

CMSC 723

Professor: Bonnie J. Dorr

Project 2

Topic and Attitude Detection in Transcripts

Georg Apitz
George Caragea
Adam Perer

Fall 2004

1 Project Description

In this paper, we present our work which focuses on automatically discovering topics and attitudes present in conversations. By using computational linguistic techniques, we devise an initial solution to this problem. A robust solution to this problem would serve many applications. For instance, discovery of topics and attitudes could prove to be a useful summary tool. Rather than read an entire conversation, users could be initially presented with the main topics being spoken about, and the positions of each of the participants. In addition to summaries, there are also potential applications for enhanced searching. For instance, if a user was interested in finding people who support universal health care in a debate, this query could be constructed with our system in place.

We begin our paper by providing a brief review of some of the seminal works we uncovered from a literature review. Next we present our algorithms for both topic and attitude detection. Later, we demonstrate our system by displaying a sample of topics and attitudes detected in transcripts from news broadcasts. Finally, we provide an evaluation of our system by comparing the output to human-judged topics and attitudes. In the appendix we describe more details about the implementation and provide instructions how to run our application.

2 Bibliographical Review

The existing literature concerning the topic gives a good overview of the broad interest in this research question and also gives insights to existing work and applied techniques.

The work described in [2] gives an excellent overview over the different motivations for topic and attitude extraction from existing corpora. As mentioned earlier one aspect for that is customer care and a certain level of quality management for conversations with customers, another motivation are automated responses from computer systems to utterances of language and this should be in the best case a language as natural as possible. On the written and transcribed side there is a huge interest by historians in content of existing documents and here especially in connection to the search of this existing material. For example historians do not only want to find letters from X to his secretary but more specifically letters where he complains about missing money from his accounts. To be able to do this kind of information retrieval we need content information from the retrieved documents. By providing this kind of content information searches in all kinds of areas could be drastically sped up. The work of Morris and Hirst [6] describes the aspect of non-classical lexical semantic relations that are important to detect relations between words or common topics of a special group of words. They proclaim that it is not enough to just look at the word classes but also at the dependencies and relationships between the words in the particular setting.

A very extensive study concerning the topic was done by Deviller et al. [5]. They conducted a study to automatically detect and tag attitude and emotion in telephone conversation with stock brokers. The main motivation is to create a quality management tool for customer care that can

control the satisfaction of customers. Use of special tags to mark attitude and emotions. Since they are looking at telephone conversations non-verbal information can be used such as pitch and heavy breathing for example. Things that can be applied to our approach are frequency of words and lexical variance, meaning what kind of words are used (swear, praise etc.) to infer the attitude.

Also, Delannoy's paper: "What are the points? What are the stances?" [1] does not look very insightful for the topic described here it delivers and confirms some of the basic approaches one can take. He used a two step process to infer the actual discussed topics from the corpora. First he compiled a topic list, that was independent from the corpora and then he tried to correlate words from that topic list to utterances from the corpora. He mainly focuses on press articles and information retrieval whereas our environment of a talk-show calls for a more interactive approach.

Wiebe et al. explain a comparison of a corpus that is on one hand annotated with respect of opinion by humans and on the other hand by an application. The application is implemented in a way so that it can use human annotated corpora to learn from them, a feature that is in general and also for our approach helpful since it extends the flexibility of the whole system. The general procedure of the learning algorithm described in [4] is similar to what we propose, in a sense that a step by step analysis leads to more and more detailed information about the corpus. [4] uses a 5 step process for their learning architecture, where the first step is a tokenization and sentence splitting, the next step is a further tokenization based on pre-learned tokens, next they use stemmers to stem the extracted tokens. After that a shallow parser is applied which constructs the syntactic structure of the document and last a special tool is applied that annotate the occurrence of named entities such as people, organizations and locations. This research as well confirms a multi-step approach by applying one step to the data and then iterate over it. This is what we did with our transcripts.

The basic idea in [3] is to extract content information from questions to allow automated answers to these questions. This approach is interesting for us since the questions are tokenized and analyzed in respect to their meaning. For their example they use so called information extraction which extracts the most important and most meaningful parts of the questions and gives a good indication about the intention of the question. Similarly, we can use the most important words/phrases in the transcripts to infer the possible topics of a paragraph or for the whole transcript.

3 Description of the Algorithms

3.1 Topic Detection Algorithm

In our system, we define topics as single keywords that play a pivotal role in the conversation. It is possible for a corpus to have several topics. For instance, in a conversation where significant time was spent debating about Iraq, Morals and Taxes, we would expect our system to find these three keywords using our topic detection algorithm.

Our algorithm begins by calculating statistics about the conversation we wish to analyze. We assume the conversations are already in plain-text form, such as the transcripts from CNN's Crossfire, which we make use of later in our demonstration. We produce a histogram of the frequency of words

present in the corpus. In other words, we find all of the unique words in the corpus and count how many times they occur. This provides us with a first glimpse at what the popular terms were that were used in a conversation. Obviously, this will include words that are present in all conversations in the English language, such as the, be, and and so on. Therefore we needed a way to rule out some of these common words so they wouldn't interfere with the detection of topics.

Our method for achieving this feature was to make use of WordNet [11]. WordNet is a lexical database for the English language. Each word in WordNet is assigned a sense number that is based on the frequency of use in their original semantically tagged corpora. These numbers are ranked based upon how often they occurred in these documents. Although the corpora used by WordNet to generate these numbers were not specifically transcripts of conversations, they will provide us with rankings of frequently used words in typical English documents. Despite this, we believe that these rankings are extremely valuable and useful, as we believe there is still a great overlap between frequently used words in conversation versus frequently used words in written text.

Our algorithm uses these ideas to provide a ranking between the words, and choosing the top N as the conversation topics. Currently, we are using $N = 5$ for our evaluation, but if this number proves to be inappropriate, it can easily be changed to some arbitrary value. For each unique word used in the conversation we compute the ratio between the number of appearances in the whole transcript and the number of appearances in the WordNet corpora; this ratio is used for the ranking of the words, the greater the ratio, the higher the rank of the word.

Likelihood estimation techniques like the one we are implementing have a known problem with sparse data, which in our case represents the words that do not appear at all in the WordNet database. To give these words a proper ranking, several smoothing techniques exist. Some of the most commonly known are *Add-one smoothing*, *Witten-Bell Smoothing* and *Backoff Smoothing*. While the two latter ones are more accurate and can deal with n-grams in the most general case, due to the simplicity of its implementation and the intuitive approach, we chose to use *Add-one smoothing*.

This smoothing technique solves the sparse data problem by just adding a fixed positive amount (usually chosen to be one) to the counts of all words, seen and unseen. It is easy to see that this approach now assigns a more relevant rank to the unseen words, but it has the disadvantage of shifting too much probability mass toward the sparse data. For the scope of our project nevertheless, we have found even this simple technique to give satisfying results.

By normalizing the frequent words in the corpus with the frequencies in WordNet, we are able to distinguish words that were talked about a lot in the conversations, yet words that are often not a part of everyday conversations. This implies that these unusual words must have been the topic of conversation. Nevertheless, this approach makes the results of the evaluation extremely sensitive to the conventions that were made when annotating the WordNet corpora, as explained in the results section.

3.2 Attitude Detection Algorithm

The second part of our project is to detect attitudes in the conversations with regards to the topic. For example, suppose there is a conversation in which the main topic is the current president. In the debate, one person supports the leader, whereas the other participant does not. We propose that our algorithm would be able to take each person's responses about a particular topic, and decide if they believe its good, bad or neutral.

Before we can even begin to judge whether a person is for or against the topic, we first needed to isolate the parts of conversation that were specifically about the topic. Obviously, topics can change throughout conversations and we didn't want a person's support or doubt about other topics to interfere with our attitude detection algorithm. So for each topic, we only ran our attitude detection algorithm on the responses containing the topic words. Furthermore, since different participants in the conversation can have different attitudes, we had to be careful that we were only judging a particular persons responses, and not their colleagues or opponents. For this, we simply used the annotated structure of the CNN transcripts to inform us who was speaking. However, to make our system more robust and not reliant on annotation, we could incorporate our classmates projects who were working on the Speaker Turn Detection problem.

Once these segmentations described above were performed, we could then begin our attitude detection algorithm. It is based upon a theory presented by Kamps et al. [8]. In the paper, they mention an idea by Osgood et al in their book, The Measurement of Meaning [7]. Osgood suggests that adjectives describing a concept can be evaluated by comparing them to the adjectives 'good' and 'bad'. This idea lends itself to the power of WordNet, as it would be possible to compare a word to 'good' and 'bad' using the structure of the synonymy relation. The synonymy relation connects words with similar meanings, so the minimal distance between two words says something about their meaning [8]. Thus, using the distance between a word and 'good' and a word and 'bad' would give us a clue as to how good or bad the description was.

An example of this is the word 'great'. Using our measurement, it receives a distance of 37 when compared to 'good'. This contrasts the distance of 173 it receives when being compared to 'bad'. Since the distance was significantly shorter to reach 'good' than 'bad', we would classify this word as a word that supports the topic, e.g. 'The president is great'.

We calculated these values with help from a Perl package that implemented a number of measures of semantic relatedness [9]. We used this package to compute the distances between words using WordNet and a measure proposed in [10]. We computed the distance from 'good' and 'bad' for every unique word in each of the corpora we were working with.

Since the distance value returned by the mentioned package is a positive number ranging from 1 to a approximately 200. To compute the positive or negative connotation of a word X , we use the formula:

$$att(X) = \frac{1}{dist_{good}} - \frac{1}{dist_{bad}},$$

where $dist_{good}$ and $dist_{bad}$ are the distances to the words 'good' and 'bad'. For any word X :

$$att(X) \in (-1, 1)$$

Corpus 1	Corpus 2	Corpus 3
Canadians	going	Columbia
States	when	George
Kerry	telling	going
George	Waxman	said
John	Notre	when

Table 1: Topics found by the Topic Detection algorithm

The intuition behind this is that the larger the distance a word X has from 'good', the lower its positive connotation is, and the smaller distance to 'bad' it has, the bigger is the negative connotation of X . Therefore, if a word is positive, it should get a positive value closer to +1, while if it is negative, the value assigned by this algorithm should be negative, and closer to -1. A word that has a neutral connotation (does not say anything about the speaker's attitude) should get a score that might be positive or negative, but in the close vicinity of 0.

We would then take the responses from a particular person about each topic and iterate through all the words, summing up the values for all words, positive, negative and neutral. If the overall score was significantly positive, we would tag the attitude as being in support of the topic. Conversely, if the overall score was significantly negative, we would tag the attitude as negative. If the score wasn't significant in either direction, the attitude would be tagged as neutral.

4 Results

In order to test our algorithms, we used transcripts from CNN's Crossfire. We chose transcripts from this show because the participants are typically combative with each other. The people on the show often have different ideologies and argue with each other to get their point across. Throughout the course of the show, they usually debate about several different main topics. This type of conversation is a perfect domain to demonstrating the success of our algorithms, due to its consistency of various topics, each with varying attitudes.

The transcripts were taken from <http://transcripts.cnn.com/TRANSCRIPTS/cf.html> and were converted into plain text. We chose the three most recent transcripts at the time (December 1-3), without regard to content.

4.1 Evaluation of Topic Detection Algorithm

After applying our topic detection algorithm, the five top topics for each corpus are shown in table 1. Independently of our topic detection algorithm, a member of our team manually produced five keyword topics for each of three corpora. These results are shown in table 2.

It should be noted that it was a hard decision to classify an entire thirty-minute conversation

Corpus 1	Corpus 2	Corpus 3
Canada	Sex	Kerik
Bush	Abstinence	Security
Ohio	Pregnancy	Columbia
Iraq	Lies	Jackson
Kerry	Notre	Ney

Table 2: Topics found by the human

into five distinct topics. This exercise was not only useful to generate an evaluation, but also to develop an appreciation of the task-at-hand for our system.

For corpus 1, our system found one of the five topics found produced by human forces. However, an additional two were very similar (Canada versus Canadians, George versus Bush). Below, we suggest improvements of how we could immediately take into account these factors. Similarly, on both corpora 2 and 3, we also had a 20% success rate. Despite these low percentages, the initial results did seem promising, as we list potential enhancements below.

As mentioned earlier, making use of the WordNet corpora makes the results of our algorithms highly sensitive to the rules and conventions that the authors have used to create that database. For example, in the WordNet database, determinants are completely left out (e.g. the,who), while other common words, like prepositions have very low frequency count (e.g. when). Since these kind of words appear rather frequently in our transcripts, our algorithm will output a unreasonably high rank for them, in most cases including them in the topic list.

To get meaningful results, we chose to implement some heuristic rules in the attitude detection algorithm. One such rule is that we disregard words that are shorter than three characters, and we do not compute a rank for them. Although of course this rule could affect the results by not detecting some particular topics in some extreme cases, the very low probability that a two or three letter word could represent a topic made us confident that this decision was justified. We note that this is a limitation of the WordNet database rather than of our algorithms.

As with any such approach to analyze written text, the results could significantly benefit from performing some form of morphological analysis. One such example is stemming, which would cause all the inflected forms of the words to be correctly identified with the form they are stored in the WordNet database. As an illustration, our results have been negatively influenced by the fact that 'said' and 'telling' tend to appear quite often in the conversation transcripts, while WordNet only includes 'say' and 'tell'. Not being able to identify these causes our algorithm to assign them a high ranking, and even include them as topics. This could be one direction of future refinement of our implementation.

Another way morphological analysis could improve the quality of our results would be to correctly identify and decompose contracted words such as l'm, we'll and let's. Most of these do not appear in WordNet as well, therefore allowing the algorithm to rank them would not be appropriate. This has justified the introduction of another heuristic rule, disregarding words that contain the apostrophe.

Topic	Begala	Blackwell	Carlson	Cohen	Mowbray
Canadians	N		N		+
States			N	+	+
Kerry		N	+		
George			N	+	
John	N		+		

Table 3: Attitudes found by the attitude detection algorithm - Corpus 1

The introduction of this rule could prevent topics such as al'Quaeda to be detected, thus making this issue another opportunity for future improvement.

A known NLP issue is the recognition of named entities, i.e. the ability to correctly recognize and identify proper names like John Kerry as denoting one entity rather than two separate words. Our project has this limitation too, therefore in this example causing John and Kerry to be ranked independently. Since both words are proper nouns, they will probably get high rankings (neither is included in WordNet), and there is a chance both will find their way in the top ranked topics.

This has an interesting consequence: since the same problem arises if the name George Bush appears in the transcript, both the first and last name are ranked independently. Nevertheless, since we chose to ignore capitalization when checking the WordNet frequency database, Bush will get a much lower ranking than George, ('bush' being a common noun, that appears in the WordNet corpora) thus making the first name much more likely to appear as a topic. This is an interesting open question and an active area of research, which is outside the scope of the current project.

4.2 Evaluation of Attitude Detection Algorithm

Our attitude detection algorithm was also subject to evaluation. We generated attitudes for each participant for every topic they spoke about. Our algorithm returned numerical results calculated from the summation described in Section 3.2. If the number returned was greater than 0.5, we considered it a positive attitude. If the number was less than -0.5, we considered it negative. If it fell in between these thresholds, we considered it neutral. tables 3, 4 and 5 show the results. Note that in the presented tables, '+' implies a positive attitude, '-' implies negative, and 'N' implies neutral. An empty spot implies the speaker did not make any remarks about the given topic.

Human results were also generated so that we could conduct a full evaluation. The human produced a positive, negative and neutral rating based upon the same segments of text the algorithm was using to produce its score. Therefore, the human used no more or no less context than what our algorithm was fed. Also, due to problems with our topic detection algorithm, some topics found had no clear way to be judged as positive or negative in any context. For example, it is unclear if the topic when could ever be judged in a positive or negative manner. Due to this, the topics that fell into this category were simply assigned neutral opinions for all of the speakers who spoke about them. The results are displayed in tables 6, 7 and 8.

Topic	Carville	Falwell	Ireland	Wood
going	N	N	N	N
when	+	N		N
telling	N	N	N	
Waxman	N			N
Notre	+			

Table 4: Attitudes found by the attitude detection algorithm - Corpus 2

Topic	Carville	Liddy	Norton	Pence
Columbia	+	+	N	
George	N	N		N
going	N	N	N	N
said	N	+	N	N
when	+			N

Table 5: Attitudes found by the attitude detection algorithm - Corpus 3

Topic	Begala	Blackwell	Carlson	Cohen	Mowbray
Canadians	N		-		N
States			+	N	N
Kerry		-	-		
George			N	+	
John	N		-		

Table 6: Attitudes found by the human - Corpus 1

Topic	Carville	Falwell	Ireland	Wood
going	N	N	N	N
when	N	N		N
telling	N	N	N	
Waxman	-			N
Notre	-			

Table 7: Attitudes found by the human - Corpus 2

Topic	Carville	Liddy	Norton	Pence
Columbia	+	-	N	
George	-	N		-
going	N	N	N	N
said	N	N	N	N
when	N			N

Table 8: Attitudes found by the human - Corpus 3

When comparing the results between our system and the human, our system was correct 62.5% of the time. It has relatively low success in Corpus 1 (4 of 12 correct), but was more impressive in Corpus 2 (10 of 13 correct) and in Corpus 3 (11 of 15 correct).

There are many results to speculate on why these results were inconsistent. Osgood's [7] strategy of comparing words to 'good' and 'bad' is interesting, but it probably makes little sense to apply this formula to all words. This strategy should probably only be applied to adjectives, as the relationship of a specific noun or verb to 'good' probably provides very little meaning. By tagging words by their part-of-speech, we could have only summed up the words describing the topic. It would be interesting to investigate the improvement this affect would have in future work.

Another possible reason for inconsistent results is due to the limited number of words our algorithm could use to judge its attitude. The algorithm only received responses of conversations that contained the topic keyword. However, it is clear that attitudes could be expressed about a topic without explicitly mentioning the topics name, or by referring to it as a pronoun or nickname. Providing more relevant context to the system would give it the potential to have more useful predictions.

An interesting finding is that our system judged no found attitudes to be negative. In fact, only one summation returned by our system was negative, albeit insignificant. We were very surprised by this result. We speculate this may be due to the structure of words in WordNet. We suspect that the word 'good' may be more central in WordNet's representation of words than 'bad'. Thus, a random word on average would have a shorter distance to 'good' than 'bad'. Further investigation, perhaps by visualizing the WordNet hierarchy, would provide evidence as if this was the case or not.

5 Conclusion

For this project, we developed algorithms for obtaining topics and attitudes present in conversations. Our algorithms were constructed based on insights gained during class and present in related literature. However, it is clear that they still need much improvement in order to provide reliable results. The process of producing topics and attitudes by the independent human for our evaluation was quite challenging in itself, which leads us to believe that is certainly not an easy problem to solve. However, given more time, resources and experience in the field, we believe our algorithms could be

much improved.

An initial concern we had was due to the fact that we used word frequencies present in WordNet to determine the unique words in the conversation. However, the WordNet word frequencies were generated from a corpus of written text, not spoken text, and this may have subtle implications in the results. Furthermore, we ranked topics by how often a unique word would occur. However, it is obvious that participants could be speaking about a topic without referencing it by the same name. Our system does not recognize what the pronouns reference and what synonyms may exist. One enhancement that could help our system would be to implement stemming. For instance, one of the corpora focused a lot about Canadian themes. Words such as Canada, Canadians, Canadian, Canada's, etc., were all grouped as separate topics. Stemming would have perhaps solved this problem and allowed the true topic of Canada to present itself.

The attitude detection algorithm certainly had its faults too. Using Osgood's [7] strategy of comparing words to 'good' and 'bad' provided us an initial strategy to gauge attitudes, but a complex problem such as this requires a more sophisticated solution. This strategy should first of all only be applied to adjectives, as the relationship of a specific noun or verb to 'good' probably provides very little meaning. However, determining part-of-speech was outside the scope of our project, so we decided to perform the distance measurement on all the words of each sentence. This seems to create a lot of noise and consequently, our algorithm produced erratic results. The attitude detection algorithm also suffered from a scoping problem. It would only judge attitudes based on the responses that contained the topic keyword. It is clear that attitudes may have been expressed in responses outside of this set, thus not giving the algorithm an accurate representation.

Completing this project was both intellectually challenging and fun. This was the first effort by all of the authors to conduct a serious NLP research effort. It allowed us to witness firsthand the typical challenges faced by researchers in this field. Overall, we were pleased to work on such an assignment as it provided us further appreciation for the impressive work that's been done in the field of natural language processing.

A Implementation details

This project was mainly implemented in Java, with a little help from Perl to automate some of the tasks.

One of the first steps in working with transcripts is, that they need to be parsed in some sense and structured in a way that they can be processed further. Several approaches are possible for this task, the transcript could be treated with part-of-speech tagging and then processed further. For this project we decided to use a two step approach in structuring the transcripts which did not involve POS-tagging for reasons of time and feasibility in the framework of the project. In the first step we process the transcripts and organize them into a hashmap according to their topic. The topics are found using the algorithm described in section 3. The second step of the organization of the transcripts takes this hashmap and parses it according to topic and speaker creating another hashmap that contains what each speaker said to each of our prioritized topics. For the attitude detection this hashmap is used to find speaker attitude toward certain topics. The whole process is done in real-time using a pre-computed database as described in the following paragraph.

The reason for using a pre-computed database was that the most intensive computing task required was estimating the distance induced by the synonymy relation between a word and the adjectives 'good' and 'bad'. These distances were required for all the words in our corpora, because they serve as the input to the attitude detection algorithm.

To estimate these distances, we have used a Perl package called `Wordnet::Similarity` which is freely available [9]. On a reasonable machine, this process takes about 10-15 seconds per word pair, this is a total of ca. 15 hours for our corpora, thus making any attempt for an on-line computation completely unfeasible. We have therefore chosen to run this as an off-line computation, and creating a database with all unique words in all the transcripts we have analyzed and, their estimated WordNet distance to the words 'good' and 'bad'. The implementation for this off-line process can be found under the `src/attitude` subdirectory in our submission tree.

The rest of the computation is completely implemented in Java, and can be found under the `src/` subdirectory. The main class here is `TopicDetection`, and it should be run using one of the three provided transcripts as a command line argument, for example:

```
> java TopicDetection ../transcripts/12_01_crossfire.txt
```

The output includes the top 5 topics detected, as well as a list of the topics each speaker has touched and a floating point number which describes the attitude of that speaker about the respective topic.

This part of the implementation requires that the WordNet database with word frequencies is in some specified path (currently hardcoded as `../WNetFiles/cntlist`); the output from the attitude detection offline computation is also required to be available (in the current version the paths are hardcoded to `../attitude/good_distance.txt` and `../attitude/bad_distance.txt`

References

- [1] *"What are the points? What are the stances?"*
Delannoy, J.-F.
Workshop on Human language technology, Conf of the Assn. for Comptl. Linguistics (ACL)
Toulouse, France, July 2001
- [2] *Emerging Language Technologies and the Rediscovery of the Past: A Research Agenda*
Crane, G, Batsheva, K.
May 2003
http://www.ercim.org/publication/workshop_reports.html
- [3] *Using natural language questions in information systems*
Ekeklint, S
GSLT - Graduate School of Language Technology, Växjö university, Sweden
2004
- [4] *Recognizing and Organizing Opinions Expressed in the World Press.*
Wiebe, J. et al.
200 AAAI Spring Symposium on New Directions in Question Answering
- [5] *Annotation and detection of emotion in a task-oriented human-human dialog corpus*
Devillers L., Vasilescu I., and Lori Lamel
SLE workshop, Edinburgh, Dec 2002
- [6] *Non-Classical Lexical Semantic Relations*
Morris J. and Graeme Hirst
Computational Lexical Semantics Workshop at HLT-NAACL 2004
- [7] *The Measurement of Meaning*
Osgood, C. E., G. J. Succi, and P. H. Tannenbaum
University of Illinois Press, Urbana IL, 1957
- [8] *Using WordNet to measure semantic orientations of adjectives*
J. Kamps, M. Marx, R.J. Mokken, and M. de Rijke
Proceedings of the Fourth International Conference on Language Resources and Evaluation
(LREC 2004), Volume IV, pages 1115-1118, 2004
- [9] *WordNet::Similarity*
Siddharth Patwardhan
<http://wn-similarity.sourceforge.net/>
- [10] *Using information content to evaluate semantic similarity*
Resnik P.

In Proceedings of the 14th International Joint Conference on Artificial Intelligence, pages 448-453, Montreal, 1995

[11] *WordNet: An on-line lexical database*

Miller, G. A.

International Journal of Lexicography, 3(4):235–312. Special Issue. 1990.