# Revisiting Again Document Length Hypotheses TREC-2004 Genomics Track Experiments at Patolis

Sumio FUJITA
PATOLIS Corporation
2-4-29, Shiohama, Koto-ku,
Tokyo 135-0043, Japan

## Abstract

The TREC-2004 Genomics track evaluation experiments at Patolis Corporation are described with a focus on the document length issues in different retrieval models such as TF*IDF or probabilistic language modeling approaches.

In the genomics ad hoc retrieval task, combination of pseudo-relevance feedback and reference database feedback is applied.

For the triage sub-task, we trained a SVM classifier using leave-one-out-cross-validation, and calibrated parameters to be optimal against the training set.

## Keywords

Document length, language modeling for information retrieval, pseudo-relevance feedback, reference database feedback, MeSH, LocusLink, MEDLINE, support vector machines.

## 1. Introduction

The TREC-2004 Genomics track evaluation experiments at the Patolis Corporation group are described. The track consists of the ad hoc retrieval task and the categorization task.

The ad hoc retrieval task is designed to simulate the subject topic retrieval against a 10 year subset (4,591,008 records) of the MEDLINE bibliographic database. 50 official (and other 5 sample) search topics are derived from interviews on real biology researchers. Relevance assessments were carried out using the conventional pooling method and each pooled documents are judged as definitely relevant (DR), possibly relevant (PR) or not relevant (NR) against the information needs. Documents rated DR or PR are considered as relevant in official evaluations.

Participants are asked to submit up to two sets of top 1000 relevance ranked list of documents retrieved by either automatically or manually constructed queries from given search topics. There are no specific restrictions using data resources.

The other task is the categorization task, which actually consists of three subtasks namely triage, annotation hierarchy and annotation hierarchy plus evidence. We participated in the triage subtask where participants are asked to identify papers deemed to have experimental evidences warranting annotation with GO codes from the collection of articles of three journals over two years. The document collection is a subset of these articles filtered through by the "mouse trap" method.

## 2. System Description

Our evaluation environment: the PLLS system developed based on the Lemur toolkit 2.0.1 for indexing system [15]; the PostgreSQL RDB system is integrated for treating bibliographic information. The system is operated on a dual CPU PC server(Xeon 3.20GHz, 4GB RAM) running RedHat Linux.

The document collections are indexed wholly automatically, and converted to inverted index files of terms.

### 2.1 Indexing Language

The words are indexed either stemmed by a porter stemmer or indexed by their appearing forms. Stopword elimination by InQuery stop list is applied.

### 2.2 Retrieval Models

The following two retrieval models are implemented:

-TF*IDF with Okapi BM25 TF [17][18] (BM25 TF*IDF hereafter)

BM25 TF is incorporated in the dot-product matching function between TF*IDF weighted vectors. Typical parameters like k1, b can be adjusted.

Instead of the Okapi IDF: log(N-df(t)+0.5/df(t)+0.5) that gets a negative value when df(t) is very large, we adopted a standard IDF adjusted by the k4 parameter. This is slightly different from the implementation in the Lemur toolkit. The same weighting is applied for the query part but with a different value for k1 and without length normalization i.e. b=0.

Such a dot-product matching between BM25 TF*IDF weighted vectors is applied successfully to TREC web ad hoc search task characterized by very short queries and various lengths of documents where subdocument based retrieval is applied [5][6].

$$w(d,t) = (k4 + \log\frac{N}{df(t)})\frac{(k1+1)freq(d,t)}{k1((1-b)+b\frac{dl_d}{avdl})+freq(d,t)}$$

$d$ : document

$t$ : term

$N$ : total number of documents in the collection

$df(t)$ : number of documents where t appears

$freq(d,t)$ : number of occurrence of t in d

-KL-divergence of probabilistic language models with Dirichlet prior smoothing (KL-Dir hereafter) [23]
For the KL-divergence model, the detail is described in Section 3.

## 2.3 Reference Database Feedback Strategies

Besides traditional pseudo relevance feedback, "reference database" feedback methods from the MeSH entry database and the LocusLink summary description database, are applied for expanding query terms. This technique is applied in the TREC-9 Web track by Fujita [5] where the queries are very short and even noisy. In the Web track, it was effective with very short queries but not with longer queries.

In the genomics track, expansions of gene symbol variation and technical term variation are intended using this technique. There is naturally another option to expand a query with such synonymous word groups: extracting exact alias symbol groups from the "ALIAS SYMBOL" field of LocusLink records (from the MH and SY fields of MeSH records as well), gene symbol thesauri are generated. Given an occurrence of a word in the generated thesauri, the query is expanded by the group of synonymous words in the thesauri.

We applied more "relaxed" expansions, where indexing each LocusLink or MeSH record as one document and retrieving the best matched documents against the original query and extracting terms from some of the best match documents. Not only synonymous words but also words from summary sentences are added to the query.

The system submits the original query generated automatically from topic descriptions against the reference databases, and takes the top n(=1) document(s) from the ranked list for term extraction. The term selection module extracts salient terms from these pseudo-relevant documents and adds them to the query vector.

## 2.4 Pseudo-Relevance Feedback Strategies

Pseudo-relevance feedback is applied in both official runs and other unofficial runs.

Rocchio feedback [19] for BM25 TF*IDF and the mixture model query update method for KL-divergence retrieval model [24] (unofficial runs), are adopted. The parameters such as the number of documents for the pseudo relevant set, the number of terms to feedback, some score cutoff threshold values and mixture coefficients of feedback terms against original terms are decided by pre-submission experiments using five sample topic sets and the corresponding relevance judgment file provided by the organizers.

## 2.5 Query Expansion in Summary

Reference database feedback procedures and the pseudo-relevance feedback are sequentially applied.

The system submits the original query generated automatically from each topic description against two reference databases consequently, and makes two groups of documents. In this case, we used only the top one document from each reference database for term extraction. The terms extracted from these documents are added to the query vector. Then the expanded query vector is submitted against the target database and the pseudo-relevance feedback is applied preceding the final search.

# 3. Language Modeling for IR

Uses of probabilistic language models in information retrieval intended to adopt a theoretically motivated retrieval model given that recent probabilistic approaches tend to use too many heuristics.

Ponte and Croft first applied a document unigram model to compute the probability of the given query to be generated from a document [16].

In TREC-7, Hiemstra and Kraaij [8] introduced linear interpolation of local and global probabilities while Miller et al. [14] used hidden Markov model to mixture two distributions. Berger and Lafferty [1] proposed a statistical translation as a model of user's distillation process from an information need into a succinct query.

## 3.1 Basic Model

The adopted model is simple: estimate a language model for each document and rank documents by the likelihood of generating the submitted query. This is exactly a retrieval version of a Naïve Bayes classifier, which estimates a language model for each class and ranks classes by the likelihood of generating the document to be classified. Applying Bayes' theorem for p(d|q), and eliminating document independent part, we have:

$$p(d \mid q) \propto p(d)p(q \mid d)$$

Assuming a simple uni-gram model of documents, p(q|d) is:

$$p(q \mid d) = \prod_i p(q_i \mid d)$$

Taking the logarithm, the retrieval function becomes:

$$\log(p(d)p(q \mid d)) = \log p(d) + \sum_i \log p(q_i \mid d)$$

A document dependent prior probability p(d) can be either a uniform probability or any document dependent factors that may affect the relevance such as document length or hyper link related information. Assuming a uniform prior probability and dropping the first term, transforming the summation over query term positions into a summation over words in the vocabulary, dividing by the query length, we have:

$$\sum_{w \in V} p(w \mid q) \log(p(w \mid d))$$

This is exactly the negative cross entropy of a query language model with a document language model, which measures the difference between the two probability distributions and this is equivalent to KL-divergence of a query language model from a document language model in view of ranking documents against the given query.

## 3.2 Smoothing Methods

Zhai and Lafferty presented that a smoothing method plays a crucial role in language modeling IR [23].

They analyzed the role of smoothing in language modeling IR from two aspects: to avoid zero probabilities for unseen words and "to accommodate generation of common words in a query". In this respect, smoothing plays a role similar to IDF in TF*IDF weighting. They proposed three types of smoothing strategies including the Jelinek-Mercer method i.e. simple linear combination of an estimated document model and a background model p(w|C), the Baysean smoothing using Dirichlet Priors method that computes maximum a posteriori parameter values with a Dirichlet prior ( i.e. a kind of the Laplace smoothing ), and the absolute discount method.

The Jelinek-Mercer method is:

$$p_\lambda(w \mid d) = (1-\lambda)p_{ml}(w \mid d) + \lambda p(w \mid C)$$

The Dirichlet-Prior method is:

$$p_\mu(w \mid d) = \frac{freq(w,d) + \mu p(w \mid C)}{|d| + \mu}$$

The smoothing factor in the first case is λ while μ/|d|+ μ in the second case. Document length is taken into consideration in the Dirichlet-Prior smoothing: as p(w|C) is divided by the document length, scores of longer documents are more penalized than the Jelinek-Mercer smoothing.

We utilized the implementation in the Lemur toolkit.

## 3.3 Document Dependent Priors

On the other hands, any document dependent and typically query independent factors that may affect the relevance can be taken into consideration by the scoring process as document prior probabilities.

Some studies suggest that document length is a good choice in TREC experiments since it is predictive of relevance against the TREC test set [14][20].

The following document length dependent probability is applied where μ is smoothing factor.

| Run description | Index | RefTerms | Mean Avg. Prec. | R-Prec. |
|---|---|---|---|---|
| TF*IDF (pllsgen4a1) | - | Strong | 0.3689 | 0.3932 |
| TF*IDF | Porter | Strong | 0.3902 | 0.429 |
| TF*IDF | - | Weak | 0.3793 | 0.4018 |
| TF*IDF (pllsgen4a2) | Porter | Weak | 0.4075 | 0.4366 |

**Table 1: Performance of official runs and their baseline runs**

| | pllsgen4a1 | pllsgen4a2 |
|---|---|---|
| K1 | 0.1 | 0.4 |
| B | 0.8 | 0.8 |
| K4 | 0.1 | 0.1 |
| # FB docs | 7 | 7 |
| # FB terms | 30 | 30 |
| <TITLE> Coeff. | 1.0 | 1.0 |
| <NEED> Coeff. | 1.0 | 0.9 |
| <CONTEXT> Coeff. | 0.5 | 0.5 |
| MeSH Coeff. | 0.04 | 0.02 |
| LocusLink Coeff. | 0.04 | 0.02 |
| Feedback Coeff. | 0.1 | 0.1 |

**Table 2: Parameters of official runs**

$$p(d) = \frac{|d| + \mu 2}{\sum_{d' \in D} |d'| + \mu 2}$$

## 4. Ad Hoc Retrieval Task

### 4.1 Official Runs

We submitted two automatic runs as follows:

pllsgen4a1: BM25TF*IDF, long query, pseudo relevance feedback, reference database feedback, stopwords elimination, without stemming.
pllsgen4a2: BM25TF*IDF, long query, pseudo relevance feedback, reference database feedback, stopwords elimination, with a porter stemmer.

Table 1 shows the performance of official runs and some comparative runs.
By using Porter stemmer in indexing, statistically significant (t-test, $p<0.05$) improvements of the MAP values are observed.
In TREC-9, we explained our approach utilizing the "foreground vs background" metaphor, where foreground terms denote directly the subject concept of the information need and background terms connote the subject topic.
When utilizing such expanded longer queries, differentiating weights of query terms according to the "foregroundness" i.e. source of the terms, makes considerable difference in effectiveness.
Table 2 shows parameters of the official runs and "XXX coeff." indicates the weights for the terms from each source, i.e. <TITLE>, <NEED> and <CONTEXT> fields of topic descriptions, MeSH and LocusLink reference databases, and pseudo-relevance feedback.

### 4.2 Post-Submission Experiments

We did not afford to submit KL-dir runs because of our experiences in NTCIR-4 [7]. We had impressions that it tends to retrieve shorter documents than it should do. This causes slightly poorer performance in test collection based evaluation where usually relevance assessments tend to prefer longer documents.
Table 3 shows the performance comparison combining pseudo-relevance feedback and reference database feedback as well as different retrieval models TF*IDF/KL-Dir on the basis of the pllsgen4a2 setting.

The pseudo relevance feedback procedure contributes to 4.39% to 2.00 % of consistent improvements in average precision in all cases.
The reference database feedback procedure improves MAP consistently but as slightly as 0.97% to 0.44%.
The improvement gained by the combination of pseudo-relevance feedback and reference database feedback is 4.57% for TF*IDF runs and 2.53% for KL-Dir runs.
The rates of improvements are modest in comparison with our past experiences in the TREC-9 Web track utilizing very short queries (+17%) [5] and in the TREC 2001 Web track (+21.4%) [6].
One of the reasons why the gains from feedbacks are small is that full-length queries are utilized where all three topic fields are combined and comparatively rich term sets are generated. Such observation is consistent with our past experiences utilizing various length queries in TREC-9 [5] and in NTCIR-1 [4].
The difference is not statistically significant but unofficial KL-Dir runs consistently better than their TF*IDF counterparts (2.40% to 0.05%) even though no parameter tuning was done.
By some parameter tuning in post-submission experiments, the best MAP as high as 0.4264 is

| Run description | Ref | PFB | AvgPrec | P@10 | ALRD |
|---|---|---|---|---|---|
| TF*IDF+porter (pllsgen4a2) | Yes | Yes | 0.4075 | 0.6040 | 265.7 |
| TF*IDF+porter | Yes | No | 0.3915 | 0.5900 | 263.5 |
| TF*IDF+porter | No | Yes | 0.4068 | 0.5960 | 265.2 |
| TF*IDF+porter | No | No | 0.3897 | 0.5780 | 263.2 |
| TF*IDF+porter(Best) | Yes | Yes | 0.4127 | 0.6200 | 269.1 |
| KL-Dir+porter | Yes | Yes | 0.4088 | 0.6180 | 269.2 |
| KL-Dir+porter | Yes | No | 0.4009 | 0.6140 | 268.9 |
| KL-Dir+porter | No | Yes | 0.407 | 0.6160 | 269.5 |
| KL-Dir+porter | No | No | 0.3987 | 0.6100 | 269.1 |
| KL-Dir+porter(Best) | Yes | Yes | 0.4264 | 0.6460 | 269.7 |

**Table 3: Performance comparison in post-submission experiments with long queries**

| Run description | Ref | PFB | AvgPrec | P@10 | ALRD |
|---|---|---|---|---|---|
| TF*IDF+porter (pllsgen4a2) | Yes | Yes | 0.3476 | 0.5160 | 261.7 |
| TF*IDF+porter | Yes | No | 0.3165 | 0.5240 | 250.4 |
| TF*IDF+porter | No | Yes | 0.3502 | 0.5140 | 260.1 |
| TF*IDF+porter | No | No | 0.3090 | 0.5100 | 244.0 |
| KL-Dir+porter | Yes | Yes | 0.3239 | 0.5040 | 256.5 |
| KL-Dir+porter | Yes | No | 0.3196 | 0.5080 | 260.4 |
| KL-Dir+porter | No | Yes | 0.3213 | 0.5020 | 256.3 |
| KL-Dir+porter | No | No | 0.3174 | 0.5060 | 280.8 |

**Table 4: Performance comparison in post-submission experiments with Title only queries**

achieved.

Using document length priors always harms the performance. Giving a large value to μ2(e.g. 100000 i.e. making p(d) flat against document length), the performance is approaching to the baseline of uniform priors but still remains below it.

The parameter μ2works just like the slope parameter in the pivoted normalization. But in this case, no document length normalization other than the one incorporated in the Dirichlet smoothing was needed.

Table 4 shows the experiments with the title only queries where feedback gains are larger than the long query runs. In fact the pseudo feedback contributes to maximum 13.3% in a BM25TF*IDF run. On the other hands, KL-Dir runs are not so much improved by the pseudo feedback, because the mixture model feedback is sensitive to the interpolation parameter by which the original query model and feedback model are mixtured. After readjusting the interpolation parameter, the best KL-Dir run achieved 0.3567 of MAP with pseudo feedback, which is better than the best BM25TF*IDF run.

## 5. Document Length Issues

We comparatively studied the behavior of two different retrieval models, namely TF*IDF with BM25 TF and KL-divergence with Dirichlet smoothing in NTCIR-3 and NTCIR-4 Japanese newspaper and patent test collections [7].

Both retrieval models reasonably worked well against the Patent test collections, which is in some sense technico-scientific documents while BM25 TF*IDF outperformed KL-Dir against the newspaper test collections. After some analyses, we found out that this discrepancy is caused by the different behavior of two retrieval models against different lengths of documents. In brief, KL-Dir tended to retrieve shorter documents than BM25 TF*IDF.

The question is why it worked for some test collections but not for other test collections.

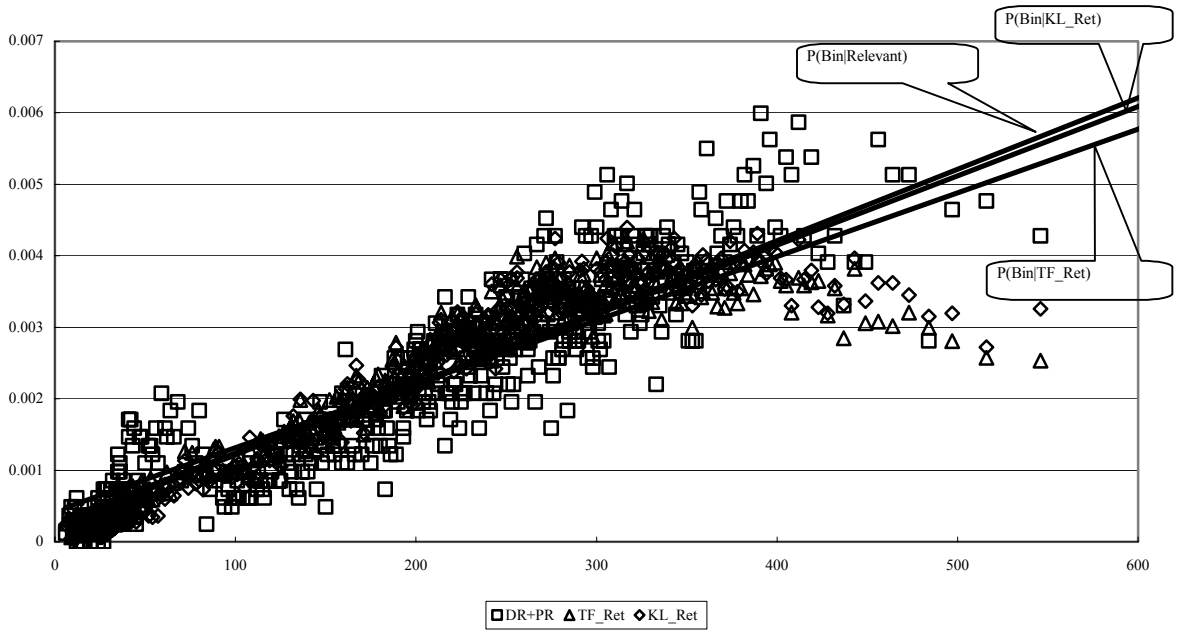


**Figure 1: p(Bin|Relevant) and p(Bin|Retrieved) by BM25TF*IDF and KL-Dir, plotted against the median bin length in the MEDLINE Collection**

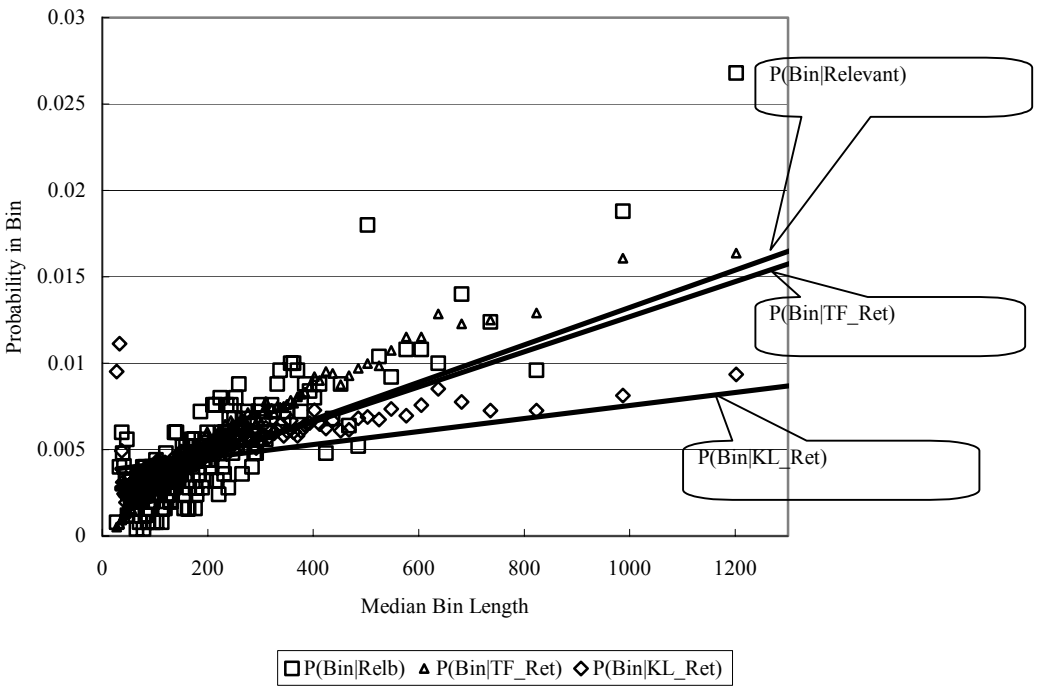


**Figure 2: p(Bin|Relevant) and p(Bin|Retrieved) by BM25TF*IDF and KL-Dir, plotted against the median bin length in the NTCIR-3 CLIR-J-J Collection**

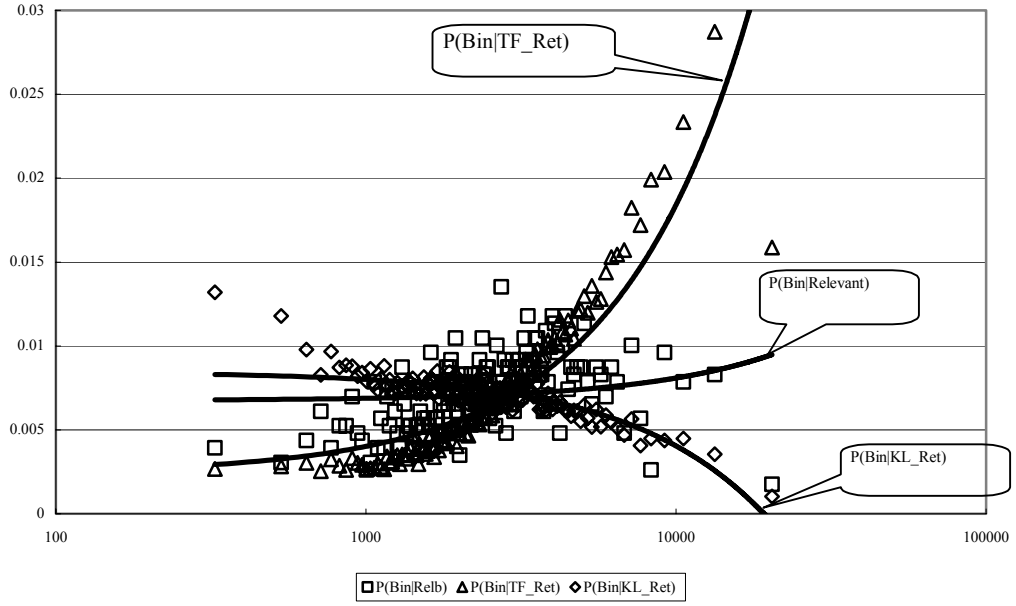


**Figure 3: p(Bin|Relevant) and p(Bin|Retrieved) by BM25TF*IDF and KL-Dir, plotted against the median bin length in the NTCIR-3 Patent Collection**

## 5.1 Document Length Hypotheses

The question to be asked here is why longer documents are longer than shorter ones? Though this question may sound as a tautology, it is not. The problem is to know how each document differs in length.

If longer documents have more information, they may be more likely to be relevant against diverse queries, so that it is fair to get a higher matching score.

Robertson and Walker [17] postulated two hypotheses to explain different length of documents namely the "Scope hypothesis" and the "Verbosity hypothesis".

The "Scope hypothesis" considers a long document as a concatenation of a number of unrelated short documents while the "Verbosity hypothesis" assumes that a long document covers the same scope as a short document but it uses more words. These two hypotheses represent the extreme cases and real documents are always the mixture of the two cases.

The natural consequence of adopting the "Scope hypothesis" is that a long document is more likely to be relevant irrespective of search requests since it covers more subject topics than a shorter one. Robertson and Walker assume that the "Verbosity hypothesis" implies that document properties such as relevance and eliteness are independent of document length.

Longer documents are more informative than shorter ones even the subject coverage is the same and also there is the minimum amount of information for a document in order to be relevant against any information needs. Such information amount issues make longer documents more likely to be relevant even under the "Verbosity hypothesis".

## 5.2 Likelihood of Relevance/Retrieved in Diverse Test Collections

To validate the document length hypotheses, different types of document collections are examined by re-applying the analyses against the TREC test collections described by Singhal et al. [20][21].

The MEDLINE document collection(PubMED Abstract 1994-2003: 4,591,008 documents) are put into bins of 5,000 documents in the order of the length of documents counted by the number of indexed terms.

We utilized 8268 "topic-relevant document" pairs for 50 topics of the test collection. Possibly relevant (PR) documents are included in these pairs in order to augment the data. From these pairs, p(d in $Bin_i$| d is relevant) for each i-th bin is computed.

From 50,000 "topic-retrieved document" pairs from retrieval result lists against the test collection, p(d in $Bin_i$| d is retrieved) is computed.

Figure 1 shows p(Bin|Relevant) and p(Bin|Retrieved) by BM25TF*IDF and KL-Dir, plotted against the median document length in each bin, in the MEDLINE Collection.

In Figure 1, approximation curves of plotted dots by a linear function indicate that the ratio of "KL-Dir retrieved"-"document length" (P(Bin|d is Retrieved by KL-Dir)) is almost overlapped on the ratio of "relevance"-"document length" (P(Bin|Relevant)) while the graph of "BM25TF*IDF retrieved"-"document length" (P(Bin|d is Retrieved by BM25TF*IDF)) is slightly below the graph of P(Bin|Relevant).

We have never observed such a situation where KL-dir tends to retrieve longer documents than BM25TF*IDF.

## 5.3 Typical Examples of "Scope Hypothesis" and "Verbosity Hypothesis"

Figure 2 shows the same analyses against the NTCIR-3 CLIR-J-J (Mainichi newspapers) collection and Figure 3, Patent test collection [12][2][9].

Newspaper documents are typically a case of the "scope hypothesis", like TREC collections, where the longer documents necessarily mention more subject topics (see the graph from NTCIR-3 CLIR J-J in Figure 2).

Patent documents may be seen as a case of the "verbosity hypothesis", where longer documents use more words to describe a specific subject topic. As required by the "Unity of Invention" principle, a patent document is about a single subject so that the document length may not affect relevance or elitness

| | NTCIR-3 CLIR-J-J | NTCIR-4 CLIR-J-J | NTCIR-3 Patent | NTCIR-4 Patent | TREC 2004 MEDLINE |
|---|---|---|---|---|---|
| A docs/ DR | 315(167%) | 308(159%) | 3164(109%) | 3137(127%) | 291(150%) |
| AB docs/ DR+PR | 290(153%) | 289(150%) | 3075(106%) | 2946(119%) | 271(140%) |
| ABCD/ judged | 232(123%) | 277(143%) | 3123(107%) | 3321(134%) | 257(132%) |
| All docs | 189(100%) | 193(100%) | 2906(100%) | 2478(100%) | 194(100%) |

**Table 5: Average document length of relevant(A)/definitely relevant(DR), partially relevant(AB)/possibly relevant(PR), pooled documents(ABCD/judged) and the whole collection(All docs) counted by the number of indexed terms**

(see the example from NTCIR-3 Patent in Figure 3).
The curve of "BM25TF*IDF retrieved"-"document length" (P(Bin|d is Retrieved by BM25TF*IDF)) increases linearly while the curve of KL-Dir is almost flat.
In summary, BM25TF*IDF always tends to retrieve longer documents and this may be optimal against newspaper documents while KL-Dir tends to retrieve much shorter documents. KL-Dir seems to be over-penalizing the matching scores of long documents since the approximation curves of P(Bin|d is Retrieved by KL-Dir) is almost flat or even decreasing against document length in Figure 2.
In the case of the MEDLINE collection, it seems difficult to say which hypothesis is adequate to assume. Scientific articles tend to concentrate on one specific subject topic irrespective of their length so that they fall into the "Verbosity hypothesis" in view of relevance against a certain subject topic. Some MEDLINE records are extremely short and no abstract is provided, although some of them are assessed as relevant to some topics. Such records are also found in the Mainichi newspaper collection but they are excluded from the NTCIR-3 CLIR-J-J evaluation.
Despite such biases, the MEDLINE collection seems to close to the Japanese newspaper collections (see Table 5) rather than the Patent collections.

## 6. Triage Task

We participated in the triage subtask of the categorization task.
Each document in training/test sets is represented as terms weighted by log TF, and we trained a classifier of soft margin linear support vector machines (SVM hereafter) by using SVM_light [10]. Each document representation is expanded by MeSH terms from the Medline records and Gene expressions identified by a gene name tagger developed and made available by Tanabe et al. [22].
After examining diverse kind of term weighting such as Log(TF)*IDF, BM25TF*IDF, Log(TF), Boolean, P(t|d), we adopted the normalized Log(TF) weighting.
It was not at all clear that the combinations of what feature sets and what weighting methods work well with SVM learning, we were completely groping for optimum utility values by leave-one-out-cross-validation (LOOCV hereafter, SVM_light options: -x 1 –o 1) against the training set.
On top of that, there are some SVM_light parameters to be determined empirically [11].
The parameter j: cost factor, by which training errors on positive examples outweight errors on negative examples, is fixed at 20 since the official utility measure multiplies 20 on the number of true positive examples.
The parameter c: trade off between training error and margin, is adjusted empirically by LOOCV on training examples. Because of the fear to over-fitting, this parameter, which works as a threshold is a little bit decreased from the optimum value against training examples.
This approach is contrary to that of Lewis [13], who changed and optimized the j parameter and gave default values to the c parameter with normalized vectors, in TREC 2001 filtering.
In LOOCV against the training set, the best utility is achieved by C=0.0001505 and J=20 and the result set from this setting is submitted as pllsgen4t1. Other three result sets where the C value is slightly decreased (the threshold is relaxed) and one set where C is increased (threshold is tightened) are submitted. (see Table 6)

| Run Tag<br>C value | Utility by LOOCV | Official utility | F-score | #TP | #FP | #FN |
|---|---|---|---|---|---|---|
| pllsgen4t1<br>0.0001505 | 0.5999 | 0.5302 | 0.2730 | 295 | 1446 | 125 |
| pllsgen4t2<br>0.00013 | 0.5945 | 0.5363 | 0.2645 | 304 | 1575 | 116 |
| pllsgen4t3<br>0.0001 | 0.5941 | 0.5494 | 0.2496 | 323 | 1845 | 97 |
| pllsgen4t4<br>0.00007 | 0.5640 | 0.5424 | 0.2186 | 349 | 2424 | 71 |
| pllsgen4t5<br>0.00016 | 0.5900 | 0.5320 | 0.2785 | 293 | 1391 | 127 |

**Table 6: Performance of triage official runs**

| Feature set | Weighting | J | C | Best Utility by LOOCV | Utility against Test set |
|---|---|---|---|---|---|
| Full text terms, Gene Entities, MeSH terms (=pllsgen4t1) | Log(TF)/Log(AvgTF) | 20 | 0.0001505 | 0.5999 | 0.5305 |
| Full text terms, MeSH terms | Log(TF)/Log(AvgTF) | 20 | 0.0001552 | 0.5996 | 0.5305 |
| Full text terms, Gene Entities, MeSH terms | Log(TF) | 20 | 0.0000175 | 0.5992 | 0.5415 |
| Full text terms | Log(TF)/Log(AvgTF) | 20 | 0.00012 | 0.5805 | 0.5250 |
| Gene Entities, MeSH terms | Log(TF)/Log(AvgTF) | 20 | 0.000041 | 0.5736 | 0.5067 |
| Full text terms, Gene Entities, MeSH terms Polynomial Kernel (d=3) | Log(TF)/Log(AvgTF) | 20 | 0.00000000026 | 0.5556 | 0.5037 |
| Full text terms, Gene Entities, MeSH terms | Log(TF)*IDF/Norm | 20 | 0.0453 | 0.5535 | 0.4862 |
| Full text terms, Gene Entities, MeSH terms | Log(TF)*IDF | 20 | 0.00000107 | 0.5512 | 0.4856 |
| Full text terms, Gene Entities, MeSH terms | TF | 20 | 0.0000005 | 0.5496 | 0.5130 |
| Full text terms, Gene Entities, MeSH terms | P(t\|d) | 20 | 6 | 0.5417 | 0.5205 |
| Gene Entities, MeSH terms | Bool | 20 | 0.000083 | 0.5336 | 0.4551 |
| Full text terms, Gene Entities, MeSH terms | BM25TF*IDF | 20 | 0.000003 | 0.5305 | 0.4685 |
| Full text terms, Gene Entities, MeSH terms | Bool | 20 | 0.00008 | 0.5305 | 0.4711 |

**Table 7: Performance by the decreasing order of the utility value in LOOCV against the training set**

According to the LOOCV against the training set shown in Table 7, the following observations are drawn in view of weighting methods.

-In summary, IDF weighting does not help while any kinds of TF weighting helps.

-Log(TF) is better than raw TF while average normalized Log(TF) is almost same as the simple Log(TF).

For the feature sets, combining the full text terms, gene entities and MeSH terms is effective but even the combinations of two of them work reasonably well.

Anyway, the C parameter tuning is a very time and labor intensive work so that we need some automatic hill-climbing parameter calibration given enough computing power.

We shall examine normalized vectors to see if it helps for an easier parameter tuning.

As our official runs show, the parameters achieving the best utility in LOOCV against the training set are usually over-fitted, the threshold should be relaxed. It is not clear how much it should be relaxed. As each document should be processed separately as the task definition, a delivery ratio basis threshold calibration [3] is not applicable here.

For the classifier of pllsgen4t1, which achieved the best utility measure in LOOCV against the training set, the number of support vectors is 4959 against 5837 training examples and the number of misclassified examples amount for 1351. These suggest that the training set with adopted feature sets is not a good example to apply SVMs.

## 7. Conclusions

TREC-2004 genomics track evaluation experiments at the Patolis corporation group are described.

The following observations are drawn from these experiments:

For the ad hoc retrieval task, we submitted BM25TF*IDF runs and examined some language modeling runs using KL-divergence with Dirichlet smoothing. KL-Dir runs tend to perform better than BM25TF*IDF runs, which was a rare case in our past experiences. We analyzed the test collection characteristic examining likelihood of relevance/retrieved against different document lengths and find out that the KL-Dir retrieved likelihood overlapped better on the relevance likelihood than that

of BM25TF*IDF, which was also the rare case according to our experiences.

In future, we will examine more the behavior of two retrieval models against diverse test collections and hopefully induce a better length normalization for language modeling retrieval methods.

In the triage subtask, we trained a SVM classifier using LOOCV against the training set. Despite the only one binary classifier to be trained, efforts for parameter calibration are considerable so that we need to consider more automated ways to calibrate parameters.

## REFERENCES


[1] Berger,A. and Lafferty,J. 1999. Information retrieval as statistical translation. In *Proceedings of the 1999 ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, CA, 222-229.

[2] Chen, K.H., Chen, H.H., Kishida, K., Kuriyama, K., Kanodo, N., Lee, S., Myaeng, S.H., Eguchi, K. and Kim, H. 2002. Overview of CLIR Task at the Third NTCIR Workshop, In *Working notes of the third NTCIR workshop meeting Part I Overview*, 23-60.

[3] Evans, D. A., Shanahan, J., Tong, X., Roma, N., Stoica, E., Sheftel, V., Montgomery, J., Bennett, J., Fujita, S. and Grefenstette, G., 2002. Topic-Specific Optimization and Structuring—A report on CLARIT TREC-2001 Experiments. In *NIST Special Publication 500-250:The Tenth Text REtrieval Conference (TREC 2001)*, 132-141.

[4] Fujita, S. 1999. Notes on Phrasal Indexing—JSCB Evaluation Experiments at NTCIR AD HOC. In *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, 101-108.

[5] Fujita, S. 2000. Reflections on "Aboutness"—TREC-9 Evaluaton Experiments at Justsystem. In *NIST Special Publication 500-249: the Ninth Text REtrieval Conference(TREC-9)*, 281-289.

[6] Fujita, S. 2001. More reflections on "Aboutness"—TREC-2001 Evaluaton Experiments at Justsystem , In *NIST Special Publication 500-250:The Tenth Text REtrieval Conference (TREC 2001)*, 331-338.

[7] Fujita, S. 2004. Revisiting the Document Length Hypotheses --NTCIR-4 CLIR and Patent Experiments at Patolis. In *Working notes of the fourth NTCIR workshop meeting*, 238-245.

[8] Hiemstra, D. and Kraaij, W. 1998. Twenty-one at TREC-7: Ad-hoc and cross-language track. In *NIST Special Publication 500-242:The Seventh Text REtrieval Conference (TREC 7),* 227-238.

[9] Iwayama, M. , Fujii, A. , Kando, N. and Takano, A. 2002. Overview of Patent Retrieval Task at NTCIR-3, In *Working notes of the third NTCIR workshop meeting Part I Overview*, 67-76.

[10] Joachims, T. Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

[11] Joachims, T. Learning to Classify Text Using Support Vector Machines. Dissertation, Kluwer, 2002.

[12] Kando, N. 2002. Overview of the Third NTCIR Workshop, In *Working notes of the third NTCIR workshop meeting Part I Overview*, 1-16.

[13] Lewis, D. D. 2001. Applying Support Vector Machines to the TREC-2001 Batch Filtering and Routing Tasks. In *NIST Special Publication 500-250:The Tenth Text REtrieval Conference (TREC 2001),* 286-292.

[14] Miller, D., H., Leek, T., and Schwartz, R. 1999. A hidden Markov model information retrieval system, In *Proceedings of the 1999 ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, CA, 214–221.

[15] Ogilvie,O. and Callan,J. 2002. Experiments Using the Lemur Toolkit, In *NIST Special Publication 500-250: The Tenth Text REtrieval Conference (TREC 2001)*,103-108.

[16] Ponte, J. and Croft, W. B. 1998. A language modeling approach to information retrieval, In *Proceedings of the 1998 ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, 275–281.

[17] Robertson, S.E. and Walker S. 1994. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In *Proceedings of the 1994 ACM SIGIR Conference on Research and Development in Information Retrieval*, Dublin, Ireland, 232-241.

[18] Robertson, S.E., Walker S., Jones S., Hancock-Beaulieu, M.M. and Gatford, M. 1995. Okapi at TREC-3. In *Proceedings of the Third Text REtrieval Conference(TREC-3)*, NIST Special Publication 500-225, Washington D.C., 109-126.

[19] Rocchio, J.J. 1971. Relevance feedback in information retrieval, In The SMART Retrieval System: Experiments in Automatic Document

Processing, G. Salton ed. Prentice-Hall, Englewood Cliffs, NJ, 313-323.

[20] Singhal, A., Buckley, C., and Mitra, M. 1996. Pivoted document length normalization. In *Proceedings of the 1996 ACM SIGIR Conference on Research and Development in Information Retrieval*, Zurich, Switzerland, 21–29.

[21] Singhal, A., Salton, G., Mitra, M. and Buckley, C. 1996. Document Length Normalization. In *Information Processing & Management*, Vol 32, No. 5, pp.619-633.

[22] Tanabe, L. and Wilbur, W. J. 2002. Tagging Gene and Protein Names in Biomedical Text. In *Bioinformatics*, vol. 18, no. 8, 1124-1132.

[23] Zhai, C. and Lafferty, J. 2001. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 2001 ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, LA, 334-342.

[24] Zhai, C. and Lafferty, J. 2001. Model-based feedback in the KL-divergence retrieval model. In *Proceedings of the Tenth International Conference on Information and Knowledge Management(CIKM 2001)*, Atlanta, GA, 403-410.