Computing Polarity in Sentiment Analysis Applications

Andrew Latterner

Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

12/07/13

Outline

Outline

Natural Language Processing

Natural Language Processing is a field in which algorithms are implemented in a way that human input can be parsed and analyzed.
This can include:

- Analyzing written text
- Parsing vocal input

Sentiment Analysis

Sentiment Analysis is a sub-field within Natural Language Processing that aims to:

- Detect sentiment in text
- Classify detected sentiment according to its polarity (e.g. happy or unhappy)

The polarity of an instance of text is also known as its *semantic orientation*.

Semantic Orientation Example

Parsing semantic orientation from simple sentences is relatively simple.

The film was wonderful.

We see:

- A large amount of neutral words
- A positive word (wonderful)

Therefore, we can easily conclude that the sentence is likely positive.

Difficulties in Semantic Orientation

When we introduce more complex elements within sentences, parsing orientation becomes difficult.

The film wasn't terrible by any means.

We see:

- Many neutral words
- A negative word (terrible)

However, one would not classify the above sentence as being negative.

Outline

Modifiers

Modifiers are elements that are introduced to a sentence that modify its semantic orientation. These include:

- Intensifiers
- Negators
- Irrealis Statements

Intensifiers

Intensifiers modify a word's orientation by increasing or decreasing a magnitude.

Two major types of intensifiers are amplifiers and down-toners:

- *Amplifiers* increase the magnitude of semantic orientation
- *Down-toners* decrease the magnitude of semantic orientation

Amplifiers

Amplifiers tend to increase the magnitude of the modified word's contextual polarity.

The film was very good.

In regards to this sentence, we see that:

- The word, *good*, has a positive semantic orientation
- The amplifier, *very*, increases *good's* semantic orientation

Down-toners

Down-toners decrease the magnitude of the modified word's contextual polarity.

The film was slightly boring.

In this example, we see that:

- The word, *boring*, has a negative orientation
- The down-toner, *slightly*, decreases the magnitude of *boring's* semantic orientation

Negators

Negators invert the semantic orientation of a word.

The film was not good.

In this example, we see that:

- The word, *good*, has a positive orientation
- The negator, *not*, inverts the word good to be negative
- The overall semantic orientation of the sentence is negative

Irrealis Statements

Irrealis statements are statements that do not reflect reality.

This movie could have been amazing.

In this example:

- The sentence is being primarily influenced by the positively oriented word, *amazing*
- The phrase, *could have*, implies that the sentence does not actually reflect reality
- Making any real conclusions regarding the polarity is difficult

Outline

1. Background
2. Modifiers
3. Lexicon Based Approach
4. Classification Based Approach
5. Results
6. Conclusion

Lexicon Based Approach

Taboada et al., of Simon Fraser University designed a lexicon based approach that entails:

- Using a *lexicon*, or a mapping between words and emotional ratings, to parse the orientation of a word or phrase (e.g. *Happy* $\rightarrow$ 3)
- Handling modifiers in accurate ways by analyzing sentence structure

The *lexicon* is created by compiling a list of subjective words and recording their projected orientation in the majority of cases (from -5 to 5).

Intensifiers in the Lexicon Based Approach

In the lexicon based approach, intensifiers are handled multiplicatively.

The film was very good.

So to find the orientation of the word, *good*, we:

- Find *good* in the lexicon and receive its orientation (e.g. 3)
- Apply the word *very* to that orientation (e.g. 200% of original polarity)

The new orientation would be: $(3 \times 200\%) = 6$.

Intensifiers in the Lexicon Based Approach cont.

For down-toners:

The film was somewhat uninteresting.

In this example, we would:

- Look up *uninteresting* in the lexicon and record its orientation (e.g. -2)
- Apply the down-toner, *somewhat*, to that orientation (e.g. 50% of original polarity)

The new orientation would be: $(-2 \times 50\%) =$ -1.

Negators in the Lexicon Based Approach

In the lexicon based approach to sentiment analysis, negators are found by searching backwards until:

- A negator is found. The negator is then applied to the words in that clause
- A punctuation mark or stop word (e.g. but) is found. This implies that there is no negator in that clause

Applying Negators in the Lexicon Based Approach

When applying a negator to a word, just inverting the polarity is not always accurate.

Consider two words and their respective orientations:

- Terrific $\rightarrow$ 3
- Terrible $\rightarrow$ -3

Flipping *terrific's* polarity to -3 would not be accurate given that *not terrific* and *terrible* are not equally negative.

Negators in the Lexicon Based Approach cont.

In the lexicon based approach, negators are applied in a less severe manner. The resulting polarity will be flipped, but to a lesser degree.

The actor was not terrible.

In the above example, we see:

- A negatively oriented word, *terrible* with a polarity of -3
- A negator, *not*, being applied to the word

The new polarity of the phrase would be -3 $\rightarrow$ 1.

Irrealis Statements in the Lexicon Based Approach

Finding irrealis statements entails looking for specific words associated with irrealis statements. This entails:

1. Looking for words associated with irrealis statements within the sentence or phrase (e.g. could)
2. Nullifying the polarity of words within the phrase where irrealis words are located

Outline

Classification Based Approach

Another approach to parsing sentiment from text is through classification based approaches. An algorithm designed by Wilson et al., of University of Pittsburgh entails:

- Parsing the text instance for certain pieces of data regarding the structure and contents of the text
- Using machine learning techniques to create a model that can classify a piece of text

Classification

Classification is the process of categorizing input based on past experiences with similar input.

Text classifiers are algorithms that:

- Take in a set of data by which to create classification rules from
- Provide the ability to classify new text instances based on those rules

Classification in Two Steps

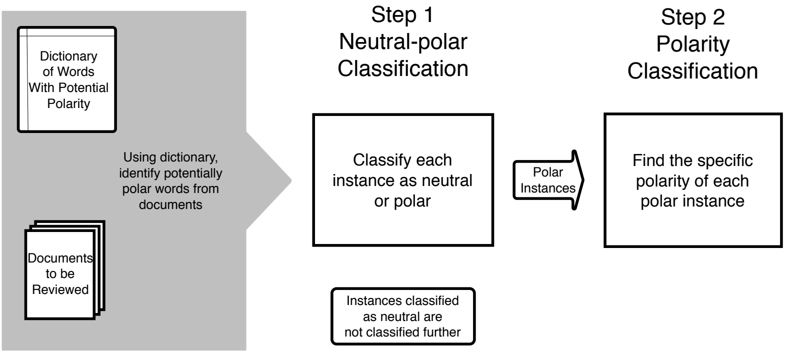

Figure : Based on Wilson et al., 2009

Neutral-Polar Classification

In the first step, the algorithm attempts to classify a word based on whether it is neutral or polar.

1. *His venomous personality was terrible*
2. *The venomous snake was yellow*

In the first example, the word *venomous* is being used in a negative context.
In the second, *venomous* is being used in a neutral context.

Types of Features

In order to classify the word, the algorithm requires a list of features of the text from which to classify. These include:

- The part of speech of the word (e.g. adjective)
- How likely a word is to be subjective
  - Strong subjective if it is subjective most of the time
  - Weak subjective if it is subjective sometimes
- The parts of speech and polarity of words surrounding the word

Dependency Tree

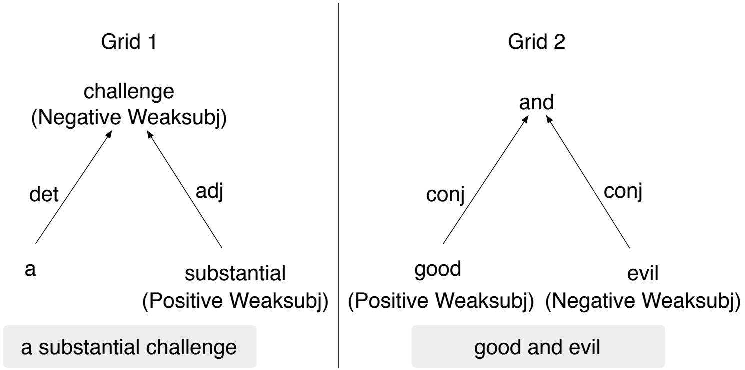

Figure : Based on Wilson et al., 2009

Polarity Classification

The second step entails classifying polar words on whether the words are positive or negative.
The features for this process are as follows:

- The features from the neutral-polar classification step
- If the word is negated
- If there are modifiers influencing the word

Outline

Results

To test the models, one must:

1. Create a set of text instances that have words with manually calculated polarities
2. Use the model to recompute the words' predicted sentiment orientation
3. Compare the computed list versus the precalculated list

F-Measure

An F-Measure is a measure of accuracy that ranges from 0 to 1. This takes into account:

- *Precision*, or the percentage of true positives per true and false positives
- *Recall*, or the percentage of true positives per true positives and false negatives

$F = 2 \times \frac{precision \times recall}{precision + recall}$.

Lexicon-Based Approach Setup

For the lexicon-based approach, the following experimental design was used:

- The data to test against was from a series of reviews (movies, music, etc.) from the website Epinion
- The algorithm was testing a setup with full modifier support and non-modifier support
- The accuracy metric used was raw accuracy

Lexicon-Based Approach Results

After running the experiment, the following results were found:

- The full modifier support system outperformed the non-modifier support by around 1% accuracy
- The full modifier system had an average accuracy of 80%

Classification-Based Approach Design

For the classification-based approach, the following experimental design was used:

- The data to test against was around 10,000 sentences from the Multi-Perspective Question Answering (MPQA) corpus
- The classification algorithms used included four distinct algorithms, some include Support Vector Learning and Boosting
- The researchers compared the use of the two-step approach versus a single step approach
- The accuracy metric used was F-Measure

Classification-Based Approach Results

The results of the experiment included:

- Using the boosting algorithm (BoosTexter) the single-step algorithm tended to outperform or perform equally to the two-step algorithm
- When using the Support Vector Learning model (SVM), the two-step did show promising results

Comparison of Approaches

The difficulty in comparing Lexicon-based and Classifier-based approaches is that they must be compared using similar data-sets. Gonçalves et al. compared these approaches using the following design:

- The data-set was a series of Twitter messages reflecting opinions on events, reviews, and sports
- The algorithm used for the Lexicon-based approach was Linguistic Inquiry and Word Count (LWIC)
- The algorithm used for the Classification-based approach was SentiStrength

Comparison Results

The following results were found:

- The F-measure for LIWC (Lexicon-based) was 0.698
- The F-measure for SentiStrength (Classification-based) was 0.765
- The Classification-based algorithm seemed to outperform the Lexicon-based algorithm

Outline

1. Background
2. Modifiers
3. Lexicon Based Approach
4. Classification Based Approach
5. Results
6. Conclusion

Conclusions

- When using the Lexicon-based approach, full modifier support seemed to yield an improvement in accuracy
- The results of experimenting with the Classification-based approach showed that the two-step approach's success was reliant on the classification algorithm used
- When comparing a Lexicon-based approach to a Classification-based approach, the Classification approach tends to perform better

Questions

References

Gonçalves, Pollyanna and Araújo, Matheus and Benevenuto, Fabrício and Cha, Meeyoung.
Comparing and combining sentiment analysis methods.
In Proceedings of the first ACM conference on Online social networks, pages 27–38, Boston, Massachusetts, USA, 2013.

Taboada, Maite and Brooke, Julian and Tofiloski, Milan and Voll, Kimberly and Stede, Manfred
Lexicon-based methods for sentiment analysis
Comput. Linguist. June 2011, pages 267–307

Wilson, Theresa and Wiebe, Janyce and Hoffmann, Paul
Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis
Comput. Linguist., September 2009, pages 399–433

Data Comparing Classifiers

	Acc	Pos-F	Neg-F	Neutral-F
BoosTexter				
two-step	74.5	47.1	57.5	83.4
one-step	74.3	49.1	59.8	82.9
SVM				
two-step	73.1	46.6	58.0	82.1
one-step	71.6	43.4	51.7	81.6