Quote Crawler README

(all programs were compiled and tested using JDK Version 1.7.0)

Sentiment Classifier Training

I wrote this program to format a training set and train a maximum *entropy model for the OpenNLP classifier to use. The training set I used was a subset (first 50,000 lines) of the twitter sentiment training data provided by Niek Sanders (http://www.sananalytics.com/lab/twitter-sentiment/). The program outputs a file called en-sentiment-model.bin.*

Compile and Run:

run javac and specify external jars

```
javac -cp "./jars/*" *.java
```

run program

```
java -cp ".:./jars/*" TrainModel [path to training data]
```

Example:

```
java -cp ".:./jars/*" TrainModel ./shortened_sent_dataset.txt
```

Crawler

The crawler crawls web pages from a specified starting point and looks for quotes within those pages. It downloads these pages to the specified directory, then creates a separate directory for indexable quote documents in ./indexable_docs

Parameters:

```
(-u) [optional] the URL from which to start the crawl; defaults to some predefined news websites
(-q) [optional] a query string, the crawler only considers pages relevant to this; defaults to an empty string
(-docs) the path name for a directory to save the downloaded pages
(-m) [optional] the maximum number of pages to download; defaults to 10
(-t) [optional] flag for generating a trace; defaults to false
```

Compile and Run:

run javac and specify external jars

```
javac -cp "./jars/*" *.java
```

run program

```
java -cp ".:./jars/*" Main -u [url] -q  [query] -docs [output] -m [max] -t
```

Example:

```
java -cp ".:./jars/*" Main -u "http://www.bloomberg.com" -q "" -docs "./crawler_test" -m 20 -t
```

Indexer

The Indexer takes in the collection of quote documents produced by the Crawler (outputted to ./indexable_docs) and produces a searchable lucene index of quotes

Compile and Run:

run javac and specify external jars

```
javac -cp "./jars/*" *.java
```

run program

```
java -cp ".:./jars/*" Indexer -index [index output] -docs [input docs]
```

Example:

```
java -cp ".:./jars/*" Indexer -index "../_index" -docs "./indexable_docs/"
```

Searcher

The searcher takes in user input from the web interface and retrieves the relevant results from the Indexer, filtered by the provided user parameters (if any)

Parameters:

```
(-q) the query to search for (can be word or phrase)
(-index) the path to the index to search
(-s) [optional] specify the speaker for the quote (limit results to a certain speaker)
(-x) [optional] lower bound for positive sentiment probability; defaults to 0.0
(-y) [optional] upper bound for positive sentiment probability; defaults to 1.0
```

Compile and Run:

run javac and specify external jars

```
javac -cp "./jars/*" *.java
```

run program

```
java -cp ".:./jars/*" Searcher -index [index] -q [query] -s [speaker]
```

Example:

```
java -cp ".:./jars/*" Searcher -index "../_index" -q "economy" -s "Morrison" -x 0.2 -y 0.9
```
