AttitudeMiner: Mining Attitude from Online Discussions

Amjad Abu-Jbara
EECS Department
University of Michigan
Ann Arbor, MI, USA
amjbara@umich.edu

Ahmed Hassan
Microsoft Research
Redmond, WA, USA
hassanam@microsoft.com

Dragomir Radev
EECS Department
University of Michigan
Ann Arbor, MI, USA
radev@umich.edu

Abstract

This demonstration presents AttitudeMiner, a system for mining attitude from online discussions. AttitudeMiner uses linguistic techniques to analyze the text exchanged between participants of online discussion threads at different levels of granularity: the word level, the sentence level, the post level, and the thread level. The goal of this analysis is to identify the polarity of the attitude the discussants carry towards one another. Attitude predictions are used to construct a signed network representation of the discussion thread. In this network, each discussant is represented by a node. An edge connects two discussants if they exchanged posts. The sign (positive or negative) of the edge is set based on the polarity of the attitude identified in the text associated with the edge. The system can be used in different applications such as: word polarity identification, identifying attitudinal sentences and their signs, signed social network extraction from text, subgroup detect in discussion. The system is publicly available for download and has an online demonstration at http://clair.eecs.umich.edu/AttitudeMiner/.

1 Introduction

The rapid growth of social media has encouraged people to interact with each other and get involved in discussions more than anytime before. The most common form of interaction on the web uses text as the main communication medium. When people discuss a topic, especially when it is a controversial one, it is normal to see situations of both agreement and disagreement among the discussants. It is even not uncommon that the big group of discussants split into two or more smaller subgroups. The members of each subgroup mostly agree and show positive attitude toward each other, while they mostly disagree with the members of opposing subgroups and possibly show negative attitude toward them. These forms of sentiment are expressed in text by using certain language constructs (e.g. use insult or negative slang to express negative attitude).

In this demonstration, we present a system that applies linguistic analysis techniques to the text of online discussions to predict the polarity of relations that develop between discussants. This analysis is done on words to identify their polarities, then on sentences to identify attitudinal sentences and the sign of attitude, then on the post level to identify the sign of an interaction, and finally on the entire thread level to identify the overall polarity of the relation. Once the polarity of the pairwise relations that develop between interacting discussants is identified, this information is then used to construct a signed network representation of the discussion thread.

The system also implements two signed network partitioning techniques that can be used to detect how the discussants split into subgroups regarding the discussion topic.

The functionality of the system is based on our previous research on word polarity identification (Hassan and Radev, 2010) and attitude identification (Hassan et al., 2010). The system is publicly available for download and has a web interface to try online[1].

This work is related to previous work in the areas of sentiment analysis and online discussion mining. Many previous systems studied the problem of identifying the polarity of individual words (Hatzivassiloglou and McKeown, 1997; Turney and Littman, 2003). Opinionfinder (Wilson et al., 2005a) is a system for mining opinions from text. Another research line focused on analyzing online discussions. For example, Lin et al. (2009) proposed a sparse coding-based model that simultaneously models the semantics and the structure of threaded discussions and Shen et al. (2006) proposed a method for exploiting the temporal information in discussion streams to identify the reply structure of the dialog. Many systems addressed the problem of extracting social networks from data (Elson et al., 2010; McCallum et al., 2007), but none of them considered both positive and negative relations.

In the rest of the paper, we describe the system architecture, implementation, usage, and its perfor-

[1] http://clair.eecs.umich.edu/AttitudeMiner/

Proceedings of the NAACL-HLT 2012: Demonstration Session, pages 33–36,
Montréal, Canada, June 3-8, 2012. ©2012 Association for Computational Linguistics

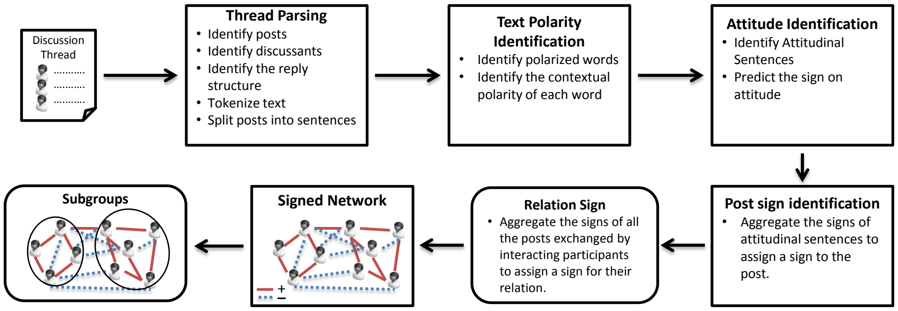

Discussion Thread
Thread Parsing
• Identify posts
• Identify discussants
• Identify the reply structure
• Tokenize text
• Split posts into sentences
Text Polarity Identification
• Identify polarized words
• Identify the contextual polarity of each word
Attitude Identification
• Identify Attitudinal Sentences
• Predict the sign on attitude
Post sign identification
• Aggregate the signs of attitudinal sentences to assign a sign to the post.
Relation Sign
• Aggregate the signs of all the posts exchanged by interacting participants to assign a sign for their relation.
Signed Network
+
−
Subgroups

Figure 1: Overview of the system processing pipeline

mance evaluation.

2 System Overview

Figure 1 shows a block diagram of the system components and the processing pipeline. The first component in the system is the *thread parsing* component which takes as input a discussion thread and parses it to identify the posts, the participants, and the reply structure of the thread. This component uses a module from CLAIRLib (Abu-Jbara and Radev, 2011) to tokenize the posts and split them into sentences.

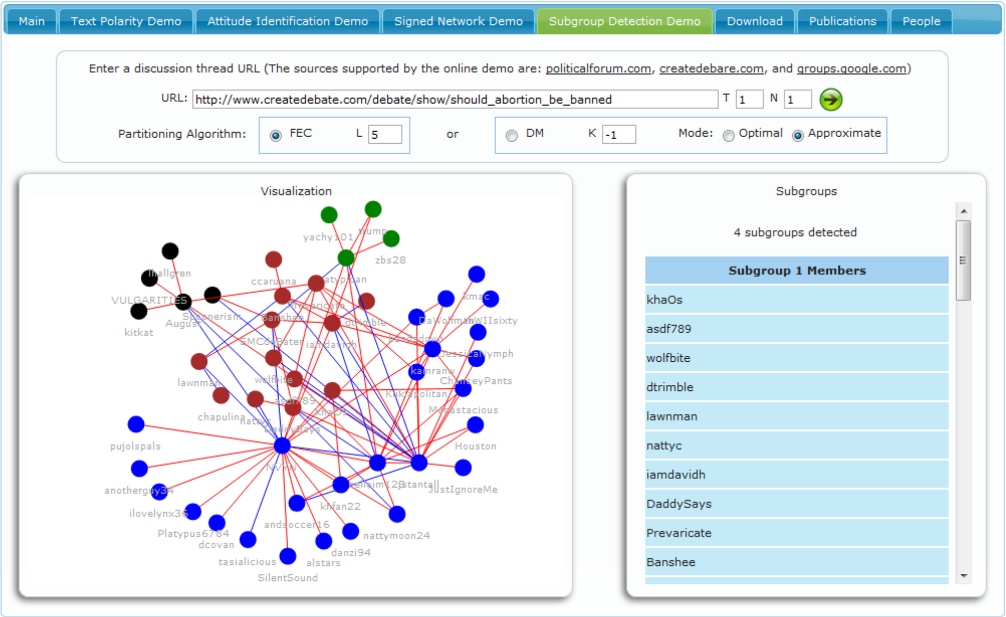
The second component in the pipeline processes the text of the posts to identify polarized words and tag them with their polarity. This component uses the publicly available tool, opinionfinder (Wilson et al., 2005a), as a framework for polarity identification. This component uses an extended polarity lexicon created by applying a random walk model to WordNet (Miller, 1995) and a set of seed polarized words. This approach is described in detail in our previous work (Hassan and Radev, 2010). The context of words is taken into consideration by running a contextual word classifier that determines whether the word is used in a polarized sense given the context (Wilson et al., 2005b). For example, a positive word appearing in a negated scope is used in a negative, rather than a positive sense.

The next component is the attitude identification component. Given a sentence, our model predicts whether it carries an attitude from the text writer toward the text recipient or not. As we are only interested in attitudes between participants, we limit our analysis to sentences that use mentions of a discussion participants (i.e. names or second person pronouns). We also discard all sentences that do not contain polarized expressions as detected by the previous component. We extract several patterns at different levels of generalization representing any given sentence. We use words, part-of-speech tags, and dependency relations. We use those patterns to build two Markov models for every kind of patterns. The first model characterizes the relation between different tokens for all patterns that correspond to sentences that have an attitude. The second model is similar to the first one, but it uses all patterns that correspond to sentences that do not have an attitude. Given a new sentence, we extract the corresponding patterns and estimate the likelihood of every pattern being generated from the two corresponding models. We then compute the likelihood ratio of the sentence under every pair of models. Notice that we have a pair of models corresponding to every type of patterns. The likelihood ratios are combined using a linear model, the parameters of which are estimated using a development dataset. Please refer to (Hassan et al., 2010) for more details about this component.

The next component works on the post level. It assigns a sign to each post based on the signs of the sentences it contains. A post is classified as negative if it has at least N_s negative sentences, otherwise it is classified as positive. The value of N_s can be chosen by the user or set to default which was estimated using a small labeled development set. The default value for N_s is 1 (i.e. if the post contains at least one negative sentence, the whole post is considered to be negative).

The next component in the pipeline uses the attitude predictions from posts to construct a signed network representation of the discussion thread. Each participant is represented by a node. An edge is created between two participants if they interacted with each other. A sign (positive or negative) is assigned to an edge based on the signs of the posts the two participants connected by the edge have exchanged. This is done by comparing the number of positive and negative posts. A negative sign is given if the two participants exchanged at least N_p negative posts. The value of N_p can be set using a development set. The default value is 1.

The last component is the subgroup identifica-



Enter a discussion thread URL (The sources supported by the online demo are: politicalforum.com, createdebate.com, and groups.google.com)

URL: http://www.createdebate.com/debate/show/should_abortion_be_banned T 1 N 1

Partitioning Algorithm: ● FEC L 5 or ○ DM K -1 Mode: ○ Optimal ● Approximate

4 subgroups detected

Subgroup 1 Members
khaOs
asdf789
wolfbite
dtrimble
lawnman
nattyc
iamdavidh
DaddySays
Prevaricate
Banshee

Figure 2: The web interface for detecting subgroups in discussions

tion component. This component provides implementations for two signed network partitioning algorithms. The first one is a greedy optimization algorithm that is based on the principals of the structural balance theory. The algorithm uses a criterion function for a local optimization partitioning such that positive links are dense within groups and negative links are dense between groups. The algorithm is described in detail in (Doreian and Mrvar, 1996). The second algorithm is FEC (Yang et al., 2007). FEC is based on an agent-based random walk model. It starts by finding a sink community, and then extracting it from the entire network based on a graph cut criteria that Yang et al. (2007) proposed. The same process is then applied recursively to the extracted community and the rest of the network.

3 Implementation Details

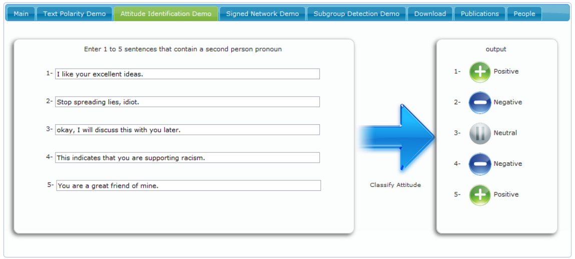
The system is implemented in Perl. Some of the components in the processing pipeline use external tools that are implemented in either Perl, Java, or Python. All the external tools come bundled with the system. The system is compatible with all the major platforms including windows, Mac OS, and all Linux distributions. The installation process is very straightforward. There is a single installation script that will install the system, install all the dependencies, and do all the required configurations. The installation requires that Java JRE, Perl, and Python be installed on the machine.

The system has a command-line interface that provides full access to the system functionality. The command-line interface can be used to run the whole pipeline or any portion of it. It can also be used to access any component directly. Each component has a corresponding script that can be run separately. The input and output specifications of each component are described in the accompanying documentation. All the parameters that control the performance of the system can also be passed through the command-line interface.

The system can process any discussion thread that is input to it in a specific XML format. The final output of the system is also in XML format. The XML schema of the input/output is described in the documentation. It is the user responsibility to write a parser that converts an online discussion thread to the expected XML format. The system package comes with three such parsers for three different discussion sites: www.politicalforum.com, groups.google.com, and www.createdebate.com.

The distribution also comes with three datasets (from three different sources) comprising a total of 300 discussion threads. The datasets are annotated with the subgroup labels of discussants. Included in the distribution as well, a script for generating a visualization of the extracted signed network and the identified subgroups.

AttitudeMiner also has a web interface that demonstrates most of its functionality. The web in-



Enter 1 to 5 sentences that contain a second person pronoun

1- I like your excellent ideas.

2- Stop spreading lies, idiot.

3- okay, I will discuss this with you later.

4- This indicates that you are supporting racism.

5- You are a great friend of mine.

output

1- Positive

2- Negative

3- Neutral

4- Negative

5- Positive

Figure 3: The web interface for identifying attitudinal sentences and their polarity

terface is intended for demonstration purposes only. No webservice is provided. Figure 2 and Figrue 3 show two screenshots for the web interface.

4 System Performance

In this section, we give a brief summary of the system performance. The method that generates the extended polarity lexicon that is used for word polarity identification achieves 88.8% accuracy as reported in (Hassan and Radev, 2010). The attitude identification component distinguishes between attitudinal and non-attitudinal sentences with 80.3% accuracy, and predicts the signs of attitudinal sentences with 97% accuracy as reported in (Hassan et al., 2010). Our evaluation for the signed network extraction component on a large annotated dataset showed that it achieves 83.5% accuracy. Finally, our experiments on an annotated discussion showed that the system can detect subgroups with 77.8% purity. The system was evaluated using a dataset with thousands of posts labeled by human annotators.

5 Conclusion

We presented of a demonstration of a social media mining system that used linguistic analysis techniques to understand the relations that develop between users in online communities. The system is capable of analyzing the text exchanged during discussions and identifying positive and negative attitudes. Positive attitude reflects a friendly relation while negative attitude is a sign of an antagonistic relation. The system can also use the attitude information to identify subgroups with a homogeneous and common focus among the discussants. The system predicts attitudes and identifies subgroups with high accuracy.

Acknowledgments

This research was funded by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through the U.S. Army Research Lab. All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of IARPA, the ODNI or the U.S. Government.

References

Amjad Abu-Jbara and Dragomir Radev. 2011. Clairlib: A toolkit for natural language processing, information retrieval, and network analysis. In *ACL-HLT 2011-Demo*, June.

Patrick Doreian and Andrej Mrvar. 1996. A partitioning approach to structural balance. *Social Networks*.

David Elson, Nicholas Dames, and Kathleen McKeown. 2010. Extracting social networks from literary fiction. In *ACL 2010*, pages 138–147, Uppsala, Sweden, July.

Ahmed Hassan and Dragomir R. Radev. 2010. Identifying text polarity using random walks. In *ACL 2010*.

Ahmed Hassan, Vahed Qazvinian, and Dragomir Radev. 2010. What's with the attitude? identifying sentences with attitude in online discussions.

Vasileios Hatzivassiloglou and Kathleen R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *EACL'97*.

Chen Lin, Jiang-Ming Yang, Rui Cai, Xin-Jing Wang, and Wei Wang. 2009. Simultaneously modeling semantics and structure of threaded discussions: a sparse coding approach and its applications. In *SIGIR '09*.

Andrew McCallum, Xuerui Wang, and Andrés Corrada-Emmanuel. 2007. Topic and role discovery in social networks with experiments on enron and academic email. *J. Artif. Int. Res.*

George A. Miller. 1995. Wordnet: A lexical database for english. *Communications of the ACM*.

Dou Shen, Qiang Yang, Jian-Tao Sun, and Zheng Chen. 2006. Thread detection in dynamic text message streams. In *SIGIR '06*, pages 35–42.

Peter Turney and Michael Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*.

Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005a. Opinionfinder: a system for subjectivity analysis. In *HLT/EMNLP - Demo*.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005b. Recognizing contextual polarity in phrase-level sentiment analysis. In *HLT/EMNLP'05*.

Bo Yang, William Cheung, and Jiming Liu. 2007. Community mining from signed social networks. *IEEE Trans. on Knowl. and Data Eng.*