Quote-Crawler Search Engine (Final Project)
Yijie Feng

URL: http://cims.nyu.edu/~yf833/cgi-bin/retriever.cgi

1. Objective:

This project aggregates comments (quotes) about a particular subject (an organization, individual, or thing) and classifies them according to sentiment (positive or negative). A crawler aggregate data from sources on the web that are relevant to a provided query (optional), extracts quotes from these pages, and outputs a set of indexable documents from these quotes. The user interacts with a a retriever program, which will return a list of quotes (ranked by relevance) grouped as either positive or negative.

The goal of this project was to create a tool that would be useful for research or data-mining purposes. By returning results that are relevant to a particular subject grouped by sentiment, researchers can base their analysis on a more restricted, relevant set of documents. High-profile individuals that are often quoted in the news could also use such a tool to manage their public image on the web.

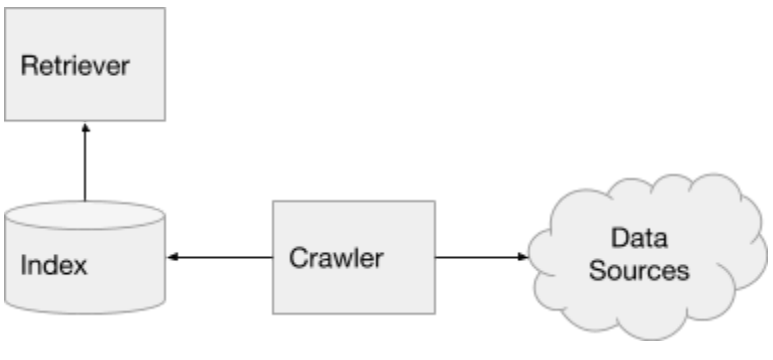
2. Architecture:

Crawler: The crawler will have several starting points and prioritizes its crawl by relevance to the provided query (optional) as it proceeds through the list of links. The crawler begins at the following starting points: www.nytimes.com, www.bloomberg.com, www.forbes.com, www.washingtonpost.com, www.latimes.com, www.huffingtonpost.com, www.usatoday.com, www.wsj.com, www.mercurynews.com, www.newday.com, www.bostonglobe.com, www.inquirer.com, www.chicagotribune.com, www.newyorker.com

(Note: not all sources yielded quotes due to incompatible page format; Bloomberg tended to have the most compatible format for its pages so many of the results are from bloomberg.com)

Retriever: The retriever program runs on a server -- the user provides input (search parameters) to the program from a web browser.

Index: The indexer is written using Lucene and takes in the output of the crawler (formatted quote documents). The indexer adds fields for sentiment, subject, speaker, source, and quote-text to be used by the retriever to filter the results.



3. System Features / Methodology

Definition of a quote

A quote is a sentence or phrase with three associated parts: a subject, a speaker, and a source. The speaker is the person (or entity) delivering the quote. The subject of a quote is not necessarily the same as the grammatical subject of the sentence (although it could be). Rather, it is defined as the main topic of the quote, determined using the method described below.

In this sentence from a New York Times article:

> *“She made New York her home and has been a real workhorse here,” said Peter Romanoff, 49, an advertising executive in Briarcliff Manor who voted for Mrs. Clinton on Tuesday.*

the speaker is Peter Romanoff, the subject is Hillary Clinton, and the source is the URL of the New York Times article.

Quotes can also be self-referential. For example, another sentence from the same new york times article:

> *“I’m hoping to do really well,” Mrs. Clinton said at an L.G.B.T. community center in Greenwich Village.*

contains a quote where the speaker is Hillary Clinton and the subject is also Hillary Clinton.

Crawling for Quotes

The crawler starts processing a page by first isolating the main text of the page from the unnecessary html and formatting content (using the boiler-pipe library). Since quotes are contained within the text of articles, we are only concerned with this part of the page.

Once the “main text” of the page is extracted, the crawler looks for quotations in text by searching for opening and closing quote characters and extracting subject and speaker information around these characters.In written English, it is common for quotation marks to be used for purposes other than quoting a person. They might be placed around an original word or phrase that the author is using, or around the titles of books or other works of writing. In order to filter out these types of quoted words/phrases, I set an arbitrary limit of 5 words for a line of text to be considered a quote by the crawler.

The crawler assumes proper punctuation and formatting for quotes. In cases where either quotation mark is missing or replaced with another character, the crawler will skip that quote or include additional text that shouldn’t be a part of the actual quote.

Identifying Speakers:

In order to identify the associated speakers for quotes, I parsed the text and tagged certain tokens as Named Entities using the OpenNLP name finder model.

In the English language, quotations in text usually follow predictable patterns such as:

1. “[quote],” said [speaker]
2. … [speaker] said, “[quote].”

To associate speakers with quotes, I looked for named entities that matched one of these patterns for the quote I was looking at. To account for modifiers and additional words that occur between the word “said,” I set a span length of 6 tokens between the start/end of the quote and where “said” occurs. I also consider words like “responds,” “replied,” “asked,” and other syncd fionyms to be equivalent to the word “said.”

The Named Entity classifier was not entirely accurate: for example, in some cases it would identify the prefix “Mr.” as being part of a named entity but not include the following name. For obvious cases like these I explicitly defined some rules to append the next token in the text to the speaker. To account for the Named Entity classifier passing over names altogether, I look for “Mr.” and “Mrs.” tokens explicitly after checking for the first two cases.

If none of the above cases are applicable for a quote, I proceed to a catch-all case where I resolve the speaker to be the closest named-entity in either direction of the quote.

Classifying Quotes by Sentiment

For sentiment classification, I trained a maximum entropy model using a subset (first 50,000 lines) of the twitter sentiment training data provided by Niek Sanders (http://www.sananalytics.com/lab/twitter-sentiment/). Data in the training set was classified using either a 1 or a 0, where 1 represents a positive sentiment and 0 represents a negative sentiment. After producing a maxent model from this training set, I used the OpenNLP Document Categorizer to classify quotes found by the web-crawler. I assigned a sentiment-score to each quote equal to the probability that the quote was positive.

Since the training set used was a collection of tweets, and the training set was not as large as it could have been, classification for quotes in text was not as accurate as it could have been.

Resolving Co-References:

Co-referencing is done to associate pronouns and other ambiguous references in the text to their proper subjects.
In order to produce more accurate counts and resolve subject references in quotes, I experimented with coreferencing using the OpenNLP Linker and WordNet dictionary. However, due to results being inconsistent and the high memory requirements involved, I decided to not go this route, and instead resolved co-references when necessary by looking for the closest named entity to a pronoun.

3. Areas where the Searching Succeeds/Fails:

Areas where Search Succeeds:

The search engine did a good job of identifying quotes that closely followed one of the quote patterns described above, where a tagged named-entity closely precedes or follows a quote after the word “said” (or a synonym of “said”).

The following two quotes from the NYTimes were produced from such a pattern:

> “I’m Warren Buffett, and I approve this message,” Mr. Buffett declares.
>
> “You have a choice in consuming more than you use,” Mr. Buffett says of calorie consumption. Referencing his own diet, he adds, “I’m a very, very happy guy.”

(crawler output)

```
QUOTE TEXT: “I’m Warren Buffett, and I approve this message, ”
QUOTE SUBJECT: Warren Buffett
QUOTE SPEAKER: Mr. Buffett
QUOTE SENTIMENT: 0.40454400530157014
QUOTE SOURCE: http://www.nytimes.com/live/warren-buffett-woodstock-capitalists-shareholder-meeting-2016/?ref=dealbook
writing..../indexable_docs/quote389_13342.html

QUOTE TEXT: “You have a choice in consuming more than you use, ”
QUOTE SUBJECT: choice
QUOTE SPEAKER: Mr. Buffett
QUOTE SENTIMENT: 0.8258542901714745
QUOTE SOURCE: http://www.nytimes.com/live/warren-buffett-woodstock-capitalists-shareholder-meeting-2016/?ref=dealbook
writing..../indexable_docs/quote390_20967.html
```

It tended to do a better job at determining the subject of a quote than the speaker since the subject does not rely on content found outside of the quote (unless it is a pronoun). A line of text might have multiple chunks that could be considered the subject of the quote, so the subject selected by the crawler is usually a reasonable approximation for the subject.

Areas where Search Fails:

Mistaking Book, Article, and Movie Titles for Quotes

The crawler had difficulty distinguishing between actual quotes by people and book, article, and movie titles. For example, the following three titles were incorrectly identified as quotes:

(crawler output)

```
QUOTE TEXT: “ When Everything Changed: The Amazing Journey of American Women from 1960 to the Present. ”
QUOTE SUBJECT: Everything Changed
QUOTE SPEAKER: Ms. Collins
QUOTE SENTIMENT: 0.7966686501984339
QUOTE SOURCE: http://www.nytimes.com/column/gail-collins

QUOTE TEXT: “America’s Women: Four Hundred Years of Dolls, Drudges, Helpmates and Heroines, ”
QUOTE SUBJECT: America’s Women
QUOTE SPEAKER: Ms. Collins
QUOTE SENTIMENT: 0.7487174339653755
QUOTE SOURCE: http://www.nytimes.com/column/gail-collins

QUOTE TEXT: “As Texas Goes: How the Lone Star State Hijacked the American Agenda, ”
QUOTE SUBJECT: Texas Goes
QUOTE SPEAKER: William Henry Harrison
QUOTE SENTIMENT: 0.8046701148745088
QUOTE SOURCE: http://www.nytimes.com/column/gail-collins
```

What makes it hard to distinguish between titles and actual quotes is that both tend to have Named Entities surrounding the quote words. For quotes, there is usually a pronoun or name a few tokens before or after the quote marks. For titles, there is usually a name before or after the quote mark. A future add-on to this project might consider sentence structure and the ratio of verbs to nouns in the sentence to distinguish between quotes and sentences.

Speaker Association Inconsistency

The Named Entity classifier struggled with certain surnames and passed over some names near the text. In these cases, it would default to the catch-all case of finding the nearest named entity (which might be very far away in the text).

For quotes that don't follow one of the standard patterns described above, the speaker association would also use the default case of finding the nearest named-entity. For example in the following quote from a NYTimes article:

> But leave it to Mr. Munger to toss off the most prickly description of the company: "Valeant was a sewer, and those who created it deserved the opprobrium they got."

The quote follows a more uncommon form, and speaker is resolved to "Merced" who is the author of the article.

General Issues Encountered:

I encountered some errors using the provided OpenNLP part-of-speech models. There were some cases where nouns and pronouns were not identified correctly which lead to incorrect identification of speaker and subject in quotes.

As noted previously, the named entity classifier was also inconsistent, especially with last names. Certain last names such as "Trump" were not recognized as part of a named entity chunk -- presumably because the word "trump" can be used in many different ways.

The crawling process itself presented some issues -- as some sites would not allow crawlers to download pages or the actual article content was not embedded in the html file itself. In these cases, the document was simply not indexable and was passed over. For documents with incorrect formatting for quotations, the crawler would fail to identify the start and end of a quote. For example in the following quote from a Bloomberg article:

Areas to Improve Upon In the Future:

Resolving co-references would improve the accuracy of subject and speaker detection. Using a co-reference linker would allow the quote-parser to link pronouns to their proper subjects by building a parse tree from the sentences in the text. The search engine currently does not do this - it identifies speakers using part-of-speech and named-entity information of the chunks near the quote. Coreference would allow the search engine to display results in a more consistent manner. For example, quotes by "Clinton," "Hillary Clinton," "She," and "Mrs. Clinton" could all be resolved to "Hillary Clinton."

5. External Software Used:

Development Tools:
IntelliJ IDE
Sublime Text 2

OpenNLP libraries:
maxent-3.0.0.jar
opennlp-tools-1.5.0.jar
nekohtml-1.9.13.jar
xerces-2.9.1.jar

Lucene libraries:
lucene-demo-5.4.1.jar
lucene-queries-5.4.1.jar
lucene-analyzers-common-5.4.1.jar
lucene-queryparser-5.4.1.jar
lucene-core-5.4.1.jar

Other libraries:

jtidy-r938.jar	(JTidy)
boilerpipe-1.2.0.jar	(boiler-pipe)
jsoup-1.9.1.jar	(JSoup)
commons-io-2.4.jar	(Apache Commons)
jwnl-1.3.3.jar	(JWNL)

OpenNLP models:

en-chunker.bin	(chunking model)
en-pos-maxent.bin	(part-of-speech model)
en-ner-person.bin	(named entity model)
en-sent.bin	(sentence detection model)
en-token.bin	(tokenization model)

6. Web Resources/References Used:

General Reference:

Mining Web Pages: Using Natural Language Processing to Understand Human Language." Safari. N.p., n.d. Web. 04 May 2016. <https://www.safaribooksonline.com/library/view/mining-the-social/9781449368180/ch05.html>.

Manning, Christopher, Prabhakar Raghavan, and Hinrich Schütze. "Introduction to Information Retrieval." Introduction to Information Retrieval. N.p., n.d. Web. 04 May 2016. <http://nlp.stanford.edu/IR-book/>.

OpenNLP Reference:

"Apache OpenNLP Developer Documentation." Apache OpenNLP Developer Documentation. N.p., 04 May 2016. Web. 04 May 2016. <https://opennlp.apache.org/documentation/1.5.3/manual/opennlp.html#opennlp>.

"Sentiment Analysis Using OpenNLP Document Categorizer." Technobium. N.p., 08 Mar. 2015. Web. 04 May 2016. <http://technobium.com/sentiment-analysis-using-opennlp-document-categorizer/>.

"Getting Started with OpenNLP 1.5.0 – Sentence Detection and Tokenizing." Dpdearing. N.p., n.d. Web. 04 May 2016. <http://blog.dpdearing.com/2011/05/opennlp-1-5-0-basics-sentence-detection-and-tokenizing/>.

boilerpipe Reference:

"Code." Google Archive. N.p., n.d. Web. 04 May 2016. <https://code.google.com/archive/p/boilerpipe/wikis/QuickStart.wiki>.

Lucene Reference:

"LuceneTutorial.com." Lucene Scoring. N.p., n.d. Web. 04 May 2016. <http://www.lucenetutorial.com/advanced-topics/scoring.html>.

Subject Resolution:

"Locating a Sentence's Topic." Nlp. N.p., n.d. Web. 04 May 2016. <http://linguistics.stackexchange.com/questions/2643/locating-a-sentences-topic>.

Parsing Quotes:

"How to Extract Quotes from the News." LingPipe Blog. N.p., 01 Oct. 2008. Web. 04 May 2016. <https://lingpipe-blog.com/2008/10/01/how-to-extract-quotes-from-the-news/>.

B. Pouliquen, R. Steinberger and C. Best, "Automatic Detection of Quotations in Multilingual News", *In Proceedings of the International Conference Recent Advances in Natural Language Processing*

Other References:

For training the sentiment classification model used by the quote-crawler, I extracted a subset of the training tweet data provided by Niek Sanders: http://www.sananalytics.com/lab/twitter-sentiment/.

I consulted Wikipedia.com for general technical reference and StackOverflow.com for debugging and installation help for some Java libraries