Opinion Finding in Blogs: A Passage-Based Language Modeling Approach

Malik Muhammad Saad Missen
Université de Toulouse
IRIT UMR 5505 CNRS
Toulouse, France
+ 33 5 61 55 72 65
Malik.Missen@irit.fr

Mohand Boughanem
Université de Toulouse
IRIT UMR 5505 CNRS
Toulouse, France
+ 33 5 61 55 74 16
Mohand.Boughanem@irit.fr

Guillaume Cabanac
Université de Toulouse
IRIT UMR 5505 CNRS
Toulouse, France
+ 33 5 61 55 72 73
Guillaume.Cabanac@irit.fr

ABSTRACT
In this work, we propose a Passage-Based Language Modeling (LM) approach for Opinion Finding in Blogs. Our decision to use Language Modeling in this work is totally based on the importance of passages in blogposts and performance LM has given in various Opinion Detection approaches. In addition to this, we propose a novel method for bi-dimensional Query Expansion with relevant and opinionated terms using Wikipedia and Relevance-Feedback mechanism respectively. Besides all this, we also compare the performance of three Passage-based document ranking functions (*Linear*, *Avg*, *Max*). For evaluation purposes, we use the data collection of TREC Blog06 with 50 topics of TREC 2006 over TREC provided best baseline with opinion finding MAP of 0.3022. Our approach gives a MAP improvement of almost 9.29% over best TREC provided baseline (baseline4).

Categories and Subject Descriptors
H.3.3 **[Information Storage and Retrieval]:** Information Search and Retrieval - *Query formulation, Relevance feedback, Search process.*

General Terms
Experimentation

Keywords
Opinion Detection, Blogs, Language Modeling, Semantic Relatedness, Passages

1. INTRODUCTION
Opinion Detection in text is of one of the most exciting and challenging task of text analysis today. This task can be performed on different levels of granularity, i.e. on word level, sentence level or on document level. As a conclusion of this task a given word, sentence or document can be declared as of opinionated nature (or subjective) or of factual nature (objective). Text with opinionated nature can further be analyzed for having negative or positive polarity of opinion and this subtask is called *Opinion Polarity Detection*.

The task of Opinion Detection becomes more difficult and challenging in Blogs. Generally in blogs, bloggers (the authors of blog) express their opinions and thoughts about something being discussed in a blog. Therefore, blogs can be considered as one of the best sources of opinions on the Web. Considering the importance of blogs regarding the task of Opinion Detection, TREC introduced a Blog track in 2006 known as TREC Blog Track with the release of blog data collection [1] including 150 different topics (over three years 2006, 2007 and 2008) and their judgments (*qrels*). Blog track has 4 tasks i.e. 1) Baseline adhoc (blog post) retrieval task 2) Opinion finding (blog post) retrieval task 3) Polarised opinion finding (blog post) retrieval task 4) Blog finding distillation task.

RIAO, April 28–30, 2010, Paris, France.

2. RELATED WORK AND MOTIVATION
Two major approaches have been used by TREC Blog participants for Opinion Detection i.e. Lexicon based approaches [2] and Machine Learning based approaches [3, 4]. In *Lexicon-based approaches*, researchers make use of polarity scores (or orientations) of the words. A final opinion score is calculated on behalf of polarities of the words (within context or without context). For example, *beautiful* has a positive prior polarity, and *horrid* has a negative prior polarity. There are various lexical resources [5, 6] available for this task. For *Machine learning approaches*, usually a classifier is trained using a set of annotated texts containing sentiment, typically employing features such as n-grams of words, part-of-speech tags, and logical forms. The details about the use of these and some other approaches can be consulted in overview papers of TREC 2006 [7], TREC 2007 [8] and TREC 2008[9].

The basic idea behind the opinion finding task is that if a document contains opinions about the topic, the document should be delivered to the user as an opinionated document i.e. two major aspects to be dealt are *Relevance* and *Opinion*. Looking at the Relevance aspect, it has been observed that a blog document does not talk about the target present in the query only. Several issues can be found being discussed in a blog document. For example, a blogpost on the topic of *US Elections* may also be discussing the *Universal Health Policies*, *Climate Change* and *Energy Crisis*. Now if an approach is considering this document as a single monolithic document then the relevance results for this document affect the overall ranking of the document. On the other hand, if we treat the document as a composition of small portions (like passages etc.), it might give us a more accurate and better ranking of the documents. Concluding from this, blog documents should better be processed on sentence or passage (or paragraph) level for the task of opinion finding. There have been many attempts already for extracting opinion or sentiments on sentence level [10, 11]. However we hypothesize that when considering the

job of opinion mining in blogs, it becomes more practical to process blog documents on paragraph level because sentence splitting is a very challenging task especially when we are dealing with blogs [12]. Lack of punctuations, capitalization and grammar mistakes make this task more difficult. Even if well-split, we may lose the context of the sentence if dealt on sentence-level. In addition, blog documents are more logically structured as paragraphs. A blogpost is split into many paragraphs and normally a comment is contained within the boundaries of a paragraph. Therefore, it becomes more logical and appropriate for us to process blog documents on paragraph level. Even it is more feasible way to work on passage level for in this context but we cannot find lot of work in the past. However, we cannot find any work that especially uses passages for the task of opinion mining. [13] and [14] are the only notable work that can be mentioned in this regard. However, [13] uses passages for topic retrieval and sentences for the task of opinion mining and [14] uses passages just for building query-specific lexicon. Lot of work already exists to justify the use of passage-based LM for our work. Passage identification and utilization in information retrieval has been the focus of research for quite some time [15, 16]. In passage-based ad hoc retrieval, either we can return the relevant passages as a result [17], or simply mark the entire document as relevant if it contains (some) relevant passage(s). In rest of the article, we describe our approach in detail, discuss results and conclude the paper with future work remarks.

3. OUR APPROACH

Our approach can also be realized in three major stages. In 1st Data Pre-processing phase, data collection is cleaned from unnecessary noisy data like removal of unnecessary HTML tags (like script, style, etc). Links are also removed in our case. For Topic-Relevance, we are using TREC provided strongest baseline and Opinion Finding approach is explained below. This baseline has a Opinion Finding MAP of 0.3022 over topics of TREC 2006 i.e. from topic 851 to 900.

3.1 Opinion-Finding

Our Opinion Detection approach consists of the following three sub-stages:

3.1.1 Query Expansion

We propose a *Two Phase Query Expansion* method in which the title of the query is used as a base to populate the query with two types of terms i.e. *relevant terms.* Examples of relevant terms for topic-851with title *March of the Penguins* are (*Luc Jacquet, Academy award, Antarctica* etc) and opinion terms are (*Hilarious, Enjoy, Emotional* etc).

3.1.1.1 Query Expansion with Relevant Terms

We use Wikipedia[1] to enrich the query with relevant terms. We extract the hyperlinked proper nouns and named entities from the Wikipage for the title of the query. At the end of this phase, we have a set of relevant terms L_{REL} of closely related terms including the title of the query.

3.1.1.2 Query Expansion with Opinion Terms

3.1.1.2.1 Relevant Passage Selection

In this step, we use L_{REL} to select relevant passages from the top *1000* relevant documents already identified using tags <*P*> and <*DIV*>. We have two choices for selection of a relevant passage i.e. either "to select all passages having only title of the query in it" or "to select all passages containing any one of the term from the set L_{REL}". We decide to go with second option so that we do not miss many relevant passages.

3.1.1.2.2 Extracting Opinion Terms

In the second stage, we extract a set of opinion words L_{OPIN} from top *10 relevant* documents marked as opinionative in *qrels*. We consider it a special case of *External Feedback* in which a user is asked to identify and mark the top *10* opinionated (we call it *Relevance*[2] *Feedback)* and top *10* non-opinionated documents (we call it *Irrelevance Feedback*). Then we remove the stop-words from these documents to make things easy and create a window of 4 words (i.e. *2* on left and *2* on right including only verbs, adjectives and adverbs) is created around each occurrence of a relevant term from L_{REL}. These *4* words are chosen to be part of our expanded query. This process is repeated with each opinionated document identified in *Relevance Feedback*. Finally, we have a set of opinion terms L_{OPIN}. Duplicates are removed from L_{OPIN} and *Subjectivity* score of each term is calculated using lexical resource *SentiWordNet* [6]**.** All terms from L_{OPIN} with subjectivity score of zero (or if they do not exist in *SWN*) are removed from the list.

$$Subj(t) = \frac{Neg(t,SWN)+Pos(t,SWN)}{|t_{sense}|} \quad \textbf{(1)}$$

In equation 1, $Neg(t, SWN)$ is the negative score of the term *t* (for all its senses), $Pos(t, SWN)$ is the positive score of term *t* (for all its senses) $|t_{sense}|$ is the total number of senses for term *t*.

3.1.2 Term Weighting

We weight opinion terms and relevant terms in two different ways. For opinion terms, we combine the subjectivity score of the terms with their frequency in the collection of 10 opinionated documents. For relevant terms, only frequency evidence is used for weighting. The parameters used for frequency calculation are *collection frequency (cf), passage frequency (pf)* and *document frequency (df) (*for all those terms with *cf* > 1 in the relevant passage collection). For opinion terms, the opinion score of each term is calculated in two different ways: First, it (labelled *ALL* in results) is calculated using equation 2 shown below; second, it (labelled *FREQ* in results) is calculated as shown in equation 3**.** If analyzed carefully then it can be noted that equation 2 gives equal importance to evidences of frequencies and subjectivity but equation 3 gives more value to subjectivity. To make the selection of more appropriate terms (that would really help to differentiate opinionated documents from non-opinionated documents) possible, we propose to use the irrelevant documents already marked (that is why we call it *Irrelevance Feedback*) during *External Feedback* process. All terms present in L_{OPIN} are checked for their existence in irrelevant documents and concerning frequencies (*cf, pf* and *df*) are calculated. Opinion scores of the terms for irrelevant documents are calculated similarly using equation 2 and equation 3 and final score of a term *t* present in L_{OPIN} is calculated using equation 4. It's obvious from equation *4*

[1] if there is no Wikipage for the given query then Google search engine is used to extract the relevant terms

[2] It is to be noted that here the term relevant *document* stands for an *opinionated document*.

that a term which is not present in irrelevant document(s) but its present in relevant document(s) will be assigned a higher final score. So as a result of using equation 4, we are giving higher scores to terms which are uniquely present in opinionated documents and not in both i.e. opinionated and non-opinionated.

$$Opin(t) = (cf * pf * df) * Subj(t) \quad \textbf{(2)}$$

$$Opin(t) = (cf * pf * df) \ if \ \ Subj(t) >= 0.5$$
$$Opin(t) = Term \text{ is dropped } if \ \ Subj(t) < 0.5 \quad \textbf{(3)}$$

$$Score_{\text{Opin}}(t) = Opin(t)_{\text{Rel + Opin}} - Opin(t)_{\text{NonRel}} \quad \textbf{(4)}$$

All terms are ranked using their final opinion score $Score_{opin}(t)$ and then top *30* opinion terms are selected for each topic in addition to their WordNet synonyms and expansions (like 'gooood' for term 'good') & contractions (like 'gud'). A term is removed from the set L_{OPIN} if answer of equation.7 results in a negative value. Same process is repeated for relevant terms of the set L_{REL} i.e. relevance score of each term is calculated for both opinionated and non-opinionated documents using *Relevance Feedback* and *Irrelevance Feedback* as we did for opinion terms. The formula used for assigning weights to relevant terms is given below in equation 5. Eventually final scores of relevance terms are calculated using equation 6.

$$\text{Rel}(t) = (cf * pf * df) \quad \textbf{(5)}$$

$$Score_{\text{Rel}}(t) = \text{Rel}(t)_{\text{Rel + Opin}} - \text{Rel}(t)_{\text{NonRel}} \quad \textbf{(6)}$$

All relevant terms are ranked according to their final relevance score $Score_{Rel}(t)$ A term is given a final relevance score of zero if answer of equation 6 results in a negative value. At the end, we merge both lists of query terms i.e. L_{REL} and L_{OPIN}.

3.1.3 Passage-Based Language Model

Using *Query-likelihood Model* [18], we estimate the probability of a query being generated by a probabilistic distribution over a fixed vocabulary induced by a document. For a query *q* and a document *d* this generation probability is often denoted *p(q|d).* The posterior probability *p(d|q)* [21] is used in order to rank documents, which can be written using Bayes' rule as

$$p(d \mid q) = \frac{p(q \mid d)\, p(d)}{p(q)} \quad \textbf{(7)}$$

Since *p(q)* is not dependent on the document and in lack of prior information *p(d)* is assumed to be uniformly distributed, the ranking task reduces to estimating *p(q|d).* For estimating the probability *p(q|d),* we use *Unigram Language Model,* which were shown to be quite effective [19, 20]. In our work, we use three passage-based documents scoring functions [15, 16] that are realized using a *Unigram Language Model* shown as below:

$$Score_{Avg}(d) = \frac{1}{|P|} \sum_{i=1}^{|P|} p(q|gi) \quad \textbf{(8)}$$

$$Score_{Max}(d) = \max_{gi \in d} p(q \mid gi) \quad \textbf{(9)}$$

$$Score_{Lin}(d) = \sum_{i=1}^{|P|} p(q|gi) \quad \textbf{(10)}$$

Where $Score_{Avg}(d)$ is the average of scores of all passages within a document *d* for a given query *q*, $Score_{Max}(d)$ is the score given to document *d* for a query *q* on behalf of one of its passages having maximum score, and $Score_{Lin}(d)$ is a linear addition of scores of all passages; $|P|$ is the total number of passages within the document *d*, g_i is the *ith* passage and $p(q|g)$ is the probability of generating query *q* from passage *g* which can also be written as shown below:

$$p(q|g) = p(t_1|g) * p(t_2|g) * ... * p(t_N|g) = \prod_{i=1}^{N} p(t_i|g) \quad \textbf{(11)}$$

Using above equation can lead to a sparse matrix. To avoid this situation, we need to use a kind of smoothing model for better results. Therefore, we use *Mixture of Language Models (MIX)* assuming that each word in the query *q* (actually expanded from passages) is generated from a mixture of three language models: the *Collection Model*, the *Document Model* and the *Passage Model* itself.

$$p(t_i|MIX) = \sum_{ti \in q} \lambda_1 p(t_i|g) + \lambda_2 p(t_i|d) + \lambda_3 p(t_i|C) \quad \textbf{(12)}$$

Where $\lambda_1 + \lambda_2 + \lambda_3 = 1$ and $p(t_i/g)$ is the probability of generating query term t_i from passage *g*, $p(t_i/d)$ is the probability of generating query term t_i from document *d* and $p(t_i/c)$ is the probability of generating query term t_i from the whole collection *C.* All three are given below in equations 13, 14 and 15.

$$p(t_i|g) = C(t_{ig}) * Score(t_i) / |T_g| \quad \textbf{(13)}$$

$$p(t_i|d) = C(t_{id}) * Score(t_i) / |T_d| \quad \textbf{(14)}$$

$$p(t_i|c) = C(t_{ic}) * Score(t_i) / |T_c| \quad \textbf{(15)}$$

Where $C(t_i|g)$, $C(t_i|d)$ and $C(t_i|c)$ are the counts of term t_i in passage g, document *d* and collection *c* respectively. $|T_g|$, $|T_d|$ and $|T_c|$ are the total number of terms in a passage *g*, document *d* and collection *c* respectively. $Score(t_i)$ is the score of each term t_i. At the end of opinion finding phase, each document is given an opinion score which is later on added with the document relevance score (given in baseline) to give us a final score for the document. Finally the documents are re-ranked according to this final score DOC_{Final}.

4. RESULTS AND CONCLUSIONS

For experimentation, we use the TREC Blog data collection (*148 GB in size*) with query topics 851-900 of TREC Blog 2006. Experiments were performed using different values of λ_1, λ_2 and λ_3 but best results obtained with lambda values of $\lambda_1 = 0.5$, $\lambda_2 = 0.3$ and $\lambda_3 = 0.2$ are given below in table 2. The metrics used for evaluation are MAP (*Mean Average Precision*) and P@10 (*Precision at top 10 Documents*). Table 1 shows the results for three different document scoring functions (see equations 8, 9, 10) using two different query term weighting functions *ALL* (equation 2) and *FREQ*(equation 3). The results show an improvement of almost **9.29% (0.3303)** in MAP over baseline results which is the best ever reported MAP over TREC Blog 2006 topics to the best of our knowledge. The previous best reported MAP over topics of

TREC Blog 2006 is 0.3221 [21]. If we look at the *MAP* results then it's very clear that the results for ranking functions *Avg* and *Max* are far beyond the results of Linear ranking function.

Table 1. MAP and P@10

Function	Metric	ALL	FREQ
Avg	**MAP**	**0.3303**	0.2735
	P@10	0.6340	0.4980
Max	**MAP**	**0.3290**	0.2636
	P@10	**0.6340**	**0.5280**
Linear	**MAP**	0.2342	**0.2418**
	P@10	0.5160	0.5400

It should be noted here that both functions i.e. *Avg* and *Max* are basically representing the score of one passage of a document while *Linear* ranking function is basically representing all the passages of a document which, in a way, proves our point that its not the whole document which can improve the opinion retrieval but it may be only one passage that might be talking about the subject of the query. Even if we do not consider that *Avg* is being calculated over scores of all passages, the difference between results of *Avg* and *Max* is very marginal for both FREQ and ALL. Now if we compare the results regarding ALL and FREQ then its evident that ALL outperformed the FREQ. The reason is quite obvious that not a large number of terms have the subjectivity score of over 0.5 and even if they have then it comes to their frequencies i.e. *collection frequency*, *document frequency* and *passage frequency*. There may be some terms that have are more subjective in their nature but those are less frequently used. While in case of *ALL*, a better balanced formula is used this combines both i.e. frequencies and subjectivity together. As far as *P@10* results are concerned, unfortunately [21] does not report the P@10 results so we cannot compare *P@10* results. TREC Blog 2008 overview paper reports some results using the same baseline that we are using but those results are reported on topics of TREC 2008. But still our *P@10* results (0.6340 using title field only) are comparable to the best P@10 reported for topics of TREC 2008 (0.6400 using title field only) especially when [9] states that TREC 2006 topics are the most difficult topics among topics of year 2006, 2007 and 2008 and topics of TREC 2008 are the easiest. However, the *P@10* results in table 1 follow the same pattern as *MAP*.

Our approach has performed well but still has few drawbacks. For example, it involves manual work that should be avoided (like in entity extraction from Wikipedia and manual extensions & contractions of terms). Similarly we need to improve our passage selection criteria and query expansion method. Also experimentation is needed on more data collections to make our approach more reliable. As for future work, we plan to use document homogeneity factors and check their impact on the results. One of the major issues that concerns opinionated document retrieval is its consistency in performance on baselines of different strengths. We consider this issue a big problem and we are working to propose such approach that adapts itself according to the strength of the baseline and adjusts accordingly. An approach with a good balance of weights between opinion and topic relevance is our need.

5. REFERENCES

[1] C. Macdonald and I. Ounis. The TREC Blogs06 collection: Creating and analyzing a blog test collection. Technical Report TR-2006-224, University of Glasgow, 2006.

[2] B. Ernsting, W. Weerkamp, and M. de Rijke, "The University of Amsterdam at the TREC 2007 Blog Track", TREC 2007 Blog Track

[3] G. Zhou, H. Joshi and C.Bayrak, "Topic Categorization for Relevancy and Opinion Detection", TREC 2007 Blog Track

[4] S. Gerani, M. Carman, and F. Crestani, "Investigating Learning Approaches for Blog Post Opinion Retrieval", ECIR 2009, Springer, Toulouse France, 1-3 April 2009

[5] T. Wilson, J. Wiebe, and Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis", HLT/EMNLP 2005: In Proceedings of the ACL HLT and EMNLP, Canada, October 06 - 08, 2005, pp. 347-354

[6] A. Esuli, and F. Sebastiani, "SentiWordNet: A publicly available lexical resourcefor opinion mining", LREC-06, Genova, 2006

[7] I. Ounis, M. Rijke, C. Macdonald, G. Mishne, and I. Soboroff, "Overview of the TREC-2006 Blog Track", TREC 2006 Blog Track

[8] C. Macdonald, I. Ounis, and I. Soboroff, "Overview of the TREC-2007 Blog Track"

[9] I. Ounis , C. Macdonald, and I. Soboroff, "Overview of the TREC-2008 Blog Track"

[10] Yu, H.,& Hatzivassiloglou,V. (2003).Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. EMNLP'03.

[11] McDonald, R., Hannan, K., Neylon, T., Wells, M., & Reynar, J. (2007). Structured models for fine-to-coarse sentiment analysis. In Proceedings ACL'07 (pp. 432–439)

[12] Missen M.M.S., Boughanem M., and Cabanac G., Challenges for Sentence level Opinion Detection in Blogs. ICIS 2009 Shinghai China.

[13] H. Yang, L. Si, and J. Callan. (2007.) "Knowledge transfer and opinion detection in the TREC 2006 Blog track." In TREC Blog 2006). special publication 500-272

[14] Y. Lee,S.H. Na, J. Kim, S.H. Nam, H.Y. Jung, J.H. Lee KLE at TREC 2008 Blog Track: BlogPost and Feed Retrieval: In Proceedings of TREC 2008 NIST

[15] James P. Callan. Passage-level evidence in document retrieval. In Proceedings of SIGIR,pages 302–310, 1994.

[16] Xiaoyong Liu andW. Bruce Croft. Passage retrieval based on language models. In Proceedings of CIKM'02, pages 375–382, 2002

[17] James Allan. HARD track overview in TREC 2003: High accuracy retrieval from documents. In Proceedings of the TREC-12, pages 24–37, 2003

[18] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In Proceedings of SIGIR, pages 275–281, 1998

[19] Fei Song and W. Bruce Croft. A general language model for information retrieval (poster abstract). In Proceedings of SIGIR, pages 279–280, 1999

[20] Chengxiang Zhai and John D. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In Proceedings of SIGIR, pages 334–342, 2001

[21] Kazuhiro Seki and Kuniaki Uehara. Adaptive Subjective Triggers for Opinionated Document Retrieval. In Proceedings of WSDM 2009, pp. 25-33, February 2009