University of Waterloo at TREC 2008 Blog track

Olga Vechtomova
Department of Management Sciences
Faculty of Engineering, University of Waterloo
Waterloo, Canada
ovechtom@engmail.uwaterloo.ca

Abstract

The paper reports the University of Waterloo participation in the opinion and polarity tasks of the Blog track. The proposed method uses a lexicon built from several linguistic resources. The opinion discriminating ability of each subjective lexical unit was estimated using the Kullback-Leibler divergence. The KLD scores of subjective words occurring within fixed-size windows around instances of query terms were used in calculating document scores. The described system also used a method of identifying phrases in topic titles by matching them to Wikipedia titles. The results show that both KLD-based scores of subjective lexical units and Wikipedia-matched phrases are useful techniques that help improve opinion retrieval performance.

1. Introduction

This year the University of Waterloo participated in the opinion finding and polarity tasks of the Blog track. Our approach relies on the use of a lexicon of subjective words and phrases, gathered from a variety of sources, such as FrameNet [1], Levin's verb classes [2], Hatzivassilouglou and McKeown's list of subjective adjectives [3]. The developed methods use the Kullback-Leibler divergence (KLD) [4] to weight subjective words, and factors these weights into the document score.

The opinion finding task of the Blog track of TREC began in 2006. Blog tracks 06-08 use the same document collection, but different sets of topics. The document collection includes 3.2 million permalink documents (88.8Gb), i.e. blog posts. Each topic consists of the standard TREC components: title, description and narrative, and describes the entity (e.g., a person, product, event or abstract entity) about which the searcher wants to find opinions. The objective of the opinion task is to retrieve a ranked list of blog posts, which express opinions about the entity described in the topic. The objective of the polarity task is to retrieve two separate ranked lists of documents with positive and negative opinions. Relevance judgements were performed on a 5-point scale: 0 – document is non-relevant; 1 – document is relevant, but contains no opinion on the target entity; 2 – document is relevant and contains negative opinion(s) on the target entity; 3 – document is relevant and contains mixed (both positive and negative) opinions on the target entity; 4 – document is relevant and contains positive opinion(s) on the target entity. Two types of relevance were defined in the task: topic relevance, where a document judged as 1-4 is considered relevant, and opinion relevance, where a document is considered relevant if judged as >1 in opinion task, and 2 (4) in negative (positive) polarity finding subtasks.

2. Methodology

Our approach to finding blogs containing opinions about the concept expressed in the query is a two-stage process. In the first stage, a set of documents is retrieved in response to the query using a topic-relevance ranking method, while in the second stage, this document set is re-ranked using an opinion-finding method. In the experiments reported in this paper, for the first stage we used BM25 [5] implemented in the Wumpus search engine [6].

2.1 Identifying phrases in queries

Our earlier work on retrieval of opinions from blogs [7] suggested that the use of phrases in the first stage, i.e. retrieval of the documents using a topic-relevance ranking method, yields better results than the use of single terms. In that approach we simply used user-defined phrases, i.e. the text enclosed in quotes by the user was treated as a phrase. Clearly, this cannot be always relied upon, since not all users explicitly delimit phrases in their queries. In the

present work we used a method of identifying phrases by matching them to Wikipedia titles, as described below, and which is similar to the method used in [8].

Any part of the query that matched a Wikipedia title was treated as a phrase. First, we attempted to match the entire query of *n* words, then, if unsuccessful, every subphrase of size *n-1*, then every subphrase of size *n-2* in the unmatched part of the query, and so on until we reached unigrams. The unmatched unigrams were always kept in the query. Stopwords were filtered out only from the single terms in the query. If a phrase that matched a Wikipedia title contained stopwords, they were not removed, such as in "March of the penguins" (Topic 851).

To illustrate the process of identifying phrases in the query, consider the following example: the query "business intelligence resources" (Topic 898) was split into "business intelligence" and "resources", because Wikipedia has an article with the title "business intelligence". Similarly, the query "opera software OR opera browser OR opera mobile OR opera mini" (Topic 944) was split into "opera software", "opera browser", "opera mobile" and "opera mini". The Boolean operator OR was removed because it is a stopword.

2.2 Building the subjective lexicon

The subjective lexicon used in the proposed methods was adapted from the resources described below. Wilson et al. used a similar set of subjective lexical units in classifying opinions by intensity [9].

2.2.1 Levin's verb classes [2]

Beth Levin has categorized English verbs into semantic classes. Verbs from the following classes were used in our method:

- Verbs of psychological state (e.g., amaze, fascinate, bother, impress);
- Verbs of desire (e.g., crave, yearn, need);
- Judgment verbs (e.g., acclaim, criticize, reproach).

2.2.2 FrameNet [1]

Lexical units (verbs, phrasal verbs, adjectives, etc.) from the following frames were used:

- Emotion_active (e.g., fret, fuss, lose sleep);
- Emotion_directed (e.g., affronted, delighted, resentful);
- Experiencer_object (e.g., enthrall, puzzle, trouble);
- Experiencer_subject (e.g., dissatisfied, jubilant, fed up).

2.2.3 Ballmer and Brennenstuhl speech act verbs [10]

A few verbs and expressions were manually selected from the Emotion Model (e.g., blow up, burst out laughing, grumble about).

2.2.4 Hatzivassilouglou and McKeown's subjective adjectives [3]

A list of 1336 subjective adjectives manually composed by Hatzivassiloglou and McKeown (e.g., amusing, impressive, unreliable) was used.

2.2.5 Subjective lexicon processing

After the removal of duplicates, the overall subjective lexicon gathered consisted of 1828 lexical units. For each verb and most of the phrasal verbs, whenever it made sense, we have also added past tense, gerund ("-ing" form) and third person forms. With all the grammatical word forms added, the total size of the subjective lexicon was 3182.

In the polarity subtask, we used the original polarity tags when they were provided by the lexicon authors. For example, Hatzivassiloglou and McKeown's adjectives have polarity tags. Some of Levin's verb classes are also divided into positive and negative. We also manually added polarity tags to some of the words in other resources. Some words have no clear polarity and can be used in positive or negative sense depending on the context, such as "feel", "overwhelm", "surprised" . We did not tag such words, and hence did not use them in the polarity subtask.

2.3 Window-based co-occurrence

Our approach to opinion-based reranking consists in adjusting the *tf* weights of query terms on the basis of their co-occurrence with subjective lexical units in fixed-size windows centered around the query term occurrences. The motivation for this is that if a subjective word or phrase occurs close to a query term, it may indicate that the author expresses an opinion about the topic related to the query term. The reason, why we chose to use a fixed-size window

instead of a natural language unit, such as a sentence, is two-fold: first, a subjective word may not actually "target" the query term occurrence in a sentence, but another word in a different sentence. For instance, a subjective adjective may modify a pronoun in a different sentence, e.g., "He saw an oil painting near the window. It was beautiful." Alternatively, it may modify a noun related to the query term, for instance, when expressing an opinion about a photo camera, a person may talk about the picture quality, rather than the camera directly: "I bought a new camera. The picture quality is excellent." The second reason is practical: it is faster and less error-prone to identify windows than split the text into sentences. In the case of blogs, sentence boundary detection is a more difficult task than with, say, a newswire article: the former may use non-standard and ill-formed syntactic constructions without proper punctuation marks, and are typically in HTML format, which may not be always possible to convert to plain text correctly.

The window is defined as n words to the left and right of the query term occurrence in text. In cases where the distance between two instances of query term(s) in a document is less than n, the text span between these two query term instances is split in the middle, such that one half is attributed to the query term on the left, and the other half – to the query term on the right. This is done in order to avoid counting the same subjective word occurrence twice. In our experiments window size was set to 30, as this proved optimal on the training datasets.

2.4 Term weighting

2.4.1 Calculating KLD scores of subjective lexical units

Intuitively, subjective words that occur relatively more frequently in the known relevant and opinionated documents than in the non-relevant or relevant but non-opinionated ones are more useful in predicting opinions. Based on this intuition, we calculated scores for the units in our subjective lexicon using the Kullback-Leibler divergence (KLD).

The Kullback-Leibler divergence measures the relative entropy between two probability distributions. It has been defined in information theory [4] and was used in many information retrieval and natural language processing tasks, for example, in query expansion following pseudo-relevance feedback [11].

The KLD score of a subjective lexical unit was calculated according to Eq. 1.

$$KLD(t) = P_R(t) \cdot \log \frac{P_R(t)}{P_N(t)} \qquad (1)$$

Where: $P_R(t)$ – probability of the subjective lexical unit t occurring in the relevant documents, and calculated as $f_R(t)/R$, where $f_R(t)$ – frequency of occurrence of t in the relevant set, R – number of terms in the relevant set; $P_N(t)$ – probability of the subjective lexical unit t occurring in the non-relevant documents, and calculated as $f_N(t)/N$, where $f_N(t)$ – frequency of occurrence of t in the non-relevant set, N – number of terms in the non-relevant set.

The BLOG track 2006 and 2007 data was used for calculating KLD scores of the subjective lexicon. In the opinion finding task, the relevant set consisted of all the documents for the 100 topics with the relevance judgments of 2 (negative opinion), 3 (mixed opinion) and 4 (positive opinion). The non-relevant set consisted of all the documents with the judgments of 0 (non-relevant) and 1 (relevant, but not opinionated). In the polarity task, KLD scores were calculated separately for positive and negative subjective words. When calculating KLD scores for positive (or negative) words, the relevant set consisted of all documents with relevance judgment level 4 (or 2 for negative), while the nonrelevant consisted of documents with all other relevance judgment levels.

A KLD score was calculated for each lexical unit, not each of its grammatical word forms separately. Thus, in calculating KLD for "fascinate", the frequencies of occurrences of "fascinate", "fascinated", "fascinating" and "fascinates" were added. In total, KLD scores for 1828 lexical units were calculated. Lexical units with negative KLD scores were discarded.

2.4.2 Calculating document matching score

Our approach to the weighting of query term occurrences in the document consists of modifying the term frequency (*tf*) calculation in BM25. Instead of counting the actual frequency of a term's occurrence in the document to get *tf*, we calculate a pseudo-frequency (*pf*) value (Eqs. 2 and 3). If the query term t_i co-occurs with a subjective word within a window of n words either side of it, then we add the normalized KLD score of the subjective word to $c(t_i)$ (Eq. 2).

The idea of using pseudo-frequency weights was also used in a proximity-based document ranking method proposed in [12], which proved to be effective in an ad-hoc IR task.

$$c(t_i) = \begin{cases} 1 + \sum_{j=1}^{|J|} \frac{KLD(s_j)}{\max KLD} & \text{if } |J| > 0 \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Where: $c(t_i)$ – contribution of the i^{th} instance of the query term t to pf, $KLD(s_j)$ – KLD score of the subjective lexical unit s_j, $|J|$ – the number of subjective lexical units occurring in the window of 30 words around t_i, $maxKLD$ – the maximum KLD score for all 1828 subjective lexical units.

$$pf_t = \sum_{i=1}^{N} c(t_i) \quad (3)$$

Where: N – number of instances of the query term t in the document.

After pf is calculated for a query term, its Term Weight (TW) in the document is calculated in the same way as in the BM25 formula [5], with pf used instead of tf (Eq. 4):

$$TW_t = \frac{(k_1 + 1) \times pf_t}{k_1 \times NF + pf_t} \times idf_t \quad (4)$$

Where: k_1 is the term frequency normalisation factor, which moderates the contribution of the weight of frequent terms. If k_1=0, pf has no effect on the term weight, while the higher the value of k_1 the more effect pf has on the term weight. NF is the document length normalisation factor, and is calculated in the same way as in the BM25 document ranking function, as expressed in Eq. 5.

$$NF = (1 - b) + b \times \frac{DL}{AVDL} \quad (5)$$

Where: b is a tuning constant, DL is the document length in word counts; $AVDL$ is the average document length in the document collection.

The Document Matching Score is calculated as the sum of weights of all query terms found in the document (Eq. 6).

$$MS = \sum_{t=1}^{|Q|} TW_t \quad (6)$$

Where: $|Q|$ is the number of terms in the query.

In the runs that used Wikipedia-based phrases (see Section 3.1) in Stage 2 (opinion based reranking), we used both phrases and single terms in calculating a document matching score, i.e. we counted the occurrences of the whole phrases in the document, plus any of their component terms that occur separately. For instance, if a document contains instances of the query phrase "March of the Penguins" and instances of "penguin", the document matching score will be TW("March of the Penguins")+TW("penguin").

3. Results

3.1 Opinion finding task

For the baseline runs, we used BM25 implemented in Wumpus, which, as described earlier, was also used for Stage 1 (initial document retrieval) of our opinion retrieval algorithm. We have evaluated different values for b and k_1 parameters of BM25 on Blog 06 dataset. The values of 0.1 and 0.75 for b and k_1 respectively yielded good

performance and were therefore used in all the baseline and experimental runs reported in the paper. Two baseline runs were performed (UWbase1 and UWbase2), both of which used the title section of the topics. UWbase1 used single terms, while UWbase2 – phrases identified by matching topic titles to Wikipedia titles as outlined in 2.1 above. Two opinion runs (UWopinion1 and UWopinion2) were conducted by re-ranking UWbase1 and UWbase2 respectively using the method described in Section 2.

Blog track participants were required to submit their runs on 150 topics: 100 topics from Blog-06 and 07, and 50 new topics that were developed this year. We report the results for all 150 topics in Table 1, as well as for the new 50 topics separately in Table 2. Our performance analysis, however, is focused on the new topics. To facilitate cross-site comparison, following the baseline retrieval stage the track organisers released 5 baselines (runs NISTbaseline1-5 in Tables 1 and 2), randomly selected from the submitted baseline runs. Participants were encouraged to submit opinion runs based on as many of these baselines as possible. We submitted opinion runs based on all of these baselines (runs UWnb1Op through UWnb5Op in Tables 1 and 2). To facilitate reading of Tables 1 and 2, all baseline runs (i.e. without opinion features) are shaded in grey and each opinion run is placed immediately below its corresponding baseline.

Table 1. Results based on 150 topics

Run	Opinion relevance			Topic relevance		
	MAP	P10	Rprec	MAP	P10	Rprec
UWbase1	0.2314	0.4740	0.2859	0.2997	0.6060	0.3524
UWopinion1	0.2508	0.5707	0.2958	0.2923	0.6347	0.3416
UWbase2	0.2476	0.4647	0.3095	0.3313	0.6327	0.3890
Uwopinion2	0.2956	0.5813	0.3493	0.3589	0.7027	0.4136
NISTbaseline1	0.2639	0.4753	0.3189	0.3701	0.7307	0.4156
UWnb1Op	0.3148	0.6107	0.3613	0.3812	0.7407	0.4264
NISTbaseline2	0.2657	0.5287	0.3189	0.3382	0.7000	0.3831
UWnb2Op	0.2940	0.5933	0.3468	0.3361	0.7167	0.3835
NISTbaseline3	0.3201	0.5387	0.3647	0.4244	0.7220	0.4573
UWnb3Op	0.3202	0.5960	0.3613	0.3768	0.7173	0.4181
NISTbaseline4	0.3543	0.5580	0.3979	0.4776	0.7867	0.5092
UWnb4Op	0.3403	0.6000	0.3807	0.4057	0.7280	0.4484
NISTbaseline5	0.3147	0.5307	0.3709	0.4424	0.7793	0.4868
UWnb5Op	0.3298	0.6133	0.3772	0.3978	0.7427	0.4504

As can be seen from Table 2, the use of opinion features in UWopinion2 was useful and led to improved average performance on 50 new topics over the corresponding baseline (UWbase2) by 5% in MAP and 13% in P10 (opinion relevance). Average improvements on 150 topics (Table 1) were more substantial: 19.4% in MAP, 25% in P10 and 6.3 in R-Prec (all statistically significant, t-test, $p<.02$). Analysis of differences in average precision values by topic between UWopinion2 and UWbase2 (Figure 1) shows that the majority of topics benefited from the use of KLD-weighted subjective words in document re-ranking. A notable outlier, topic 1013 (European Union, Iceland), dropped from 0.8615 to 0.1018, causing substantial average drop in performance. A likely explanation for such poor performance is different use of phrases: in the baseline run, Wikipedia-matched title phrases were searched as fixed phrases, i.e. no partial matches were allowed. In the opinion re-ranking stage, partial matches were allowed, i.e. "European Union" would match "European", "Union" and "European Union". Thus, documents, containing many instances of "European" co-occurred with subjective vocabulary, were promoted in the ranked list, hurting performance.

Table 2. Results based on 50 new topics

Run	Opinion relevance			Topic relevance		
	MAP	P10	Rprec	MAP	P10	Rprec
UWbase1	0.2485	0.5160	0.3014	0.2897	0.5840	0.3359
UWopinion1	0.2398	0.5100	0.2820	0.2571	0.5440	0.2934
UWbase2	0.2753	0.5160	0.3391	0.3309	0.6380	0.3824
Uwopinion2	0.2892	0.5840	0.3361	0.3335	0.6580	0.3789
NISTbaseline1	0.3239	0.5800	0.3682	0.4031	0.7320	0.4345
UWnb1Op	0.3365	0.6160	0.3706	0.3841	0.6980	0.4229
NISTbaseline2	0.2640	0.5500	0.3145	0.3107	0.6480	0.3493
UWnb2Op	0.2838	0.5840	0.3247	0.3057	0.6460	0.3387
NISTbaseline3	0.3565	0.5540	0.3887	0.4344	0.6440	0.4608
UWnb3Op	0.3298	0.6060	0.3563	0.3613	0.6680	0.3884
NISTbaseline4	0.3822	0.6160	0.4284	0.4724	0.7440	0.4993
UWnb4Op	0.3381	0.6060	0.3718	0.3793	0.6700	0.4131
NISTbaseline5	0.2988	0.5300	0.3524	0.3745	0.7040	0.4170
UWnb5Op	0.3196	0.6220	0.3623	0.3549	0.6920	0.4033

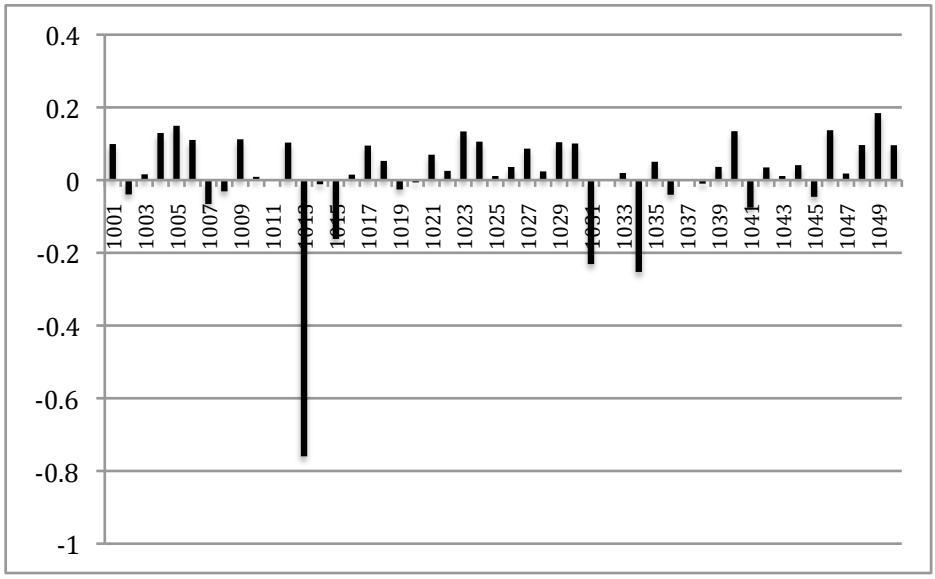

Figure 1. Difference in average precision of UWopinion2 from the baseline UWbase2.

To determine the effect of KLD-based weighting on performance, we conducted a run without KLD (UWopinion2-noKLD in Tables 3 and 4). In this run, instead of using pseudo-frequency *pf* (Eqs. 2 and 3), we used term frequency *tf*, whereby each instance of the query term *t* occurring within the window of +/- 30 words of a subjective lexical unit, contributed 1 towards the *tf* of this term. As in UWopinion2, those query term instances that did not co-occur with a subjective word in a window, were not counted towards *tf*.

Table 3. The effect of KLD on performance (150 topics)

Run	Opinion relevance			Topic relevance		
	MAP	P10	Rprec	MAP	P10	Rprec
UWbase2	0.2476	0.4647	0.3095	0.3313	0.6327	0.3890
UWopinion2	0.2956	0.5813	0.3493	0.3589	0.7027	0.4136
UWopinion2-noKLD	0.2733	0.5560	0.3298	0.3476	0.6907	0.3979

Table 4. The effect of KLD on performance (50 new topics)

Run	Opinion relevance			Topic relevance		
	MAP	P10	Rprec	MAP	P10	Rprec
UWbase2	0.2753	0.5160	0.3391	0.3309	0.6380	0.3824
UWopinion2	0.2892	0.5840	0.3361	0.3335	0.6580	0.3789
UWopinion2-noKLD	0.2732	0.5680	0.3212	0.3233	0.6620	0.3638

As can be seen from Tables 3 and 4, there is a substantial improvement from using KLD in run UWopinion2 over UWopinion2-noKLD. Based on opinion relevance judgements on the new 50 topics, there is 5.9% improvement in MAP (statistically significant, t-test, $p<.01$), 2.8% improvement in P10 (not significant) and 4.6% improvement in R-prec (significant, t-test, $p<.05$).

3.2 Polarity task

For the polarity task we used the same method as in the opinion task, with the weights for positive/negative lexical units calculated respectively on positive/negative opinion training datasets. UWpolarity1 is based on UWbase1 run, while UWpolarity2 – on UWbase2. The results for 150 topics are presented in Tables 5 and 6, while for 50 new topics – in Tables 7 and 8.

Table 5. Negative polarity results (150 topics)

Run	MAP	P10	Rprec
UWpolarity1	0.0686	0.1239	0.0935
UWpolarity2	0.0925	0.1542	0.1257

Table 6. Positive polarity results (150 topics)

Run	MAP	P10	Rprec
UWpolarity1	0.1033	0.1651	0.1384
UWpolarity2	0.1280	0.1933	0.1701

Table 7. Negative polarity results (50 new topics)

Run	MAP	P10	Rprec
UWpolarity1	0.0669	0.1104	0.0896
UWpolarity2	0.0968	0.1396	0.1200

Table 8. Positive polarity results (50 new topics)

Run	MAP	P10	Rprec
UWpolarity1	0.0893	0.1408	0.1284
UWpolarity2	0.1239	0.2041	0.1662

4. Conclusions

In this paper we presented a new method of opinion retrieval from blogs. We used a subjective lexicon gathered from a number of linguistic resources, such as FrameNet, Levin's verb classes, Ballmer and Brennestuhl speech act verbs, etc. The Kullback-Leibler divergence measure was used to weight subjective words. We also experimented with different types of queries for the first stage (document retrieval) and the second stage (opinion-based reranking). Specifically, we identified phrases in the TREC topic titles by matching them to Wikipedia titles.

The method that achieved the best overall performance (UWopinion2) used Wikipedia-based phrases in both stages and the KLD-based document reranking method. The improvement over the baseline (UWbase2) on the 50 new topics is 5%, whereas the improvement on all 150 topics is 19.4% (statistically significant, t-test, $p<.01$). Further analysis demonstrates that KLD-based weighting of subjective words is a useful factor in the ranking of opinionated documents.

References

[1] Baker, C. F., Fillmore, C. J. and Lowe, J. B. The Berkeley FrameNet project. In Proc. of the COLING-ACL, Montreal, Canada, 1998.

[2] Levin, B. English Verb Classes and Alternations. The University of Chicago Press, Chicago, 1993.

[3] Hatzivassiloglou, V. and McKeown, K. R. Predicting the semantic orientation of adjectives. In Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics, 1997, pp. 174-181.

[4] Losee, R. M. The science of information: Measurements and applications. Academic Press Prof., Inc., San Diego, CA, 1990.

[5] Spärck Jones, K., Walker, S. and Robertson, S.E. A probabilistic model of information retrieval: development and comparative experiments. Parts 1 and 2. Information Processing and Management, 2000, 36(6), 779-808, 809-840.

[6] Büttcher, S. and Clarke, C.L.A. Indexing Time vs. Query Time Trade-offs in Dynamic Information Retrieval Systems. In Proceedings of the 14th ACM Conference on Information and Knowledge Management (CIKM), Bremen, Germany, November 2005.

[7] Vechtomova O. Using Subjective Adjectives in Opinion Retrieval from Blogs. In Proceedings of the 16th Text Retrieval Conference, November 6-9, 2007, Gaithersburg, MD.

[8] MacKinnon, I. and Vechtomova, O. Improving Complex Interactive Question Answering with Wikipedia Anchor Text. In Proceedings of the 30th European Conference on Information Retrieval (ECIR), March 30 - April 3, 2008, Glasgow, Scotland.

[9] Wilson, T., Wiebe, J. and Hwa, R. Recognizing Strong and Weak Opinion Clauses. Computational Intelligence, 2006, 22(2), 73-99.

[10] Ballmer, Th. and Brennenstuhl, W. Speech Act Classification. Springer Series in Language and Communication, Vol. 8, Springer-Verlag Berlin Heidelberg, 1981.

[11] Carpineto, C., De Mori, R., Romano, G. and Bigi, B. An Information-Theoretic Approach to Automatic Query Expansion. ACM Transactions on Information Systems, 2001, 19(1), 1–27.

[12] Vechtomova O. and Karamuftuoglu M. Lexical Cohesion and Term Proximity in Document Ranking. Information Processing and Management, 2008, 44(4), 1485-1502.