News Sentiment Analysis Using R to Predict Stock Market Trends

Anurag Nagar and Michael Hahsler
Computer Science
Southern Methodist University
Dallas, TX
Topics

- Motivation
- Gathering News
- Creating News Corpus
- Gathering Sentiment
- Results
- Conclusion
- References
Motivation

- It's well known that news items have significant impact on stock indices and prices.
- Lots of previous work on finding sentiment from static text using Text Mining and NLP techniques.
- We analyze news items for sentiment using dynamic data sources – such as online news stories and streaming data such as blogs.
R Resources for Financial News

- R allows real-time news gathering using:
  - `tm` package
  - `tm` package plugins:
    - `tm.plugin.webmining`
    - `tm.plugin.sentiment`
  - `XML` package

- Allow financial news to be aggregated using sources such as Google Finance, Yahoo Finance, Twitter, etc.
R Resources for Financial News

- Creating a corpus using Google Finance:
  
  ```R
  > corpus <- WebCorpus(GoogleFinanceSource("AAPL"))
  ```

- Returns a corpus of documents with several useful attributes:
  
  - Time Stamp (Filter out old stories)
  - Heading (Find breaking news)
  - Short Description (Check if it's relevant)
  - Author (Authority?)
  - Source (Reliable source?)
Types of Corpuses

Three types of text corpuses are constructed from the news articles:

- Constructed from Filtered Sentences
- Constructed from just the Headlines
- Constructed from the Short Description Attribute
Extracting Relevant Sentences

- Our approach filters the news articles to only those sentences which contain the stock symbol.
- Instead of tagging the entire news story, we focus only on relevant sentences.

Filtered Sentence Corpus

- Used R package openNLP to break the corpus into sentences.

```r
> stock ← "AAPL"
> sentences ← sentDetect(corpus)
> filteredSentences ← sentences[grepl(stock, sentences)]
```

- **Filtered sentences** more likely to contain company specific news, analysis, and predictions.
WebCorpus allows us to look at the headlines.

```r
> sapply(corpus,FUN=function(x){attr(x,"Heading")})
```

Corpus items have a “Description” attribute

```r
> stock ← "PCLN"
> desc ← sapply(corpus,FUN=function(x) { attr(x,"Description") } )
> filteredDesc ← desc[grepl(stock,desc)]
```

filteredDesc contains stock specific current news.
Identifying Polarity of Words

- Used following sources to create list of “sentiment” words:

1. Multi-Perspective Question Answering (MPQA) Subjectivity Lexicon
   http://www.cs.pitt.edu/mpqa/subj_lexicon.html

2. List of sentiment words from R package tm.plugin.tags

3. List of sentiment words from Jeffrey Breen's tutorial
   http://jeffreybreen.wordpress.com/2011/07/04/twitter-text-mining-r-slides/
Scoring Text Corpus

- An instance (sentence, headline) is positive if the count of positive words is greater than count of negative words and vice versa.

For example, the sentence:
  “AAPL continues its phenomenal run”
is a positive sentence as count(positive) = 2 and count(negative) = 0

  “Cracks develop in PCLN”
is negative heading as count(positive) = 0 and count(negative) = 1
For an entire corpus, we count the positive and negative instances and compute the score as:

Corpus Score = \frac{\text{Positive instances}}{\text{Total instances}}

Three types of Corpus Scores:
1. Sentences Corpus Score
2. Headlines Corpus Score
3. Short Description Corpus Score
Scoring Text Corpus Code

# text is from the news, pos and neg are positive and negative word lists
scoreCorpus <- function(text, pos, neg) {
  corpus <- Corpus(VectorSource(text))
  termfreq_control <- list(removePunctuation = TRUE,
                          stemming=FALSE, stopwords=TRUE, wordLengths=c(2,100))
  dtm <- DocumentTermMatrix(corpus, control=termfreq_control)
  # term frequency matrix
  tfidf <- weightTfIdf(dtm)
  # identify positive terms
  which_pos <- Terms(dtm) %in% pos
  # identify negative terms
  which_neg <- Terms(dtm) %in% neg
  # number of positive terms in each row
  score_pos <- row_sums(dtm[, which_pos])
  # number of negative terms in each row
  score_neg <- row_sums(dtm[, which_neg])
  # number of rows having positive score makes up the net score
  net_score <- sum((score_pos – score_neg)>0)
  # length is the total number of instances in the corpus
  length <- length(score_pos – score_neg)
  score <- net_score / length
  return(score)
}
Results

- Next slides will compare Sentiment Score trends with Stock Price movement for Apple Corp (AAPL).
- Note the similarity in the shape and trend of the curves.
- Sentiment scores are able to predict the movement of stocks quite accurately.
- Sentence Sentiment scores are often more accurate because of the larger sample size.
Results – AAPL Sentences vs Stock

AAPL Sentences from 04/03 to 05/04
Results – AAPL Headlines vs Stock
Results – AAPL Description vs Stock

AAPL Description from 04/03 to 05/04

AAPL

[2012-04-03/2012-05-04]
Discussion

- Strong visual correlation between stock price movement and News Sentiment Score.
- Accuracy can be further improved by incorporating stock market specific terms into the tagging scheme.
- This scheme can be used along with other techniques to provide a very strong indicator of stock market movement.
References


