for the latter. In a series of upper-bound experiments, we compare the goodness of fit of the two models. A second series of experiments investigates the effect on the resulting retrieval quality in the fusion step of distributed retrieval. These experiments show that good estimates of the actual probability of relevance can be achieved, and the logistic model outperforms the linear one. However, retrieval quality for distributed retrieval (only merging, without resource selection) is only slightly improved by using the logistic function.

1 Introduction

Probabilistic models are widely used in information retrieval: Besides the fact that even classical 'non-probabilistic' models can be given a probabilistic interpretation [20], current language models extend classical probabilistic models by methods for considering specific representations of documents and queries. The key advantage of probabilistic models is their underlying theoretic justification, the Probability Ranking Principle (PRP) [15]. The PRP states that optimum retrieval (defined with respect to (w. r. t.) document representations) is given if the documents are ranked according to their probability $Pr(\text{rel}|d,q)$ that document d is relevant to a user query q ("probability of relevance").

For ad-hoc retrieval, probabilistic IR algorithms do not have to estimate these probabilities of relevance directly; instead it is sufficient to rank the documents according to the documents' retrieval status values (RSVs) if these are monotonically increasing with $Pr(\text{rel}|d,q)$ (where d denotes a document, q a query and rel stands for the event that this relationship is judged relevant by the user). For example the well-known binary independence retrieval model ranks documents according to the RSVs $Pr(d|\text{rel},q)/Pr(d|\neg\text{rel},q)$. Computation of these RSVs is easier than estimating the exact probabilities of relevance.

Classical probabilistic IR models are relevance-oriented, i.e. they explicitly refer to the fact that a query-document relationship is judged relevant by a user. In contrast, Rijsbergen [18] introduced a new paradigm for probabilistic information retrieval, namely

uncertain inference. This generalisation of the logical view on databases aims at computing the probability $Pr(q \leftarrow d)$ that document d logically implies a given query q ("probability of inference"). As the relationship between $Pr(q \leftarrow d)$ and $Pr(rel|q,d)$ is assumed to be monotonically increasing, it is sufficient to rank documents w. r. t. the probabilities of inference. As a consequence, little effort has been spent so far on approximating the relationship. Rijsbergen [19] proposed a linear function. In this paper we show that a logistic function yields better results. Logistic functions have been used in different application areas within IR for quite some time, e.g. for text categorisation [6] or retrieval functions [**?**, 7] (logistic variant of the model proposed in [5]).

Although estimating the probabilities of inference is sufficient for ad-hoc retrieval, the filtering task requires the probabilities of relevance, and thus a mapping function is necessary: Following the decision-theoretic justification of the PRP, let C_r ($\bar{C_r}$) denote the costs for retrieving a relevant (non-relevant) document; similarly, C_o ($\bar{C_o}$) is the cost for omitting a relevant (non-relevant) document from the retrieved set. Then the filtering task corresponds to the problem of determining the cut-off point for ranked retrieval: A document should only be presented to the user if the costs for retrieval are lower than those for omitting the document, i.e.

$$Pr(rel|q,d) \cdot C_r + [1 - Pr(rel|q,d)] \cdot \bar{C_r} < Pr(rel|q,d) \cdot C_o + [1 - Pr(rel|q,d)] \cdot \bar{C_o} \quad . \quad (1)$$

Our current research focuses on distributed IR, where we also need estimates for the probability of relevance: The decision-theoretic framework [11, 4] for resource selection (the task to determine the best collections to be searched) aims at estimating the number of relevant documents;[1] for this, the probabilities of relevance of the top-ranked documents have to be approximated. The probabilities of relevance also play an important role in the fusion step of distributed retrieval, where the documents retrieved from the selected collections have to be merged in order to get a single ranked list; for this application, we describe experiments below.

This paper is organised as follows: Section 2 mapping function), and the drawbacks of this specific function. Section 3 proposes the logistic function as an alternative function for mapping the probabilities of inference onto probabilities of relevance. Section 4 reports on the results of our evaluation; we measured the overall quality of the estimates of the real probabilities as well as their impact on retrieval quality in the context of distributed IR. Finally, Sec. 5 contains concluding remarks and an outlook on future work.

2 Probabilistic Information Retrieval

In this section we describe uncertain inference, a probabilistic generalisation of the logical view on databases, and the standard way of mapping the probability of inference onto the probability of relevance. Finally, we present the drawbacks of this specific linear mapping function.

[1] In contrast to resource-ranking algorithms like GlOSS [9, 8] or CORI [1] which only compute a matching score between collections and the given query.

2.1 Probabilistic IR as Uncertain Inference

Risjbergen's [18] paradigm of IR as uncertain inference can be seen as a generalisation of the logical view on databases, where queries and document contents are treated as logical formulae. For a given query q, the database only returns those documents d which logically imply the query, i.e. it proves $q \leftarrow d$. In order to satisfy this formula, external knowledge like a thesaurus or an ontology can be included as well.

For considering the intrinsic uncertainty of information retrieval, Rijsbergen used probabilistic inference. Thus, probabilistic IR can be interpreted as estimating the probability $Pr(q \leftarrow d)$ that the document logically implies the query. Rijsbergen pointed out that this probability should not be considered in the traditional sense, i.e. $Pr(q \leftarrow d) \not\equiv Pr(\neg d \wedge q)$, but as the conditional probability $Pr(q|d)$.

In this paper, we assume that a query q is represented as a set of terms, where each term t has a query term weight $Pr(q \leftarrow t)$. Similar, a document d is represented as a set of terms with probabilistic indexing weights $Pr(t \leftarrow d)$ for each term t.

Assuming disjointness of query terms, the widely used linear retrieval function [17, 20] can be applied for computing the probability of inference:

$$Pr(q \leftarrow d) = \sum_{t \in q} Pr(q \leftarrow t) \cdot Pr(t \leftarrow d) \quad . \tag{2}$$

2.2 Uncertain Inference and Probabilities of Relevance

So far, this model does not cope with the concept of relevance. If we assume a monotonically increasing function mapping the probability of inference onto the probability of relevance, it is sufficient to rank documents according to $Pr(q \leftarrow d)$. Thus, only little work has been done on determining a "good" mapping function.

One of the possible mappings from the probability of inference $Pr(q \leftarrow d)$ onto the probability of relevance $Pr(\text{rel}|q,d)$ is given in [19] based on the total probability theorem:

$$Pr(\text{rel}|q,d) = Pr(\text{rel}|q \leftarrow d) \cdot Pr(q \leftarrow d) + Pr(\text{rel}|\neg(q \leftarrow d)) \cdot Pr(\neg(q \leftarrow d)) \quad . \tag{3}$$

This equation can be transformed into:

$$\begin{aligned} Pr(\text{rel}|q,d) &= Pr(\text{rel}|q \leftarrow d) \cdot Pr(q \leftarrow d) + Pr(\text{rel}|\neg(q \leftarrow d)) \cdot Pr(\neg(q \leftarrow d)) && (4)\\ &= Pr(\text{rel}|q \leftarrow d) \cdot Pr(q \leftarrow d) + Pr(\text{rel}|\neg(q \leftarrow d)) \cdot [1 - Pr(q \leftarrow d)] && (5)\\ &= Pr(\text{rel}|\neg(q \leftarrow d)) + [Pr(\text{rel}|q \leftarrow d) - Pr(\text{rel}|\neg(q \leftarrow d))] \cdot Pr(q \leftarrow d) && (6)\\ &= f(Pr(q \leftarrow d)) \quad . && (7) \end{aligned}$$

Thus, we have an affine linear mapping function:

$$f(x) := Pr(\text{rel}|\neg(q \leftarrow d)) + [Pr(\text{rel}|q \leftarrow d) - Pr(\text{rel}|\neg(q \leftarrow d))] \cdot x = c_0 + c_1 \cdot x \quad . \tag{8}$$

This mapping function is query-specific. As the parameters $Pr(\text{rel}|q \leftarrow d)$ and $Pr(\text{rel}|\neg(q \leftarrow d))$ are unknown as long as no relevance judgements are available, query-independent constants are used instead.

Function (8) can be further simplified if we assume $Pr(\text{rel}|\neg(q \leftarrow d)) \approx 0$ [19]. Then, we obtain

$$f(x) := Pr(\text{rel}|q \leftarrow d) \cdot x = c_1 \cdot x \quad . \tag{9}$$

This linear mapping function is used in the original version of the decision-theoretic framework [11, 4] for resource selection.

2.3 Drawbacks of the Linear Model

The linear model described above has several drawbacks.

First, it does not ensure that the results are between 0 and 1 in the general case of $c_0, c_1 \in \mathbb{R}$. In other words, the result cannot necessarily be regarded as a probability.

Furthermore, experiments show that the relationship between the probability of inference and the probability of relevance is not a linear one (see Sec. 4.2).

Finally, the linear function $c_1 \cdot x$ has the additional disadvantage that the probability of relevance cannot be larger than the probability of inference (interpreting the linear factor c_1 as the conditional probability $Pr(\text{rel}|q \leftarrow d) \leq 1$). If we allow for an affine linear function and/or any real values $c_0, c_1 \in \mathbb{R}$, this problem can be solved. Anyway, as we show in the next two sections, even these more general functions are inappropriate for modelling the shape of the actual mapping function.

3 Logistic Functions

In this section, we propose an alternative to the linear mapping functions described above.

For this, we take a closer look on the ideal situation. Here, exactly the documents in the ranks $1, \ldots, l$ are relevant, and the documents in the remaining ranks $l+1, \ldots$ are irrelevant. Let a be the probability of inference of the documents in rank l (i.e., the lowest probability of inference of any relevant documents). Then, the relationship function should be a step function

$$f(x) := \begin{cases} 1 & , \quad x >= a, \\ 0 & , \quad x < a \end{cases} \quad . \tag{10}$$

Obviously, no information retrieval system can ensure this requirement. Thus, in general some of the top-ranked documents are irrelevant, and some of the lower-ranked documents are relevant. But, less documents with lower probabilities of inference should be relevant than documents with higher probabilities. In other words, the probability that any arbitrary document is relevant should decrease with decreasing probability of inference.

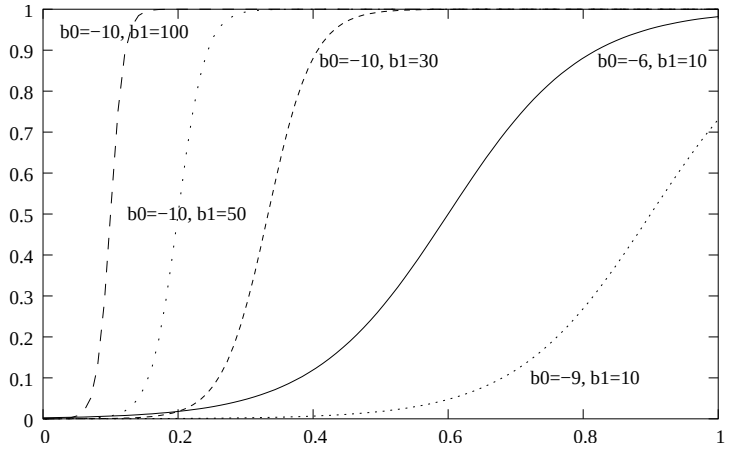
For modelling this characteristics, we want a continuous function f which approximates the discrete step function (10). Obviously, the pure and affine linear functions (8,9) are not appropriate.

Instead, one good candidate is the logistic function [2, 3]

$$f : [0,1] \rightarrow [0,1], \; f(x) := \frac{\exp(b_0 + b_1 \cdot x)}{1 + \exp(b_0 + b_1 \cdot x)} \tag{11}$$

with the two parameters b_0 and b_1. Figure 1 depicts some logistic functions with different parameters. One of the nice properties of logistic functions is that the result is always in $[0,1]$. In addition, a large variety of curves can be obtained by varying b_0 and b_1. The curve can be moved along the x axis by varying b_0; the slope can be adjusted by varying b_1.

Fig. 1. Example logistic functions



The plots in Fig. 2 (in Sec. 4.2) indicate that the logistic function outperforms the linear function. A more detailed analysis is given later in this paper.

The parameters b_0 and b_1 of the logistic function are query-specific (similar to the parameters c_0 and c_1 of the linear function).

For learning the parameters b_0 and b_1, a learning sample is required. This sample contains the tuples (document, query, probability of inference, relevance judgement). Optimum parameters can then be computed by means of regression methods. Possible optimisation criteria are maximum likelihood [6] or least-square polynomials [12]. In both cases extrema of a function (the likelihood function or the square error) have to be determined, i.e. the points where the first derivative equals zero. The resulting equation cannot be solved directly; instead, an iterative method (e.g. Newton-Raphson) has to be applied.

Usually the parameters are unknown before retrieval, and relevance feedback data is not available. Therefore, global parameters (learned with a sample based on several queries) have to be used.

4 Evaluation

This section presents the experiments we conducted. For a fair comparison, we used the affine linear variant as it has the same number of degrees of freedom as the logistic function.

In the following, we first describe the test-bed, including the documents, the queries, the relevance judgements and the learning algorithms.

Our first evaluation step is to prove our hypothesis that probabilities of inference $Pr(q \leftarrow d)$ can be mapped onto probabilities of relevance $Pr(\text{rel}|q,d)$ using a logistic function. For this, we investigate the approximation errors of the probabilities of relevance with logistic and linear functions. This is an upper bound experiment, since we use complete relevance judgements.

The second series of experiments focuses on the effect of the choice of the mapping function on the resulting retrieval quality. For this purpose, we investigate the case of distributed retrieval, where documents are merged according to the estimates of $Pr(\text{rel}|q,d)$. In this case, training and test samples are disjoint, so the outcome is representative for this type of retrieval task.

4.1 Experimental Setup

We used the TREC-123 test bed with the CMU 100 collection split [1]. The collections are of roughly the same size (about 33 megabytes), but vary in the number of documents they contain. The documents inside a collection are from the same source and the same time-frame. Table 1 depicts the summarised statistics for this 100 library test-bed.

Table 1. Summarised statistics for the 100 collection test-bed

Collection	Minimum	Average	Maximum
Documents	752	10,782	33,723
Bytes	28,070,646	33,365,514	41,796,822

The document index only contains the `<text>` sections of the documents. Terms are indexed employing a modified BM25 weighting scheme [16]:

$$P(t \leftarrow d) := \frac{tf(t,d)}{tf(t,d) + 0.5 + 1.5 \cdot \frac{dl(d)}{avgdl}} \cdot \frac{\log \frac{|DL|}{df(t)}}{\log |DL|} \quad . \tag{12}$$

Here, $tf(t,d)$ is the term frequency (number of times term t occurs in document d), $dl(d)$ denotes the document length (number of terms in document d), $avgdl$ the average document length, $|DL|$ the collection size (number of documents), and $df(t)$ the document frequency (number of documents containing term t).

We modified the standard BM25 formula by the normalisation component $1/\log|DL|$ to ensure that indexing weights are always in the closed interval $[0,1]$, and can thus be regarded as a probability.

The resulting indexing weights are rather small; but this can be compensated by the linear components of both the linear and the logistic mapping function (see also below).

Queries are based on TREC topics 51–100 and 101–150 [10], respectively. We use three different sets of queries:

1. Short queries, where we only used the `<title>` field. Short queries contain between 1 and 7 terms (average 3.3), and are similar to those submitted to a WWW search engine.
2. Mid-length queries, where we only used the `<description>` field. Here, queries typically contain between 4 and 19 terms (average 9.9), and may be used by advanced searchers.
3. Long queries, where we used all fields. These queries contain 39–185 terms (average 87.5) and are common in TREC-based evaluations.

Normalised tf values are used as query term weights:

$$P(q \leftarrow t) := \frac{tf(t,q)}{ql(q)} \quad . \tag{13}$$

Here, $tf(t,q)$ denotes the term frequency (number of times a term t occurs in query q), and $ql(q) := \sum_{t \in q} tf(t,q)$ is the query length (number of terms in query q).

For both documents and queries, terms are stemmed (using the Porter stemmer [13]), and stop words (the TREC "common words") are removed.

The relevance judgements are the standard TREC relevance judgements [10], documents with no judgement are treated as irrelevant.

In our experiments, we use the Gnuplot[2] implementation of the nonlinear least-squares (NLLS) Marquardt-Levenberg algorithm [14] for learning the parameters c_0, c_1 of the linear function and b_0, b_1 of the logistic function. NLLS does not ensure further properties of the learned parameters, i.e. we obtain optimum parameters $c_0, c_1 \in \mathbb{R}$. These parameters do not match the underlying theory

$$Pr(\text{rel}|q \leftarrow d) = c_0 + c_1 \qquad Pr(\text{rel}|\neg(q \leftarrow d)) = c_0 \tag{14}$$

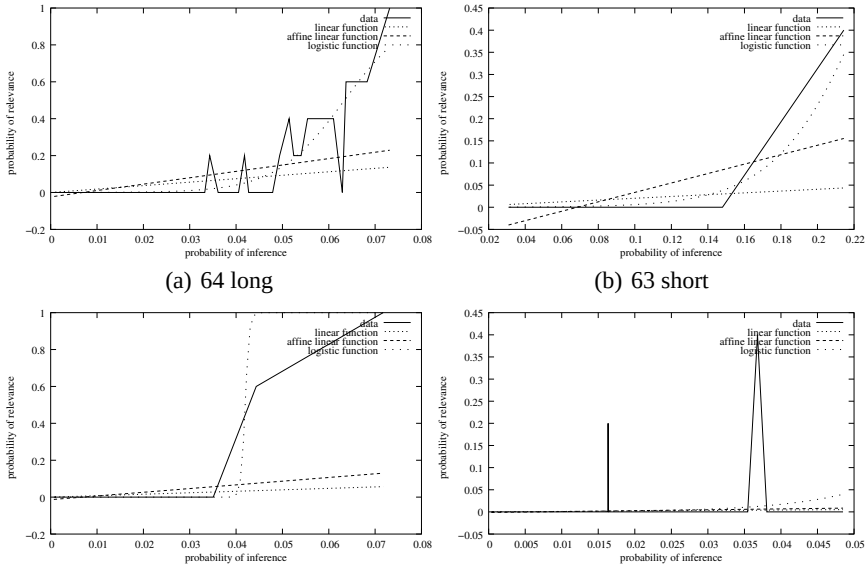
as in general $Pr(\text{rel}|q \leftarrow d)$ and $Pr(\text{rel}|\neg(q \leftarrow d))$ cannot be regarded as probabilities. However, as we learn optimum parameters, this can only lead to an increased approximation quality for the linear model (w. r. t. parameters derived from the underlying theory), and so it does not favor our new model (the logistic function).

4.2 Approximation Errors of the Mapping Functions

In our first set of experiments we evaluated our hypothesis that the mapping between probabilities of inferences $Pr(q \leftarrow d)$ and probabilities relevance $Pr(\text{rel}|q,d)$ can be approximated by a logistic function. For this purpose, we performed an upper-bound experiment where we have complete relevance judgements, and then measured the goodness of fit of the two models.

[2] `http://www.ucc.ie/gnuplot/gnuplot.html`

Fig. 2. Example queries on TREC collection ap88_6, showing average probability of relevance, logistic and linear fit



(a) 64 long (b) 63 short

(c) 78 long (d) 73 long

We randomly chose 10 out of the 100 TREC-123 collections. The characteristics of these 10 collections are rather different but representative for the whole 100 collection test-bed; the collection sizes are depicted in Tab. 1.

For each of the TREC topics 51–100 we learned query-specific parameters b_0, b_1 and c_0, c_1, respectively, and used these parameters for evaluating the same query (optimal parameters). In the result plots in Fig. 2, the x-axis denotes the probabilities of inference and the y-axis the probabilities of relevance. The three curves represent the actual relevance judgements (where probabilities of inference and the corresponding relevance judgements of the documents in ranks 1–5, 6–10, 11–15 ... are averaged), the logistic function and the affine linear function.

Where our retrieval algorithm performs well (Fig. 2a, 2b, 2c), the logistic function obviously is a better approximation than the linear functions. For other queries (here, Fig. 2d), the retrieval algorithm performs badly (i.e., the top-ranked documents are irrelevant), and the logistic function does not improve approximation quality in comparison to the linear ones.

Plots can only give a brief overview of the resulting quality. Thus, we also computed the mean square approximation error of the relevance judgement for all of the 10 selected collections for short, mid-length and long queries (pure linear, affine linear and logistic mapping function). The results are listed in Tab. 2.

Table 2. Evaluation results: mean square approximation error (in 10^{-3}) and improvement (in %)

(a) Affine linear function $c_0 + c_1 \cdot x$

		short queries			mid-length queries			long queries		
Collection	size	alin.	log.	Δ	alin.	log.	Δ	alin.	log.	Δ
ap88_6	9210	307	270	+11.9	135	121	+11.0	079	058	+26.2
ap89_5	10542	339	291	+14.1	153	127	+17.4	087	057	+34.8
ap90_4	9955	581	518	+10.7	261	229	+12.4	155	121	+21.8
ap90_6	8934	461	418	+9.4	208	188	+10.0	127	099	+21.9
ap90_7	9794	524	487	+7.1	235	218	+7.2	136	108	+21.0
patn3_5	1021	158	137	+13.2	078	072	+7.1	054	049	+10.4
wsj88_3	13602	355	318	+10.4	138	120	+12.9	074	055	+26.6
wsj90_2	10560	370	342	+7.4	158	144	+8.9	094	079	+16.3
ziff1_2	8847	207	189	+8.7	082	076	+7.1	049	042	+13.9
ziff3_9	9670	233	224	+4.1	086	083	+4.2	053	050	+6.1

(b) Linear function $c_1 \cdot x$

		short queries			mid-length queries			long queries		
Collection	size	lin.	log.	Δ	lin.	log.	Δ	alin.	log.	Δ
ap88_6	9210	330	270	+18.0	140	121	+14.0	084	058	+30.2
ap89_5	10542	381	291	+23.7	159	127	+20.5	093	057	+39.3
ap90_4	9955	635	518	+18.5	277	229	+17.5	166	121	+26.9
ap90_6	8934	501	418	+16.6	217	188	+13.4	133	099	+25.4
ap90_7	9794	554	487	+12.1	240	218	+9.4	143	108	+24.7
patn3_5	1021	165	137	+16.7	079	072	+8.7	056	049	+13.1
wsj88_3	13602	379	318	+16.0	142	120	+15.8	078	055	+30.1
wsj90_2	10560	388	342	+11.6	163	144	+11.5	097	079	+18.8
ziff1_2	8847	213	189	+11.1	083	076	+8.5	050	042	+15.8
ziff3_9	9670	242	224	+7.8	089	083	+6.4	055	050	+9.1

Using a logistic function instead of a pure or affine linear one always reduces the mean square error; for the affine linear function, the improvement ranges between 4.1% (collection ziff3_9, mid-length queries) to 34.8% (collection ap89_5, long queries). The improvement is higher for long queries than for short and mid-length queries; the same is true for the approximation error (this seems due to the fact that the retrieval algorithms performs better for long queries, i.e. more relevant documents are top-ranked). Obviously, the difference is significant (assuming a simple sign test over the ten subcollections).

The reported differences between the pure/affine linear and logistic mapping functions are slightly biased in favor of the linear function, as we map results of the linear function outside $[0,1]$ onto the corresponding margin (as we learned parameters $c_0, c_1 \in \mathbb{R}$ for the linear mapping function, thus the range is not restricted to $[0,1]$). Without this mapping, the quality is slightly worse.

As a consequence, the bad quality of the linear approximation of the relationship between the probability of inference and the probability of relevance cannot be blamed on the fact that the results of the linear function are not always in $[0,1]$.

In addition, Fig. 3 shows the mean square error when retrieving $1,\ldots,100$ documents (as we are more interested in the top-ranked documents). The average is computed over all 10 selected collections and topics 51–100; computation is done separately for short, mid-length and long queries.

One can see that the logistic function performs much better for the top-ranked documents (which are typically the most interesting documents in retrieval). The difference between the two approximation errors decreases for the low-ranked documents; here the two linear functions and the logistic function are very close to zero, and most of the documents are irrelevant (yielding a very low square error for the lower ranks).

These results show a significant improvement for the logistic function, especially for the most interesting ranks.

4.3 Retrieval Quality in Distributed IR

In our second set of experiments, we want to test whether a logistic relationship function improves quality for distributed IR.

All 100 TREC-123 collections are used for this experiment. For every collection, we learned the mapping functions (i.e., their parameters b_0, b_1 and c_o, c_1, respectively) with TREC topics 51–100 (only the first 30 documents per query) and evaluated them with topics 101–150 and vice versa (cross evaluation). In this experiment, we are not interested in the influence of resource selection, thus retrieval is performed always on all collections (without resource selection), the top 30 documents of each collection are retrieved, and the resulting 3,000 documents are merged according to the approximations of $Pr(\text{rel}|q,d)$ (with collection-specific parameters).

Average precision at ranks 5, 10, 15, 20 and 30 is given in Tab. 3. Table 3(a) contains the results for the affine linear function $c_0 + c_1 \cdot x$; Table 3(b) contains the results for the linear function $c_1 \cdot x$.

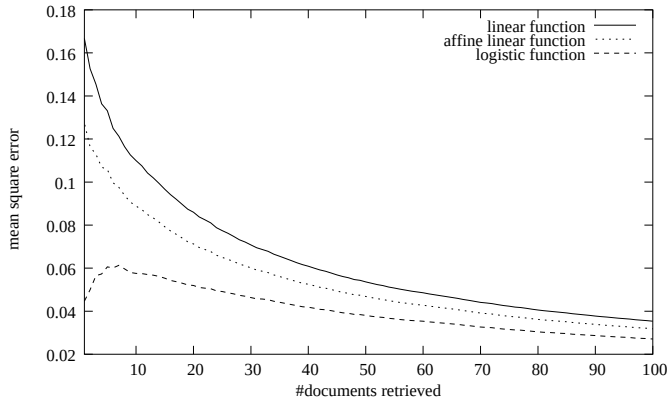
The results show a significant improvement when an affine linear function is used instead of a pure linear one (particularly for mid-length and long query). This is interesting as it contrasts the assumption that the conditional probability $Pr(\text{rel}|\neg(q \leftarrow d))$ can be neglected.

Precision in the top ranks can also be increased by using a logistic mapping function instead of an affine linear one, although this improvement is quite small (and in a few cases precision also decreases).

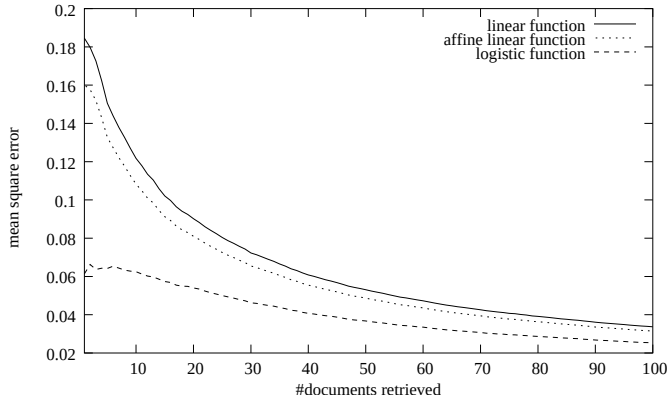
In addition, recall/precision graphs for the linear, the affine linear as well as the logistic mapping function are given in Fig. 4.

Here the recall-precision graphs of the affine linear function are much better than the graphs for the pure linear one (especially for the mid-length and long queries). The difference between the graphs for the affine linear function and the logistic function is rather small, however on average we obtained slight improvements by using the logistic function.

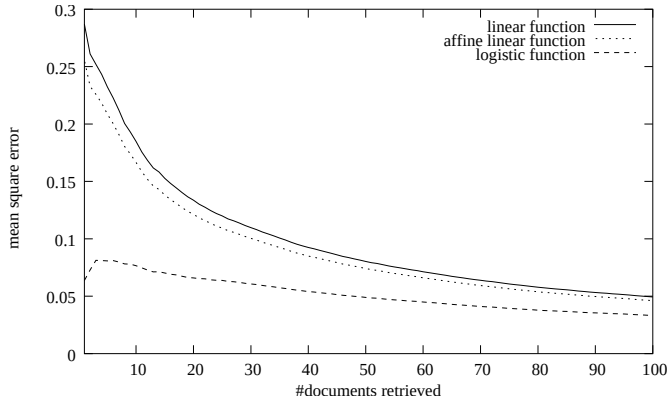
Fig. 3. Mean square approximation error when retrieving $1, \ldots, 100$ documents



(a) short queries

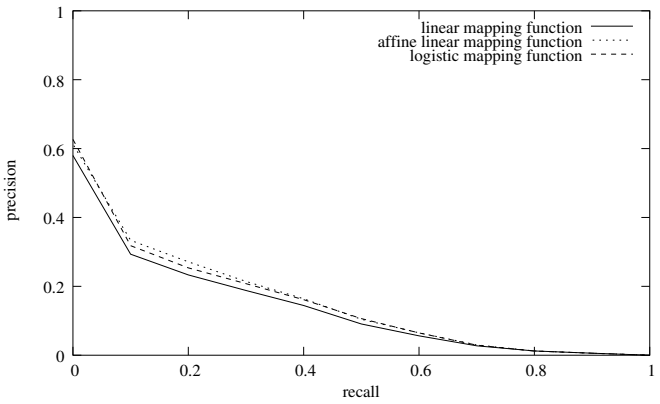


(b) mid-length queries

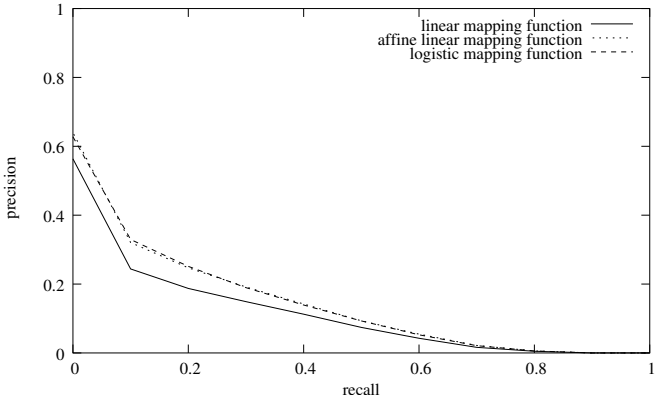


(c) long queries

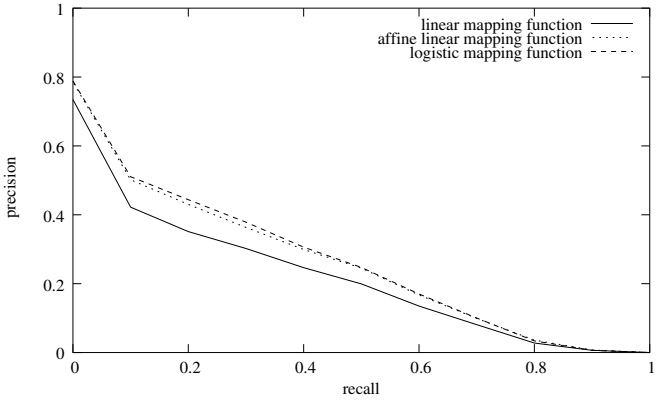
Fig. 4. Recall/precision for logistic and linear function



(a) short queries



(b) mid-length queries



(c) long queries

Table 3. Average precision at given ranks and improvement (in %)

(a) Affine linear function $c_0 + c_1 \cdot x$

	short queries			mid-length queries			long queries		
Rank	alin.	log.	Δ	alin.	log.	Δ	alin.	log.	Δ
5	0.4000	0.3860	-3.5	0.4200	0.4240	+1.0	0.5940	0.5660	-4.7
10	0.3850	0.3800	-1.3	0.3940	0.4050	+2.8	0.5670	0.5560	-1.9
15	0.3840	0.3787	-1.4	0.3820	0.3967	+3.8	0.5467	0.5473	+0.1
20	0.3835	0.3745	-2.3	0.3790	0.3910	+3.2	0.5375	0.5460	+1.6
30	0.3703	0.3567	-3.7	0.3693	0.3687	-0.2	0.5177	0.5320	+2.8

(b) Linear function $c_1 \cdot x$

	short queries			mid-length queries			long queries		
Rank	lin.	log.	Δ	lin.	log.	Δ	lin.	log.	Δ
5	0.3620	0.3860	+6.6	0.3580	0.4240	+18.4	0.5300	0.5660	+6.8
10	0.3730	0.3800	+1.9	0.3350	0.4050	+20.9	0.5030	0.5560	+10.5
15	0.3587	0.3787	+5.6	0.3127	0.3967	+26.9	0.4787	0.5473	+14.3
20	0.3495	0.3745	+7.2	0.3080	0.3910	+26.9	0.4755	0.5460	+14.8
30	0.3297	0.3567	+8.2	0.2930	0.3687	+25.8	0.4563	0.5320	+16.6

5 Conclusion

In this paper, we have investigated the relationship between the probability of inference $Pr(q \leftarrow d)$ and the probability of relevance $Pr(\text{rel}|q,d)$. In the past, little effort has beeen spent on estimating the latter, since the most popular retrieval task—ad-hoc retrieval—needs only a monotonic function of this probability in order to yield a ranking according to the probability ranking principle.

However, advanced IR applications are based on estimates of the actual probabilities of relevance, e.g. for approximating the number of relevant documents in the result set for resource selection (decision-theoretic framework [11, 4]) or for merging the documents retrieved from the selected collections into a single ranked list.

Rijsbergen proposed a linear mapping function for modelling the relationship between the probability of inference and the probability of relevance. Here we showed that this approach has only a moderate approximation quality, in particular in the top-ranked documents.

In this paper we prososed the use of a logistic function. The logistic function can be justified from a theoretical point of view, as it is a continuous approximation of the discrete step function (the ideal relationship). Our experiments showed a significant approximation improvement compared to a linear function. Quality for distributed retrieval (without resource selection) can be significantly improved using an affine linear instead of a linear function, and the results for the logistic function are slightly better than those for the affine linear one (in addition to the nice properties of the logistic function, e.g. the fact that the resulting values are always between zeo and one).

In future, we will investigate in more detail how many documents per query and how many queries are required for obtaining good parameters.

Furthermore, we will extend the decision-theoretic framework [4] so that it can use a logistic mapping function [11] and evaluate the quality compared to the linear function and other resource selection methods. The experiments we conducted for distributed retrieval did not include any resource selection, but this is not realistic. Thus we will investigate whether we can improve retrieval quality when using a logistic function for resource selection.

6 Acknowledgements

This work is supported by the EU commission under grant IST-2000-26061 (project MIND).

A major part of this work was performed while both authors where affiliated at the University of Dortmund.

References

[1] J. Callan and M. Connell. Query-based sampling of text databases. *ACM Transactions on Information Systems*, 19(2):97–130, 2001.

[2] S. Fienberg. *The Analysis of Cross-Classified Categorial Data*. MIT Press, Cambridge, Mass., 2. edition, 1980.

[3] D. Freeman. *Applied Categorial Data Analysis*. Dekker, New York, 1987.

[4] N. Fuhr. A decision-theoretic approach to database selection in networked IR. *ACM Transactions on Information Systems*, 17(3):229–249, 1999.

[5] N. Fuhr and C. Buckley. A probabilistic learning approach for document indexing. *ACM Transactions on Information Systems*, 9(3):223–248, 1991.

[6] N. Fuhr and U. Pfeifer. Combining model-oriented and description-oriented approaches for probabilistic indexing. pages 46–56, New York, 1991. ACM.

[7] F. C. Gey. Inferring probability of relevance using the method of logistic regression. In B. W. Croft and C. J. van Rijsbergen, editors, *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–231, London, et al., 1994. Springer-Verlag.

[8] L. Gravano and H. Garcia-Molina. Generalizing GIOSS to vector-space databases and broker hierarchies. pages 78–89, Los Altos, California, 1995. Morgan Kaufman.

[9] L. Gravano, H. Garcia-Molina, and A. Tomasic. The effectiveness of GlOSS for the text database discovery problem. In R. T. Snodgrass and M. Winslett, editors, *Proceedings of the 1994 ACM SIGMOD. International Conference on Management of Data.*, pages 126–137, New York, 1994. ACM.

[10] D. Harman, editor. *The Second Text REtrieval Conference (TREC-2)*, Gaithersburg, Md. 20899, 1994. National Institute of Standards and Technology.

[11] H. Nottelmann and N. Fuhr. MIND resource selection framework and methods. Technical report, Universität Dortmund, Feb. 2002.

[12] M. Pollmann. Entwicklung und untersuchung von verbesserten probabilistischen indexierungsfunktionen für freitext-indexierung. Master's thesis, Universität Dortmund, Fachbereich Informatik, 1993.

[13] M. Porter. An algorithm for suffix stripping. *Program*, 14:130–137, July 1980.

[14] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, editors. *Nested Relations and Complex Objects in Databases*. Cambridge University Press, 1992.

[15] S. Robertson. The probability ranking principle in IR. *Journal of Documentation*, 33:294–304, 1977.

[16] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at TREC. In *Text REtrieval Conference*, pages 21–30, 1992.

[17] H. Turtle and W. Croft. Efficient probabilistic inference for text retrieval. pages 644–661, Paris, France, 1991. Centre de Hautes Etudes Internationales d'Informatique Documentaire (CID).

[18] C. J. van Rijsbergen. A non-classical logic for information retrieval. *The Computer Journal*, 29(6):481–485, 1986.

[19] C. J. van Rijsbergen. Probabilistic retrieval revisited. *The Computer Journal*, 35(3):291–298, 1992.

[20] S. Wong and Y. Yao. On modeling information retrieval with probabilistic inference. *ACM Transactions on Information Systems*, 13(1):38–68, 1995.