Sentiment Analysis and its practice in Twitter corpus

Min Li

Shen, Dan, and Catherine Baudin. "Sentiment Analysis in Practice." ICDM’ 11 Tutorial
Outline

• Introduction
• Sentiment Analysis Approaches
• Applications in Twitter corpus
Introduction

• Sentiment analysis (also known as opinion mining) refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials.

• Facts vs. Opinions

• Opinions have targets (objects and their attributes) on which opinions are expressed
Introduction

• <object, attribute, orientation, opinion holder, time>
  – Some may be implied due to pronouns, context
  – not all five are needed

• Some difficulties:
  – Direct opinion vs. Comparative opinion
  – Implicit opinion vs. facts (neutral)
SA Approaches

• A hierarchical method
  Step 1: Sentiment identification(subjective identification)
  Step 2: Sentiment orientation classification
• Document-level SA (product reviews, blogs)
• (Sentence-level SA )
• (Attribute-level) (extract the object attributes)
  More granular, more difficult.
Document level Sentiment Analysis

• Tasks: identify if the document expresses opinions and if yes classify the document into positive or negative based on the overall sentiments expressed by opinion holders

• Assumptions:
  – The document is opinionated on a single object
  – The opinions are from a single opinion holder
A simple method-Counting Opinion Words

• Opinion/polarity words: dominating indicators of sentiments, especially adjectives, adverbs, and verbs.
  – Eg. “I absolutely love this camera. It is amazing!”.

• Assign orientation score (+1, -1) to all words

• The orientation score of the document is the sum of orientation scores of all opinion words found

• How to get opinion words:
  – Automatic finding opinion words [Turney, ACL2002]
  – Bootstrapping, dictionary based or corpus word clustering
  – Use some preextracted ones: lexicon from General Inquirer MPQA, SentiWordNet
Automatically Finding Opinion Words [Turney, ACL2002]

- Data: reviews from epinions.com on automobiles, banks, movies and travel destinations

- Step 1. perform part of speech (POS) tagging and extract phrases containing adjectives and adverbs based on manually specified patterns

**Table 1. Patterns of POS tags for extracting two-word phrases**

<table>
<thead>
<tr>
<th>First word</th>
<th>Second word</th>
<th>Third word (Not Extracted)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. JJ</td>
<td>NN or NNS</td>
<td>anything</td>
</tr>
<tr>
<td>2. RB, RBR, or RBS</td>
<td>JJ</td>
<td>not NN nor NNS</td>
</tr>
<tr>
<td>3. JJ</td>
<td>JJ</td>
<td>not NN nor NNS</td>
</tr>
<tr>
<td>4. NN or NNS</td>
<td>JJ</td>
<td>not NN nor NNS</td>
</tr>
<tr>
<td>5. RB, RBR, or RBS</td>
<td>VB, VBD, VBN, or VBG</td>
<td>anything</td>
</tr>
</tbody>
</table>

- e.g. extract two consecutive words if the first word is adjective, the second is a noun and the third (which is not extracted) is anything: “this camera produces beautiful pictures”
Pointwise Mutual Information (PMI)

• Step 2. estimate the orientation of each extracted phrase using the PMI measure
  
  – PMI is the amount of information that we acquire about the presence of one of the words when we observe the other

\[
PMI(\text{term}_1, \text{term}_2) = \log_2 \left( \frac{Pr(\text{term}_1 \land \text{term}_2)}{Pr(\text{term}_1) Pr(\text{term}_2)} \right).
\]

  – The opinion orientation (OO) of a phrase is computed based on its association with the positive reference word “excellent” and its association with the negative reference word “poor”

\[
oo(\text{phrase}) = PMI(\text{phrase}, \text{“excellent”}) - PMI(\text{phrase}, \text{“poor”}).
\]
PMI (cont.)

- Estimate probabilities with number of hits of search query
  - For each search query, search engine returns the number of relevant documents to the query, which is the number of hits

- Turney used AltaVista which had a NEAR operator, which constrains the search to documents that contain the words within 10 words of one another, in either order

\[
oo(\text{phrase}) = \log_2 \left( \frac{\text{hits(phrase \ NEAR \ "excellent")} \cdot \text{hits("poor")}}{\text{hits(phrase \ NEAR \ "poor")} \cdot \text{hits("excellent")}} \right)
\]

- Examples:
  - "low fees", "JJ NNS", 0.333
  - "unethical practices", "JJ NNS", -8.484
  - "low funds", "JJ NNS", -6.843
Step 3. Compute the average semantic orientation of all phrases in the review

- Classify as positive (recommended) or negative (not recommended) based on the sign of the average

- Final classification accuracy:
  - Automobiles – 84%
  - Banks – 80%
  - Movies – 66%
  - Travel destinations – 71%

Note: Recent variations use more than two words to determine the orientation
Rule based method

• Subject {like | adore | want | work} TOPIC positive, e.g.
  – “I like the old camera”
• TOPIC GAP_0_3 NOT work negative, e.g.
  – “The new search still does not work”
• Can handle negation
• Expensive and limited
Supervised Learning

• A machine learning technique: find patterns in known examples and apply to new documents
  – Training and testing examples
  – A set of data features to represent documents

• Popular methods may apply
  – Naïve Bayes
  – SVM
  – Logistic Regression
Feature extraction

• Terms and their frequency
  – Unigram and more general n-grams
  – Word position information
  – Term frequency (tf-idf weighting)

• Part of speech tags (adj. more important)

• Syntactic dependency
  – Syntactic parsing tree (OpenNLP)

• More comprehensive attributes, see [Pang and Lee, FATIR2008]

• Domain Adaptation, see [Aue & Gamon, RANLP2005; Blitzer et al, ACL 2007; Yang et al, TREC2006]
Sentiment Analysis in Twitter

• Short and more ambiguous
  (more casual, sarcasm)
• Lack contexts
• Prone to introduce more targets
  (Need more work on determining the object)
• Additional information provided
  (follower-followee relations, related tweets, hashtags)
An example


• A nice procedure for Sentiment Analysis on Twitter Corpus
Background

• Problems in state-of-art literature:
  – Adopt a target – independent strategy, assign irrelevant sentiments to given target (about 40% error introduced)
    e.g. "People everywhere love Windows & vista. Bill Gates“
    Windows 7 is much better than Vista!
  – Only take the tweet to be classified into consideration (ignore related tweets)

• Improve the framework by:
  – Incorporating target-dependent features
  – Take relative tweets into consideration
Approach Overview

- Subjective classification (subjective vs neutral)
- Polarity classification (positive or negative about the target)
- Graph based optimization
Approach Overview

• Preprocessing
  Tweet normalization, POS tagging, word stemming, syntactic parsing

• Target-independent features
  1. Content features, including words, punctuation, emoticons, and hashtags (hashtags are provided by the author to indicate the topic of the tweet).
  2. Sentiment lexicon features, indicating how many positive or negative words are included in the tweet according to a predefined lexicon.
Approach Overview

• Extended targets extraction

Express sentiments about a target by commenting not on the target itself but on some related things of the target

e.g. I am passionate about Apple technologies

  – Adding mentions co-referring to the target as new extended targets by co-reference resolution tool
  – Identify the top K nouns and noun phrases which have the strongest association with the target. (PMI)
  – Extracting head nouns of all extended targets whose PMI are above a threshold (Apple Technologies)
Approach Overview

• Target-dependent features
  Use syntactic parse tree with rules to generate binary features
  Handling word sequence, negation, comparison here.

  Example features:
  love_arg for I love iPhone
  arg_v_well for iPhone works better.
  (adj)_cp_arg for GalaxyS4 is better than iPhone

• Binary feature indicating whether or not the tweet contains at least one of the above target-dependent features (Overcome sparcity)
Approach Overview

• Graph based optimization
  1. retweets
  2. Tweets containing the target and published by the same person
  3. Tweets replying to or replied by the tweet to be classified.

\[ p(c \mid \tau, G) = p(c \mid \tau) \sum_{N(d)} p(c \mid N(d)) p(N(d)) \]

N(d) is a specific assignment of Sentiment labels to all immediate neighbors

• Iterative estimation method
Experiments

• Corpus
  • Selected 5 popular queries of these kinds:
    \{Obama, Google, iPad, Lakers, Lady Gaga\}.
    Finally obtain 459 positive, 268 negative, 1212 neutral tweets

• human error rate
  100 tweets labeled by two people
  86 the same 14% human error rate
Experiments

• Subjectivity classification result
  – 727 subjective and 727 neutral (randomly chosen)
  – 10-fold cross-validation

<table>
<thead>
<tr>
<th>Features</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content features</td>
<td>61.1</td>
</tr>
<tr>
<td>+ Sentiment lexicon features</td>
<td>63.8</td>
</tr>
<tr>
<td>+ Target-dependent features</td>
<td>68.2</td>
</tr>
<tr>
<td>Re-implementation of (Barbosa and Feng, 2010)</td>
<td>60.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Target</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact target</td>
<td>65.6</td>
</tr>
<tr>
<td>+ all extended targets</td>
<td>68.2</td>
</tr>
<tr>
<td>- co-references</td>
<td>68.0</td>
</tr>
<tr>
<td>- targets found by PMI</td>
<td>67.8</td>
</tr>
<tr>
<td>- head nouns</td>
<td>67.3</td>
</tr>
</tbody>
</table>
Experiments

• Polarity Classification Result
  – 268 positive tweets and 268 negative ones

<table>
<thead>
<tr>
<th>Features</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content features</td>
<td>78.8</td>
</tr>
<tr>
<td>+ Sentiment lexicon features</td>
<td>84.2</td>
</tr>
<tr>
<td>+ Target-dependent features</td>
<td>85.6</td>
</tr>
<tr>
<td>Re-implementation of (Barbosa and Feng, 2010)</td>
<td>83.9</td>
</tr>
</tbody>
</table>
Experiments

• Graph-based Optimization

Corpus info:

<table>
<thead>
<tr>
<th>Relation type</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Published by the same person</td>
<td>41.6</td>
</tr>
<tr>
<td>Retweet</td>
<td>23.0</td>
</tr>
<tr>
<td>Reply</td>
<td>21.0</td>
</tr>
<tr>
<td>All</td>
<td>66.2</td>
</tr>
</tbody>
</table>

Results:

Overall accuracy increases from 66.0% to 68.3%
Conclusion

• Learn from the entire framework
• Possible future works:
  – Relation types within extended targets
  – Introduce more targets besides extended targets
    Eg. I love iPhone; retweets: I don’t think so
  – More relations between accounts
• Thank you for your time
• Q & A