Using Twitter as a source of information for stock market prediction

Argimiro Arratia
argimiro@lsi.upc.edu

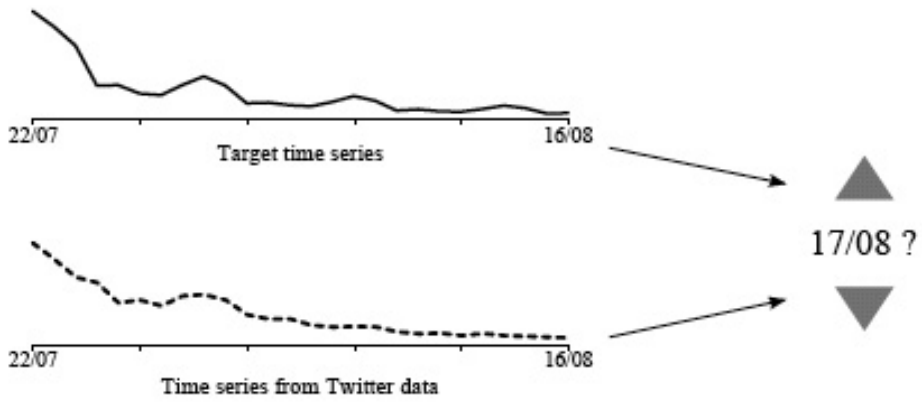
LSI, Univ. Politécnica de Catalunya, Barcelona
(Joint work with Marta Arias and Ramón Xuriguera)

Motivations and Goals

- News impact our decision-making
- Public mood or sentiment influence stock market prices
- Does it really?

If we can quantify the public sentiment towards a certain stock or the market in general, can we use this measure to predict prices?

- We center attention on measuring sentiment of users of social networking platforms, and particularly **Twitter**
- Hypothesis:
  (Volume) More messages $\mapsto$ more variance (Volatility)?
- Hypothesis:
  (Sentiment index) negative/positive $\mapsto$ decrease/increase benefits (returns)?



Related work

J. Bollen, et al. *Twitter mood predicts the stock market*, arXiv 14/10/2010.

- Uses OpinionFinder a general purpose software package for text content analysis that produces a emotional polarity (+/-) of sentences. Google Profile of Mood States, another tool for assessing sentiment within text from various mood dimensions.
- Applies only one prediction model: Self-organizing Fuzzy Neural Network. (Justification: "the relation between public mood and stock market values is almost certainly non linear".)
- Uses a DB of tweets from Feb to Dec. 2008.
- "All data sets and methods are available on our project web site" (op.cit p 3)

Not so! No possibility of replicating the authors experiments ... [2011-05-17]: $40 Million Twitter-Based Hedge Fund Now Open for Business!!
(http://mashable.com/2011/05/17/twitter-based-hedge-fund/)

Twitter

@TEDNews
TED News

RT @TEDchris: Mind-shifting #TED talk on the evolution of language from Mark Pagel http://on.ted.com/Pagel

3 Aug via TweetDeck

- Online service that allows to build social networks based on microblogging
- Messages or tweets up to 140 characters
- Special or reserved symbols: @ for replying; # (hashtags) for topics assignment or keywords; RT (*retweet*) for sharing with followers; URLs.

Tweet retrieval

There are no public data sets available and Twitter have imposed several restrictions on retrieving on-line posted tweets.
We were forced to create our own data set with limitations.
Begun on 22 March 2011. Use a Streaming API:

- One HTTP connection is kept alive to retrieve tweets as they are posted
- Filter the stream by keyword or user

Alternatives: Google Realtime Search returned search results from Twitter or Facebook (past data). But went offline due to expiration of Google's agreement with Twitter.
Buy data from official Twitter reseller. Very, very expensive!

Sentiment Analysis

Goal: Determine whether a message contains positive or negative impressions on a given subject

↙ ↘

Subjectivity Recognition	Polarity Detection

- Begin with creating a labelled corpora for supervised training a sentiment classifier
- We apply a recent pictorial tagging idea [Bifet et al, 2010]: Create a dataset of tweets that are automatically labelled positive if contains a smiley of the form: `:-)` ,`;-D` or negative if contains `:-(`.
  (Formally, consider all regular expressions from `[:=8][ -]?[)D]` for positive, and from `[:=8][ -]?(` for negative.)

Sentiment Analysis

- Expand the training datasets by Feature Lists of words: classify by frequent words with at least 5 occurrences, and topic (e.g. a stock's ticker).
- Of course! Clean the data previously:
  - Remove duplicates (mostly in retweets);
  - Remove stopwords (e.g. pronouns, prepositions, many verbs);
  - Stemming (reduce words to their roots by removing suffixes);
  - Negation handling (use tagging: not good $\mapsto$ NOT_good);
- Relevance filtering: text containing some of the keywords is no guarantee of its relevance for the subject we want to classify. E.g.
  "I really love eating an apple"
  "Apple stock soared above $404 today"

Latent Dirichlet Allocation (LDA) filtering [Blei et al 2002]

LDA is a generative model used for topic modelling.
Basic idea: docs. consist of mixture of topics in different proportions determined by the histogram. A multisided coin is thrown for each word in the doc. assigning topics according to their distribution.
E.g.: we can say that the tweet above is a mixture of *politics, sociology* and *sound/music*.

LDA. Experimental results

- Created datasets of tweets as mixture of two topics (*relevant /not relevant*)
- 300K tweet collection. Tested with 20K tweets
- High precision (83.3%), low recall (65.4%)

Datasets of tweets for training classifier: *English*, *Multilanguage*, *Stock related*

Sentiment Classifier

- Multinomial Naïve Bayes: Given D the set of all docs. and C set of class labels, a Naive Bayes classifier will assign a doc. d the class with the highest conditional probability given that document. (d is represented as n-dim. vector $(w_1, \ldots, w_n)$ of Bernoulli-distributed variables indicating whether word w_i occurs in d). The naive assumption is that the presence of a particular feature of a class is unrelated to the presence of any other feature.
- Binary Classification (positive/negative)
- Accuracy: 76.49% on noisy set (English), 79.5% for twittersentiment set (by frequency of words related to topic)

Sentiment Index

A time series consisting of the daily percentage of positive tweets (over the total number of tweets posted concerning a subject, e.g. SP500)

Financial Time Series

Goal

To predict stock's **return** and **volatility**

Focus on three technological companies: AAPL, MSFT, GOOG and two indices: OEX (S&P100), VIX (S&P500 implicit volatility).

Returns
- Computed from Adjusted Close
- Log–normally distributed
- Log returns

Volatility
- Computed from log returns
- Exponential Weighted Moving Average

Forecasting time series

Models for prediction

- Linear Regression
- Neural Networks (feedforward)
- Support Vector Machines (with kernel polynomial, radial or sigmoid)

Model Assessment

- Nonlinearity: we use White neural network test for neglected nonlinearity to asses the possible nonlinear relationship between two time series
- Causality: apply Granger test of causality, both parametric and non-parametric, to the two time series and for different lags.
- Prequential evaluation: Our data is from short period, we cannot spare an independent set for training. Use past data to update/retrain the classifier and predict the direction of the time series for the following day.

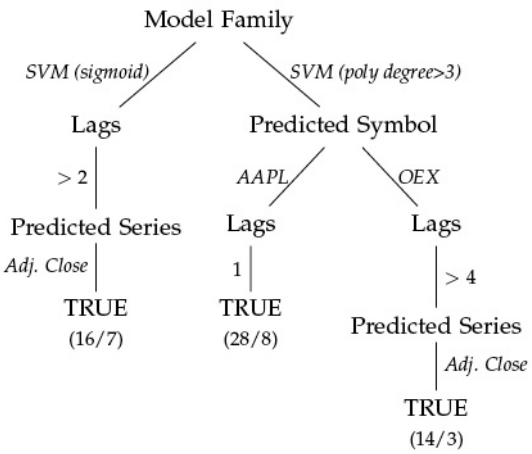
Experimental set up

Combining all different parameters give us 6820 different experiments.

Summary trees

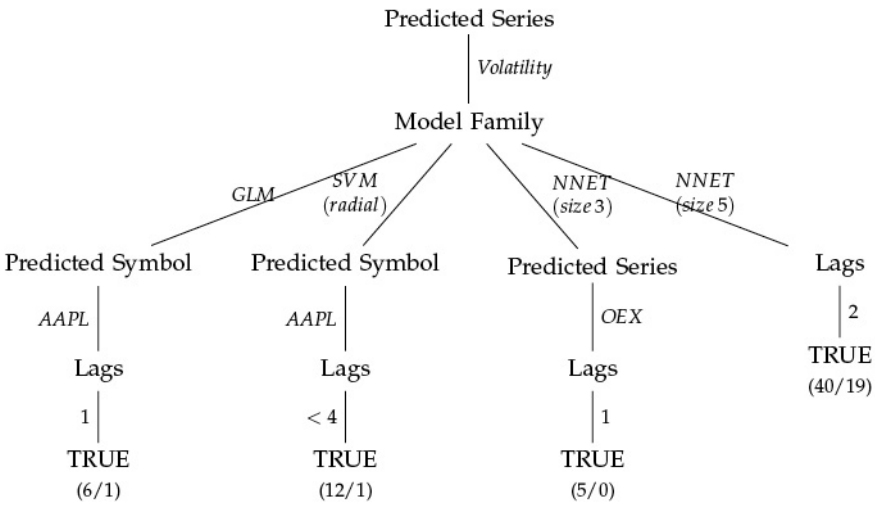
A decision tree, elaborated with `REPTree` (Weka), built by selecting attributes that give a higher performance gain

Adding the sentiment index to model



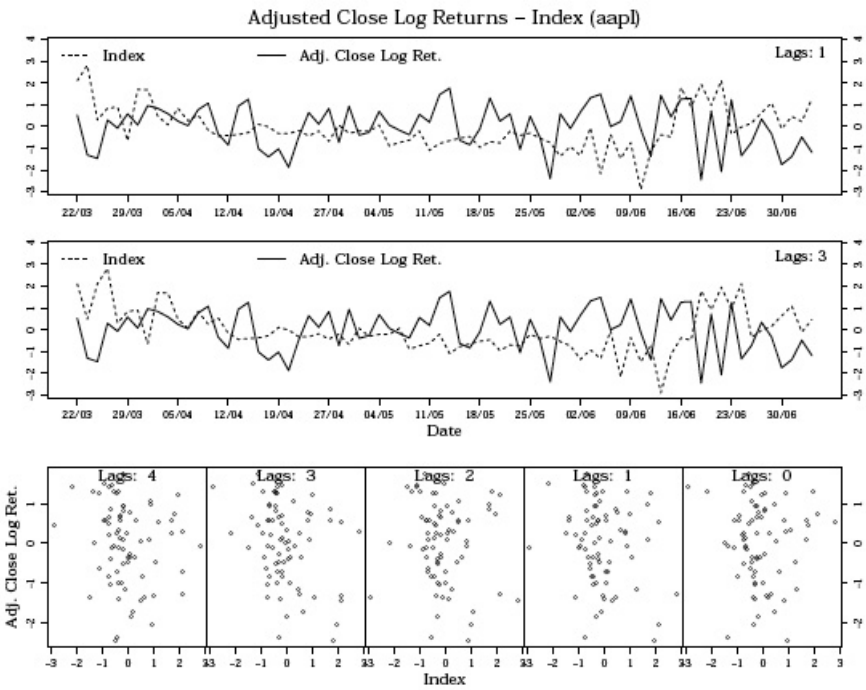
Experimental results

Adding tweet volume to model

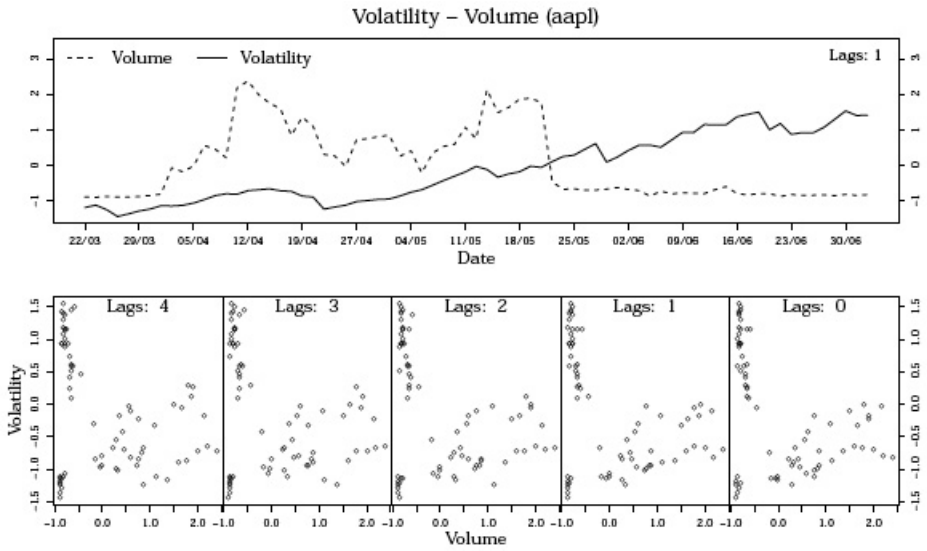


The trees give an indication of the attributes that most influent the increase on accuracy. We note that neither the classifiers nor the LDA filtering are at first levels of tree

Experimental Results (Apple)



Experimental Results



Forecasting Evaluation

Kappa statistics, Accuracy values for the most successful model according to summary tree: the SVM.

Predicting the direction of price with an SVM of poly kernel ...

Table 1: with/without Sentiment index (with = Imp.)

Lag	Correct	Misclassified	Accuracy	Imp. Accuracy	Kappa	Imp. Kappa
1	48	25	0.6027	0.6575	0.2097	0.3167
2	47	25	0.6666	0.6527	0.3338	0.3044
3	44	27	0.6338	0.6197	0.2686	0.2419
4	48	22	0.6285	0.6857	0.2565	0.3729
5	41	28	0.6521	0.5942	0.3053	0.1889

Table 2: with/without Sentiment index and tweet volume

Lag	Correct	Misclassified	Accuracy	Imp. Accuracy	Kappa	Imp. Kappa
1	49	25	0.60811	0.66216	0.21622	0.32432
2	48	25	0.65753	0.65753	0.31520	0.31468
3	45	27	0.62500	0.62500	0.25000	0.25000
4	48	23	0.61972	0.67606	0.23898	0.35275
5	42	28	0.64286	0.60000	0.28571	0.20000