Twitter Sentiment Analysis

Sawyer Anderson, Dublin Anondson, Kanat Bektemirov, Neal Roberts

Abstract [Sawyer Anderson, Dublin Anondson, Neal Roberts]
Twitter contains a wealth of data about people, products, and events, but aggregating that data into meaningful, useful, and relevant information is challenging, particularly given the 140-character limit on tweets. This web-based application will allow users to view the overall sentiment surrounding a keyword, hashtag, or username by determining the sentiment of tweets relevant to the search and aggregating these sentiments into an easily interpreted timeline representation, with additional context pertaining to segments of the timeline, such as related news stories. This project will provide users with access to information necessary to make well-informed decisions based on public sentiment, as well as provide insight into what might have driven that sentiment.

It is expected that there will be two primary research-intensive aspects to this project. First and foremost, analyzing the sentiment of a particular tweet (that is, determining if a tweet is overall positive or negative, as relates to the relevant keywords). Second, providing time-sensitive context about a search, especially taking into account the sentiment score from a search, will require experimenting with a number of different methods to determine what is most effective.

1.0 Problem [Sawyer Anderson, Dublin Anondson, Neal Roberts]
Twitter is an online social networking and microblogging service that allows users to post 140-character “Tweets” which are then publically visible. Twitter has over 270 million users, with over 500 million Tweets sent and 2.1 billion search engine queries served per day [1]. Due to its size, Twitter contains a vast trove of data about people, companies, products, sports teams, current events, and other topics. However, as with any crowdsourced dataset, translating that data into meaningful and searchable information is difficult.

Users want to be able to not only search for Tweets relevant to their interests or needs, but also to be presented with contextual information about the sentiment surrounding their search. The lack of such an interface to the relevant information on Twitter is a hindrance to education, research, decision-making, and economic progress.
2.0 Objective [Sawyer Anderson]

The objective of this project is to create a web application that provides easy access to historical sentiment information from Twitter, and to provide context to that information, such as news articles related to changes in sentiment.

3.0 Background [Kanat Bektemirov]

3.1 Key Concepts

Sentiment analysis is the process of classifying the attitude of some text as positive, negative, or neutral. There has been a lot of research in this area, but not many with useful results. This project will perform multiple experiments using several classification techniques such as Naive Bayes, Support Vector Machines, and k-Nearest Neighbors in order to get the most accurate results.

3.2 Related Work

Some research has been done on sentiment analysis of a particular text/tweet, but not much about aggregating the data. The following are some tools and related work that provide some background information and direction to the project.

- Dan Jurafsky, Stanford Professor on sentiment analysis and natural language processing, provides an in-depth look at standard sentiment analysis techniques and terms [3]
- Natural Language ToolKit - text processing and analysis library [4]

Additionally, a number of web services exist to perform sentiment analysis on aggregated Twitter data, but most simply use pre-defined lists of binary positive and negative keywords instead of a more learnable mechanism.

4.0 Design [Sawyer Anderson, Dublin Anondson, Kanat Bektemirov, Neal Roberts]

4.1 Requirements and/or Use Cases and/or Design Goals

- Requirements
  - A randomly-sampled, sufficiently large dataset of Tweets over a specified time period
  - Each Tweet in the dataset should include, at minimum:
    - The <= 140-character raw text of the Tweet
    - The date and time stamp of the Tweet
    - The username of the Tweeting user
  - A web interface for analyzing and searching the dataset which supports search by keywords, hashtags, and users.
  - Users can specify a time range over which to search and analyze sentiment
  - Display a normalized graph of sentiment of the search over the user-specified time period
  - Provide links to news articles from the specified time period which might be relevant to or explain the displayed sentiment
Use Cases
  o Search and display
    ▪ User navigates to the web page and is presented with a search box and corresponding search button, as well as a method of specifying a time frame to search within
    ▪ User types in a string search query composed of keywords, hashtags (any term preceded by the # character), and usernames (any term preceded by the @ character), specifies a time frame (default is date of first tweet in dataset to date of last tweet in dataset) and clicks search or presses the return key
    ▪ Application runs the query against the dataset, analyzes sentiment across relevant Tweets, and searches for related news articles.
    ▪ User is presented with a graph of sentiment over the specified timeframe, as well as a list of links to related news articles from the timeframe.
Use Case: Sentiment Term Search
Author: Dublin Anondson
Primary Actor: User
Goal in context: Search a term and have a chart showcasing sentiment presented to the user.
Preconditions: Knowing the search term and ability to access the internet.
Trigger: The user enters a search term.
Scenario:
1. User: Presented with a search bar.
3. Program: Term analyzed for its sentiment score and a chart is constructed displaying the trend.
Exceptions:
2. Lack of sentiment data for a query term.
Priority: Medium
Channel to actor: Graphical User Interface (GUI)
Frequency of use: Up to many times per day
Secondary actors: The search bar
Channels to secondary actors: Input via keyboard
Open issues:
1. Analyzing sarcasm or word misspellings.
2. How far back will the sentiment be reaching for?
4.2 High Level Architecture

4.2.1 Repository Layer (Dataset and Index Storage)
For initial design and testing purposes, the dataset will need to contain a sampling of tweets related to a small set of queries/topics. We propose to begin with a dataset of roughly 50,000 Tweets.

The dataset will be indexed and stored in a form that allows for text searching and querying. That is, it will be indexed. The current candidate technology for indexing and querying the dataset is Apache’s Lucene project, which is a full-featured, open-source indexing and retrieval project.

4.2.2 Service Layer (Indexing, Analysis, and Context Lookup)
The service layer will be primarily responsible for:

1. Indexing the dataset
2. Querying the search terms and timeframe against the dataset
3. Performing sentiment analysis on the returned tweets
4. Retrieving related news articles and other context information

4.2.3 Controller Layer (Formatting and Translation)
The controller layer is the topmost layer between the application and the interface presented to the user. It is responsible for receiving user input and outputting formatted information to the front-end user interface.
Due to its ease-of-use and well-defined standards, the controller layer will be implemented as a REST API.

4.2.4 User/Web Interface

The front-end of the application, the user interface displays the information to and receives input from the user, which is communicated over the REST API. Primarily HTML, Javascript, and CSS.

4.3 Risks

<table>
<thead>
<tr>
<th>Risk</th>
<th>Risk Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twitter linguistics</td>
<td>The classifier can be manually trained on which emojis, abbreviations, and other twitter terminology are usually classified as positive or negative mood.</td>
</tr>
<tr>
<td>Handling Sarcasm</td>
<td>The classifier will be sarcasm un-aware.</td>
</tr>
<tr>
<td>Service might misrepresent sentiment on which businesses and users might rely</td>
<td>The service is not guaranteed to provide accurate results; thus, the service will explicitly outline in Terms of Service that the results are provided “as is.”</td>
</tr>
</tbody>
</table>

4.4 Tasks

1. Understand/gain background
   - Research and choose an appropriate method of scoring a Tweet’s sentiment
   - Determine the best method of including time-sensitive context information relevant to a search

2. Design
   - User interface: design the method of presenting the sentiment graph and context links in an easily-understood manner.
   - Analysis: once sufficient background research has been performed, design how and in what order analysis steps should be performed.

3. Implementation
   - Implementation will consist first of providing an interface into analysis from the randomly sampled dataset, then using more up-to-date results via Twitter’s live APIs

4. Test
   - Testing will consist of running multiple search queries over the dataset and sanity-checking the output sentiment graph and context links

5. Demonstrate
   - Success of the application will be determined by the usefulness and relevance of the presented information, as well as by the scope of analyzable queries.

6. Document
• Final documentation will include a report about analysis and context-search methods used, and the accuracy/relevance of each method.

4.5 Schedule

<table>
<thead>
<tr>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Understanding</td>
<td></td>
</tr>
<tr>
<td>2. Design</td>
<td></td>
</tr>
<tr>
<td>3. Implement</td>
<td></td>
</tr>
<tr>
<td>4. Test</td>
<td></td>
</tr>
<tr>
<td>5. Demonstrate</td>
<td></td>
</tr>
<tr>
<td>6. Document</td>
<td></td>
</tr>
</tbody>
</table>

4.6 Deliverables

• A feature-complete design document of the project
• All code written, as well as a link to the test dataset
• A final report of chosen design, architecture, tested analysis methods, and results

5.0 Key Personnel

Sawyer Anderson – Anderson is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed relevant courses in Information Retrieval and Data Mining, as well as internships at SOAPware, Inc performing web development, Microsoft working on the Cortana personal assistant, and Intentional Software Corporation building a group productivity application on a domain-specific language platform. Anderson will be primarily responsible for research, design, and implementation surrounding the analysis portion of the project.

Dublin Anondson – Anondson is a senior Computer Science major in the CSCE department at the University of Arkansas. He is currently enrolled in the relevant courses of Information Retrieval and has currently completed Algorithms and Database Management. Anondson will be responsible for the front end UI of the website as well as performing testing.

Kanat Bektemirov - Bektemirov is a senior Computer Science major in the CSCE Department at the University of Arkansas. He has completed relevant courses in machine learning, data mining, and information retrieval. He has also interned as a software engineer at places like Google, Amazon Web Services, Amazon Payments, and SOAPware. He will primarily be responsible for the sentiment analysis scoring part of the project. Bektemirov will also guide the rest of the team with relevant technology pieces that will be used throughout the project.

Neal Roberts – Roberts is a senior Computer Science major in the CSCE department at the University of Arkansas. He has completed relevant courses in information retrieval and algorithms. Roberts will assist with the research, implementation and design of the project, as well as dataset gathering and tuning.
Dr. Susan Gauch, Project Advisor - Dr. Gauch joined the University of Arkansas to become the Head of Computer Science and Computer Engineering. Her primary research field is intelligent information retrieval, personalization and web search, and ontology construction. She has worked with Freshman Engineering Program students on basic Twitter sentiment analysis in 2014.

6.0 References


