A Language Modeling Approach to Information Retrieval

Jay M. Ponte and W. Bruce Croft
Computer Science Department
University of Massachusetts, Amherst
{ponte, croft}@cs.umass.edu

Abstract Models of document indexing and document retrieval have been extensively studied. The integration of these two classes of models has been the goal of several researchers but it is a very difficult problem. We argue that much of the reason for this is the lack of an adequate indexing model. This suggests that perhaps a better indexing model would help solve the problem. However, we feel that making unwarranted parametric assumptions will not lead to better retrieval performance. Furthermore, making prior assumptions about the similarity of documents is not warranted either. Instead, we propose an approach to retrieval based on probabilistic language modeling. We estimate models for each document individually. Our approach to modeling is non-parametric and integrates document indexing and document retrieval into a single model. One advantage of our approach is that collection statistics which are used heuristically in many other retrieval models are an integral part of our model. We have implemented our model and tested it empirically. Our approach significantly outperforms standard *tf.idf* weighting on two different collections and query sets.

1 Introduction

Over the past three decades, probabilistic models of document retrieval have been studied extensively. In general, these approaches can be characterized as methods of estimating the probability of relevance of documents to user queries. One component of a probabilistic retrieval model is the indexing model, i.e., a model of the assignment of indexing terms to documents. We argue that the current indexing models have not led to improved retrieval results. We believe this is due to two unwarranted assumptions made by these models. We have taken a different approach based on non-parametric estimation that allows us to relax these assumptions. We have implemented our approach and empirical results on two different collections and query sets are significantly better than the standard *tf.idf* method of retrieval. Now we take a brief look at some existing models of document indexing.

We begin our discussion of indexing models with the 2-Poisson model, due to Bookstein and Swanson [1] and also to Harter [7]. By analogy to manual indexing, the task was to assign a subset of words contained in a document (the 'specialty words') as indexing terms. The probability model was intended to indicate the useful indexing terms by means of the differences in their rate of occurrence in documents 'elite' for a given term, i.e., a document that would satisfy a user posing that single term as a query, vs. those without the property of eliteness.

The success of the 2-Poisson model has been somewhat limited but it should be noted that Robertson's *tf*, which has been quite successful, was intended to behave similarly to the 2-Poisson model [12].

Other researchers have proposed a mixture model of more than two Poisson distributions in order to better fit the observed data. Margulis proposed the n-Poisson model and tested the idea empirically [10]. The conclusion of this study was that a mixture of n-Poisson distributions provides a very close fit to the data. In a certain sense, this is not surprising. For large values of n one can fit a very complex distribution arbitrarily closely by a mixture of n parametric models if one has enough data to estimate the parameters [18]. However, what is somewhat surprising is the closeness of fit for relatively small values of n reported by Margulis [10].

Nevertheless, the n-Poisson model has not brought about increased retrieval effectiveness in spite of the close fit to the data. In any event, the semantics of the underlying distributions are less obvious in the n-Poisson case as compared to the 2-Poisson case where they model the concept of eliteness.

Apart from the adequacy of of the available indexing models, estimating the parameters of these models is a difficult problem. Researchers have looked at this problem from a variety of perspectives and we will discuss several of these of these approaches in section 2. In addition, as previously mentioned, many of the current indexing models make assumptions about the data that we feel are unwarranted.

- The parametric assumption.
- Documents are members of pre-defined classes.

In our approach we relax these two assumptions. Rather than making parametric assumptions, as is done in the 2-Poisson model it is assumed that terms follow a mixture of two Poisson distributions, as Silverman said, "the data will be allowed to speak for themselves [16]." We feel that it is unnecessary to construct a parametric model of the data when we have the actual data. Instead, we rely on non-parametric methods.

Regarding the second assumption, the 2-Poisson model was originally based on the idea of 'eliteness' [7]. It was assumed that a document elite for a given term would

satisfy a user if the user posed that single term as a query. Since that time, the prevailing view has come to be that multiple term queries are more realistic. In general, this requires a combinatorial explosion of elite sets for all possible subsets of terms in the collection. We take the view that each query needs to be looked at individually and that documents will not necessarily fall cleanly into elite and non-elite sets.

In order to relax these assumptions and to avoid the difficulties imposed by separate indexing and retrieval models, we have developed an approach to retrieval based on probabilistic language modeling. Our approach provides a conceptually simple but explanatory model of retrieval.

At this time, we should make clear what we mean by the word 'model.' In our view, the word 'model' is used in information retrieval in two senses. The first sense denotes an abstraction of the retrieval task itself. The best example of this is the vector space model which allows one to talk about the task of retrieval apart from implementation details such as storage media, and data structures [15]. A second sense of the word 'model' is the probabilistic sense where it refers to an explanatory model of the data. This was intention behind the 2-Poisson model.

We add a third sense of the word when we refer to language modeling. The phrase 'language model' is used by the speech recognition community to refer to a probability distribution that captures the statistical regularities of the generation of language [21]. In the context of the retrieval task, we can treat the generation of queries as a random process. Generally speaking, language models for speech attempt to predict the probability of the next word in an ordered sequence. For the purposes of document retrieval, one can model occurrences at the document level without regard to sequential effects and will be the approach taken here. It is also possible to model local predictive effects for features such as phrases but that will be left for future work. Regarding query generation as a random process, it is not the case that queries really are generated randomly, but it is the case that retrieval systems are not endowed with knowledge of the generation process. Instead, we will treat language generation as a random process modeled by a probability distribution and focus on the estimation of probabilities as a means of achieving effective retrieval.

Our approach to retrieval is to infer a language model for each document and to estimate the probability of generating the query according to each of these models. We then rank the documents according to these probabilities. This means that our data model and our discriminant function for retrieval are one and the same. The intuition behind our approach is that, in our view, users have a reasonable idea of terms that are likely to occur in documents of interest and will choose query terms that distinguish these documents from others in the collection, an intuition discussed in more detail in section 5. By focusing on the query generation probability as opposed to the probability of relevance, our model does not require us to make a set of inferences for indexing and a separate set of inferences for retrieval.

Most retrieval systems use term frequency, document frequency and document length statistics. Typically these are used to compute a *tf.idf* score with document length normalization. An example of this is the INQUERY ranking formula shown in section 4.3.

In our approach, collection statistics such as term frequency, document length and document frequency are integral parts of the language model and are not used heuristically as in many other approaches. For this reason, we do not use the standard *tf* and *idf* scores. In addition, length normalization is implicit in the calculation of the probabilities and does not have to be done in an *ad hoc* manner.

The remainder of the paper is organized as follows. In section 2 we review some existing retrieval models. Section 3 describes a language modeling approach that closely parallels the standard approach to IR. Section 4 shows the effectiveness of this model empirically. Finally we offer concluding remarks and describe future directions of this work.

2 Previous Work

As mentioned previously. the standard probabilistic indexing model is the 2-Poisson model. One of the assumptions of the model was that a subset of terms occurring in a document would be useful for indexing. According to Harter [7], such words can be identified by their distribution and thereby assigned as indexing terms. Documents were assumed to be of approximately equal length, a reasonable assumption for the data used in the initial studies [7]. This model is somewhat similar to ours if one views the probability of term assignment as analogous to the term generation probability. The two main differences are that we do not make distributional assumptions and we do not not distinguish a subset of specialty words or assume a preexisting classification of documents into elite and non-elite sets.

Two well known probabilistic approaches to retrieval are the Robertson and Sparck Jones model [14] and the Croft and Harper model [3]. Both of these models estimate the probability of relevance of each document to the query. Our approach differs in that we do not focus on relevance except to the extent that the process of query production is correlated with it.

An additional probabilistic model is that of Fuhr [4]. A notable feature of the Fuhr model is the integration of indexing and retrieval models. The main difference between this approach and ours is that in the Fuhr model the collection statistics are used in a heuristic fashion in order to estimate the probabilities of assigning concepts to documents. In our approach, we are able to avoid using heuristic methods since we are not inferring concepts from terms.

Another recent probabilistic approach is the INQUERY inference network model by Turtle and Croft [19]. Similar to the Fuhr model, Turtle and Croft integrate indexing and retrieval by making inferences of concepts from features. Features include words, phrases and more complex structured features. Evidence from multiple feature sets and multiple queries can be combined by means of a Bayesian network in order to infer the probability that the information need of the user has been met. This distinction between information need and query is a notable feature of this model. As previously noted, in our approach, we have shifted our emphasis from probability of relevance to probability of query production. We assume these are correlated but do not currently attempt to model that correlation explicitly. We will discuss this point further in section 5.

In section 3 we will discuss our probability estimation procedure. One statistic that we will be using is the average probability of term occurrence. A similar statistic was used by Kwok [9] for a different purpose. Kwok used

the unnormalized average tf to estimate the importance of a term with respect to the query. In our approach we use the average of tf normalized by document length in the estimation of the generation probability.

Wong and Yao [20] proposed a model in which they represented documents according to a probability distribution. They then developed two separate approaches to retrieval, one based on utility theory and the other based on information theory. Regarding the probability distribution, Wong and Yao use a maximum likelihood estimator for term probabilities. In our approach, we use a more robust estimator. Wong and Yao's utility and information theoretic retrieval models are somewhat analogous to other approaches to retrieval in that they have an indexing model apart from their retrieval model. Terms are associated with documents according to the maximum likelihood probability estimate and the discriminant is a utility theoretic or information theoretic function of this estimate. In our approach we have been able to avoid this extra complexity and perform retrieval according to a single probabilistic model.

The most similar approach to the one we have taken is that of Kalt [8]. In this model, documents are assumed to be generated by a stochastic process; a multinomial model. The task Kalt investigated was text classification. Each document was treated as a sample from a language model representing the class of that document. In this model, tf and document length are both integral parts of the model rather than being heuristics as they are in many other models. The discriminant function is taken to be the maximum likelihood estimator of the query given the document's language model. Note that the 'query' in this case was inferred from the training set in the context of the classification task.

While our approach is conceptually somewhat different, it is clearly related to the Kalt approach and shares the desirable property that the collection statistics are integral parts of the model. In our initial study we have deliberately not made any distributional assumptions. Another major difference from the Kalt approach is that we do not rely on the maximum likelihood estimator but instead use a more robust estimator. Next, we do not assume that documents were necessarily drawn from k language models representing the k classes of interest. Instead, we make a weaker assumption that we can get estimates of each document's language model individually without making inferences about the class membership of documents. We then use these models to compute the query generation probability. We now describe the development of our approach.

3 Model Description

As mentioned, we infer a language model for each document and rank according to our estimate of producing the query according to that model. We would like to estimate $\hat{p}(Q|M_d)$, the probability of the query given the language model of document d.

The maximum likelihood estimate of the probability of term t under the term distribution for document d is:

$$\hat{p}_{ml}(t|M_d) = \frac{tf_{(t,d)}}{dl_d}$$

where $tf_{(t,d)}$ is the raw term frequency of term t in document d and dl_d is the total number of tokens in document d. We assume that given a particular language model that the query terms occur independently. This gives rise to the ranking formula $\prod_{t \in Q} \hat{p}_{ml}(t,d)$ for each document. There are several problems with this estimator. The most obvious practical problem is that we do not wish to assign a probability of zero to a document that is missing one or more of the query terms. In addition to this practical consideration, from a probabilistic perspective, it is a somewhat radical assumption to infer that $p(t|M_d) = 0$. I.e., the fact that we have not seen it does not make it impossible. Instead, we make the assumption that a non-occurring term is possible, but no more likely than what would be expected by chance in the collection. I.e., $\frac{cf_t}{cs}$, where cf_t is the raw count of term t in the collection and cs is the raw collection size or the total number of tokens in the collection. This provides us with a more reasonable distribution and circumvents the practical problem. It should be noted that in homogeneous databases, one may need to use a more careful estimate of the collection probability since, in some cases the absence of a very frequently occurring word (i.e. a word with the characteristics of a stopword) could conceivably contribute more to the score of a document in which it does not occur. This is not a problem in the collections we have studied as they are heterogeneous in nature and stopwords have been removed, but we plan to address this issue in the future to be sure that our approach will be immune to these pathological cases.

The other problem with this estimator may be less obvious. If we could get an arbitrary sized sample of data from M_d we could be reasonably confident in the maximum likelihood estimator. However we only have a document sized sample from that distribution. To circumvent this problem we are going to need an estimate from a larger amount of data. That estimate is:

$$\hat{p}_{avg}(t) = \frac{\left(\sum_{d_{(t \in d)}} p_{ml}(t|M_d)\right)}{df_t}$$

where df_t is the document frequency of t. In other words, the mean probability of t in documents containing it. This is a robust statistic in the sense that we have a lot more data from which to estimate it but it too has a problem. We cannot and are not assuming that every document containing t is drawn from the same language model and so there is some risk in using the mean to estimate $p(t|M_d)$ furthermore if we used the mean by itself, there would be no distinction between documents with different term frequencies.

In order to benefit from the robustness of this estimator and to minimize the risk we will model the risk for a term t in a document d using the geometric distribution [22] as follows:

$$\hat{R}_{t,d} = \left(\frac{1.0}{(1.0 + \overline{f}_t)}\right) \times \left(\frac{\overline{f}_t}{(1.0 + \overline{f}_t)}\right)^{tf_{t,d}}$$

where $\overline{f}_t$ is the the mean term frequency of term t in documents where t occurs, i.e., $p_{avg}(t) \times dl_d$. Another way to say this is that $\overline{f}_t$ is the term count one would get if the term occurred at the average rate. The intuition behind this formula is that as the tf gets further away from the normalized mean, the mean probability becomes riskier to use as an estimate. For a somewhat related use of the geometric distribution see [5].

Now we will use this risk function as a mixing parameter in our calculation of $\hat{p}(Q|M_d)$, our estimate of the probability of producing the query for a given document model as follows:

Let,

$$\hat{p}(t|M_d) = \begin{cases} p_{ml}(t,d)^{(1.0-\hat{R}_{t,d})} \times p_{avg}(t)^{\hat{R}_{t,d}} & \text{if } tf_{(t,d)} > 0 \\ \frac{cf_t}{cs} & \text{otherwise} \end{cases}$$

Then,

$$\hat{p}(Q|M_d) = \prod_{t \in Q} \hat{p}(t|M_d) \times \prod_{t \notin Q} 1.0 - \hat{p}(t|M_d)$$

In this formula, the first term is the probability of producing the terms in the query and the second term is the probability of not producing other terms. Notice the risk function, $\hat{R}_{t,d}$ and the background probability $\frac{cf_t}{cs}$ mentioned earlier. We compute this function for each candidate document and rank accordingly. Now we describe our empirical results.

4 Empirical Results

4.1 Data

We performed recall/precision experiments on two data sets. The first set was TREC topics 202-250 on TREC disks 2 and 3, the TREC 4 ad hoc task and the second was TREC topics 51-100 on TREC disk 3 using the concept fields. We chose these query sets because they are quite different from each other. The 51-100 concept fields are essentially lists of good terms while topics 202-250 are 'natural language' queries consisting of one sentence each.

4.2 Implementation

We have implemented a research prototype retrieval engine known as Labrador to test our approach. This engine was originally implemented as a high throughput retrieval system in the context of our previous work on topic segmentation [13]. For these experiments, the system does tokenization, stopping and stemming in the usual way. We have implemented both standard *tf.idf* weighting as well our language modeling approach.

4.3 Recall/Precision Experiments

Table 1 shows the results for TREC topics 202-250 on TREC disks 2 and 3. In the figure we see the eleven point recall/precision results as well as the non-interpolated average precision and precision figures for the top N documents for several values of N. The first two columns compare the baseline result to our new approach. The baseline result was obtained using the INQUERY ranking formula which uses Robertson's *tf* score and a standard *idf* score and is defined as follows:

$$tfbel_{t,d} = \frac{tf_{t,d}}{tf_{t,d} + 0.5 + 1.5\frac{len_d}{avgdoclen}}$$

$$idf_t = \log(\frac{N+0.5}{docf_t}) / \log(N+1)$$

The third column reports the percent change. Column four is of the form I/D where I is the count of queries for which performance improved using the new method and D is count of queries for which performance was different. Column five reports significance values according to the sign test and column six does likewise according to the Wilcoxon test. The entries in these two columns marked with a star indicate a statistically significant difference at the 0.05 level. Note that these are one sided tests.

	tf.idf	LM	%chg	I/D	Sign	Wilc.
Rel:	6501	6501				
Rret.:	3201	3364	+5.09	36/43	0.0000★	0.0002★
Prec.						
0.00	0.7439	0.7590	+2.0	10/22	0.7383	0.5709
0.10	0.4521	0.4910	+8.6	24/42	0.2204	0.0761
0.20	0.3514	0.4045	+15.1	27/44	0.0871	0.0081★
0.30	0.2761	0.3342	+21.0	28/43	0.0330★	0.0054★
0.40	0.2093	0.2572	+22.9	25/39	0.0541	0.0158★
0.50	0.1558	0.2061	+32.3	24/35	0.0205★	0.0018★
0.60	0.1024	0.1405	+37.1	22/27	0.0008★	0.0027★
0.70	0.0451	0.0760	+68.7	13/15	0.0037★	0.0062★
0.80	0.0160	0.0432	+169.6	9/10	0.0107★	0.0035★
0.90	0.0033	0.0063	+89.3	2/3	0.5000	undef
1.00	0.0028	0.0050	+76.9	2/3	0.5000	undef
Avg:	0.1868	0.2233	+19.55	32/49	0.0222★	0.0003★
Prec.						
5	0.4939	0.5020	+1.7	10/21	0.6682	0.4106
10	0.4449	0.4898	+10.1	22/30	0.0081★	0.0154★
15	0.3932	0.4435	+12.8	19/26	0.0145★	0.0038★
20	0.3643	0.4051	+11.2	22/34	0.0607	0.0218★
30	0.3313	0.3707	+11.9	28/41	0.0138★	0.0070★
100	0.2157	0.2500	+15.9	32/42	0.0005★	0.0003★
200	0.1655	0.1903	+15.0	35/44	0.0001★	0.0000★
500	0.1004	0.1119	+11.4	36/44	0.0000★	0.0000★
1000	0.0653	0.0687	+5.1	36/43	0.0000★	0.0002★
RPr	0.2473	0.2876	+16.32	34/43	0.0001★	0.0000★

Table 1: Comparison of *tf.idf* to the language modeling approach on TREC queries 202-250 on TREC disks 2 and 3.

Notice that on the eleven point recall/precision section, the language modeling approach achieves better precision at all levels of recall, significantly at several levels. Also notice that there is a significant improvement in recall, uninterpolated average precision and R-precision, the precision after R documents where R is equal to the number of relevant documents for each query. On the second part of the figure there is again an improvement at all levels of recall, most of them statistically significant.

The results for TREC queries 51-100 on TREC disk 3 are shown in table 2. Once again we see improvement in precision at all levels of recall on the eleven point chart and we also see improvement in precision for each level of recall by document count. Several levels show a significant improvement.

4.4 Improving the Basic Model

We now try to improve our probability estimates since, according to our model, this should yield better retrieval performance. A simple improvement of the estimate developed in section 3 is to smooth the estimates of the average probability for terms with low document frequency. The estimate of the average probability of these terms is based on a small amount of data and so could be sensitive to outliers.

In order to correct for this, we binned the low frequency data by document frequency and used the binned estimate for the average. We used a cutoff of $df = 100$ for low frequency terms, though it turned out that the cutoff is not critical.

	tf.idf	LM	%chg	I/D	Sign	Wilc.
Rel:	10485	10485				
Rret.:	5818	6105	+4.93	32/42	0.0005*	0.0003*
Prec.						
0.00	0.7274	0.7805	+7.3	10/22	0.7383	0.2961
0.10	0.4861	0.5002	+2.9	26/44	0.1456	0.1017
0.20	0.3898	0.4088	+4.9	24/45	0.3830	0.1405
0.30	0.3352	0.3626	+8.2	28/47	0.1215	0.0277*
0.40	0.2826	0.3064	+8.4	25/45	0.2757	0.0286*
0.50	0.2163	0.2512	+16.2	26/40	0.0403*	0.0007*
0.60	0.1561	0.1798	+15.2	20/30	0.0494*	0.0025*
0.70	0.0913	0.1109	+21.5	14/22	0.1431	0.0288*
0.80	0.0510	0.0529	+3.7	8/13	0.2905	0.2108
0.90	0.0179	0.0152	−14.9	1/4	0.3125	undef
1.00	0.0005	0.0004	−11.9	1/2	0.7500	undef
Avg:	0.2286	0.2486	+8.74	32/50	0.0325*	0.0015*
Prec.						
5	0.5320	0.5960	+12.0	15/21	0.0392*	0.0125*
10	0.5080	0.5260	+3.5	14/30	0.7077	0.1938
15	0.4933	0.5053	+2.4	14/28	0.5747	0.3002
20	0.4670	0.4890	+4.7	16/34	0.6962	0.1260
30	0.4293	0.4593	+7.0	20/32	0.1077	0.0095*
100	0.3344	0.3562	+6.5	29/45	0.0362*	0.0076*
200	0.2670	0.2852	+6.8	29/44	0.0244*	0.0009*
500	0.1797	0.1881	+4.7	30/42	0.0040*	0.0011*
1000	0.1164	0.1221	+4.9	32/42	0.0005*	0.0003*
RPr	0.2836	0.3013	+6.24	30/43	0.0069*	0.0052*

Table 2: Comparison of *tf.idf* to the language modeling approach on TREC queries 51-100 on TREC disk 3.

This new estimate of the average is incorporated into our ranking formula as before and rerun on TREC queries 202-250 against TREC disks 2 and 3 and the results are shown in table 3.

The results show a statistically significant improvement in precision at several levels of recall. The average precision is also improved. Running the new model on our second query set, TREC queries 51-100 against TREC disk 3, we get the result shown in table 4.

Again we see significant improvements, albeit modest ones, at several levels of recall and on average. Our conjecture is that that smaller improvement on this query set is due to the longer average query length as compared to the other query set. It appears that for low frequency terms, the effects on the average due to outliers is just as likely to overestimate as it is to underestimate and so these effects cancel each other out with more terms in the query. However, this is only a conjecture, the verification of which we leave for future work.

5 Conclusions and Future Work

We have presented a novel way of looking at the problem of text retrieval based on probabilistic language modeling that is both conceptually simple and explanatory.

We feel that our model will provide effective retrieval and can be improved to the extent that the following conditions can be met:

1. Our language models are accurate representations of the data.
2. Users understand our approach to retrieval.
3. Users have a some sense of term distribution.

We feel that condition one has been met reasonably well by the approach we have taken in this study. However, we also feel that our models can and should be improved. Our current language models do not incorporate any knowledge of the language generation process. It is

	LM	LM2	%chg	I/D	Sign	Wilc.
Rel:	6501	6501				
Rret.:	3364	3350	−0.42	16/33	0.5000	0.4432
Prec.						
0.00	0.7590	0.7717	+1.7	11/17	0.1662	0.1137
0.10	0.4910	0.5115	+4.2	25/41	0.1055	0.0194*
0.20	0.4045	0.4137	+2.3	23/42	0.3220	0.2100
0.30	0.3342	0.3539	+5.9	26/42	0.0821	0.0275*
0.40	0.2572	0.2709	+5.3	23/37	0.0939	0.0420*
0.50	0.2061	0.2164	+5.0	23/33	0.0175*	0.0222*
0.60	0.1405	0.1405	−0.0	15/24	0.9242	0.8197
0.70	0.0760	0.0724	−4.8	4/14	0.0898	0.0886
0.80	0.0432	0.0450	+4.1	5/9	0.5000	undef
0.90	0.0063	0.0065	+4.6	2/3	0.5000	undef
1.00	0.0050	0.0040	−19.1	2/3	0.8750	undef
Avg:	0.2233	0.2318	+3.81	34/49	0.0047*	0.0055*
Prec.						
5	0.5020	0.5469	+8.9	13/17	0.0245*	0.0176*
10	0.4898	0.5082	+3.7	12/22	0.4159	0.1532
15	0.4435	0.4571	+3.1	14/23	0.2024	0.1007
20	0.4051	0.4235	+4.5	18/25	0.0216*	0.0083*
30	0.3707	0.3755	+1.3	16/34	0.6962	0.3222
100	0.2500	0.2655	+6.2	28/39	0.0047*	0.0005*
200	0.1903	0.1932	+1.5	18/30	0.1808	0.1226
500	0.1119	0.1128	+0.8	21/37	0.2557	0.1615
1000	0.0687	0.0684	−0.4	16/33	0.5000	0.4432
RPr	0.2876	0.2928	+1.79	19/34	0.3038	0.1485

Table 3: Comparison of the original language modeling approach to the new language modeling approach on TREC queries 202-250 on TREC disks 2 and 3.

	LM	LM2	%chg	I/D	Sign	Wilc.
Rel:	10485	10485				
Rret.:	6105	6107	+0.03	3/5	0.5000	undef
Prec.						
0.00	0.7805	0.7807	+0.0	3/5	0.5000	undef
0.10	0.5002	0.5038	+0.7	16/20	0.0059*	0.0020*
0.20	0.4088	0.4093	+0.1	16/25	0.1148	0.0959
0.30	0.3626	0.3634	+0.2	13/18	0.0481*	0.0238*
0.40	0.3064	0.3077	+0.4	16/24	0.0758	0.0198*
0.50	0.2512	0.2505	−0.3	11/25	0.3450	0.2059
0.60	0.1798	0.1777	−1.2	10/20	0.5881	0.3826
0.70	0.1109	0.1113	+0.3	9/12	0.0730	0.0356*
0.80	0.0529	0.0530	+0.1	5/8	0.3633	undef
0.90	0.0152	0.0154	+0.9	1/2	0.7500	undef
1.00	0.0004	0.0004	+0.4	1/1	0.5000	undef
Avg:	0.2486	0.2488	+0.08	28/39	0.0047*	0.0254*
Prec.						
5	0.5960	0.6000	+0.7	1/1	0.5000	undef
10	0.5260	0.5260	+0.0	0/0	1.0000	undef
15	0.5053	0.5093	+0.8	3/3	0.1250	undef
20	0.4890	0.4920	+0.6	4/5	0.1875	undef
30	0.4593	0.4613	+0.4	5/8	0.3633	undef
100	0.3562	0.3568	+0.2	5/7	0.2266	undef
200	0.2852	0.2859	+0.2	8/11	0.1133	0.0548
500	0.1881	0.1884	+0.1	6/9	0.2539	undef
1000	0.1221	0.1221	+0.0	3/5	0.5000	undef
RPr	0.3013	0.3011	−0.08	8/12	0.9270	0.7349

Table 4: Comparison of the original language modeling approach to the new language modeling approach on TREC queries 51-100 on TREC disk 3.

possible that additional knowledge added to the models will yield better estimates.

Regarding point two, we feel that our model is simple enough to be explained to users at an intuitive level and that the understanding of it will facilitate the formation of better queries. It is not that users will need or want to know the details of the model but it is more the case that if users have a general understanding of how the system works, they will be able to use it more effectively. Users are typically instructed to pose natural language descriptions of their information needs as queries. A user that understands our model will tend to think in terms of which words will help the system distinguish the documents of interest from everything else. We feel that if we can get users to think in this manner they will be able to formulate queries that will better express their information needs in manner useful to the retrieval system.

Regarding point three, in order for users to identify useful words, we feel that they would benefit from a sense of how the words are distributed in the collection. Again, it is not the case that users need or want to know the term distribution in detail, but a sense of which terms are more likely to be useful would be beneficial. We can imagine a variety of both textual and graphical tools to help users get a better sense of the distribution of terms. This especially important to win over expert users who often prefer boolean retrieval because they like the sense of control over the search [2].

Regarding the results of our study, performance on two different query sets was better than that obtained by *tf.idf* weighting. However, the improvement in performance is not the main point. More significant is that a different approach to retrieval has been shown to be effective. The ability to think about retrieval in a new way can lead to insights that would be less obvious in other approaches. Of course, the converse is is also true, and so rather than viewing our approach as a competing model, we view it as a one of a number of tools for investigating retrieval.

Our second set of experiments showed that using simple smoothing yields results significantly better than baseline on both query sets. This is an example of an insight gained from our approach that is not an obvious consequence of other approaches. It is also possible that a more elaborate smoothing technique or perhaps other techniques such as data transformation would improve results further. We plan to investigate these matters in the future.

We also need to address the estimate of default probability. As mentioned, our current estimator could in some strange cases assign a higher probability to a non-occurring query term. This could only happen in cases of very commonly occuring terms, i.e., terms which are not likely to be useful, however, we feel we should address this problem in order to insure the robustness of our model in such cases. Our approach will be to apply a smoothing based estimator that will guarantee a default probability estimate that cannot exceed the lowest estimate that we assign to a document in which a given term occurs.

Finally, the generative language model seems to be an intuitive way to think about query expansion techniques such as relevance feedback or local feedback. We intend to derive these techniques from our model rather than attempting to explain existing techniques.

6 Acknowledgments

The authors would like to thank Warren Greiff for his comments on several aspects of this work and for numerous useful comments on an early draft. Thanks to Ron Papka for his remarks on an early draft of this paper. Also, thanks to Tom Kalt for several useful discussions.

This material is based on work supported in part by the National Science Foundation, Library of Congress and Department of Commerce under cooperative agreement number EEC-9209623. This material is also based on work supported in part by United States Patent and Trademark Office and Defense Advanced Research Projects Agency/ITO under ARPA order number D468, issued by ESC/AXS contract number F19628-95-C-0235. Any opinions, findings and conclusions or recommendations expressed in this material are the authors and do not necessarily reflect those of the sponsors.

References

[1] Bookstein, A. and D. Swanson. "Probabilistic models for automatic indexing" Journal for the American Society for Information Science. v.25 no.5 pp. 312-318, 1976.

[2] Byrd, D. Personal Communication. 1998.

[3] Croft, W. B. and D. J. Harper. "Using probabilistic models of document retrieval without relevance information." Journal of Documentation, 35, 1979 (pp. 285-295).

[4] Fuhr, N. "Models for Retrieval with Probabilistic Indexing" Information Processing and Management. v. 25, no. 1 1989.

[5] Ghosh, M. J., T. Hwang and K. W. Tsui. "Construction of Improved Estimators in Multiparameter Estimation for Discrete Exponential Families." Annals of Statistics, v. 11, 351-367.

[6] Greenwood, M. and G. U. Yule. "An Inquiry into the Nature of Frequency Distribution Representative of Multiple Happenings with Particular Reference to the Occurrence of Multiple Attacks of Disease or of Repeated Accidents." Journal of the Royal Statistical Society. v. 83, pp. 255-279, 1920.

[7] Harter, S. P. "A Probabilistic Approach to Automatic Keyword Indexing" Journal of the American Society for Information Science, July-August, 1975.

[8] Kalt, T. "A New Probabilistic Model of Text Classification and Retrieval", CIIR Tech. Report No. 78, 1996.

[9] Kwok, K. L. "A New Method of Weighting Query Terms for Ad-Hoc Retrieval", In proceedings of ACM SIGIR 1996, pp 187-195.

[10] Margulis, E. L. "Modeling documents with multiple Poisson distributions." Information Processing and Management v. 29 no. 2 pp. 215-227, 1993.

[11] Parzen, E. "On estimation of a probability density function and mode." Annals of Mathematical Statistics, vol. 33, 1962.

[12] Robertson, S. E. and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In proceedings of ACM SIGIR 1994. pp. 232-241.

[13] Ponte, J. M. and W. B. Croft. "Text Segmentation by Topic," in Proceedings of the First European Conference on Research and Advanced Technology for Digital Libraries, 1997.

[14] Robertson, S. E. and K. Sparck Jones. "Relevance Weighting Of Search Terms," Journal of the American Society for Information Science, vol. 27, 1977.

[15] Salton, G. *Automatic Text Processing.* Addison Wesley, 1989.

[16] Silverman, B. W. *Density Estimation for Statistics and Data Analysis* Chapman and Hall, 1986.

[17] Terrell, G. R. and D. W. Scott. "Oversmoothed Nonparametric Density Estimators" Journal of the American Statistical Association. Vol. 3, Number 389, 1985.

[18] Titterington, D. M., U. E. Makov and A. F. M Smith. *Statistical Analysis of Finite Mixture Distributions* John Wiley and Sons, 1985.

[19] Turtle H. and W. B. Croft. "Efficient Probabilistic Inference for Text Retrieval," Proceedings of RIAO 3, 1991.

[20] Wong, S. K. M. and Y. Y. Yao. "A Probability Distribution Model for Information Retrieval" Information Processing and Management, v. 25 no. 1 pp. 39-53, 1989.

[21] Yamron, J. "Topic Detection and Tracking Segmentation Task" In proceedings of The Topic Detection and Tracking Workshop, Oct. 1997.

[22] Zelen, M. and N. Severo. "Probability Functions" Handbook of Mathematical Functions. M. Abramowitz and I. A. Stegun ed. National Bureau of Standards Applied Mathematics Series No. 55. 1964.