Opinion Mining in Hungarian based on textual and graphical clues

GÁBOR BEREND, RICHÁRD FARKAS
Department of Informatics
University of Szeged
Árpád tér 2., Szeged, H-6720
HUNGARY
Berend.Gabor@stud.u-szeged.hu, rfarkas@inf.u-szeged.hu

Abstract: - Opinion Mining aims at recognizing and categorizing or extracting opinions found in unstructured text resources and is one of the most dynamically evolving subdiscipline of Computational Linguistics showing some resemblance to document classification and information extraction tasks. In this paper we propose a novel approach in Opinion Mining which combines Machine Learning models based on traditional textual and graphical clues as well. By examining subjective messages in a given forum topic dealing with a specific voting question, our system makes a prediction about the opinion of unknown people, which can be utilized to predict the forthcoming result of a referendum. The novelty of the work is that beside the regular textual clues (i.e. uni-bigrams), decisions are enhanced by using knowledge derived from a so-called response graph, which represents the interactions between the forum members. Our experimental results showed that with the help of such a graph we were able to achieve better results and significantly outperform the baseline accuracy. The promising results have reinforced our expectations that such an application can be easily adapted to any future Opinion Mining task in the election domain.

Key-Words: - sentiment analysis, opinion mining, text mining, information extraction, data mining, graph-based techniques

1 Introduction

Opinion Mining [1] aims at detecting and possibly extracting opinions and polarity about a certain topic from unstructured texts. This task has been receiving increasing academic interest in Natural Language Processing mainly for a decade. This phenomenon is due to the fact that nowadays people are more likely to share their emotions and opinions toward various topics, thus, the amount of information on web sites (i.e. blogs and forums) that reflects the user's opinion has seen a remarkable growth in size [2].

From these rich sources of opinions valuable information can be extracted, which can help, for instance, political parties to design their campaign programme or companies to get feedback about their product structure based on opinions expressed on the internet.

The main task of Opinion Mining is similar to that of information retrieval to some content. However, the target is to capture not the objective content but the subjective sentiments, opinions and their polarity expressed in the text.

Nevertheless, identifying polar phrases, which expresses its author's emotions towards a certain topic seems to be a yet unsolved and challenging problem because of the subjective and context-sensitive nature of opinion evaluation. Several attempts have been made to help determining the polarity of phrases [3, 4].

This paper describes a system which classifies each forum member of a forum topic containing discussion on the necessity and judgments about a Hungarian referendum. By reliably classifying the forum members, the aim of our application is to predict the opinions of unknown people, thus to forecast the outcome of a forthcoming election.

2 Related work

Within Computational Linguistics, works dealing with the topic of Opinion Mining have only become a relevant part of the academic interest in the past few years and there has been no previous work on Opinion Mining for Hungarian language.

Our target application shows similarity to Kim and Hovy (2007) [5] where their aim was to predict the results of the forthcoming Canadian elections by collecting predictive opinions and deriving generalized features from them. We also worked on the election domain but in Hungarian, however, the main difference between the two works is that we were interested in personal, subjective opinions towards the topic (e.g. "I strongly reject this issue and I will definitely say no at the referendum.") instead of predictive opinions (e.g. "I think Democrats will win.").

Other works such as Kobayashi et al. (2007) [6] focus on customer opinion extraction. Their task consisted of extracting *aspect-evaluation* and *aspect-of* relations from

unstructured weblog posts on product reviews. They used contextual and context-independent clues as well.

Since one of the main challenges of Opinion Mining is to determine polarity expressed in the text, several language resources have been developed to support this task. For instance, Esuli et al. (2006) [3] created SentiWordNet, ordering a triplet of numbers to each synset of the Princeton WordNet, describing its objectivity, positive and negative emotional charge. Kaji et al. (2007) [4] automatically collected a list of polar words and phrases based on structural analysis of massive collection of Japanese HTML documents.

3 Dataset

Previously, there has been no database dedicated to Opinion Mining in Hungarian language. Therefore, we had to create a corpus on our own. The data for further processing were gathered from the posts of the forum topic of the Hungarian government portal (www.magyarorszag.hu) dealing with the referendum about dual citizenship[1].

3.1 Annotation guidelines

We downloaded all the 1294 forum posts from the three month period preceding the referendum and these were annotated by two independent linguists so that we could measure the consistency of the annotation. Since we were interested in the future vote of the author of each comment, annotators were told to label them independently according to the most likely vote their composer would give. Based on this, we determined three categories of comments, i.e. *irrelevant*, *supporting* and *rejecting* ones.

However, preliminary results showed us that a significant proportion of the posts belonged to another class, namely those stating that they would intentionally vote invalidly because they did not like the idea of asking such a question in a referendum. So, finally we had to classify the posts into four groups (*irrelevant*, *supporting*, *rejecting* and *invalid*).

Comments labeled differently by the annotators were collected and given to a third linguist, who made the final decision on the ambiguous annotations. In this way our disambiguated dataset consisting of 1294 documents from 85 authors was yielded.

By aggregating the labeled, individual posts based on their authors, we automatically determined the orientation of each author. The following aggregation procedure was executed: an author got irrelevant label only if he had nothing but irrelevant posts, otherwise he got the non-irrelevant label out of which he possessed the most (in case of possible equality we decided on the label of the author's latest comment).

3.2 Annotation results

The considerably subjective nature of determining one's polarity towards the referendum often expressed only by hints or irony ended up in 299 differently labeled document out of the 1294 ones. This means 76.89% inter-annotator agreement rate and a 0.487 κ-measure [7] for the post-level annotation, while the author-level annotation reaches a 72.94% agreement rate and a κ-measure of 0.613.

Due to the general interpretation of κ-measure, scores between 0.4 and 0.6 are considered moderate while scores between 0.6 and 0.8 are regarded as substantial inter-annotator agreement.

The following table contains some statistics on the contingency of the author-level annotation:

		Annotator A				
		Supp	Rej	Inv	Irr	Sum
Annotator B	Supp	16 18,82%	2 2,35%	2 2,35%	6 7,06%	26 30,56%
	Rej	3 3,53%	20 23,53%	-	4 4,71%	27 31,76%
	Inv	-	-	4 4,71%	-	4 4,71%
	Irr	2 2,35%	4 4,71%	-	22 25,88%	28 32,94%
	Sum	21 24,71%	26 30,59%	6 7,06%	32 37,65%	85 100%

Table 1. Contingency table of the author-level annotation (Supportive, Rejecting, Invalid, Irrelevant)

From the κ-measures it can be clearly seen that the categorization of forum members can be executed with more reliability than that of an individual post. Thus, the previously mentioned accuracy of 72.94% among annotators in the case of author-level annotation can be dealt as a theoretical upper bound for our automatic system.

4 Methods

In our approach towards classifying forum members of the given topic we combined the results of two machine learning methods. One of them was based on the traditional Vector Space Model while the other one was trained on data derived from a so-called interaction or response graph.

[1] http://www.kettosallampolgarsag.mtaki.hu/english.html

4.1 Vector Space Model

When dealing with textual clues we used Vector Space Model, the most commonly used representation for documents. This model describes each document as a vector, based on the presence of terms contributing to the model. We carried out various filtering steps on the terms of the Vector Space Model trying to eliminate noise from the data and to build a better performing model that way.

Upon building our model we used word uni- and bigrams in the feature space. These terms were tf-idf normalized in order to get the most relevant subset of them. After the normalization we handled the occurrence of a term in a binary way, which means that if a term was absent in a document, it got 0 value in the vector representing it, otherwise that value was set to 1.

Besides tf-idf normalization further filtering and preprocessing steps were carried out. In a preprocessing step we lemmatized texts, which can be of use since we get the root of a word by detaching it from its suffixes. In morphologically rich languages such as Hungarian, it might be very useful since the very same root of a noun has 268 different forms due to its possible suffixes [8]. However, when lemmatizing texts, we can also lose nuances of meaning, so we have to pay special attention to overgeneralization [9].

Named entities occurred mainly in salutations (e.g. "Dear Andrew") and other irrelevant forms which definitely do not contribute to the classification of forum members, so we decided to replace each named entity to term *NAMED_ENTITY*. We simply considered a sequence of tokens as a named entity if it started with a capital letter not at the beginning of a sentence. However, if a named entity was recognized inside a sentence, it was also treated as a named entity even if it occurred at the beginning of a sentence.

We also investigated the part-of-speech (POS) of each token. Its utility, on the one hand, is that homonymous words having different POS-codes can be distinguished. On the other hand, if we know the POS of a word, we can make filterings based on this knowledge: for instance, we can exclude from our model articles or auxiliary words which lack any special meaning in the vast majority of cases.

Stop word and token length filterings were also executed, which could helped us not to build in irrelevant terms into the learning model.

We tried trigger word based filtering as well. Based on our empirical experiences we concluded that if a lemma from *yes*, *no* or *support* is included in a post, it can be stated with very high certainty that the post was relevant to the topic of the referendum.

After performing various combinations of the above mentioned preprocessing and filtering procedures we used C4.5 decision trees [10] for the machine learning process.

4.2 Interaction graph

Our idea was to make use of our hypothesis that people representing different views in the debate would comment more frequently on each other's posts compared to others.

Thus, we composed a weighted, directed graph, in which each vertex is mapped to a person and the weight of an *edge(A, B)* corresponds to the number of person *B*'s replies towards person *A*. We gained this information from the HTML structure of the pages, but it is worth knowing that not everyone indicated if he was replying to another post and some people did not use this feature correctly (e.g. addressed replies to themselves, however, such loops in the graph were omitted).

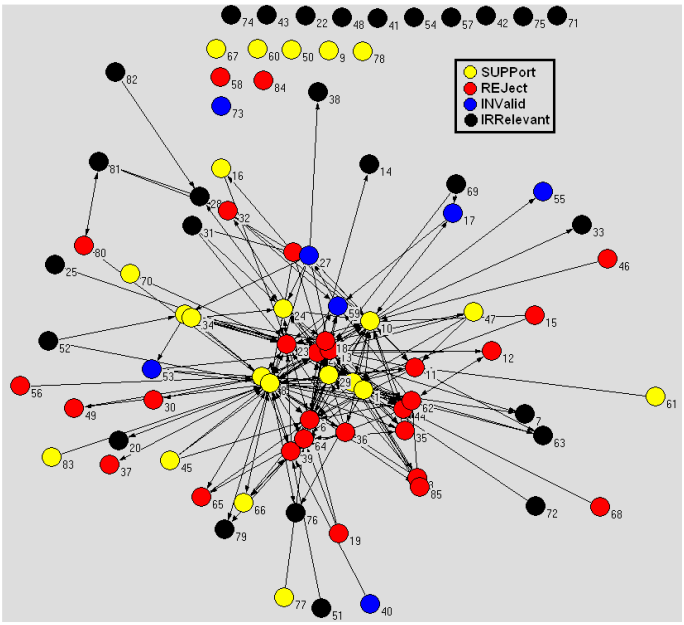

Figure 1. Visualization of the interaction graph

From the graph we extracted various characteristics that were used for creating feature vectors. These features consisted of the number of total/rejecting/supporting neighbors, the number of incoming and outcoming edges as well as the ratio of rejecting and supporting vertexes one and two distance away.

These purely graphical features were extended by others characteristic of forum members, for instance, posting frequency, the number of post or the date of the first/last post and the time elapsed between them.

4.3 Combination of the methods

Since the above mentioned methods have different advantages and different disadvantages, it seemed to be obvious to combine their results.

Predictions of the Vector Space Model based machine learning model tend to achieve better results on *irrelevant* class, but it performs fairly well at other class labels as well.

Results of the response graph performs better at recognizing relevant forum members (belonging to *support*, *reject*, *invalid* classes) in those cases where users have more posts than the average number of posts per forum member.(15.22).

So, in case a forum user had more than 15 posts, we accepted the prediction of the interaction graph, otherwise we chose the Vector Space Model based prediction, improving the accuracy of our system in this way.

5 Results and discussion

As mentioned previously, human annotator agreement on our task is 72.94%, which can be regarded as a theoretical upper bound of our application. For a baseline method we chose the simple rule which assigns the most common class (i.e. *reject*) to all of the forum members. It achieved an accuracy of 34.11%.

Because of the relatively small database available we decided to use one-leave out evaluation in our experiments to see how it would perform when trying to determine the polarity of an unknown forum user.

For the Vector Space Model based machine learning the best results were achieved when using uni- and bigrams along with named entity and stop word filtering. Applying these filtering methods we managed to improve the accuracy of the Vector Space Model-based predictions from 51.76% to 65.88%.

The detailed results of the best-performing predictions, using C4.5 decision tree are included in the following table:

	Recall	**Precision**	**F-measure**
Irrelevant	0.889	0.686	0.774
Support	0.455	0.625	0.526
Reject	0.724	0.656	0.689
Invalid	0.143	0.5	0.222
Total	0.6588		

Table 2. Breakdown of the Vector Space Model-based results according to the four classes

Using the features derived from the response graph, we achieved an accuracy of 55.29% and got the following results for our predictions:

	Recall	**Precision**	**F-measure**
Irrelevant	0.778	0.538	0.636
Support	0.364	0.727	0.485
Reject	0.621	0.514	0.563
Invalid	0.0	0.0	0.0
Total	0.5529		

Table 3. Breakdown of the response graph-based results according to the four classes

Results after combining the two methods can be seen in the following tables:

	ETALON				
	Irr	**Supp**	**Rej**	**Inv**	***Sum***
Irrelevant	24	5	4	2	35
Support	1	13	2	2	18
Reject	2	4	23	2	31
Invalid	0	0	0	1	1
Sum	27	22	29	7	85

Table 4. Confusion matrix of the final predictions

	Recall	**Precision**	**F-measure**
Irrelevant	0,888889	0,685714	0,774194
Support	0,590909	0,722222	0,65
Reject	0,793103	0,741935	0,766667
Invalid	0,142857	1	0,25
Total	0.7176		

Table 5. Breakdown of the final results according to the four classes

Our results have shown that with the help of filtering and preprocessing of texts the accuracy of the Vector Space Model could be improved by 5.88%.

As our experiments show, using graphical clues alone is less effective than using textual ones. However, if we consider the fact that it yields bad predictions mainly in those cases where the forum members had only a few posts and in the case of frequently posting people it outperforms the accuracy of the Vector Space Model (regarding those forum members who have more posts than the threshold, Vector Space Model-based predictions achieve 62.5% accuracy, while the response graph-based one performs at 93.75%). So, we can definitely say that it is worth combining the results of the two models.

In such a manner we could reach further improvement in our results ending up in an overall accuracy of 71.76%, which means that we managed to outperform our baseline value of 34% by more than 37%, and we also successfully approximated the inter annotator accuracy of 72.94%.

It is also interesting to mention that the ratio of the *supporting* forum members among the most relevant

classes (*supporting* + *rejecting*) in the etalon dataset was 0.57 and the very same ratio resulted in 0.63 based on our predictions. Thus, the results of our system might be used for giving a rough approximation on the outcome of such a forum-based debate.

6 Conclusions

In our paper we proposed a novel approach in Opinion Mining which enhances traditional Natural Language Processing techniques by exploiting valuable information extracted from response graphs based on the interactions of users.

The importance of this work is multifold. Since there were no previous works on opinion mining in Hungarian, we had to make the first Hungarian corpus entirely dedicated to such a task. Hopefully, our promising results will inspire others in Hungary as well to deal with this kind of problem. More importantly, our Opinion Mining system significantly outperformed the baseline system and achieved comparable results to the inter-annotation agreement. This reinforces our belief that such systems can be successfully applied in the case of other similar tasks in economic and political domains.

We believe that further improvements can be done by a more sophisticated named entity recognition, which identifies members and organizations of political sides and handle them in a special way as opposed to other rather irrelevant named entities. Another possibility of getting better results is to handle intra-sentence, intra-post anaphoras and automatic detection of interactions among authors as well.

Acknowledgments

The authors wish to thank the three annotators, Attila Almási, Kata Tóth and Norbert Kiss for their devoted efforts. This work was supported in part by the NKTH grant of the Jedlik Ányos R&D Programme 2007 (project codenames TUDORKA7 and TEXTREND) of the Hungarian government.

References:

[1] Anindya Ghose, Panagiotis G Ipeirotis, Arun Sundararajan, Opinion Mining Using Econometrics: A Case Study on Reputation Systems, *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 2007, pp.416-423

[2] Robert Metcalfe, Metcalfe's Law: A network becomes more valuable as it reaches more users, *Infoworld*, 1995

[3] Andrea Esuli, Fabrizio Sebastiani, SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining, *Proceedings of LREC-06, the 5th Conference on Language Resources and Evaluation,* 2006, pp.417-422

[4] Nobuhiro Kaji, Masaru Kitsuregawa, Building Lexicon for Sentiment Analysis from Massive Collection of HTML Documents, *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007, pp. 1075-1083

[5] Soo-Min Kim, Eduard Hovy, Crystal: Analyzing Predictive Opinions on the Web, *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007, pp. 1056-1064

[6] Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, Extracting Aspect-Evaluation and Aspect-Of Relations in Opinion Mining, *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007, pp. 1065-1074

[7] Joan E Carletta, Assessing agreement on classification tasks: The kappa statistic, *Computational Linguistics,* No. 22, 1996, pp. 249-254

[8] Richard Farkas and Veronika Vincze and Istvan Nagy and Róbert Ormandi and Gyorgy Szarvas and Attila Almasi: Web-based lemmatisation of Named Entities, *In Proceedings of TSD*, 2008

[9] Kushal Dave, Steve Lawrence, David M. Pennock, Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews, *WWW '03: Proceedings of the twelfth international conference on World Wide Web*, 2003, pp. 519-528

[10] Robert Quinlan, *C4.5: Programs for Machine Learning,* Morgan Kaufmann Publishers, 1993