Opinion Analysis for Business Intelligence Applications

Adam Funk
Department of Computer Science
University of Sheffield
Sheffield, UK
a.funk@dcs.shef.ac.uk

Yaoyong Li
Department of Computer Science
University of Sheffield
Sheffield, UK
y.li@dcs.shef.ac.uk

Horacio Saggion
Department of Computer Science
University of Sheffield
Sheffield, UK
saggion@dcs.shef.ac.uk

Kalina Bontcheva
Department of Computer Science
University of Sheffield
Sheffield, UK
k.bontcheva@dcs.shef.ac.uk

Christian Leibold
Semantics Technology Institute
University of Innsbruck
Innsbruck, Austria
christian.leibold@sti2.at

ABSTRACT

More than ever before, business analysts have access to public forums in which opinions and sentiments about companies, products, and policies are expressed in unstructured form. Mining information from public sources is of great importance to many business intelligence applications such as credit rating or company reputation.

We have implemented a supervised machine-learning system which uses linguistic information to classify text by rating (good or bad, for example, or 1 to 5 stars). In an evaluation we have obtained good results in comparison with the state-of-the-art in opinion mining.

We are further developing the system to classify each text according to a "qualitative variable" category from an ontology specially developed for Business Intelligence (BI). This work will allow us to generate RDF statements to populate a knowledge base for BI.

Keywords

Opinion analysis, sentiment extraction, business intelligence

1. INTRODUCTION

Work on opinion mining has been fuelled by the development of evaluation programs such as the Text Retrieval Conference (TREC) 2006 track on blog mining for opinion retrieval or the National Institute of Informatics Test Collection for Information Retrieval (NTCIR) Workshop on Evaluation of Information Access Technologies[1] and now the new Text Analysis Conference[2] (TAC) with a track on opinion summarization.

Opinion mining consists of several different problems, such as determining whether each segment of text (sentence, paragraph, or section) is "opinionated" or not; identifying the opinion-holder (the person or organization who expresses the opinion)[3]; and determining how positive or negative each opinion is. For business intelligence, it is also useful to classify each opinion according to the aspect of the business or transaction describes: e.g., product quality, ordering, or integrity.

Opinion analysis helps to assess the limitations of particular products and then exploit this information in the development of improved products or services. It also helps enterprises to understand their customers as well as to plan future products and services. Another applications is extracting opinions from legacy data, such as scientific articles, in which opinions about previous works are usually stated. [15]

Given the abundance of reviews on the World Wide Web about products, especially with the more recent proliferation of blogs and other Web 2.0 services, one application is to identify for a given entity its features and then identify what is being said about them (positive or negative statements), in order to compile summaries of opinions about particular entities or features. This information is then compiled in order to produce a textual summary together with statistics about what has been said.

The target application to which we are contributing will store the results of our work and that of other partners in a shared knowledge base, using RDF[4] for interoperability and integration with the rest of the MUSING[5] business intelli-

[1] http://research.nii.ac.jp/ntcir/workshop/

[2] http://www.nist.gov/tac/

[3] This can also be treated as an information extraction problem. [16]

[4] http://www.w3.org/RDF/

[5] http://www.musing.eu/, "Multi-Industry Semantic-Based Business Intelligence"

gence suite.

The MUSING project is applying human language technology such as ontology-based extraction in the context of business intelligence applications.[17] Business intelligence (BI) is the process of finding, gathering, aggregating, and analyzing information to support decision-making. It has become evident to business analysts that qualitative information plays an important role in many BI applications. One such application in MUSING is a reputation teller that aims to collect and organize opinions about business entities (organizations, people, products, etc.). In MUSING, information is organized in a domain ontology, which the information extraction systems target. In particular a sub-ontology in MUSING models subjective information such as reputation, reliability, and quality. The objective of our reputation teller application is to identify statements which reflect these concepts and track them over time in order to create an accurate picture of a business entity.

Here we present initial work on analysing language for that application. In particular, we aim in the experiments described below to establish the reliability and utility of simple linguistic features of classified texts for rapid supervised learning, in order to carry out opinion mining for business intelligence applications in various domains in which some annotated data are available—as is often the case with reviews on the web.

2. RELATED WORK

Classifying product reviews is a common problem in opinion mining: the goal is to identify for a given entity its features and the positive or negative statements expressed then identify what is being said about each of them. This information is then compiled in order to produce textual summaries together with statistics about the frequency of positive, negative, and neutral statements. A variety of techniques have been used here including supervised [9] and unsupervised [7, 18, 19, 20] machine-learning.

Language resources such as SentiWordNet have recently been developed for the research community. [5] Some approaches to opinion mining involve predefined gazetteers of positive and negative "opinion words", whereas Turney's well-known method [18] determined the semantic orientation of lexemes by calculating their Pointwise Mutual Information (PMI, based on probability of collocations [2]) to the reference words *excellent* and *poor*. More recent work on product reviews in particular involved the identification of words referring to *implicit and explicit features*. [11] Naturally, the work based on unsupervised learning has relied on *a priori* information.

Turney's work [18] on sentiment analysis achieved a classification accuracy of 74%, which we treat as a benchmark for good results.

The work presented here aims to avoid relying on *a priori* information but to use a data-driven approach based on rapid natural language processing (NLP) techniques as input to a machine learning tool. Our continuing and future work also aims to classify the texts by type according to a BI ontology.

3. BINARY CLASSIFICATION

We began by investigating a relatively simple problem of classifying paragraphs of text from the web as expressing either a positive or a negative opinion.

3.1 Corpus collection and preprocessing

We crawled web pages on a consumer forum[6] and collected a corpus of HTML documents, each containing in particular a comment (a paragraph of natural-language text) and a *thumbs-up* or *thumbs-down* rating, both entered by one of the forum's users.

Each rating was represented by an `<img>` tag pointing to a GIF cartoon of a thumbs-up or thumbs-down gesture, with an `alt` attribute of `Consumer ThumbsUp` or `Consumer ThumbsDown`, respectively.

We preprocessed each HTML document to identify the comment text (based on the HTML tags and CSS attributes) and annotate it with a feature indicating the rating (*thumbs-up* or *thumbs-down*) according to the adjacent `<img>` tag.

The resulting corpus consisted of 92 documents, each containing one instance (review) for classification. The distribution of ratings in the corpus was 67% *thumbs-down* and 33% *thumbs-up.*

3.2 Methodology

We then treated this as a straightforward binary classification problem: to train the support vector machine (SVM) engine [8] implemented in GATE [3] to classify each marked comment span as either *thumbs-up* or *thumbs-down*, based on *n*-grams of simple linguistic features of the text it contains. Unlike many other opinion-classification studies we did not use any predefined word-lists or specialized lexical resources, but allowed the machine-learning techniques used to infer the values of words implicitly from the training data.

We carried out the linguistic annotation by applying some of GATE's standard natural language processing (NLP) components for English: the sentence-splitter, the tokenizer, the morphological analyser (lemmatizer), and the Hepple part-of-speech tagger [6]. These processes segmented the review text into sentences and tokens and added the following features to each token (these abbreviations for the features will be used in subsequent discussions of our experiments and results):

string the original, unmodified text of the token;

root the lemmatized, lower-case form of the token (for example, *run* is the root feature for *run*, *runs*, *ran*, and *Running*);

category the part-of-speech (POS) tag, a symbol that represents a grammatical category such as determiner, present-tense verb, past-tense verb, singular noun, etc.)[7];

[6] `http://www.clik2complaints.co.uk`

[7] Our POS tagger uses the Wall Street Journal corpus's tagset.

orth a code representing the token's combination of upper- and lower-case letters[8] (if it has been classified as a word).

We then carried out training and evaluation with 10-fold cross-validation over the corpus described in Section 3.1, in order to classify each review text as *thumbs-up* or *thumbs-down* based on SVM analysis of n-grams of various combinations of the token features listed above. Table 1 summarizes the standard information extraction measurements from this series of experiments.[9]

3.3 Evaluation

From these results we can make the following general observations.

- The combination of *category* and *orth* produced relatively poor results—as expected, because it is semantically empty.
- Increasing the number of features does not necessarily improve performance, because it can make the training data sparse.
- Increasing the value of n in the n-gram can decrease performance, as is often the case with SVM machine-learning techniques (as in [14], for example).
- The unigram results obtained this way compare favourably with the 74% accuracy benchmark for the binary classification of movie review texts. [18]

Table 2 shows the detailed evaluation results by category for the three best analyses. As these breakdowns show, these experiments erred in the negative direction; i.e., it tended to misclassify *thumbs-up* texts as *thumbs-down* more often than the other way. (This is also true for the others listed in Table 1 but not reported in more detail here.)

This directional error is understandable because the dataset is inherently biased that way (67% *thumbs-down*, as mentioned in Section 3.1). Nonetheless, we consider 80% overall accuracy to be a good achievement using only simple token-level features.

4. FIVE-WAY CLASSIFICATION

Given the success of our approach to binary classification, we now consider a more complicated problem: five-way classification of product and company reviews (using discrete ratings from *1-star* to *5-star*).

4.1 Corpus collection and preprocessing

We crawled web pages on another consumer forum[10] and collected a corpus of HTML pages, each containing a number of separate comments product or company reviews. Each review consisted of a paragraph or two of natural-language text entered by one of the forum's users and the same user's rating of the company from one to five stars.

Each rating was represented by an `<img>` tag pointing to a GIF image of a row of one to five adjacent stars, with an `alt` attribute of `1 Star Review`, `2 Star Review`, etc.

We preprocessed each HTML document to identify each unit of comment text (based on the HTML tags and CSS attributes) and annotate it with a feature indicating the rating (from *1-star* to *5-star*) according to the adjacent `<img>` tag.

The resulting corpus consisted of 600 documents containing approximately 7300 classification instances, with ratings distributed unevenly as shown in Table 3.

4.2 Methodology

We treated this too as a straightforward classification problem: to train the same SVM engine to assign one of the five possible features to each comment span, based on combinations the same simple linguistic features of the text.

We carried out SVM training and evaluation with 5-fold cross-validation over the corpus described in Section 4.1, using various combinations of token features as in the binary set of experiments (Section 3.2).

Because of the much greater memory and processing time required to deal with the larger corpus, and since our previous experiments had indicated (as expected) that using bigrams, trigrams, and combinations of three features would not improve the results, we limited this set of experiments to unigrams of one or two features.

Table 4 summarizes the standard information extraction measurements for this series of experiments.

4.3 Evaluation

Even for five-way classification we obtained reasonably good overall results—around 74%. Unfortunately, as the detailed analysis of the two best results in Table 5 shows, the scores were very good only for the extreme classifications, *1-star* and *5-star*, whereas the scores for *2-star* and *3-star* in particular were quite low. (The detailed results for the other two experiments were similar.)

We attribute this uneven performance partly to the unbalanced distribution of ratings in our dataset (see Table 3) as well as to the inherent fuzziness of mid-range, subjective ratings. In other words, the opinions associated with *2-*, *3-*, and *4-star* ratings are less "opinionated" than *1-* and *5-star* ratings and therefore less clearly bounded. Table 6 shows a few examples of "vague" review texts.

The precision and recall scores in the *2-*, *3-*, and *4-star* categories also suggest that the classification errors occur mainly within these three mid-range classes; of course, misclassifying a *3-star* text as *2-star*, for example, is much less serious than misclassifying it as *1-star*.

It is also worth noting that an SVM engine treats these rat-

[8] `upperInitial`, `allCaps`, `lowerCase`, or `mixedCaps`

[9] *Precision* (P) is the proportion of automatically tagged labels of a particular type that were correct; *recall* (R)is the proportion of items with that type that were actually found by the system; and $F_1 = 2PR/(P+R)$ (the harmonic mean of precision and recall). See [12], for example, for details.

[10] `http://www.pricegrabber.co.uk`

Table 1: Overall evaluation of *thumbs-up/down* classification

n	Token features used	F_1 %		
		thumbs-down	*thumbs-up*	overall
1	string	85.0	51.7	78.9
1	root	85.1	50.0	78.9
1	string, category	84.2	50.0	77.8
1	root, category	84.1	50.7	77.8
1	string, orth	85.0	51.7	78.9
1	root, orth	85.8	53.0	80.0
1	category, orth	78.5	7.7	66.7
1	string, category, orth	84.2	50.0	77.8
1	root, category, orth	84.2	50.0	77.8
2	string	81.1	33.2	72.2
2	root	81.1	31.5	72.2
2	string, orth	81.1	33.2	72.2
2	root, category	80.5	28.2	71.1
2	root, orth	80.5	28.2	71.1
3	string	78.8	13.5	67.8
3	root	78.4	10.7	66.7
3	root, category	78.8	13.5	67.8

Table 2: Detailed results of the best binary classifications

n	Features used	Rating	Precision %	Recall %	F_1 %
1	root, orth	*thumbs-down*	77.2	98.8	85.8
		thumbs-up	85.0	44.2	53.0
		overall	80.0	80.0	80.0
1	root	*thumbs-down*	76.1	98.8	85.1
		thumbs-up	85.0	40.8	50.0
		overall	78.9	78.9	78.9
1	string	*thumbs-down*	76.2	98.8	85.0
		thumbs-up	85.0	42.5	51.2
		overall	78.9	78.9	78.9

Table 3: Distribution of ratings in the *1–5 star* dataset

Rating	% of instances
1-star	7.8%
2-star	2.3%
3-star	3.2%
4-star	18.9%
5-star	67.9%

Table 4: Overall evaluation of *1–5 star* classification

n	Token features used	F_1 % by rating					
		1-star	*2-star*	*3-star*	*4-star*	*5-star*	overall
1	root	79.9	1.8	5.8	22.5	85.1	74.9
1	string	78.0	2.4	7.2	23.7	84.6	74.1
1	root, category	77.0	24.0	7.3	24.3	84.3	73.7
1	root, orth	77.8	4.8	7.6	23.7	84.8	74.6

Table 5: Detailed results of the best *1–5 star* classifications

n	Features used	Rating	Precision %	Recall %	F_1 %
1	root	*1-star*	80.6	80.0	79.9
		2-star	30.0	0.9	1.8
		3-star	44.8	3.1	5.8
		4-star	44.1	15.1	22.5
		5-star	79.0	92.5	85.2
		overall	77.0	73.0	74.9
1	root, orth	*1-star*	78.9	77.5	77.8
		2-star	46.7	2.6	4.8
		3-star	65.0	4.1	7.6
		4-star	46.9	15.9	23.7
		5-star	78.7	92.3	84.8
		overall	76.6	72.7	74.6

Table 6: Examples of 2- and 3-star review texts which are difficult to classify

Rating	Text
2-star	My personal details were not retained and when asked for an ‘order number’ on the survey I could find no trace of it. Not a very pleasing shop. I have in the past been very pleased.
3-star	Navigation is not intuitive. It seems to be logically structured but the cues are too brief or assumptive. I took twice as long as with some alternative sites.
3-star	The secure server didnt work first time so I had to go through again and reenter half my info again before it worked. It did work in the end and I hope to receive the goods soon.

ings as a set of five arbitrary strings rather than as sequential numeric values.

5. CURRENT AND FUTURE WORK

This set of experiments indicates that we can successfully classify short texts by rating (the positive or negative value of the opinions) using machine-learning based on quick and simple analysis with a standard NLP toolkit—and without relying on predefined lists of opinion words.

In current and future work, we will treat identifying the opinion-holder (as mentioned in Section 1) and the subject of the review (the company or product) as standard information extraction and named-entity recognition (NER) problems (areas in which we also have experience). [1, 4]

Future experiments along these lines will build on the techniques presented here and will be applied to a wider variety of text sources, particularly those more relevant to business intelligence. We will also incorporate our other our previous work with SVM machine-learning on linguistic information for opinion analysis of the MPQA and NTCIR-6 corpora [10], and investigate the use of SVM active learning in order to minimize the training data required and benefit from some human domain expertise. For comparison we also intend to test weakly supervised methods such as the probabilistic model for hedge classification [13], and later using unlabelled data and transductive SVM in a move towards unsupervised learning.

To apply these developments to a BI system, we also need to classify the texts by type. For this purpose we are manually annotating the corpora described above so that each review is tagged with one of the subclasses of *Qualitative-Variable* in the *company* ontology, as shown in Figure 1. The MUSING ontology for business intelligence extends the PROTON System and Top modules.[11]

Once we have manually annotated the corpora, we will apply the techniques as in Sections 3 and 4 above to the problem of classifying review texts by qualitative categories.

Combining the text classification approach evaluated here with NER and the ontological classification of the *QualitativeVariables* will allow us to export useful reputation information as RDF statements, such as those shown in Figure 2, to a shared knowledge base for the MUSING project's *Reputation Teller* pilot service. Further approaches that we develop successfully will also be integrated into this system.

6. ACKNOWLEDGEMENTS

This work is partially supported by the EU-funded MUSING project (IST-2004-027097).

7. REFERENCES

[1] K. Bontcheva, M. Dimitrov, D. Maynard, V. Tablan, and H. Cunningham. Shallow Methods for Named Entity Coreference Resolution. In *Chaînes de*

[11] `http://proton.semanticweb.org/`

```
protons:Entity
       |  ...
       \--protont:Object
                 |  ...
                 \--company:QualitativeVariable
                      |--company:CompanyDevelopment
                      |--company:CreditDecision
                      |--company:CreditWorthinessIndex
                      |--company:GeneralOpinion
                      |--company:Identity
                      |--company:Imagination
                      |--company:Integrity
                      |--company:OrderSituation
                      |--company:PaymentExperience
                      |--company:Quality
                      |--company:Reliability
                      |--company:SocialResponsibility
                      |--company:TechnicalInnovation
                      \--company:ValueForMoney
```

Figure 1: Excerpt from the MUSING ontology's class hierarchy showing *Qualitative Variable* and its subclasses

```
<rdf:RDF
   xmlns:company="http://musing.deri.at/ontologies/v0.8/general/company#"
   xmlns:bach="http://musing.deri.at/ontologies/v0.8/general/bach"
   xmlns:protonu="http://proton.semanticweb.org/2005/04/protonu#"
   xmlns:protont="http://proton.semanticweb.org/2005/04/protont#"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
>

<company:Corporation rdf:ID="ACME">
  <company:hasReputation>
    <company:Reputation rdf:ID="Reputation_ACME">
      <rdfs:label rdf:datatype="xsd:string">Acme</rdfs:label>
    </company:Reputation>
  </company:hasReputation>
  <rdfs:label rdf:datatype="xsd:string">Acme</rdfs:label>
</company:Corporation>

<company:Rating rdf:ID="Rating_Acme">
  <bach:hasQualitativeVariable rdf:resource="#ValueForMoney" />
  <rdfs:label rdf:datatype="xsd:string">Acme</rdfs:label>
</company:Rating>

<company:Reliability rdf:ID="ValueForMoney_ACME">
  <bach:hasQualitativeValue rdf:datatype="xsd:float">-1.5</bach:hasQualitativeValue>
  <rdfs:label rdf:datatype="xsd:string">Acme</rdfs:label>
</company:Reliability>

</rdf:RDF>
```

Figure 2: RDF statements representing a negative *Reliability* rating for the *Acme* corporation

références et résolveurs d'anaphores, workshop TALN 2002, Nancy, France, 2002. http://gate.ac.uk/sale/taln02/taln-ws-coref.pdf.

[2] K. W. Church and P. Hanks. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics*, 16(1):22–29, 1990.

[3] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, 2002.

[4] M. Dimitrov, K. Bontcheva, H. Cunningham, and D. Maynard. A Light-weight Approach to Coreference Resolution for Named Entities in Text. In A. Branco, T. McEnery, and R. Mitkov, editors, *Anaphora Processing: Linguistic, Cognitive and Computational Modelling*. John Benjamins, 2004.

[5] A. Esuli and F. Sebastiani. SENTIWORDNET: A publicly available lexical resource for opinion mining. In *Proceedings of LREC 2006*, 2006.

[6] M. Hepple. Independence and Commitment: Assumptions for Rapid Training and Execution of Rule-based Part-of-Speech Taggers. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, Hong Kong, October 2000.

[7] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, New York, NY, USA, 2004. ACM.

[8] Y. Li, K. Bontcheva, and H. Cunningham. An SVM Based Learning Algorithm for Information Extraction. Machine Learning Workshop, Sheffield, 2004. http://gate.ac.uk/sale/ml-ws04/mlw2004.pdf.

[9] Y. Li, K. Bontcheva, and H. Cunningham. Cost Sensitive Evaluation Measures for F-term Patent Classification. In *The First International Workshop on Evaluating Information Access (EVIA 2007)*, pages 44–53, May 2007.

[10] Y. Li, K. Bontcheva, and H. Cunningham. Experiments of opinion analysis on the corpora MPQA and NTCIR-6. In *Proceedings of the Sixth NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*, pages 323–329, May 2007.

[11] B. Liu, M. Hu, and J. Cheng. Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of the 14th international conference on World Wide Web (WWW '05)*, pages 342–351, New York, NY, USA, 2005. ACM.

[12] C. D. Manning and H. Schütze. Evaluation measures. In *Foundations of statistical natural language processing*, chapter 8.1, pages 267–271. Cambridge, MA, MIT Press, 1999.

[13] B. Medlock and T. Briscoe. Weakly supervised learning for hedge classification in scientific literature. In *Proceedings of the ACL 2007*, 2007.

[14] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment Classification Using Machine Learning Techniques. In *Proceedings of the 2002 Conference on EMNLP*, pages 79–86, 2002.

[15] S. S. Piao, S. Ananiadou, Y. Tsuruoka, Y. Sasaki, and J. McNaught. Mining opinion polarity relations of citations. In *Proceedings of the 7th International Workshop on Computational Semantics*, 2007.

[16] E. Riloff, C. Schafer, and D. Yarowsky. Inducing information extraction systems for new languages via cross-language projection. In *Proceedings of the 19th international conference on Computational linguistics*, pages 1–7, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[17] H. Saggion, A. Funk, D. Maynard, and K. Bontcheva. Ontology-based information extraction for business applications. In *Proceedings of the 6th International Semantic Web Conference (ISWC 2007)*, Busan, Korea, November 2007.

[18] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL '02)*, pages 417–424, Morristown, NJ, USA, July 2002. Association for Computational Linguistics.

[19] T. Zagibalov and J. Carroll. Unsupervised classification of sentiment and objectivity in chinese text. In *Proceedings of IJCNLP 2008*, Hyderabad, India, January 2008.

[20] L. Zhuang, F. Jing, and X.-Y. Zhu. Movie review mining and summarization. In *CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 43–50, New York, NY, USA, 2006. ACM.