Contribution to Stock Price Prediction using a Sentiment Analysis Framework
Contents

• Data Set
• Method
• Further Research and Application
Data Set

- Targeted financial news websites with publicly available content
  - Financial Times
  - SeekingAlpha
- Data set consists of 560,000 articles over 10 years (2004-2014)
Data flow for classification task

1. Automated Data Download using Python Mechanize
2. Parsing of HTML articles using BeautifulSoup and Storage into local DB
3. Traditional server
4. Classification Process using NLTK and SentiWordNet
Classification

- Bayesian classification model in conjunction with bag of words concept
- Generate a score for each word using SentiWordNet database after classification into positive and negative
- Bag of words model is a simple natural language processing mechanism and provides a baseline for correlating sentiment against stock price
- We leverage Python’s NLTK library which provides many tools for linguistic programming
Training the Model

• Before applying the model we must “train” it to recognize positivity and negativity in language - specifically financial language.

• Take a subset of data ~ 20,000 articles and manually classify them into positive and negative by leveraging Amazon Mechanical Turks.

• Next step is to remove stop words and conjunctions which have little meaning to the overall article such as “I, who, whom...” - this is accomplished by using the NLTK stop word corpus.

• Finally we apply the bag of words approach to create a frequency distribution of positive and negative words.
Classifying the data

- Given an article we tokenize it into sentences
- Within a sentence we use a Naive Bayesian Classification scheme to decide if a word is positive or negative
- Mathematical Problem Statement

Assume we have two distributions \( C_j, j \in 1, 2 \)
An unseen instance of a word \( d \)

\[
P(C_j | d) = \frac{p(d | c_j) p(c_j)}{p(d)}
\]

Where:
- \( p(c_j | d) \) is the probability that instance \( d \) is in distribution \( C_j \).
- \( p(d | c_j) \) is probability of generating instance \( d \) given distribution \( C_j \).
- \( p(c_j) \) is probability of distribution \( c_j \) occurring, in our case \( \frac{1}{2} \).
- \( p(d) \) is probability of instance \( d \) occurring
Creating Wordnets

• For each given word we create a wordnet

• To do so we first have to create a “synset” - set of distinctive cognitive synonyms

• E.g. for the word profit

  • [Synset('net_income.n.01'),
    Synset('profit.n.02'),
    Synset('profit.v.01'),
    Synset('profit.v.02')]

• Based on these synsets we can create a wordnet - a lexical and conceptual link between synsets

• These are ordered by the hyperonymy/hyponymy relation
Assigning a Score

• Given a wordnet our next task is to *disambiguate word senses* and then assign a sentiment score to the appropriate word.

• Returning to our example with profit let us consider the following statement “Company XYZ reported earnings that resulted in large profits”

• Here earnings are related to profits so we consider the synsets of earnings and profits.

• For earnings the synsets are [Synset('net_income.n.01'), Synset('wage.n.01'), Synset('gain.v.08'), Synset('earn.v.02')]

• For profits the synsets are [Synset('net_income.n.01'), Synset('profit.n.02'), Synset('profit.v.01'), Synset('profit.v.02')]

• We then create wordnets for each synset and determine similarity by taking the shortest path between the wordnets - in this case we have a perfect match.
Application to Stock Price Prediction

- Based on our method we generate a sentiment score for the overall market and use it as a regressor variable in order to predict stock price movement.

- Based on our observations the change in sentiment on a day over day basis has higher predictive power than the instantaneous sentiment.
Further Research and Application

- Apply a high pass filter to sentiment score as large swings seem to have higher predictive power.
- Fit the data using a time series based approach which is able to model lags between sentiment and stock price movement.
- Apply this data to a wider set of financial instruments.
  - It is possible in the equity market the sentiment value has already worked itself into the price because of transparent access to stock prices.
  - Instruments which are harder to price, and do not have publicly available quotes may benefit more.
1. W. Yih, J. Goodman, V. Carvalho, “Finding Advertising Keywords in Webpages”

2. J. Cushing, R. Hastings, “Introducing Computational linguistics with NLTK”

3. H. Kang, S. J. Yoo, and D. Han, "Senti-lexicon and improved Naïve Bayes Algorithms for Sentiment Analysis of Restaurant Reviews”