For Academics

Is the code open source?

Sentiment140 isn't open source, but there are resources with open source code with a similar implementation:

- Text Classification for Sentiment Analysis by Jacob Perkins
- TwitGraph by Ran Tavory
- Twitter sentiment analysis using Python and NLTK by Laurent Luce
- Twitter Sentiment Corpus by Niek Sanders

What algorithm are you using?

We are using a Maximum Entropy classifier. Read about our machine learning approach.

Where is the training data?

You can download our training data:

- Stanford link
- Google Drive link

What is the format of the training data?

The data is a CSV with emoticons removed. Data file format has 6 fields:
0 - the polarity of the tweet (0 = negative, 2 = neutral, 4 = positive)
1 - the id of the tweet (2087)
2 - the date of the tweet (Sat May 16 23:58:44 UTC 2009)
3 - the query (lyx). If there is no query, then this value is NO_QUERY.
4 - the user that tweeted (robotickilldozr)
5 - the text of the tweet (Lyx is cool)

If you use this data, please cite Sentiment140 as your source.

How was your data collected and annotated?

Our approach was unique because our training data was automatically created, as opposed to having humans manual annotate tweets. In our approach, we assume that any tweet with positive emoticons, like :), were positive, and tweets with negative emoticons, like :(, were negative. We used the Twitter Search API to collect these tweets by using keyword search. This is described in our paper.

Where is the tweet corpus for Spanish?

Unfortunately, we do not provide the Spanish data set.

What did you use to build this?

We built this using the following technologies:

- Twitter API
- Amazon EC2 (for the backend)
- Google Closure (for the JavaScript library)
- Google Visualization API (for the annotated timeline)
- Google Charts API (for the pie and bar charts)
- Google Sites (for this documentation)
- Google Spreadsheets (for our feedback form)
- Google Analytics

Want to discuss ideas?

You're welcome to discuss ideas on the Sentiment140 forum.

Do you have any project ideas?

If you are new to the field of sentiment analysis, we recommend reading the following by Pang and Lee:
Opinion mining and sentiment analysis

There are still many unsolved problems in sentiment analysis. If you're interested, you can help us by working on one of the problems below.

- **Building a classifier for subjective vs. objective tweets.** We've focused mostly on classifying positive vs. negative correctly. We haven't looked at classifying tweets with sentiment vs. no sentiment very closely.
- **Handling negation.** Words like no, not, and never are difficult to handle properly.
    - Relevant papers:
        - Isaac G. Councill, Ryan McDonald, and Leonid Velikovich. 2010. What's great and what's not: learning to classify the scope of negation for improved sentiment analysis. [pdf]
        - Potts, Christopher. 2010. On the negativity of negation. [pdf]
- **Handling comparisons.** Our bag of words model doesn't handle comparisons very well. For example, in the phrase "Stanford is better than Berkeley", the tweet would be considered positive for both Stanford and Berkeley using our bag of words model because it doesn't take into account the relation towards "better".
- **The "aboutness" problem.** Given a tweet, automatically detect if the sentiment is towards an entity.
    - Example:
        - about the term [Google]: "I love Google."
        - not about the term [Google]: "You should Google that."
    - Relevant papers:
        - Target-dependent Twitter Sentiment Classification [pdf]
- **Determine context switches.** Sometimes tweets contain two different ideas. It would be good to be able to segment these two different ideas out. Here's an example: "Just chomped my way through a massive apple, was pretty tasty. Now for work. Business revision."
- **Building an accurate parser for tweets.** Dependency parsers, like the Stanford Parser, doesn't handle ungrammatical text very well because they were trained on corpuses like the Wall Street Journal . It would be great to develop a parser that can handle informal text better.

- **Sarcasm detection.**
- **Topic classification for tweets.**
- **Tag clouds.** Given a list of positive and negative tweets, what are the most meaningful words to put in a tag cloud?
- **Applying sentiment analysis to Facebook messages.** Facebook messages don't have the same character limitations as Twitter, so it's unclear if our methodology would work on Facebook messages.
- **Internationalization.** We focus only on English sentences, but Twitter has many international users. It should be possible to use our approach to classify sentiment in other languages.

How did it start?

Sentiment140 started as a class project from Stanford University. We explored various aspects of sentiment analysis classification in the final projects for the following classes:

- CS224N Natural Language Processing in Spring 2009, taught by Chris Manning
- CS224U Natural Language Understanding in Winter 2010, taught by Dan Jurafsky and Bill MacCartney
- CS424P Social Meaning and Sentiment in Autumn 2010, taught by Chris Potts and Dan Jurafsky

Recommended Books

If you are interested in machine learning and natural language processing, you may be interested in the following books:

- Opinion Mining and Sentiment Analysis by Bo Pang and Lillian Lee. (Download)
- Foundations of Statistical Natural Language Processing by Christopher Manning and Hinrich Schuetze
- Introduction to Information Retrieval by Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze (Download)
- Speech and Language Processing by Daniel Jurafsky and James Martin

Comments

You do not have permission to add comments.