IR-Specific Searches at TREC 2007: Genomics & Blog Experiments

Claire Fautsch, Jacques Savoy

Computer Science Department, University of Neuchatel
Rue Emile-Argand, 11, CH-2009 Neuchatel (Switzerland)
{Claire.Fautsch, Jacques.Savoy}@unine.ch

ABSTRACT

This paper describes our participation in the TREC 2007 Genomics and Blog evaluation campaigns. Within these two tracks, our main intent is to go beyond simple document retrieval, using different search and filtering strategies to obtain more specific answers to user information needs. In the Genomics track, the dedicated IR system has to extract relevant text passages in support of precise user questions. This task may also be viewed as the first stage of a Question/Answering system. In the Blog track we explore various strategies for retrieving opinions from the blogsphere, which in this case involves subjective opinions about various targets entities (e.g., person, location, organization, event, product or technology). This task can be subdivided in two parts: 1) retrieve relevant information (facts) and 2) extract positive, negative or mixed opinions about the specific entity being targeted.

To achieve these objectives we evaluate retrieval effectiveness using the Okapi (BM25) and various other models derived from the *Divergence from Randomness* (DFR) paradigm, as well as a language model (LM). Through our experiments with the Genomics corpus we find that the DFR models perform clearly better than the Okapi model (relative difference of 70%) in terms of mean average precision (MAP). Using the blog corpus, we found the opposite; the Okapi model performs slightly better than both DFR models (relative difference around 5%) and LM (relative difference 7%) model.

1. INTRODUCTION

The biomedical domain presents the information retrieval (IR) community with a number of challenging problems. For the first Genomics campaign [1] for example the main objective was to retrieve bibliographic references (composed mainly of title, author names and abstract) from a large subset of the MEDLINE repository, in order to meet real user needs. Last year [2], the main goal was to retrieve text fragments or passages rather than the entire scientific article. From an IR point of view, this task lies somewhere between classical text retrieval in which search responses consists of documents (or references to these documents) and question/answering where responses consist of very short passages extracted from documents. The term "passage" is in fact not very precise, given it could refer to a paragraph, sentence, or a short window of *n* characters.

For the Blog track [3], the IR system has to retrieve relevant information from different permalink documents (URLs pointing to a specific blogging entry), representing various points of view on various domains. Unlike traditional document collections used in the IR domain, a blog is more subjective, while also being characterized by more diverse document structures and writing styles. Even though the blogsphere may contain objective information (facts), the objective of the Blog track is to find answers based on opinions rather than relevant factual information. As such, relevant answers to the request "IPhone" may include factual and technological information (relevant but unopinioned answers) but also more personalized (and subjective) aspects of the product (why it is useful, complaints about this new tool, drawbacks of using a specific function, personal experiences concerning new product, etc.). Thus, in a first step the answer would contain a ranked list of relevant documents, but in a second stage a classification procedure would subdivide them into documents not based on opinion (factual information or descriptions), or documents expressing positive, mixed or negative opinion about the target entity.

The rest of this paper is organized as follows. Section 2 depicts the main characteristics of the Genomics test-collection and how passages are derived from an article according to our definition while Section 3 describes the main features of the Blog test-collection. Section 4 describes the indexing approach and Section 5 briefly presents the three probabilistic models used to search the genomics or blogsphere. Section 6 evaluates the three IR models by applying different conditions. Finally, the main findings of this paper are presented in Section 7.

2. GENOMICS TEST-COLLECTION

The document collection used this year contains approximately 12 GB of uncompressed data, made up of 162,259 full-text publications extracted from 49 biomedical journals (for more details, see the Web site at http://ir.ohsu.edu/genomics/2006data.html). To facilitate the effective retrieval of relevant passages and not documents, the IR literature [4] defines passages according to their various types, based mainly on delimiters such as text, window or semantic markers.

In a first approach to defining passages, we processed each article in order to generate its corresponding passages. As passage delimiters, we assigned the following HTML tags: H1, H2, H3, H4, H5, H6, P, BR, HR, TABLE, TD, TH, TR, OL, and UL.

```
<PASSAGE>
<FN> /raid/Genomics/peds/12118078.html
<ID> 12118078.23
<SO> 28541
<L> 978
<TGN> p
<R> false
<TITLE> Alterations in the Mouse and Human
Proteome Caused by Huntington's disease
<TX> In addition to the cytoplasmic brain
fraction that was used in the above experiments,
proteins solubilized by urea and detergent
treatment, yielding an extract enriched in
membrane proteins, as well as DNA-binding
proteins released by DNase, were screened to
expand the range of protein classes studied. In
both fractions no additional proteins were
consistently different between R6/2 and control
mice (data not shown). AAT was present at low
amounts in the membrane fraction and
undetectable in the fraction of proteins
released by DNase in control mice, arguing for a
mainly cytoplasmic localization of the protein
(data not shown). ABC was found in all three
fractions. A consistently lower expression of
ABC and AAT expression below the detection limit
were detected in R6/2 samples in all three
fractions (data not shown).
</PASSAGE>
```

Figure 1. Example of generated passage

Figure 1 shows an example of a passage that might be generated. All our passages are structured according to the following set of fields.

- FN (article filename path),
- ID (passage identifier),
- SO (start offset),
- L (passage length in bytes),
- TGN (tag name from which the passage was extracted),
- R (indicates whether or not the passage is identified as a reference),
- TITLE (title of article),
- TX (passage contents).

Following the filtering of all passages containing fewer than 10 words, the resulting collection contained exactly 10,700,925 passages from which 1,275,132 (11.9%) were marked as references.

For a second passage definition we used the sentence level and reused the subdivision structure applied at Erasmus MC - University Medical Center Rotterdam (the Netherlands) (see the Web site www.biosemantics.org).

This collection consisted also of 36 topics (numbered #200 to #235) corresponding to the real information needs commonly expressed by biologists (see Figure 2 for examples). Each topic relates to one of the 14 possible biological entity types (e.g., antibodies, diseases, mutations, pathways, tumor types, signs or symptoms). This information could thus be used to automatically enlarge the submitted query.

```
<ID> 200
<QUESTION> What serum [PROTEINS] change
expression in association with high disease
activity in lupus?

<ID> 214
<QUESTION> What [GENES] are involved axon
guidance in C.elegans

<ID> 232
<QUESTION>  What [DRUGS] inhibit HIV type 1
infection?
```

Figure 2. Examples of three topics (genomics corpus)

3. BLOG TEST-COLLECTION

The Blog test collection contains approximately 148 GB of uncompressed data, made up of 4,293,732 documents extracted from three sources: 753,681 feeds (or 17.6%), 3,215,171 permalinks (74.9%) and 324,880 homepages (7.6%). Their size is as follows; 38.6 GB for feeds (or 26.1%), 88.8 GB for permalinks (60%) and 20.8 GB for the homepages (14.1%). In this evaluation campaign only the permalink part is used. This corpus was crawled between Dec. 2005 and Feb. 2006 (for more information see: http://ir.dcs.gla.ac.uk/test_collections/).

Figure 3 depicts two examples of blog documents, showing their date, URL source and permalink structure at the beginning of each document. Some information extracted during the crawl is placed after the **<DOCHDR>** tag. Additional pertinent information follows after the **<DATA>** tag, along with ad links, name sequences (e.g., authors, countries, cities) plus various menu or site map items. Finally there is some factual information, such some of the locations where various different opinions can be found.

```
<DOC>
<DOCNO> BLOG06-20051212-051-0007599288
<DATE_XML> 2005-10-06T14:33:40+0000
<FEEDNO> BLOG06-feed-063542
<FEEDURL> http://
contentcentricblog.typepad.com/ecourts/index.rdf
<PERMALINK>
http://contentcentricblog.typepad.com/ecourts/20
05/10/efiling_launche.html#
<DOCHDR> …
Date: Fri, 30 Dec 2005 06:23:55 GMT
Accept-Ranges: bytes
Server: Apache
Vary: Accept-Encoding,User-Agent
Content-Type: text/html; charset=utf-8
…
<DATA>
electronic Filing &amp;amp; Service for Courts
…
October 06, 2005
eFiling Launches in Canada
Toronto, Ontario, Oct.03 /CCNMatthews/ -
LexisNexis Canada Inc., a leading provider of
comprehensive and authoritative legal, news, and
business information and tailored applications
to legal and corporate researchers, today
announced the launch of an electronic filing
pilot project with the Courts
…
```

Figure 3. Example of LexisNexis blog page

```
<DOC>
<DOCNO> BLOG06-20060212-023-0012022784
<DATE_XML> 2006-02-10T19:08:00+0000
<FEEDNO> BLOG06-feed-055676
<FEEDURL> http://
lawprofessors.typepad.com/law_librarian_blog/ind
ex.rdf#
<PERMALINK>
http://lawprofessors.typepad.com/law_librarian_b
log/2006/02/free_district_c.html#
<DOCHDR> …
Connection: close
Date: Wed, 08 Mar 2006 14:33:59 GMT …
<DATA>
Law Librarian Blog

Blog Editor
Joe Hodnicki
 Associate Director for Library Operations
 Univ. of Cincinnati Law Library
…
News from PACER   :

&amp;amp;quot;In the spirit of the E-Government Act
of 2002, modifications have been made to the
District Court CM/ECF system to provide PACER
customers with access to written opinions free
of charge

The modifications also allow PACER customers to
search for written opinions using a new report
that is free of charge. Written opinions have
been defined by the Judicial Conference as
&amp;amp;quot;any document issued by a judge or
judges of the court sitting in that capacity,
that sets forth a reasoned explanation for a
court's decision.&amp;amp;quot; …
```

Figure 4. Example of blog document

During this evaluation campaign a set of 50 topics (Topics #901 to #950) was created from this corpus. Like last year (Topics #851 to #900) they express user information needs extracted from a commercial search engine blog log, such as the examples shown in Figure 5.

```
<ID> 916
<TITLE> dice.com
<DESC> Find opinions concerning dice.com,
an on-line job search site.
<NARR> Opinions on dice.com's effectiveness
are relevant. Mention of its problems is
relevant. Recounting an experience using
dice.com is relevant. Simply mentioning it
as a possible tool is not relevant.

<ID> 928
<TITLE> “big love”
<DESC> Find opinions regarding the HBO
television show "Big Love".
<NARR> All statements of opinion regarding
the HBO production "Big Love" are relevant.
Statements of opinion about HBO or actors
in the show are relevant provided that "Big
Love" is mentioned.

<ID> 937
<TITLE> LexisNexis
<DESC> Find opinions about the information
service LexisNexis.
<NARR> Relevant documents will provide
opinions about the information service
LexisNexis. Documents that are obviously
sponsored by LexisNexis are considered to
be spam and not relevant.
```

Figure 5. Three examples of Blog track topics

Based on relevance assessments (relevant facts & opinions, or relevance value ≥ 1) made on this test collection, we listed 12,187 correct answers. The mean number of relevant web pages per topic is 243.74 (median: 208; standard deviation: 186.0). Topic #939 (“Beggin' Strips”) returned the minimal number of pertinent passages (16) while Topic #903 (“Steve jobs”) produced the greatest number of relevant passages (710).

Based on opinion-based relevance assessments (2 ≤ relevance value ≤ 4), we found 7,000 correct opinions. The mean number of relevant web pages per topic is 140.0 (median: 109.5; standard deviation: 123.456). Topic #910 (“Aperto Networks”) and Topic #950 (“Hitachi Data Systems”) returned a minimal number of pertinent passages (4) while Topic #903 (“Steve jobs”) produced the most relevant passages (496).

The polarity of opinions pertaining to target entities could be divided into three groups: negative (relevance value = 2), mixed (relevance value = 3) or positive (relevance value = 4) opinion. From an analysis of negative opinions only (relevance value = 2), we found 1,844 correct answers (mean: 40.087, median: 22.5, min: 1 (Topic #909 “Barilla”, #934 “cointreau”, #948 “sorbonne” or #950 “Hitachi Data Systems”), max: 189

(Topic #912, "nasa"), standard deviation: 45.12). Topic #901 ("jstor"), #910 ("Aperto Networks"), #914 ("northernvoice") and #925 ("mashup camp") obtained no positive opinions.

For positive opinions only (relevance value = 4), we found 2,960 correct answers (mean: 59.2, median: 49.5, min: 1 (Topic #950, "Hitachi Data Systems"), max: 234 (Topic #903, "Steve jobs"), standard deviation: 53.98). Finally for mixed opinions only (relevance value = 3), we found 2,196 correct answers (mean: 47.74, median: 22, min: 1 (Topic #901, "jstor", and Topic #925, "mashup camp"), max: 196 (Topic #946, "tivo"), standard deviation: 50.74).

4. INDEXING APPROACHES

To index documents or queries, we applied the indexing method described in Section 4.1. To derive orthographic variations of protein or gene names that could be included in topics, we used the algorithm described in Section 4.2.

4.1 Document Indexing

As a natural approach to indexing and searching both corpora, we chose words as the indexing units. As such our lexical analyzer applies the followings steps to process the input. First, the text is tokenized (using spaces or punctuation marks), simple acronyms are normalized (e.g., D.N.A. is converted into DNA) and hyphenated terms are also broken up into their components. For example, a word such as "COUP-TF1" generates three different forms, namely "COUP", "TF1" and the original form "COUP-TF1". Second, uppercase letters are transformed into their lowercase forms. Third, stopwords are filtered out using the SMART list (571 entries). Fourth, with the *S-stemmer* algorithm [5] based on three rules, we remove the final '-s' (the most common plural suffix for the English language). This choice is based on the experiments we did over previous years [6], [7] which demonstrate that out of the four evaluated stemmers (Lovins, *S-stemmer*, Porter and SMART) the *S-stemmer* provided the best retrieval effectiveness.

For the Blog task we also considered a second tokenization procedure. For example we noticed that in certain blogs there are rather long sequences of identical letters such as "aaaaah" and thus we retained only the first three letters, transforming it into "aaah".

4.2 Generation of Orthographic Variants

As is known, in biomedical literature there can be several orthographic variants [8] representing a given name, generally introduced for a variety of reasons:

1) Typographic errors and misspellings (e.g. "retreival" and "retrieval") or cognitive (e.g., "ecstasy", "extasy", or "ecstacy"; "occurence" or "occurrence");
2) Alternative punctuation and tokenization, mainly due to the lack of a naming convention (e.g. "Nur77", "Nurr-77" or "Nurr 77");
3) Regional language variations, such as British and American English (e.g. "colour" or "color", "grey" or "gray", etc.)
4) Transliteration of foreign names (e.g., "Crohn" and "Krohn" or "Creutzfeld-Jakob" and "Creutzfeldt-Jacob");
5) Morphological variations (inflections or derivations) which could be resolved by using a stemmer.

During previous TREC campaigns, many methods were proposed for resolving problems with orthographic variations, as for example [9]. The algorithms proposed were usually rule-based and were essentially concerned with secondary causes such as those described above (e.g., see [10]).

In order to automatically find a ranked list of alternative spellings for each search word, we modified the Lucene [11] Spell Checker[1]. In its initial stage this tool required a lexicon containing the correct spelling, so in our case we used the words extracted from the TREC 2005 corpus, a large subset of the MEDLINE collection. We then introduced a single term or a short sequence of words, limited in the current case to two terms. The spellchecker thus responded by returning a ranked list of the top 100 hits extracted from the given lexicon. In our case we used the following formula to re-ranked this list according to the minimal *edit-distance* measure and its length, calculated for each candidate considered a variant of the original (misspelled) term submitted:

Score = 1 – [edit-distance / length(term)]

When the two similar candidates were deemed to be equal (which occurred relatively frequently), they were ordered according to popularity (or *df*, document frequency), ranging from most to less frequent.

For each topic available in this TREC campaign, we submitted each search word or group of two successive words to the spellchecker engine. As shown in Figure 6, the spelling candidates were then re-sequenced by the *edit* and *df* measure and automatically added to the topic following the <BISPLELL-*n*> tag (followed by the alternative number).

In Figure 6, the *input* attribute describes the term submitted to the spellchecker. The *score* attribute refers to the final score achieved by the alternative term.

We then used the WordNet thesaurus to automatically enlarge the query. As shown in Figure 6 for the entity in question and the tag <ENTITY-EXPANSION> we could add

[1] http://wiki.apache.org/jakarta-lucene/SpellChecker

synonyms (e.g. “dna” for Topic #214) or morphologically related terms (e.g., “signal signaling signalize singnalise” to the term “signal”), and modifications such as these were made for 30 out of 50 queries. Finally for the tag <MEDICAL-TERM> we added synonyms from the question words extracted from the WordNet thesaurus. The number of added synonyms is relatively low (e.g., 20 words for the 50 queries under the tag <MEDICAL-TERM>).

```
<ID>  200
<ENTITY>  PROTEINS
<ENTITY-EXPANSION>
<QUESTION>  What serum PROTEINS change
expression in association with high disease
activity in lupus
<MEDICAL-TERM>
<BISPELL-1 input="serum proteins" score="0.86"
freq="1">  serum-proteina
<BISPELL-2 input="serum proteins" score="0.85"
freq="15">  serum-protein
<BISPELL-1 input="disease activity" score="0.94"
freq="3">  disease-activity

<ID>  214
<ENTITY>  GENES
<ENTITY-EXPANSION>  dna
<QUESTION>  What GENES are involved axon
guidance in C.elegans
<MEDICAL-TERM>
<BISPELL-1 input="axon guidance" score="0.92"
freq="5">  axon-guidance
```

Figure 6. Example of two topics, their orthographic variants and their WordNet expansions

5. RETRIEVAL MODELS

In our evaluations we conducted experiments by applying the single IR models described in Section 5.1 or by merging the result lists computed by various single IR models as explained in Section 5.2 (data fusion).

5.1 Single IR Models

To begin our evaluation we considered three probabilistic retrieval models. As a first approach, we used the Okapi (BM25) model [12], evaluating the document D_i score for the current query Q using the following formula:

$$Score(D_i, Q) = \sum_{t_j \in q} qtf \cdot \log\left(\frac{n - df_j}{df_j}\right) \cdot \frac{(k_1 + 1) \cdot tf_{ij}}{K + tf_{ij}}, \quad (1)$$

$$\text{where } K = k_1 \cdot \left[(1 - b) + b \cdot (l_i / avdl)\right]$$

in which the constant *advl* was fixed at 839 for the Blog corpus and 14 with sentences (Genomics) or 63 with our passage delimitation (Genomics), b was set to either 0.4 (Blog), 0.55 (Genomics, passages), or 0.35 (Genomics, sentences) and $k_1 = 1.4$ (Blog) or 1.2 (Genomics).

As a second approach, we implemented various models derived from the *Divergence from Randomness* (DFR) paradigm [13]. In this case, the document score was evaluated as:

$$Score(D_i, Q) = \sum_{t_j \in q} qtf \cdot w_{ij} \quad (2)$$

where *qtf* denotes the frequency of term t_j in query Q, and the weight w_{ij} of term t_j in document D_i is based on combining two information measures as follows:

$$w_{ij} = Inf^1_{ij} \cdot Inf^2_{ij} = -\log_2[Prob^1_{ij}(tf)] \cdot (1 - Prob^2_{ij}(tf))$$

As a first model, we implemented the PB2 scheme, defined by the following equations:

$$Inf^1_{ij} = -\log_2[(e^{-\lambda_j} \cdot \lambda_j^{tf_{ij}})/tf_{ij}!] \quad \text{with } \lambda_j = tc_j / n \quad (3)$$

$$Prob^2_{ij} = 1 - [(tc_j + 1) / (df_j \cdot (tfn_{ij} + 1))] \quad \text{with } tfn_{ij} = tf_{ij} \cdot \log_2[1 + ((c \cdot mean\ dl) / l_i)] \quad (4)$$

where tc_j indicates the number of occurrences of term t_j in the collection, l_i the length (number of indexing terms) of document D_i, *mean dl* is the average document length (fixed at 839 for the Blog, or 63 for the Genomics), n the number of documents in the corpus, and c a constant (= 5 for the Blog or the Genomics sentences or to 9.5 for the Genomics passages).

For the second model PL2, the implementation of $Prob^1_{ij}$ is given by Equation 3, and $Prob^2_{ij}$ by Equation 4, as shown below:

$$Prob^2_{ij} = tfn_{ij} / (tfn_{ij} + 1) \quad (4)$$

where λ_j and tfn_{ij} were defined previously.

For the third model called IneC2, the implementation is given by the following two equations:

$$Inf^1_{ij} = tfn_{ij} \cdot \log_2[(n+1) / (n_e + 0{,}5)] \quad \text{with } n_e = n \cdot [1 - [(n-1)/n]^{tc_j}] \quad (5)$$

$$Prob^2_{ij} = 1 - [(tc_j + 1) / (df_j \cdot (tfn_{ij} + 1))] \quad (6)$$

where n, tc_j and tfn_{ij} were defined previously, and df_j indicates the number of documents in which the term t_j occurs.

A third approach we considered was based on a statistical language model (LM) [14], [15], where probability estimates would be estimated directly, based on occurrence frequencies in document D_i or corpus C. According to this language model paradigm, various implementation and smoothing methods could be considered, although in this study we adopted the model proposed by Hiemstra [15] as described in Equation 7, combining an estimate based on document ($P[t_j \mid D_i]$) and on corpus ($P[t_j \mid C]$).

$$P[D_i \mid Q] = P[D_i] \cdot \prod_{t_j \in Q} [\lambda_j \cdot P[t_j \mid D_i] + (1 - \lambda_j) \cdot P[t_j \mid C]]$$
$$\text{with } P[t_j \mid D_i] = tf_{ij}/l_i \quad \text{and } P[t_j \mid C] = df_j/lc$$
$$\text{and with } lc = \sum_k df_k \quad (7)$$

where λ_j is a smoothing factor (constant for all indexing terms t_j, and usually fixed at 0.35) and *lc* an estimate of the size of the corpus C.

5.2 Combining Different IR Models

It is assumed that combining different search models would improve retrieval effectiveness, due to the fact that each document representation might retrieve pertinent items not retrieved by others and thus increase overall recall [16]. In this current study we combined three probabilistic models representing both the parametric (Okapi and DFR) and non-parametric (language model or LM) approaches. Various fusion operators have been suggested to perform these combinations, such as the "Sum RSV" operator, where the combined document score (or the final retrieval status value) is simply the sum of the retrieval status value (RSV_k) for the corresponding document D_k computed by each single indexing scheme [17].

$$\text{Z-score } RSV_k = [((RSV_k - Mean^i) / Stdev^i) + \delta^i],$$
$$\delta^i = ((Mean^i - Min^i) / Stdev^i) \quad (8)$$

This year, we only used the Z-Score operator (shown in Eq. 8) to combine two or more single runs. To do this we needed to compute the average RSV_k value (denoted $Mean^i$) and the standard deviation (denoted $Stdev^i$) for each *i*th result list. These values could then be used to normalize the retrieval status for each document D_k found in the *i*th result list through computing the deviation for RSV_k with respect to the mean ($Mean^i$). Of course another method would be to weight the relative contribution of each retrieval scheme by assigning a different α_i value to each retrieval model.

6. EVALUATION

To evaluate our various search strategies, we used the tool provided by the organizers, based on the TREC_EVAL method to measure retrieval effectiveness. Based on the retrieval of 1,000 passages per query, this program computed different performance measures (e.g., the MAP). For the Blog collection, we limited our investigation to the opinion-finding task, namely the retrieval of information on the target entities without classifying them as positive, negative or mixed. For the Genomics task, the MAP was used in three different types of granularity at the document, passage and passage2 levels, and also at the feature level.

6.1 Genomics Official Runs

Table 1 provides a description of our three official runs within the Genomics task. These runs were based on the two probabilistic models (Okapi & I(n)B2) and include some of the search features described previously. First we listed the I(n)B2 model with the WordNet expansions (see Figure 6 for an example). In our second official run we applied WordNet thesaurus expansions and for our third we considered orthographic variants resulting from WordNet expansions.

Run name	IR model	Passage defined by
UniNE1	I(n)B2 + WordNet Exp.	**<P </P**
UniNE2	Okapi + WordNet Okapi + reranking I(n)B2 + WordNet	sentence
UniNE3	I(n)B2 + WordNet + Spell. Okapi + WordNet I(n)B2 + WordNet	**<P </P**

Table 1. Description of official runs (Genomics track)

Run name	MAP document	MAP passage2	MAP aspect	Passage defined by
Okapi	0.1486	0.0190	0.0633	**<P </P**
Okapi	0.1289	0.0089	0.0740	sentence
I(n)B2	0.2533	0.0907	0.2036	**<P </P**
I(n)B2	0.1508	0.0193	0.0952	sentence
Okapi+WN	0.1690	0.0287	0.0388	**<P </P**
Okapi+WN	0.1566	0.0166	0.0896	sentence
I(n)B2+WN	0.2777	0.0998	0.2177	**<P </P**
I(n)B2+WN	0.1978	0.0347	0.1227	sentence
Okapi+Spell	0.1462	0.01883	0.0602	**<P </P**
Okapi+Spell	0.1219	0.0084	0.0683	sentence
I(n)B2+Spell	0.2510	0.0902	0.2019	**<P </P**
I(n)B2+Spell	0.1538	0.0179	0.0850	sentence
Okapi+WN+Sp	0.1671	0.02819	0.0707	**<P </P**
Okapi+WN+Sp	0.1509	0.0159	0.0875	sentence
I(n)B2+WN+S	0.2765	0.0983	0.2177	**<P </P**
I(n)B2+WN+S	0.1961	0.0328	0.1188	sentence
UniNE1	**0.2777**	**0.0988**	**0.2189**	**<P </P**
UniNE2	0.1903	0.0278	0.1102	sentence
UniNE3	0.2710	0.0978	0.2043	**<P </P**

Table 2. Official Genomic track results and their components

Table 2 lists the evaluation results for our three official runs, together with their various components. Listed first in this table are the single IR models (Okapi & I(n)B2), and then these same models with the WordNet (WN) query expansion option (lines 5 to 8). In lines 9 and 12 we used the Okapi and I(n)B2 models along with spelling variations of the search terms, and finally we evaluated the Okapi and I(n)B2 approaches with both WordNet and orthographic variant expansions (lines 13 and 16). Our three official runs thus combined IR models based on the Z-score approach (see Section 5.2).

The results listed in Table 2 show that through using the WordNet thesaurus, we could enlarge the query (both with synonyms and morphological related terms) and improve the MAP results (from 9.6% to 31.2% in relative values). For example, with the I(n)B2 model, the MAP increases from 0.2533 to 0.2777 (+9.6%). Including orthographic variants tend to hurt slightly the MAP values (from -5.4% to 2%). When compared to the use of passage segmentation (denoted **<P** **</P** in Table 2), the use of sentences as passages was clearly not a good idea. Applying the document-based MAP, our best run (UniNE1) produced performances that were 30 times better than the median of all submitted runs.

6.2 Opinion-Finding Official Runs

To search information in the blogsphere, we based our official runs on three IR systems, namely the probabilistic Okapi model, the language model (LM) and models derived from the *Divergence from Randomness* (DFR) paradigm. See Table 3 for an evaluation of these different IR approaches and three query formulations (T, TD and TDN). In this case we considered all factorial web pages to be relevant (relevance value, rv=1) and all documents comprising various opinions (negative rv=2, mixed rv=3 or positive rv=4) concerning the specified target entity.

IR Model	T	TD	TDN
Okapi	**0.3585**	0.4003	0.3965
DFR-PL2	0.3568	**0.4033**	**0.3942**
DFR-IneC2	0.3398	0.3849	0.3771
DFR-I(n)B2	0.3397	0.3770	0.3606
DFR-PB2	0.3365	0.3767	0.3617
LM	0.3331	0.3808	0.3812

Table 3. Fact and opinion evaluations of the single IR models (Blog, three query formulations)

This table illustrates how the Okapi or the DFR-PL2 approaches produced the best results, albeit with rather small differences. Through adding the descriptive part in the query formulation we might improve the MAP by 12.5% in mean. Also worth noting is that increasing the query from TD to TDN does not necessarily improve the MAP values (mean decrease of -2.2%). Table 4 lists our six official runs for the Blog track Table 5 lists our official results.

Our official results for the Blog track tend to indicate that simple IR models perform better than more complex search strategies. With the TD query formulation for example, combining two IR models for the UniNEblog3 run produced an MAP of 0.4034, while under the same conditions the DFR-PB2 by itself model achieved an MAP of 0.4033 (see Table 3).

Run name	IR model
UniNEblog1	Okapi
UniNEblog2	DFR-PL2
UniNEblog3	DFR-PB2 + Okapi & Rocchio 5/50
UniNEblog4	LM (λ=0.35) + DFR-PL2
UniNEblog5	DFR In2C2 + Okapi (5-gram) + LM (λ=0.35, three letters)
UniNEblog6	LM (λ=0.35)

Table 4. Description of official Blog track results

Run name	QUERY	RELEVANT	POLARITY
UniNEblog1	T	0.3585	0.2770
UniNEblog2	TDN	0.3942	0.2898
UniNEblog3	TD	**0.4034**	**0.3049**
UniNEblog4	T	0.3467	0.2659
UniNEblog5	TD	0.3892	0.2972
UniNEblog6	TD	0.3808	0.3016

Table 5. Official results of the Blog track results

6.3 Difficult Topics in the Blog Track

Table 6 lists the top five most difficult topics of our best performing runs and also provides a better picture of the problems encountered when our systems searched the Blog track (UniNEblog3).

Topic ID	AP	Main explanation
#916	0.0005	Too many spam
#937	0.0049	Discrimination fails
#928	0.0177	Stopword list too large
#921	0.0373	Discrimination fails
#929	0.0571	Discrimination fails

Table 6. The most difficult topics in our best runs (UniNEblog3)

Because this search model does not account for noun phrases, there was a decrease in retrieval effectiveness due to our inability to impose the presence of two (or more) search terms. With title-only queries such as Topic #929 ("Brand manager"), Topic #921 ("Christianity Today") or Topic #928 ("Big Love") for example, the presence of both terms in the web page should be imposed and thus ensure their retrieval. Our IR models tend to extract many documents because one of the search terms has a high term frequency.

A second problem is our extended stopword list. In order to ignore HTML-tags (which may have passed the parsing step) and also to remove very frequent blog words, we added a few terms to our stopword list (e.g., big, com). In

Topic #928 ("Big Love") or Topic #916 ("dice.com") however this reduced the underling query to the single term "love" or "dice", meaning that such a query would not effectively retrieve and rank highly relevant web pages.

For Topic #916 ("dice.com"), our IR systems encountered a problem related to spam. Given that "dice.com" was reduced to "dice", most retrieved documents at the top of the result list assigned very high term frequency to the term "dice". Most of the spam blogs retrieved thus had the same content, being a list of popular internet searches containing terms such as "dice game", "dodecahedron dice" or "Dice Games and Rules", all of which originate from the same server (newgreatblogs.com).

For Topic #937 ("LexisNexis") most of the highly ranked yet non-relevant web pages were retrieved from the same blog (lawprofessors.typepad.com/law_librarian_blog), which contains numerous links to the LexisNexis web site. The outcome was an increase in the *tf* component for those pages, providing them with higher ranks. Unfortunately we cannot simply ignore these pages because they originate from a blog that also contains some relevant documents.

7. CONCLUSION

During this TREC 2007 Genomic evaluation campaign we evaluated various indexing and search strategies. The empirical evidence collected shows that the DFR-I(n)B2 model tends to perform better than the Okapi probabilistic model (0.2533 vs. 0.1486, document-based MAP). The inclusion of orthographic variants for search words (or two-word query sequences) does not really improve retrieval effectiveness, at least as implemented in our system (e.g., with the I(n)B2 model, from 0.2533 to 0.2510). Enlarging query formulations by adding synonyms or morphological related words extracted from the WordNet thesaurus results in better MAP (e.g., from 0.2533 to 2777 using the I(n)B2 model). Our passage segmentation approach was clearly more efficient than an approach based on sentences.

In the Blog track (limited in our case to retrieving opinions on a target entity), we find that the Okapi or the DFR-PL2 search models tend to produce the best MAP for certain query formulations. For example with the T query formulation we obtained a MAP of 0.3585 for the Okapi model compared to 0.3331 for the language model (-7.1%). By including the topic's descriptive part, this formulation increases the MAP by around 12% in mean (e.g., Okapi 0.3585 vs. 0.4003). Including the narrative part however tends to hurt the MAP (mean decrease around -2%). Moreover, simple IR models tend to produce retrieval performance similar to that of more complex IR strategies, such as those combining two ranked lists. When using TD queries for example the DFR-PL2 produces a MAP of 0.4033 while with a combined run (DFR-PB2 and Okapi plus pseudo-relevance feedback) a similar MAP (0.4034) resulted. In an effort to improve the MAP, we analyzed various difficult topics and their result lists. From an analysis of these resultant ranked lists we concluded that accounting for noun phrases (e.g., "Brand manager", "Big Love") or at least accounting for the presence of the two (or more) search terms in the retrieved web page may improve the MAP.

ACKNOWLEDGMENTS

This research was supported in part by the Swiss NSF under Grant #200021-113273.

8. REFERENCES

[1] Hersh, W.R., Cohen, A.M., Yang, J., Bhuptiraju, R.T., Roberts, P., & Hearst, M. TREC 2005 genomics track overview. In *Proceedings of TREC-2005.* Gaithersburg (MA), 2006.

[2] Hersh, W.R., Cohen, A.M., Roberts, P., & Rekapalli, H.K. TREC 2006 genomics track overview. In *Proceedings of TREC-2006.* Gaithersburg (MA), NIST Publication #500-272, 2007.

[3] Ounis, I., de Rijke, M., Macdonald, C., Gilad Mishne, G., & Soboroff, I. Overview of the TREC-2006 blog track. In *Proceedings of TREC-2006.* Gaithersburg (MA), NIST Publication #500-272, 2007.

[4] Kaszkiel, M., & Zobel, J. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology*, 52(4), 2001, 344-364.

[5] Harman, D. How effective is suffixing? *Journal of the American Society for Information Science*, 42(1), 1991, 7-15.

[6] Abdou, S., Ruch, P., & Savoy, J. Evaluation of stemming, query expansion and manual indexing approaches for the genomic task. In *Proceedings TREC-2005*, Gaithersburg (MA), 2006, 863-871.

[7] Abdou, S., & Savoy, J. Report on the TREC 2006 genomics experiment. In *Proceedings TREC-2006*, Gaithersburg (MA), NIST Publication #500-272, 2007.

[8] Yu, H., & Agichtein, E. Extracting synonymous gene and protein terms from biological literature. *Bioinformatics*, 19(1), 2003, i340-i349.

[9] Huang, X., Zhong, M., & Si, L. York University at TREC 2005: Genomics track. In *Proceedings of TREC-2005*. Gaithersburg (MA), 2006.

[10] Cohen, A.M. Unsupervised gene/protein named entity normalization using automatically extracted

dictionaries. In *Proceeding ACL-ISMB*, Detroit (MI), 2005, 17-24.

[11] Gospodnetic, O., & Hatcher, E. *Lucene in Action*. Manning Publications, 2004

[12] Robertson, S.E., Walker, S., & Beaulieu, M. Experimentation as a way of life: Okapi at TREC. *Information Processing & Management*, 36(1), 2000, 95-108.

[13] Amati, G., & van Rijsbergen, C.J. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM-Transactions on Information Systems*, 20(4), 2002, 357-389.

[14] Hiemstra, D. Using language models for information retrieval. CTIT Ph.D. Thesis, 2000.

[15] Hiemstra, D. Term-specific smoothing for the language modeling approach to information retrieval. In *Proceedings of the ACM-SIGIR*, 2002, 35-41.

[16] Vogt, C.C. & Cottrell, G.W. Fusion via a linear combination of scores. *IR Journal*, 1(3), 1999, 151-173.

[17] Fox, E.A. & Shaw, J.A. Combination of multiple searches. In *Proceedings TREC-2*, Gaithersburg (MA), NIST Publication #500-215, 1994, 243-249.