Blog Mining through Opinionated Words

Giuseppe Attardi[1]
Dipartimento di Informatica
Università di Pisa
attardi@di.unipi.it

Maria Simi
Dipartimento di Informatica
Università di Pisa
simi@di.unipi.it

Abstract

Intent mining is a special kind of document analysis whose goal is to assess the attitude of the document author with respect to a given subject. Opinion mining is a kind of intent mining where the attitude is a positive or negative opinion. Most systems tackle the problem with a two step approach, an information retrieval followed by a postprocess or filter phase to identify opinionated blogs. We explored a single stage approach to opinion mining, retrieving opinionated documents ranked with a special ranking function which exploits an index enriched with opinion tags. A set of subjective words are used as tags for identifying opinionated sentences. Subjective words are marked as "opinionated" and are used in the retrieval phase to boost the rank of documents containing them. In indexing the collection, we recovered the relevant content from the blog permalink pages, exploiting HTML metadata about the generator and heuristics to remove irrelevant parts from the body. The index also contains information about the occurrence of opinionated words, extracted from an analysis of WordNet glosses. The experiments compared the precision of normal queries with respect to queries which included as constraint the proximity to an opinionated word. The results show a significant improvement in precision for both topic relevance and opinion relevance.

1. Introduction

When searching the Web to find a solution for a technical problem, it is often frustrating to be referred to pages where other people ask about the same problem but nobody offers a solution. This happens for instance when looking for a fix to an error using as query the error message displayed by an application. One has to dig through a number of result pages where people ask the same question before finding one that reports an actual solution to the problem. This is an example where it would be useful if results were marked or grouped according to what they intend to express, such as: problem (*description*, *solution*), agreement (*assent*, *dissent*), preference (*likes*, *dislikes*), statement (*claim*, *denial*). We may call this *intent classification* as a generalization of *sentiment classification* which focuses on opinions, like preference or agreement.

The ability to identify and group documents by intent may lead to new tools for knowledge discovery, for instance for generating a research survey that collects relevant opinions on a subject, for determining prevalent judgments about products or technologies, for analyzing reviews, for gathering motivations and arguments from court decision making or lawmaking debates, for analyzing linkages in medical abstracts to discover drug interactions.

2. Approach

While traditional text classification tries to assign predefined categories to a document, such as

[1] Currently in sabbatical at Yahoo! Research Barcelona, Carrer Ocata 1, Barcelona, Spain

spam/no-spam for e-mail, sentiment or intent identification is a different and challenging task whose goal is the assessment of the writer's attitude toward a subject. Examples include categorization of customer e-mails and reviews by types of claims, modalities or subjectivities.

Learning algorithms for text classification typically represent text as a bag-of-words, where the word order and syntactic relations appearing in the original text are ignored. Despite such naive representation, text classification systems have been quite successful [6].

We experimented with a simple extension of this approach, which identifies subjective words in the documents considered to be carrying an opinion bias, and uses searches which detect their presence and a ranking measure that takes their presence into account.

Most systems tackle opinion mining with a two step approach: an information retrieval step followed by a postprocessing or filtering phase which tries to identify opinionated blogs among those ranked relevant by the IR engine.

Our approach instead consists in a single stage, retrieving opinionated documents ranked with a special ranking function which exploits an index enriched with opinion tags. Opinion mining is so reduced to search, exploiting the efficiency and effectiveness of inverted indexes. The processing of the whole test set for the TREC 2006 Blog task required just 6.28 seconds.

Integrating opinion mining within search has also the possible advantage of avoiding missing documents that the IR engine might overlook, not having a sufficiently high score. For example a document that only mentions an entity once, but then provides a lot of opinionated remarks, might achieve a lower score than a document that mentions the entity often, without expressing any opinion. Since the relevant documents may be hundred thousands, the postprocessing stage might not have a chance to consider it.

The effectiveness of the approach hinges on the ability to identify suitable ways to enrich the index annotating documents expressing opinions. Since we did not have training data from previous editions of the task, that would help identify most common ways to express opinions in blogs, we resorted to a generic solution, choosing the annotations from a general dictionary like WordNet.

Despite this, our TREC experiments show that the approach is effective in selecting opinionated blogs with respect to the baseline of normal IR.

The approach of enriching the index can be extended to tackle other issues, for instance synonyms. Searching using synonyms can be done by looking up the synonyms in a dictionary and then performing several searches with the various synonyms. Besides inefficiencies, this sometimes decreases precision, because of word ambiguities. Synonyms could instead be added as tags in a document, performing word sense disambiguation and only including those appropriate to the sense in the document. The additional cost of adding extra terms occurrences to the index are limited, adding a few bytes per occurrence in the compressed index representation. The extra cost during indexing is fully compensated by a much faster query speed.

3. Opinion Mining

The opinion retrieval task at TREC 2006 involves locating blog posts that express an opinion about a given target. The target can be a "traditional" named entity -- a name of a person, location, or organization -- but also a concept (such as a type of technology), a product name, or an event.

The TREC 2006 Blog task provides a collection of blogs for comparing and evaluating opinion mining systems. This is the first edition of the task, so there are no training data available to participants. The TREC Blog06 collection is just a collection of crawled feeds and blog pages. The answers to a set of 50 topics, submitted by all the participants, have been pooled and judged by human experts from NIST. After the TREC 2006 Conference, TREC will make available the list of all these relevance judgments. This list can be used for training and tuning the next version of our Opinion Classifier.

In this paper we describe how we tackled the task of locating opinionated blogs and discuss ideas for the more general task of intent classification.

4. Indexing the collection

The TREC Blog6 collection consists in over 3 million blogs collected from over 100,000 feeds crawled during a period of 3 months.

The feeds are pages in RSS/Atom format. Each RSS feed represents a single channel, with metadata for title, URL, description, generator, language and a list of items. Each item contains elements such as title, URL for the content, URL for the comments, description, date, creator and category. Atom feeds use slightly different naming but contain similar metadata and items.

4.1. Identifying content

One major issue was how to recover the content of each blog, since the standard for RSS 2.0 does not provide for the inclusion of the content in the feed itself. The 'Content' extension module allows including content within an item, but this is rarely used in the collection. Some feeds include the whole content in the description element, even though this field is meant to provide a short synopsis of the content. Hence the content must often be taken from the referred blog page. Unfortunately blog pages are messed up with all sort of extra information besides the blog post and the readers' comments: pages often include annotated lists of previous posts, lists of similar related pages, navigation bars, side bars, advertising, etc. If we indexed the page as a normal HTML page, all the text in these parts would end up in the index, leading to results with poor relevance.

For identifying the proper post content within a blog page, we used three strategies. The first strategy is to use the content element from the feed, when available. In order to do this, we created an index for the feeds. When indexing a blog permalink, we check whether the feed where it came from contains a content element: in this case we use that element as the content for indexing.

The second strategy is to deal specially with blogs generated by programs which follow well defined markup rules allowing the post's content to be identified.

Of the total 551,763 million blogs: 282,982 were produced with Blogger, 101,355 with WordPress, 99,100 with LiveJournal, 9,267 with MovableType, 3,562 with Technorati, 1,869 with UserLand, 626 with FeedCreator. Each generator creates pages with a specific markup style. For instance WordPress encloses the posts within a `<div class=”post”>` element, and the proper content within a `<div class=”storycontent”>` element. Post in Spanish instead are enclosed in a `<div class=”texto”>` element. Comments are enclosed in a `<ol class="commentlist">` element. Blogger instead uses div's `post`, `post-body` and `<div id=”comments”>` to enclose comments. For these most used content generators, we created a list of elements to be included.

We exploited the features of the customizable HTML reader in IXE [5], which allows providing a list of elements, element classes or element ids to skip or to include during indexing. For instance, using these parameters in IXE configuration file:

```
IncludeElement  Blogger div.post
 div.comments
IncludeElement  WordPress* div.post
 ol#commentlist
```

we direct IXE to limit indexing to `div` elements with class name `post` or `comments` for pages generated by Blogger or `div` elements with class name `post` or `ol` elements with id `commentlist` generated from any version of WordPress.

The third strategy is used for handling the remaining cases, excluding elements which are considered not part of the post. For example:

```
ExcludeElement  div.*link*
 div.side* div.*bar
ExcludeElement  div#header div#nav*
```

excludes from indexing any `div` whose class name contains `link`, starts with `side` or ends with `bar` as well as any `div` whose `id` name is `header` or starts with `nav`. Fortunately enough, many content generators do indeed use markup of this kind, so that with a list of about 50 elements to exclude, we avoid most of the irrelevant parts.

4.2. Avoiding spam blogs

Another problem is the presence of spam blogs, also called *splogs*, i.e. fake blog pages which

contain advertising or other irrelevant content used just to promote affiliated sites, which are often disreputable. For instance we detected in the collection a large number of splogs from the domain blogspot.com, which hosts a free blog posting service by Google. To avoid splogs, we used a black list of URLs from `http://www.splogspot.com/`. Any blog from that list is assigned a document rank of 0 during indexing, so that it will not normally appear in the search results.

Finally, pages are written in several languages, but only the English blogs are considered relevant according to TREC blog track guidelines. RSS includes a metadata field for language. However it is not used consistently and hence it is quite inaccurate. One could apply a language detector to identify the language, but in the case of blog posts, which are often quite short, also this method is not sufficiently accurate.

4.3. Tagging subjective words

In order to facilitate finding opinionated blogs, we enriched the index with tags for words, i.e. the index does not contain only words but also an overlay of tags for each word. One tag is the OPINIONATED tag, which is associated to subjective words considered to be carrying an opinion bias.

We tagged as opinionated a subset of the list of words SentiWordNet [7]. SentiWordNet was created from WordNet, starting from two seed sets of positive and negative terms, expanded by means of synonyms, antonyms and other semantic relations. Subjective terms were then represented as feature vectors consisting of terms in their description and glosses and used to train a statistical classifier. All words in WordNet were classified, producing the list SentiWordNet, consisting in 115,341 words marked with positive and negative orientation scores ranging from 0 to 1. We extracted from SentiWordNet a subset of 8,427 opinionated words, by selecting those whose orientation strength is above a threshold of 0.4.

5. Search Strategy

The inclusion of tags for opinionated words in the index allows performing proximity searches of the type:

```
content matches proximity 6
 [OPINIONATED:* 'George Bush']
```

which will return all documents where any (i.e. *) opinionated word occurs within 6 terms from the phrase `'George Bush'`.

We plan to refine the approach by exploiting an English parser [3], in order to detect whether the opinionated term refers indeed to Bush, rather than to another entity in the same sentence.

6. Results

We performed a few experiments using the TREC 2006 Blog topics number 851 to 900. These topics range from controversial or beloved political figures (e.g. Abramoff, Bush, Ann Coulter), to performers (Jon Steward), to movies or tv shows (March of the Penguins, Arrested Development, Life on Mars, Oprah Winfrey), to products (MacBook pro, Blackberry, Shimano) to political subjects (nuclear power, jihad). Each topic consists in a title, plus a description and a narrative. For example, topic 951 is as follows:

```
<title>"March of the Penguins"
<desc>Provide opinion of the film
 documentary "March of the
 Penguins".
<narr>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.
```

We performed a baseline run with queries made just from title words joined in AND. A second run used the same words but added a proximity operator with distance 6 to an opinionated word. The third run used an AND combination of title words plus an OR of description words. For the fourth run we used queries made from title words within proximity 6 from opinionated words plus an OR of description words.

For instance topic number 895 was dealt with the query:

```
content matches proximity 6
 [OPINIONATED:* Oprah] (Oprah |
 Winfrey | tv | show)
```

The TREC evaluators considered overall 19,891 documents as relevant for all topics and 11,530 as opinionated on those topics. According to this evaluation our runs obtained the following scores for precision at five (p@5):

run	topic		opinion	
	relevant	p@5	relevant	p@5
title	6150	56.80	3566	33.60
title + opinionated	4287	54.40	2500	32.80
title + description	5874	61.60	3293	36.00
title + opinionated + description	4290	69.60	2469	47.60

Results indicate that the opinionated words analysis provides a significant improvement on title+description queries.

We should note that the so called opinionated words, being extracted from a general dictionary, are not very specific and include terms such as ‘like’, ‘hate’, ‘not’ but also ‘want’ and ‘wish’. We hope that a better list might be obtained from training on the Blog 06 collection itself, exploiting the qrels from this year evaluation.

Opinionated words have a positive effect also in identifying documents relevant to a topic. A possible explanation is that they prune documents in which the query terms do not appear in fully formed sentences: for instance they appear in an anchor link.

The presence of opinionated words has a large effect (+11.6 %) on the precision of retrieval of opinionated documents: this improvement however is with respect to a relatively low score of 36%.

While queries made from title and description achieved a reasonable 61.60% precision at 5, not many of these are opinionated. This suggests that other criteria need to be used to find them, as we propose in section 8.

7. Performance

Our system was designed for speed in retrieval, hence it uses AND queries with proximity and a specialized search engine as mentioned earlier.

The system achieved a score of 47.60 for p@5, which is the third best result, after the University of Chicago with 52.00% and University of Amsterdam with 48.80%. The use of AND queries explains why the system did not rank as well according to Mean Average Precision, since fewer documents were typically returned.

On the other hand the system achieved a quite good retrieval speed, being able to process the whole set of 50 topics in just 6.28 seconds on a 2.8 GHz Linux PC with 2 MB of memory. The system with top p@5 score required several hors to complete the task [9].

8. Intent Retrieval

The Opinion Classifier will be used as part of an intent retrieval engine, which uses a specialized Passage Retrieval system in order to retrieve candidate sentences about a target.

The Passage Retrieval system [4] supports keyword searches based on a traditional inverted word-document index as well as searches for opinionated words as described above, but returns passages rather than documents where the keywords occur. The Passage Retrieval index also contains annotations about Named Entities and can be queried specifying the search for a term representing a named entity. This allows distinguishing for instance the term `'Apple'` used to mean an organization rather than a fruit. Each candidate result sentence is given a score computed by a combination of a classical IR similarity metric (PL2 [2], a variant of the well known BM25) and a distance metric on the target proximity. Anaphora is handled in a crude way by the Passage Retrieval which computes a score for the sentence based on the distance between the candidate sentence and one where the target occurs.

For intent classification, we plan to use techniques based on extracting dependency relations, which have proven to be more effective than traditional bag-of-word approaches [7].

Dependency relations allow distinguishing statements of opposite polarity, e.g. “I liked the movie”, and “I didn’t like it”. Dependency relations represent the linguistic structure of sentences, as an alternative to phrase structure trees, as in classical linguistics.

The features used by the intent classifier will include sub-patterns corresponding to dependency relations.

Since intent/attitudes can be expressed in very different ways depending on the domain [1], as a preliminary step the training documents will be mined to extract: an intent vocabulary and frequent sub-patterns corresponding to dependency relations and including terms from the vocabulary. A classifier is then trained on an annotated corpus using such frequent sub-patterns as features [8].

The intent mining task will be performed by retrieving sentences from the Passage Retrieval that contain the given target tagged as an entity in proximity to an opinionated word. Each retrieved sentence will be parsed and a set of sub-patterns extracted as features and used to classify the sentence. The result will consist of a list of sentences with associated notes about the presence of opinions about the given target and their polarity.

9. Conclusions

We presented the opinion search engine which we built for the TREC 2006 Blog Task. The engine is based on an enhanced index which maintains annotations on words denoting in particular whether the word is opinionated. Results are ranked according to the presence of opinionated words in the proximity of query terms.

Results from the TREC evaluation showed better then average precision and in particular significant improvements in precision when exploiting annotations on opinionated words.

We plan to use this engine as part of an opinion retrieval engine which extracts candidate sentences containing an opinion using a specialized Passage Retrieval index which maintains also annotations on Named Entities.

A final filtering stage will involve an Intent Classifier to assess whether each candidate sentence indeed contains an opinion on the requested subject.

Acknowledgements

We thank Andrea Esuli and Fabrizio Sebastiani for making available to us the list of opinionated words they produced from WordNet. Massimiliano Ciaramita provided suggestions to improve the presentation.

10. References

[1] K. Abe, S. Kawasoe, T. Asai, H. Arimura, S. Arikawa. 2002. Optimized substructure discovery for semi-structured data. *Proc. of 6th PKDD*, 1–14.

[2] G. Amati and C.J. Van Rijsbergen. 2002. Probabilistic models of information retrieval based on measuring divergence from randomness, *TOIS* **20**(4):357–389.

[3] G. Attardi. 2006. Experiments with a Multilanguage non-projective dependency parser. In *Proc. of the Tenth CoNLL*.

[4] G. Attardi, A. Cisternino, F. Formica, M. Simi, A. Tommasi, C. Zavattari. 2001. PIQASso: PIsa Question Answering System *Proceedings of Text Retrieval Conference (Trec-10)*, 599-607, NIST, Gaithersburg (MD).

[5] G. Attardi, A. Cisternino. 2001. Template Metaprogramming an Object Interface to Relational Tables, *Reflection 2001*, *LNCS* **2192**, 266-267, Springer-Verlag, Berlin.

[6] S. Dumais, J. Platt, D. Heckerman and M. Sahami. 1998.Inductive learning algorithms and representations for text categorization. In *Proceedings of CIKM*, 148–155.

[7] A. Esuli, F. Sebastiani. 2005. Determining the semantic orientation of terms through gloss classification. *CIKM 2005*: 617-624.

[8] S. Matsumoto, H. Takamura, M. Okumura. 2005. Sentiment Classification Using Word Sub-sequences and Dependency Sub-trees. In: T.B. Ho, D. Cheung & H. Li (eds), *Proceeding of PAKDD'05, the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*. *LNCS*, vol. 3518.

[9] W. Zhang. 2006. Personal communication.