UniNE at TREC 2008: Fact and Opinion Retrieval in the Blogsphere

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 Blog track at the TREC 2008 evaluation campaign. The Blog track goes beyond simple document retrieval, its main goal is to identify opinionated blog posts and assign a polarity measure (positive, negative or mixed) to these information items. Available topics cover various target entities, such as people, location or product for example. This year's Blog task may be subdivided into three parts: First, retrieve relevant information (facts & opinionated documents), second extract only opinionated documents (either positive, negative or mixed) and third classify opinionated documents as having a positive or negative polarity.

For the first part of our participation we evaluate different indexing strategies as well as various retrieval models such as Okapi (BM25) and two models derived from the *Divergence from Randomness* (DFR) paradigm. For the opinion and polarity detection part, we use two different approaches, an additive and a logistic-based model using characteristic terms to discriminate between various opinion classes.

1. INTRODUCTION

In the Blog track [1] the retrieval unit consists of permalink documents, which are URLs pointing to a specific blog entry. In contrast to a corpus extracted from scientific papers or a news collection, blogposts are more subjective in nature and contain several points of view on various domains. Written by different kinds of users, a post retrieved following the request "TomTom" for might contain factual information about the navigation system, such as software specifications for example, but it might also contain more subjective information about the product such as ease of use. The ultimate goal of the Blog track is to find opinionated documents rather than present a ranked list of relevant documents containing either objective (facts) or subjective (opinions) content. Thus, in a first step the system would retrieve a set of relevant documents but then a second step this set would be separated into two subsets, one containing the documents without any opinions (facts) and the second containing documents expressing positive, negative or mixed opinions on the target entity. Finally the mixed-opinion documents would be eliminated and the positive and negative opinionated documents separated. Later in this paper, the documents retrieved during the first step will be referenced as a baseline or factual retrieval.

The rest of this paper is organized as follows. Section 2 describes the main features of the test-collection used. Section 3 explains the indexing approaches used and Section 4 introduces the models used for factual retrieval. In Section 5 we explain our opinion and polarity detection algorithms. Section 6 evaluates the different approaches as well as our official runs. The principal findings of our experiments are presented in Section 7.

2. BLOG TEST-COLLECTION

The Blog test collection contains approximately 148 GB of uncompressed data, consisting 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 sizes are 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%). Only the permalink part is used in this evaluation campaign. This corpus was crawled between Dec. 2005 and Feb. 2006 (for more information see: http://ir.dcs.gla.ac.uk/test_collections/).

Figures 1 and 2 show two blog document examples, including the date, URL source and permalink structures at the beginning of each document. Some information extracted during the crawl is placed after the **<DOCHDR>** tag. Additional pertinent information is placed after the **<DATA>** tag, along with ad links, name sequences (e.g., authors, countries, cities) plus various menu or site map items. Finally some factual information is included, such as some locations where various opinions can be found. The data of interest to us follows the <**DATA>** tag.

```
<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; 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 1. 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   :

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 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. …
```

Figure 2. Example of blog document

During this evaluation campaign a set of 50 new topics (Topics #1001 to #1050) as well as 100 old topics from 2006 and 2007 (respectively Topics #851 to #900 and #901 to #950) were used. They were created from this corpus and express user information needs extracted from the log of a commercial search engine blog. Some examples are shown in Figure 3.

```
<num> Number: 851
<title> "March of the Penguins"
<desc> Description:
Provide opinion of the film documentary
"March of the Penguins".
<narr> Narrative:
Relevant documents should include opinions
concerning the film documentary "March of
the Penguins".  Articles or comments about
penguins outside the context of this film
documentary are not relevant.

<num> Number: 941
<title> "teri hatcher"
<desc> Description:
Find opinions about the actress Teri
Hatcher.
<narr> Narrative:
All statements of opinion regarding the
persona or work of film and television
actress Teri Hatcher are relevant.

<num> Number: 1040
<title> TomTom
<desc> Description:
What do people think about the TomTom GPS
navigation system?
<narr> Narrative:
How well does the TomTom GPS navigation
system meets the needs of its users?
Discussion of innovative features of the
system, whether designed by the
manufacturer or adapted by the users, are
relevant.
```

Figure 3. Three examples of Blog track topics

Based on relevance assessments (relevant facts & opinions, or relevance value ≥ 1) made on this test collection, we listed 43,813 correct answers. The mean number of relevant web pages per topic is 285.11 (median: 240.5; standard deviation: 222.08). Topic #1013 (“Iceland European Union”) returned the minimal number of pertinent passages (12) while Topic #872 (“brokeback mountain”) produced the greatest number of relevant passages (950).

Based on opinion-based relevance assessments (2 ≤ relevance value ≤ 4), we found 27,327 correct opinionated posts. The mean number of relevant web pages per topic is 175.99 (median: 138; standard deviation: 169.66). Topic #877 (“sonic food industry”), Topic #910 (“Aperto Networks”) and Topic #950 (“Hitachi Data Systems”) returned a minimal number of pertinent passages (4) while Topic #869 (“Muhammad cartoon”) produced the greatest number of relevant posts (826).

The opinion referring to the target entity and contained in a retrieved blogpost may be negative (relevance value = 2), mixed (relevance value = 3) or positive (relevance value = 4). From an analysis of negative opinions only (relevance value = 2), we found 8,340 correct answers (mean: 54.08; median: 33; min: 0; max: 533; standard deviation: 80.20). For positive opinions only (relevance value = 4), we found 10,457 correct answers (mean: 66.42, median: 46; min: 0; max: 392; standard deviation: 68.99). Finally for mixed opinions only (relevance value = 3), we found 8,530 correct answers (mean: 55.48; median: 23; min: 0; max: 455; standard deviation: 82.33). Thus it seems that the test collection tends to contain, in mean, more positive opinions (mean 66.42) than it does either mixed (mean: 55.48) or negative opinions (mean: 54.08) related to the target entity.

3. INDEXING APPROACHES

We used two different indexing approaches to index documents and queries. As a first and natural approach we chose words as indexing units and their generation was done in three steps. First, the text is tokenized (using spaces or punctuation marks), hyphenated terms are broken up into their components and acronyms are normalized (e.g., U.S. is converted into US). Second, uppercase letters are transformed into their lowercase forms and third, stop words are filtered out using the SMART list (571 entries). Based on the result of our previous experiments within the Blog track [2] or Genomics search [3], we decided not to use a stemming technique.

In its indexing units our second indexing strategy uses single words and also compound constructions, with the latter being those composed of two consecutive words. For example in the Query #1037 *"New York Philharmonic Orchestra"* we generated the following indexing units after stopword elimination: "york," "philharmonic," "orchestra," "york philharmonic," "philharmonic orchestra" ("new" is included in the stoplist). We decided to use this given the large number of queries containing proper names or company names such as "David Irving" (#1042), "George Clooney" (#1050) or "Christianity Today" (#921) for example should be considered as one single entity for both indexing and retrieval. Once again we did not apply any stemming procedure.

4. FACTUAL RETRIEVAL

The first step in the Blog task was factual retrieval. To create our baseline runs (factual retrieval) we used different single IR models as described in Section 4.1. To produce more effective ranked results lists we applied different blind query expansion approaches as discussed in Section 4.2. Finally, we merged different isolated runs using a data fusion approach as presented in Section 4.3. This final ranked list of retrieved items was used as our baseline (classical *ad hoc* search).

4.1 Single IR Models

We considered three probabilistic retrieval models for our evaluation. As a first approach we used the Okapi (BM25) model [4], evaluating the document D_i score for the current query Q by applying the following formula:

$$Score(D_i, Q) = \sum_{t_j \in q} qtf_j \cdot \log\left(\frac{n - df_j}{df_j}\right) \cdot \frac{(k_1 + 1) \cdot tf_{ij}}{K + tf_{ij}}, \qquad (1)$$
$$\text{where } K = k_1 \cdot \left[(1 - b) + b \cdot (l_i / avdl)\right]$$

in which the constant *avdl* was fixed at 837 for the word-based indexing and at 1622 for our compound-based indexing. For both indexes the constant *b* was set to 0.4 and k_1 to 1.4.

As a second approach, we implemented two models derived from the *Divergence from Randomness* (DFR) paradigm [5]. In this case, the document score was evaluated as:

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

where qtf_j denotes the frequency of term t_j in query Q, and the weight w_{ij} of term t_j in document D_i was based on a combination of 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:

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

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

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* the average document length (fixed at 837 for word-based respectively at 1622 for compound-based indexing approach), *n* the number of documents in the corpus, and *c* a constant (fixed at 5).

For the second model PL2, the implementation of $Prob^1_{ij}$ is given by Equation 3, and $Prob^2_{ij}$ by Equation 5, as shown below:

$$Prob^2_{ij} = tfn_{ij} / (tfn_{ij} + 1) \qquad (5)$$

where λ_j and tfn_{ij} were defined previously.

4.2 Query Expansion Approaches

In an effort to improve retrieval effectiveness, various query expansion techniques were suggested [6], [3], and in our case we chose two of them. The first uses a blind query expansion based on Rocchio's method [7], wherein the system would add the top *m* most important terms extracted from the top *k* documents retrieved in the original query. As a second query expansion approach we used Wikipedia[1] to enrich those queries based on terms extracted from a source different from the blogs. The title of the original topic description was sent to Wikipedia and the ten most frequent words from the first retrieved article were added to the original query.

4.3 Combining Different IR Models

It was 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. On the other hand, we might assume that an item retrieved by many different indexing and/or search strategies would have a greater chance of being relevant for the query submitted [8], [9].

To combine two or more single runs, we applied the Z-Score operator [10] defined as:

$$Z-Score\ RSV_k = \sum_j \left[\frac{RSV_k^j - Mean^j}{Stdev^j} + \delta_j \right] \qquad (6)$$

with $\delta^i = ((Mean^j - Min^j) / Stdev^j)$

In this formula, the final document score (or its retrieval status value RSV_k) for a given document D_k is the sum of the standardized document score computed for all isolated retrieval systems. This later value was defined as the document score for the corresponding document D_k achieved by the *j*th run (RSV_k^j) minus the corresponding mean (denoted $Mean^j$) and divided by the standard deviation (denoted $Stdev^j$).

5. OPINION AND POLARITY DETECTION

Following the baseline retrieval, the goal was to separate the retrieved documents into two classes, namely opinionated and non-opinionated documents, and then in a subsequent step assign a polarity to the opinionated documents.

In our view, opinion and polarity detection are closely related. Thus, after performing the baseline retrieval, our system would automatically judge the first 1,000 documents retrieved. For each retrieved document the system may classify it as positive, negative, mixed or neutral (the underlying document contains only factual information). To achieve this we calculated a score for each possible outcome class (positive, negative, mixed, and neutral), and then the highest of these four scores determined the choice of a final classification.

Then for each document in the baseline we looked up the document in the judged set to obtain its classification. If the document was not there it was classified as *unjudged.*

Documents classified as positive, mixed or negative were considered to be opinionated, while neutral and unjudged documents were considered as non-opinionated. This classification also gave the document's polarity (positive or negative).

To calculate the classification scores, we used two different approaches, both being based on Muller's method for identifying a text's characteristic vocabulary [11], as described in Section 5.1. We then presented our two suggested approaches, the additive model in Section 5.2 and the logistic approach in Section 5.3.

5.1 Characteristic Vocabulary

In Muller's approach the basic idea is to use Z-score (or standard score) to determine which terms can properly characterize a document, when compared to other documents. To do so we needed a general corpus denoted *C,* containing a documents subset *S* for which we wanted to identify the characteristic vocabulary. For each term *t* in the subset *S* we calculated a Z-Score by applying Equation (7).

$$Z-Score(t) = \frac{f' - n'\mathrm{Prob}(t)}{\sqrt{n' \cdot \mathrm{Prob}(t) \cdot (1-\mathrm{Prob}(t))}} \qquad (7)$$

where *f'* was the observed number of occurrences of the term *t* in the document set *S,* and *n'* the size of *S.* Prob(t) is the probability of the occurrence of the term *t* in the entire collection *C.* This probability can be estimated according to the Maximum Likelihood Estimation (MLE) principle as Prob(t) = *f*/*n*, with *f* being the number of occurrences of *t* in *C* and *n* the size of *C.* Thus in Equation 7, we compared the expected number of occurrences of term *t* according to a binomial process (mean = *n'* · Prob(t)) with the observed number of occurrences in the subset *S* (denoted *f'*). In this binomial process the variance is defined as *n'* · Prob(t) · (1-Prob(t)) and the corresponding standard deviation becomes the denominator of Equation 7.

Terms having a Z-score between –ε and +ε would be considered as general terms occurring with the same frequencies in both the entire corpus *C* and the subset *S.* The constant ε represents a threshold limit that was fixed at 3 in our experiments. On the other hand, terms having an absolute value for the Z-score higher than ε are considered overused (positive Z-score) or underused (negative Z-score) compared to the entire corpus *C.* Such terms therefore may be used to characterize the subset *S.*

In our case, we created the whole corpus *C* using all 150 queries available. For each query the 1,000 first retrieved documents would be included in *C.* Using the relevance assessments available for these queries (queries #850 to

[1] http://www.wikipedia.org/

#950), we created four subsets, based on positive, negative, mixed or neutral documents, and thus identified the characteristic vocabulary for each of these polarities. For each possible classification, we now had a set of characteristic terms with their Z-score.

Defining the vocabulary characterizing the four different classes in one step, and in the second step it is to compute an overall score, as presented in the following section.

5.2 Additive Model

In our first approach we used characteristic term statistics to calculate the corresponding polarity score for each document. The scores were calculated by applying following formulae:

$$
\begin{aligned}
Pos\_score &= \frac{\#PosOver}{\#PosOver + \#PosUnder} \\
Neg\_score &= \frac{\#NegOver}{\#NegOver + \#NegUnder} \\
Mix\_score &= \frac{\#MixOver}{\#MixOver + \#MixUnder} \\
Neutral\_score &= \frac{\#NeuOver}{\#NeuOver + \#NeuUnder}
\end{aligned}
\tag{8}
$$

in which *#PosOver* indicated the number of terms in the evaluated document that tended to be overused in positive documents (i.e. Z-score > ε) while *#PosUnder* indicated the number of terms that tended to be underused in the class of positive documents (i.e. Z-score < -ε). Similarly, we defined the variables *#NegOver, #NegUnder, #MixOver, #MixUnder, #NeuOver, #NeuUnder,* but for their respective categories, namely negative, mixed and neutral.

The idea behind this first model is simply assigning the category to each document for which the underlying document has relatively the largest sum of overused terms. Usually, the presence of many overused terms belonging to a particular class is sufficient to assign this class to the corresponding document.

5.3 Logistic Regression

As a second classification approach we used logistic regression [12] to combine different sources of evidence. For each possible classification, we built a logistic regression model based on twelve covariates and fitted them using training queries #850 to #950 (for which the relevant assessments were available). Four of the twelve covariates are *SumPos, SumNeg, SumMix, SumNeu* (the sum of the Z-scores for all overused and underused terms for each respective category). As additional explanatory variables, we also use the 8 variables defined in Section 5.2, namely *#PosOver, #PosUnder, #NegOver, #NegUnder, #MixOver, #MixUnder, #NeuOver,* and *#NeuUnder*. The score is defined as the logit transformation π(x) given by each logistic regression model is defined as:

$$\pi(x) = \frac{e^{\beta_0 + \sum_{i=1}^{12} \beta_i x_i}}{1 + e^{\beta_0 + \sum_{i=1}^{12} \beta_i x_i}} \tag{9}$$

where β_i are the coefficients obtained from the fitting and x_i the variables. These coefficients reflect the relative importance of each explanatory variable in the final score.

For each document, we compute the π(x) corresponding to the four possible categories and for the final decision we need simply to classify the post according to the maximum π(x) value. This approach accounts for the fact that some explanatory variables may have more importance than others in assigning the correct category. We must recognize however that the length of the underlying document (or post) is not directly taken into account in our model. Our underling assumption is that all documents have a similar number of indexing tokens. As a final step we could simplify our logistic model by ignoring explanatory variables having a coefficient estimate (β_i) close to zero and for which a statistical test cannot reject the hypotheses that the real coefficient β_i = 0.

6. EVALUATION

To evaluate our various IR schemes, we adopted mean average precision (MAP) computed by `trec_eval` software to measure the retrieval performance (based on a maximum of 1,000 retrieved records). As the Blog task is composed of three distinct subtasks, namely the *ad hoc* retrieval task, the opinion retrieval task and the polarity task, we will present these subtasks in the three following sections.

6.1 Baseline Ad hoc Retrieval Task

A first step in the Blog track was the *ad hoc* retrieval task, where participants were asked to retrieve relevant information about a specified target. These runs also served as baselines for opinion and polarity detection. In addition the organizers provided 5 more baseline runs to facilitate comparisons between the various participants' opinion and polarity detection strategies. We based our official runs on two different indexes (single words under the label "W" and compound construction under the label "comp." see Section 3) and on two different probabilistic models (see Section 4). We evaluated these different approaches under three query formulations, T (title only), TD (title and description) and TD$^+$. In the latter case, the system received the same TD topic formulation as previously but during the query representation process the system built a phrase query from the topic description's title section. Table 1 shows the results and Table 2 the results of our two different query expansion techniques.

Model	T		TD		TD$^+$	
	comp.	W	comp.	W	comp.	W
Okapi	**0.374**	0.337	**0.403**	0.372	0.400	0.390
PL2	0.368	0.336	0.398	0.378	0.396	0.392
PB2	0.362	0.321	0.394	0.358	0.374	0.380

Table 1. MAP of different IR models (*ad hoc* search) (Blog, T & TD query formulations)

As shown in Table 1 the performance for the Okapi and the DFR schemes is almost the same, with the Okapi perhaps having a slight advantage. This table also shows that using compound indexing approach (word pairs) or phrase (from the title section of the query) increases the performance. This can be explained by the fact that in the underling test collection numerous queries contain statements that should appear together or close together in the retrieved documents, such as names (e.g. #892 "Jim Moran", #902 "Steve Jobs" or #931 "fort mcmurray") or concepts (e.g. #1041 "federal shield law"). Finally it can also be observed that adding the descriptive part (D) in the query formulation might improve the MAP.

	T	
	comp.	W
Okapi (baseline)	0.374	0.336
Rocchio 5 doc/ 10 terms	0.387	**0.344**
Rocchio 5 doc/ 20 terms	0.386	0.331
Rocchio 5 doc/ 100 terms		0.253
Rocchio 10 doc/ 10 terms	0.384	0.343
Rocchio 10 doc/ 20 terms	**0.390**	0.339
Rocchio 10 doc/ 100 terms		0.277
Wikipedia	0.387	0.342

Table 2. Okapi model with various blind query expansions

Table 2 shows that Rocchio's blind query expansion might slightly improve the results, but only if a small number of terms is considered. When adding a higher number of terms to the original query, the system tends to include more frequent terms such as navigational terms (e.g. "home", "back", "next") that are not related to the original topic formulation. The resulting MAP tends therefore to decrease. Using Wikipedia as an external source of potentially useful search terms only slightly improves the results (an average improvement of +2.75% on MAP).

Table 3 lists our two official baseline runs for the Blog track and Table 4 the MAP for both the topic (or *ad hoc*) search and opinion search for our two official baseline runs, as well as for the additional five baseline runs provided by the organizers.

Run Name	Query	Index	Model	Expansion
UniNEBlog1	T	comp.	Okapi	Rocc. 5/20
	TD	comp.	PL2	none
	TD$^+$	W	PB2	none
UniNEBlog2	T	comp.	Okapi	Wikipedia
	T	comp.	Okapi	Rocc. 5/10

Table 3. Description of our two official baseline runs for *ad hoc* search

Run Name	Topic MAP	Opinion MAP
UniNEBlog1	**0.424**	**0.320**
UniNEBlog2	0.402	0.306
Baseline 1	0.370	0.263
Baseline 2	0.338	0.265
Baseline 3	0.424	0.320
Baseline 4	**0.477**	**0.354**
Baseline 5	0.442	0.314

Table 4. *Ad hoc* topic and opinion relevancy results for baseline runs

6.2 Opinion retrieval

In this subtask participants were asked to retrieve blog posts expressing an opinion about a given entity and then to discard factual posts. The evaluation measure adopted for the MAP meant the system was to produce a ranked list of retrieved items. The opinion expressed could either be positive, negative or mixed. Our opinion retrieval runs were based on our two baselines described in Section 6.1 as well as on the five baselines provided by the organizers. To detect opinion we used two approaches: Z-Score (denoted Z in the following tables) and logistic regression (denoted LR). This resulted in a total of 14 official runs. Table 5 lists the top three results for each of our opinion detection approaches.

Compared to the baseline results shown in Table 4 (under the column "Opinion MAP"), adding our opinion detection approaches after the factual retrieval process tended to hurt the MAP performance. For example, the run UniNEBlog1 achieved a MAP of 0.320 without any opinion detection and only 0.309 when using our simple additive model (-3.4%) or 0.224 with our logistic approach (-30%).

This was probably due to the fact that during the opinion detection phase we removed all the documents judged by our system to be non-opinionated. Ignoring such documents thus produced a list clearly comprising less than 1,000 documents. Finally, Table 5 shows that having a better baseline also provides a better opinion run and that for opinion detection our simple additive model performed slightly better than the logistic regression approach (+36.47% on opinion MAP).

RunName	Baseline	Topic	Opinion
UniNEopLR1	UniNEBlog1	0.230	0.224
UniNEopLRb4	baseline 4	0.228	0.228
UniNEopLR2	UniNEBlog2	0.220	0.212
UniNEopZ1	UniNEBlog1	0.393	0.309
UniNEopZb4	baseline 4	**0.419**	**0.327**
UniNEopZ2	UniNEBlog2	0.373	0.296

Table 5. MAP of both ad hoc search and opinion detection

6.3 Polarity Task

In this third part of the Blog task, the system retrieved opinionated posts separated into a ranked list of positive and negative opinionated documents. Documents containing mixed opinions were not to be considered. The evaluation was done based on the MAP value, and separately for documents classified as positive and negative. As for the opinion retrieval task, we applied our two approaches in order to detect polarity in the baseline runs. Those documents that our system judged as belonging to either of the mixed or neutral categories were eliminated.

Table 6 lists the three best results (over 12 official runs) for each classification task. It is worth mentioning that for the positive classification task, we had 149 queries and for the negative opinionated detection only 142 queries provided at least one good response. The resulting MAP values were relatively low compared to the previous opinionated blog detection run (see Table 5).

For our official runs using logistic regression, we did not classify the documents into four categories (positive, negative, mixed and neutral) but instead into only three (positive, negative, mixed). This meant that instead of calculating four polarity scores, we calculated only three and assigned polarity to the highest one. Table 7 shows the results for the logistic regression approach, with three (without neutral) and four (with neutral) classifications.

RunName	Baseline	Positive MAP	Negative MAP
UniNEpolLRb4	baseline 4	0.102	0.055
UniNEpolLR1	UniNEBlog1	**0.103**	0.057
UniNEpolLRb5	baseline 5	0.102	0.055
UniNEpolZb5	baseline 4	0.070	**0.061**
UniNEpolZ5	baseline 5	0.067	0.058
UniNEpolZ3	baseline 3	0.067	0.063

Table 6. MAP evaluation for polarity detection

Baseline	With neutral		Without neutral	
	Positive	Negative	Positive	negative
UniNEBlog1	0.065	0.046	0.103	0.057
UniNEBlog2	0.064	0.042	0.102	0.051

Table 7. Logistic regression approach with three or four classifications

Using only three classification categories instead of four had a positive impact on performance, as can be seen from an examination of Table 7 (logistic regression method only). Most documents classified as "neutral" in the four-classification approach were then eliminated. When we considered only three categories, these documents were mainly classified as positive. This phenomenon also explains the differences in positive and negative MAP in our official runs when logistic regression was used (see Table 6).

7. CONCLUSION

During this TREC 2008 Blog evaluation campaign we evaluated various indexing and search strategies, as well as two different opinion and polarity detection approaches.

For the factual or *ad hoc* baseline retrieval we examined the underlying characteristics of this corpus with the compound indexing scheme that would hopefully improve precision measures. Compared to the standard approach in which isolated words were used as indexing units, in the MAP we obtained there was a +11.1% average increase for title only queries, as well as a +7.7% increase for title and description topic formulations. These results strengthen the assumption that for Blog queries such a precision-oriented feature could be useful. In further research, we might consider using longer tokens sequences as indexing unit, rather than just word pairs. Longer queries such as #1037 "New York Philharmonic Orchestra" or #1008 "UN Commission on Human Rights" might for example obtain better precision.

For the opinion and polarity tasks, we applied our two approaches to the given baselines as well as to two of our own baselines. We noticed that applying no opinion detection provides better results than applying any one of our detection approaches. This was partially due to the fact that during opinion detection we eliminated some documents, either because they were judged "neutral" or because they were not contained in the judged pool of documents ("unjudged").

In a further step we will try to rerank the baselines instead of simply removing documents judged as non-opinionated. A second improvement to our approach could be judging each document at the retrieval phase instead of first creating a pool of judged documents. In this case we would no longer have any documents classified as "unjudged" although more hardware resources would be required. Polarity detection basically suffers from the same problem as opinion detection. Finally, we can conclude that having a good factual baseline is the most important part of opinion and polarity detection.

ACKNOWLEDGMENTS

The authors would also like to thank the TREC Blog task organizers for their efforts in developing this specific test-collection. This research was supported in part by the Swiss NSF under Grant #200021-113273.

8. REFERENCES

[1] 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.

[2] Fautsch C, & Savoy J. IR-Specific Searches at TREC 2007: Genomics & Blog Experiments. In *Proceedings TREC-2007*, NIST publication #500-274, Gaithersburg (MD), 2008.

[3] Abdou S., & Savoy J. Searching in Medline: Stemming, Query Expansion, and Manual Indexing Evaluation. *Information Processing & Management*, 44(2), 2008, 781-789.

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

[5] 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.

[6] Efthimiadis, E.N. Query expansion. *Annual Review of Information Science and Technology*, 31, 1996, p: 121-187

[7] Rocchio, J.J.Jr. Relevance feedback in information Retrieval. In G. Salton (Ed.): *The SMART Retrieval System*. Prentice-Hall Inc., Englewood Cliffs (NJ), 1971, 313-323.

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

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

[10] Savoy, J. Combining Multiple Strategies for Effective Cross-Language Retrieval. *IR Journal*, 7(1-2), 2004, 121-148.

[11] Muller, C. *Principe et methodes de statistique lexicale*. Honoré Champion, Paris, 1992.

[12] Hosmer D., Leneshow S., *Applied Logistic Regression*. Wiley Interscience, New York, 2000.