Available online at www.sciencedirect.com

ScienceDirect

The Journal of Finance and Data Science 4 (2018) 120–137

http://www.keaipublishing.com/en/journals/jfds/

Financial news predicts stock market volatility better than close price

Adam Atkins*, Mahesan Niranjan, Enrico Gerding

Electronics and Computer Science, University of Southampton, UK

Received 3 September 2017; revised 19 January 2018; accepted 1 February 2018
Available online 8 February 2018

Abstract

The behaviour of time series data from financial markets is influenced by a rich mixture of quantitative information from the dynamics of the system, captured in its past behaviour, and qualitative information about the underlying fundamentals arriving via various forms of news feeds. Pattern recognition of financial data using an effective combination of these two types of information is of much interest nowadays, and is addressed in several academic disciplines as well as by practitioners. Recent literature has focused much effort on the use of news-derived information to predict the direction of movement of a stock, *i.e.* posed as a classification problem, or the precise value of a future asset price, *i.e.* posed as a regression problem. Here, we show that information extracted from news sources is better at predicting the direction of underlying asset *volatility* movement, or its second order statistics, rather than its direction of price movement. We show empirical results by constructing machine learning models of Latent Dirichlet Allocation to represent information from news feeds, and simple naïve Bayes classifiers to predict the direction of movements. Empirical results show that the average directional prediction accuracy for volatility, on arrival of new information, is 56%, while that of the asset close price is no better than random at 49%. We evaluate these results using a range of stocks and stock indices in the US market, using a reliable news source as input. We conclude that volatility movements are more predictable than asset price movements when using financial news as machine learning input, and hence could potentially be exploited in pricing derivatives contracts via quantifying volatility.

Keywords: Machine learning; Natural language processing; Volatility forecasting; Technical analysis; Computational finance

1. Introduction

Data from financial markets offer challenging signal processing problems and have attracted much interest from quantitative researchers and traders. The underlying complexity of the financial system that generates such data is enormous, and hence data arising from the system can show both non-linear and non-stationary characteristics. With easy availability of electronic data, there is significant recent growth in their analyses, and such analyses translating

* Corresponding author.
E-mail addresses: adam@adamatkins.co.uk (A. Atkins), mn@ecs.soton.ac.uk (M. Niranjan), eg@ecs.soton.ac.uk (E. Gerding).
Peer review under responsibility of China Science Publishing & Media Ltd.

https://doi.org/10.1016/j.jfds.2018.02.002

into automated trading systems, with one estimate suggesting algorithmic trades accounted for 85% of market volume in 2012, compared to just 15% in 2003.[1] The derivatives markets, too, have grown significantly with the number of U.S.-listed equity and index options averaging 16.5 million contracts daily across the first five months of 2017[2] and the global exchange-traded options market was valued at over 41 thousand billion US dollars in December 2016.[3]

Optimisation of portfolios based on expected returns from financial instruments (such as stocks) and uncertainties in their price movements (*e.g.*[4–6]), pricing derivatives and other complex contracts based on some predicted future value of an underlying asset,[7] and general time series analysis using a myriad of univariate and multivariate analytical techniques[8] have been explored in literature, spanning a range of disciplines including statistics, quantitative finance, signal processing and machine learning. These tools of *quantitative modelling*, elementary forms of which are captured in the practice of *technical trading*, complement what is termed *fundamental analysis*[9] in which qualitative information about an asset is used in combination with macroeconomic information to make predictions about its value. Qualitative information used in fundamental analysis includes periodic reports published by companies on their profitability, payments of dividends etc. as well as news about them and the economic environment, available via news feeds. More advanced data-driven models including artificial neural networks and support vector machines have been used in combination with traditional methods of technical analysis (*e.g.*[10–14]), and varying degrees of success have been claimed.

An intriguing development that has significant bearing on the above is in natural language processing and understanding, whereby news feeds and other sources of textual information can be automatically mined and relevant information extracted.[15] Information extracted from text has been successfully applied in several disciplines; IBM's question answering system Watson,[16] and the construction of gene regulatory networks[17] are examples of this. Coupled with this is the exponential increase in information generated and propagated via social media, with a much wider and faster reach to individual investors and professional practitioners.[18]

There are many examples of applying text mining to news data relating to the stock market (*e.g.*[19–21]), with a particular emphasis on the prediction of market close prices. An example is the work of Gidofalvi,[22] which, similar to our own, uses a naïve Bayes classifier to predict close price direction on an intraday basis using news as input. In addition to news feeds, sources of textual and behavioural data used by researchers include StockTwits, GoogleTrends and even Wikipedia page views.[23–25] Analysis of Google trends query volumes led to the discovery and formation of "patterns that may be interpreted as 'early warning signs' of stock market moves".[25]

In terms of extracting information from social media, the work of Bollen et al had a significant impact in the literature.[26] These authors derived a six dimensional representation of mood (Calm, Alert, Sure, Vital, Kind, and Happy) by mining large volumes of tweets.[a] The resulting characterization of mood was used to predict the direction of movement of the Dow Jones Index. However, their claim of predicting daily up/down market movement at an accuracy of 86.7% was met with broad skepticism in the computational finance community. Firstly, the prediction accuracy reported was the best of 8 models attempted in the study and could be viewed as selective reporting. Secondly, the testing period considered by the authors was very short, at only 15 trading days in total, with $\frac{13}{15}$ days predicted correctly. The uncertainty in this result is bound to be high and not explored by the authors. Lastly, the authors test Granger causality between sentiment and financial time series. However, Granger causality assumes stationary covariance (mean and variance unchanged) over time, which is not the case for market prices, exhibiting non-stationary characteristics. The authors do not explain whether or not they have adjusted for this assumption, for instance by constructing data windows.

Despite this, the very idea of relating information extracted from news feeds to market movement via the intermediate step of characterizing sentiments is an elegant one. Overall emotions, both at the level of groups of traders and at the level of wider society, are bound to influence the behaviour of financial markets. While the behaviour of individual investors might be heavily modulated by context, Bollen et al correctly postulate that an overall collective signal which influences trends in market behaviour may exist and this may be recoverable by mining messages exchanged on social media. We address some of the concerns in the work of Bollen et al by testing on large, heterogeneous data sets.

All of the work reviewed above aims to predict market prices or changes in prices from literature derived information. To the best of our knowledge, with the sole exception of,[27] using information mined from news to predict

[a] www.twitter.com.

second order statistics such as volatility, or even volume traded in the markets has not been attempted. Forecasting and modelling volatility, however, has been of interest in quantitative modelling of financial data, with a rich literature on statistical models and algorithms (*e.g.*[28–30]). The family of conditional heteroskedasticity models (*e.g.* GARCH) are examples of time series models with stochastic dynamics imposed on residual variances. Data-driven approaches to modelling volatilities include the use of advanced machine learning methods such as support vector machines, to predict volatility changes from previous values of the volatility time series.[31] There is some work in the literature in combining time series modelling of volatilities (GARCH) in combination with news-derived data (*e.g.*[32,33]). A standalone model in[34] attempts to correlate StockTwits posting volumes to volatility, though with no underlying information extraction from natural language processing. The relationship between notes in federal reserve board meeting minutes to market volatility has been explored in,[35] and is the attempt closest to the work in this paper. The work reported in[27] compares news-driven linear regression with GARCH modelling, but the authors make no attempt at comparing the predictions against predicting asset price movement.

In this paper, we explore the hypothesis that news-derived information is likely to have a greater effect on the second order characteristic of market volatility than on the asset values or their direction of movement. We carry out an empirical study and show evidence in support of this hypothesis. We construct a Latent Dirichlet Allocation (LDA) model, which is effective in natural language feature reduction, though proves to be computationally expensive to train. Classification is achieved using a naïve Bayes algorithm, which performs well despite the simplistic assumption of feature independence.

The novel contribution we present is in making predictions about asset price and volatility changes from news-derived information alone, as opposed to integrating with a time series model. This results (see the section on results) in the prediction of volatilities giving rise to a small, yet statistically significant, signal, and the direction of movement of the asset price itself performing no better than random. Further, our work is illustrated on a much larger set of data than what we find as usual in the literature. We assess the performance of our model against a range of benchmarks, including a technical analysis model making use of only time series data. We take account of non-stationarity in the data firstly by applying a decay function to weight news in the feature vector according to recency, and secondly by repeatedly training and testing models on sliding temporal windows, and presenting results that quantify the variation obtained across these different periods. In making a systematic comparison between predicting volatility and predicting close price movements, accounting for non-stationarity in a systematic way and in the scale of the empirical evaluation, our work goes substantially beyond what has been touched upon in.[27]

2. Data and inference algorithms

2.1. Financial data

For empirical evaluation of ideas, we used two stock indices (NASDAQ Composite and Dow Jones Industrial Average) and two equities (Goldman Sachs and J. P. Morgan), all covering the range 09 September 2011 to 07 September 2012. For purposes of analysis and dealing with non-stationarity, this data was split into four sections (see Appendix A).

An empirical observation claimed in[22] is that on average, there is a 20 min time lag between release of news and quantifiable impact on markets. After some tuning on a small subset of data, we implemented all our predictions during 60 min periods. This strikes a balance between excessive sparsity of the feature vector and the effective time-lag over which news has effect on the market, as selecting a short temporal window of analysis reduces the chance of observing a term in the vocabulary while taking a long window may smooth out the dynamic nature of relevant information.

Minute-by-minute resolution intraday data available online at a start-up quantitative trading website[b] called 'The Bonnot Gang' was downloaded for the above assets during the period 09 September 2011 to 07 September 2012. Periods of data with missing values were excluded so as not to skew the volatility calculation. Missing values at the start or end of day were imputed using daily data from Quandl's Yahoo Finance database.[c]

[b] www.thebonnotgang.com/tbg/historical-data.
[c] www.quandl.com/data/YAHOO.

2.2. News data

For textual data, we used a portion of the Reuters US news archive from 09-2011 to 09-2012, as this was considered a reliable length data set, surpassing the length used in other literature (*e.g.*[26,20,36]). US news was used, as this is most relevant to the US indices and assets we are predicting. Furthermore, Reuters is considered a reliable source of financial news, with the Reuters-21578 collection being popular in the natural language processing community.[37] This particular portion was selected because, when considering US market volatility using the VIX S&P500 volatility index, it contains periods of both high and low volatility for the US markets. It is beneficial to expose the predictor to different market conditions to showcase its resilience.

Another advantage of using Reuters as a news source was that the archives were friendly to scrape from, as each day had an archive page of the form: http://www.reuters.com/resources/archive/us/YYYYMMDD.html.

2.3. Volatility estimation

Volatility was estimated as variance of log returns over each interval,[7] and updated every hour using minute by minute data, *i.e.* from 60 returns, with returns computed as ratios of close prices at adjacent points in time:

$$r_i = \log\left(\frac{S_i}{S_{i-1}}\right)$$

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(r_i - \overline{r})^2,$$

where S_i is the asset price at time i, r_i, the asset return at time i and $\overline{r}$, the mean return over the analysis window of N time points.

2.4. Machine learning

Latent Dirichlet Allocation (LDA) followed by naïve Bayes classification was the overarching machine learning model used in this study. LDA, introduced in,[38] is widely seen as a powerful way of extracting *topics* across a range of text data. More recently, outside of computational finance, the method has been used in modelling of disease subtypes in cancer, where molecular level heterogeneity is analogous to the variability seen in documents representing multiple underlying topics.[39] Naïve Bayes, though making restrictive statistical assumptions, has proved to be popular in text classification tasks due to its simplicity and high empirical performance.

2.4.1. Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA)[38] is a generative model that allows a set of documents (observations) to be categorised into their underlying topics as a method of feature reduction. A topic is a set of words, where each word has a probability of appearance in documents labelled with the topic. Each document is a mixture of corpus-wide topics, and each word is drawn from one of these topics. From a high-level, topic modelling extrapolates backwards from a set of documents to infer the topics that could have generated them - hence the generative model. The underlying algorithm for assigning topics to documents is collapsed Gibbs sampling. LDA effectively reduces features and produces a set of comprehensible topics. However, on the downside it is relatively computationally expensive (polynomial time complexity).[40–42]

2.4.2. Naïve Bayes classification

Naïve Bayes is one of a wide range of supervised learning algorithms available for pattern classification. The algorithm's popularity in text classification stems from its simplicity. Specifically, in a multivariate classification setting, naïve Bayes models each feature as being independent of the others. This is a simplistic assumption in the case of text processing because when text is represented as a normalised bag-of-word frequencies, word occurrence frequencies are likely to be correlated. However, empirical experience as well as the need for large amounts of data to

estimate all pairwise correlations at high dimensions (in p dimensions, we need to estimate $p^2/2$ correlations) robustly tend to favour the use of this classification approach.[43]

Thus, the motivation for selecting naïve Bayes as a classifier is that they are simple to implement, perform similarly to more complex methods, such as support vector machines,[44] and also require only a relatively small amount of training data to converge (estimate the parameters) as the whole covariance matrix doesn't need to be determined due to the independence assumption. Also in one case,[45] a NB-LDA approach, with a similar design to ours, produced positive results.

3. Text processing

We used Python packages urllib and requests to fetch the news data, and BeautifulSoup to parse the HTML and extract information between specified tags. The system filters news on a corpus-level using a keyword-based approach, constructing a set of financial terms and discarding articles that did not contain at least one of these terms. Appendix C shows the financial terms used in the retrieval process, which were constructed by hand using a small holdout set of news articles. Filtering keywords were added to the set until categories of news with a lower perceived relevance (*e.g.* sport and opinion) were discarded, leaving only market, political, business and world news categories in the holdout set.

Extracted text was then subject to standard natural language processing steps of tokenizing and stemming. Stop-words, short words (length < 3) and Web addresses were removed. These steps were implemented using the NLTK toolbox (WordPunctTokenizer for tokenisation and PorterStemmer for stemming). A word list constructed from NLTK, augmented by the list used in[46] was used for stop-word removal. Removal of web addresses used a regular expression matching rule.

3.1. Topic modelling

To model the semantics of the article text and reduce the number of features, we used Latent Dirichlet Allocation (LDA), using the article body and title as separate topic models and then combining the resultant topic distributions into a single feature vector. The LDA implementation used was 'Online Latent Dirichlet Allocation' within the gensim[47] Python library, which produces topic models using a 'memory-friendly' stream.[48] Memory constraints were a major consideration given the size of the corpus used. The topic modelling algorithm took approximately 30 min of computing to converge when processing each 3 month period of textual data. The resulting topic model is a number of corpus-derived topics that form a distribution. Each article is assigned a number of topics from this distribution.

We chose the following parameters for our LDA model: Topics-body = 100; Topics-title = 20; LDAPasses-body = 3; LDAPasses-title = 20; LDAChunks-body = 2000; and LDAChunks-title = 200. Hierarchical Dirichlet Process (HDP), which is a non-parametric topic model similar to LDA, was used to initialise the number of topics.

3.1.1. Example topic model

The output of LDA is a set of topics, each topic contains unigram/bigram tokens with a weighting applied to them. The weighting describes to what extent this token contributes to the overall topic.

Table B.14, Appendix B is an example topic and captures foreign affairs articles concerning the Middle East in terms of war and oil, due to stemmed tokens like: "oil", "iran", "syria", "rule", "attack", "assad", "govern", "presid", "militari", "foreign" and "al_qaeda". Note the bigram token "al_qaeda" is a major component of the topic, though irrelevant to our goal, the semantic commonality seems to be correctly extracted by the model.

3.2. Naïve Bayes prediction model

From the topic model, a list of topics was constructed of size 120 (100 body topics, 20 title topics) for each 60 min prediction interval: this list is one feature vector in the feature space defined by the topics. The feature vector for an interval is a topic-count sparse vector, representing the number of times each topic appears in articles within the interval. Some topics may appear more than once, and some not at all. The target vector is then constructed by pairing binary direction labels from market data to each feature vector: the label for the next 60 min period is used, as we are predicting this. For instance, we are using news from 09:00AM–10:00AM to predict the direction of movement

(increase/decrease) of volatility over period 10:00AM−11:00AM and direction of movement of close price from 10:00AM to 11:00AM.

Internally, a dictionary is used as a mapping between each topic and the number of occurrences of the topic over a 60 min interval. This feature vector of topic frequency has a corresponding binary target vector for each period.

With the above representation, we trained a multinomial naïve Bayes model. The model was run over 3 month periods, with the first half of data used for training and the second half for testing. Increasing the size of this temporal window reduced the accuracy of classification, on the validation set, which is clearly attributable to non-stationarity in financial markets.

Following,[49] we trimmed the temporal window over which predictions are evaluated, excluding the first hour and the last half-hour of trading in any day, as these periods tend to feature irregular patterns of trading, potentially biasing the model.

A classic problem with such textual representations in fixed dimensional vector spaces is probabilities being set to zero during intervals in which none of the terms/topics are observed. This is dealt with by smoothing, and the "Lidstone smoothing parameter" (α) was set to 0.01.[50]

3.3. Dealing with non-stationary data

Financial information is known to be non-stationary; *i.e.* any functional relationship we learn over short periods of time, the mapping between news and market movement in our case, can change with time. Hence, in estimating models over a temporal window, we might expect distant past to be of less importance than recent past. An approach we used to account for this observation was implementing a decay function to weight the news data when training the models. The 60 min window of news was split into 6 intervals and the articles were weighted in a decreasing manner across each interval as follows:

$$w_i = w_{i-1} - 0.15,$$

where $i \in \{0...5\}$ and $w_0 = 1.0$, with interval i=0 being closest to the prediction period and i=5 being furthest back (50−60 min) in time.

The weighting was then applied to the topic model by multiplying the number of topics across an interval by the weight, w. During ad-hoc testing, marginal performance improvements were observed (~0.5% accuracy) when using this weighting on the analysis window.

3.4. Bigram model

Another enhancement to the basic model was the use of bigrams, thought to capture a greater level of semantic information than unigrams. For example, in our data set, phrases such as "money_market" or "bailout_money" can be captured as a single token when represented as bigrams, whereas in the unigram model the tokens would be separated and so the context and true meaning of 'money' as being related to the money markets would be lost in the first case. Similarly, in the second case, the negative implication of money used for bailing out an entity would not be captured, and a singular unigram token 'money' might wrongly be seen as a positive one. Extracting bigrams is straightforward within the NLTK library and collocation detection was performed using pointwise mutual information as criterion. We did not find trigrams to be useful features, probably due to the inevitable data sparsity issue.

3.5. Chi-squared feature reduction

The feature set constructed in this way is of very high dimensionality. In order to reduce the dimensions, thus to deal with the problem of the "curse of dimensionality", we implemented a Chi-squared feature selection method. In our model we rank by χ^2 and select the 30 best features/topics from this.

An alternative way of dealing with high dimensions is to project the data onto the principal subspace obtained from a Principal Component Analysis (PCA). During experimental comparisons, Chi-squared feature selection produced accuracy results with less variance. Further, PCA projections, not being constrained to be positive, sometimes produced negative outputs which were incompatible with our classifier.

3.6. Luhn's cut-off

A fundamental statistical aspect of word usage in natural language relates to a skewed distribution known as Zipf's law, which states that the frequency of usage of a word is in inverse proportion to its rank in a usage frequency table. Consequently, the most frequent term occurs twice as frequently as the second most frequent term which occurs twice as frequently as the third, and so on. To avoid difficulties caused by the dominance of frequently used words in learning and rare words only appearing in the training and test sets, upper and lower cut-offs are enforced on the usage frequency table, known as Luhn cut-offs.[51] This is then reflected in the dictionary. The resulting significant words have greater discriminative power than words outside the cut-offs.

In our implementation we used: UpperCutoff-body = 0.5; UpperCutoff-title = 0.75; LowerCutoff-body = 3; and LowerCutoff-title = 2. Lower cutoffs are an absolute number of documents, and terms that occur in less than this amount were excluded. Upper cutoffs are a percentage of documents. Upper and lower cut-offs were specified using a gensim method, which filters the dictionary. The exact location of the cut-offs are tunable parameters which we set by analysing a small validation set of the data.

4. Technical analysis model

4.1. Rationale

A parallel model was constructed using features derived from the time-series, rather than using news as input for predictions. This model is a benchmark to assess whether or not volatility is an 'easier' prediction problem in general. For instance, in the case where both volatility and asset price are as predictable as one another in a time-series model, any claim that a news-based model has a greater ability to predict volatility than asset price would be made stronger, as a model using a different set of explanatory variables found no significant difference in predictability. Furthermore, comparing the performance of predictions based solely on time-series data against those based solely on news data over similar prediction periods provides some performance context for the news-based classifier, especially when making literature comparisons, as much of the historic literature is centred around using time-series data alone for such forecasts, with only more recent literature looking at alternative data.

The technical analysis model makes use of various technical indicator functions, which transform price data to provide additional value over exclusively using price data as input. Technical indicators (Appendix D) were chosen based on previous literature usage[52–56] and their combined ability to capture as many market properties and dynamics as possible, while keeping the feature vector as small as possible, to attempt to avoid problems associated with high dimensional feature sets ('The Curse of Dimensionality'). The indicators span momentum oscillators, trend identifiers, mathematical smoothing and volatility categories.

4.2. Implementation

Time series data was decimated from 1-min to 5-min resolution in order to produce 12 *open, high, low, close* (OHLC) data points over each 60 min period. Technical indicators were derived at each data point, resulting in 12 values per indicator for each 60 min period. In total, nine technical indicators encapsulate various market properties, with their calculation and rationale described in greater detail in Appendix D. On top of this, OHLC were input alongside OHLC average and volatility, although these are not strictly technical indicators, they provided some additional signal. This resulted in 15 indicators in total, producing $15 \times 12 = 180$ features for each 60 min period.

The final feature vector represents a 'lagged' input from t to $t-55$ for each input feature, with up/down volatility and close label of the next period as target, in the same manner as the news-based predictor. Technical indicator calculation was performed in the well-known TA-Lib[d] library, using Python bindings to call C++ functions. Input feature vectors were scaled to fit a Gaussian distribution with 0 mean and unit variance before training and testing of the learning algorithm took place.

[d] www.ta-lib.org

A support vector machine was favoured as the underlying predictive algorithm over other methods, due to its ability to handle relatively high dimensional data, and to position the hyperplane such that the margin between classes is maximised. This property leads to robust classification. Refer to[57] for a thorough explanation of the mathematics involved. Contrary to a similar model by Kim,[52] a linear kernel yielded better results on the model tuning data set than radial basis or polynomial kernels. A final C value of 20 was used, as too low values resulted in over-generalisation and high training error, and too high values were over-fitting the training data. Model selection and tuning occurred on the same validation set as the news-based predictor (2014/15 data), however the temporal window was larger, spanning 22-09-2011−07-09-2012. As normal, training occurred on the first half of the dataset and testing on the second half. Technical analysis model predictions are produced from 22-09 rather than 09-09 (for the news model) as the technical indicator functions require a look-back portion of data before values can be produced. Longer training and testing windows are used because, on the validation set, it was observed that the technical analysis predictor didn't exhibit the same level of non-stationary behaviour as the news predictor, so didn't require short training windows, and in fact produced better results with longer training windows. This is expected, as patterns learnt from time series data are likely to change less readily than patterns based on news data, where over time news topics evolve more freely from bullish to bearish sentiment or vice versa.

5. Benchmarks and measures

Various measures were used for performance evaluation of the news-based model (Model_N), and to test the relationship between news-derived information and market movements, briefly outlined below:

5.1. Single direction (SD)

Over short windows of time, market movements can be monotonic (Bear or Bull). Hence always predicting the direction of movement as being Up or Down is a useful benchmark.

5.2. Random walk (RW)

The random walk benchmark predicts that whatever happened in one period will happen in the next. Over long periods of time, the accuracy of random walk converges to 50%.

5.3. Technical analysis model ($Model_{TA}$)

The technical analysis model forms predictions based solely on time-series data.

5.4. Classifier performances

To quantify classification performance, we used several measures taken from information retrieval literature: accuracy, recall, precision, F1 score and Matthews correlation coefficient (MCC). With TP representing True Positives, FP representing False Positives (and TN and FN denoting True and False Negatives, respectively) these performance measures are defined as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{F1} = \frac{2(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

$$MCC = \frac{TP*TN - FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

Using a range of measures as performance metrics, instead of accuracy of classification alone to quantify results, is important in this context because the positive (Up) and negative (Down) movements in the market data can be highly unbalanced.

We used Welch's unequal variances *t*-test on the accuracy values to compare performances with the null hypothesis H_0 being that volatility is as predictable from news as close price and a *p*-value threshold of 0.05 to reject H_0. Further, we confirmed using D'Agostino and Pearson's test that both volatility and close price data were normally distributed, hence satisfying the requirement for applying the *t*-test.

6. Results

Table 1 shows the performance of predicting changes in close prices and volatilities using the different performance measures considered during the 122 60 min periods defined from September to December 2011, after excluding periods with missing data and the start/end of day periods as discussed previously. We note that in prediction accuracies and the F1 measure, model based prediction of volatility changes is significantly better than those of market close prices. Further, this was a period during which the **Dow** Index remained fairly stable while the **NASDAQ** showed a decline. This is reflected in the performances of the **SD Down** and **SD Up** predictors of close prices of these indices, both remaining at 50% for **DJI** and **SD Down** showing an accuracy of 53.3% on **NASDAQ**. **Model$_N$** predictions using news information, however, do not perform any better than these when predicting close prices. In predicting index volatilities, however, while **SD Down** and **SD Up** perform no better than chance, **Model$_N$** predictions are at 56.6% for **DJI** and 61.5% for **NASDAQ**. Similar observations can be made about the two equity close prices and volatilities, too. Similar, detailed results of predicting volatility and price movements across these securities, partitioned into four different temporal segments and evaluated by the various criteria are given in Appendix A.

Table 2 summarises the detailed results by taking weighted averages (weighted by the number of data) of each of the analysis windows and assets, a total of 1695 different test samples. The summary shows that prediction accuracies of volatilities using information derived from news are better than those of predicting close price of the underlying assets, to a high level of statistical significance. Furthermore, when comparing each of these against a random walk prediction strategy, we see that predicting volatility movements is significantly better than the RW model, whereas in predicting close prices, this is not the case. Fig. 1 also summarises the same information as boxplots and shows the average performance of random walk models (bold dot on figure).

Table 3 shows the same data and comparisons for **Model$_{TA}$**. The most notable result here is that there is no statistically significant difference in accuracy between predicting volatility and close price using time-series data. Furthermore, close price prediction accuracy significantly outperforms the random walk benchmark in this case, contrary to the results of **Model$_N$**. This helps strengthen the hypothesis that volatility is more predictable than close price when using news-derived information as input, as these results help show that it isn't the case that volatility is an 'easier' prediction problem.

Table 1
Performance of predicting directional changes in close price and volatility of various assets during the period 09/09/2011 to 07/12/2011, with first half of the period for training and the second for testing.

Security	Model$_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
Close Price									
DJI	49.2	53.3	50.0	50.0	49.2	49.0	47.1	−0.018	122
NASDAQ	53.3	59.8	46.7	53.3	53.3	54.3	52.9	0.079	122
Goldman Sachs	48.7	58.0	52.9	47.1	48.7	46.8	45.6	−0.061	119
J.P. Morgan	51.7	45.8	48.3	51.7	51.7	51.1	48.4	0.019	120
Volatility									
DJI	56.6	46.7	49.2	50.8	56.6	58.2	54.8	0.150	122
NASDAQ	61.5	44.3	50.0	50.0	61.5	63.0	60.3	0.244	122
Goldman Sachs	50.4	46.2	47.9	52.1	50.4	51.7	48.3	0.028	119
J.P. Morgan	62.5	40.0	45.0	55.0	62.5	62.2	62.1	0.235	120

Table 2
Summaries of comparisons of $\mathbf{Model_N}$ and random-walk predictions of volatility and close price. Test for significance shows clear difference between predicting volatilities and close prices from news-derived information. While $\mathbf{Model_N}$ prediction accuracies of volatilities are significantly better than that of a random-walk model, this does not hold for predicting asset price movement. $N = 16$ intervals used, containing a total of 1695 prediction periods.

	Volatility		*Close price*
Mean	55.566		49.386
StdDev	4.212		3.772
P value		0.0001 (<0.05)	
	Volatility		*RW*
Mean	55.566		41.302
StdDev	4.212		3.494
P value		0.0000 (<0.05)	
	Close price		*RW*
Mean	49.386		50.331
StdDev	3.772		5.581
P value		0.5795 (>0.05)	

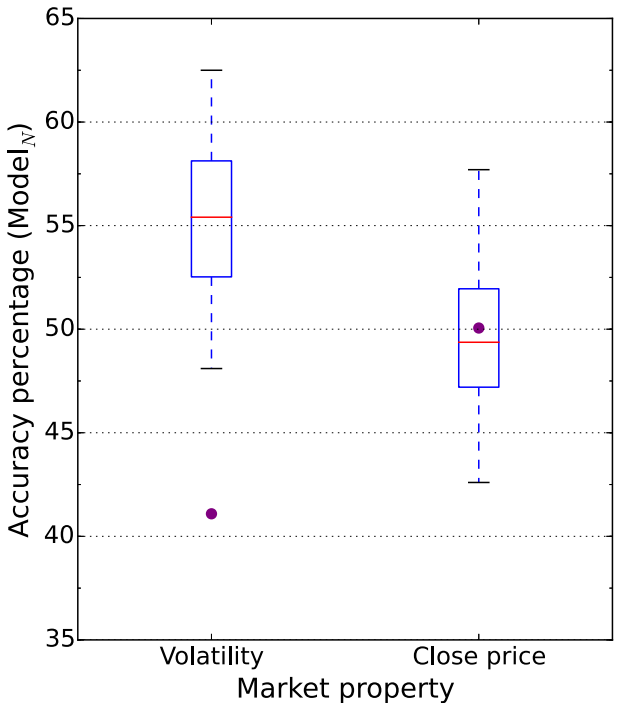

Fig. 1. Summary of prediction accuracies of volatilities and close prices over the 12 month period 09/09/2011−09/09/2012. The box plot shows the variation across the different financial instruments, evaluated at different time periods (see Appendix A for individual results). The solid horizontal lines show the mean accuracies and the filled circles show random walk accuracies of the two models.

Although the random walk benchmark is common in literature,[58] it may not be fully applicable to volatility. The volatility random walk accuracy is consistently a lot lower than the random walk for close price and far from the expected 50% convergence, due to the decreased likelihood of finding adjacent periods of increasing/decreasing volatility. A solution to this is to use pseudo-random number generation to define the random benchmark, or in our case to define a static benchmark of 50%. It should be noted that when using a 50% accuracy benchmark with zero standard deviation, *t*-test p-values were still within the same significance thresholds and were largely unchanged.

Table 3
Summaries of comparisons of $\mathbf{Model_{TA}}$ and random-walk predictions of volatility and close price. Contrary to the above, both volatility and close price $\mathbf{Model_{TA}}$ accuracies significantly out-perform the random-walk benchmark and furthermore there is no significant difference in accuracy between predicting volatility and close price using time-series data. $N = 4$ intervals used, containing a total of 2275 prediction periods.

	Volatility		*Close price*
Mean	55.533		56.375
StdDev	2.910		0.868
P value		0.6179 (>0.05)	
	Volatility		*RW*
Mean	55.533		39.940
StdDev	2.910		1.510
P value		0.0007 (<0.05)	
	Close price		*RW*
Mean	56.375		49.908
StdDev	0.868		1.819
P value		0.0030 (<0.05)	

The mean accuracy of volatility classification for $\mathbf{Model_N}$, taken across all of the time slices of evaluation and over all the considered assets is 55.6%, whilst the accuracy of classifying close price direction is 49.4%. Note that across the different assets and time intervals of evaluation, the maximum and minimum accuracies for volatility prediction are 62.5% and 48.1% whereas the corresponding values for close price predictions are 57.7% and 42.6%. Thus, while volatility is predicted with higher accuracy and the average prediction performance appears small, it is possible to see parts of data in which the claims made could be far higher. This, we believe is the case in previous literature, showing empirical results over much smaller datasets without giving details of variation shown here.

7. Conclusions and discussion

In this paper, we have shown empirical results that information extracted from textual news sources can be used in predicting directional changes in market volatility. In particular, we have shown that the changes in volatility are better predicted than the changes in the close price of an asset or an index of assets. Though the inability to predict close price movement any better than random contradicts previously published results, our results suggest that information in news, influencing markets via sentiment-driven behaviour essentially affects second order statistics of the financial system.

The result that second order statistics better relate to news can indeed be exploited in trading, via pricing of derivatives. Our current research focuses on the relationship between news-derived information and implied volatilities of underlying stocks, computed by inversion of option pricing formulae (*e.g.* Black-Scholes,[7]). We are also interested in the relationship to explicit models of volatility, such as stochastic volatility models, where we believe parameters characterizing the dynamics of volatility can be extracted from news-derived information. Comparisons should be made between forecasted volatility and the implied and realised volatility of near term at-the-money options, to ascertain whether or not volatility movements are expected, and priced into the options market, or if there exists some exploitable inefficiency.

It is important to note that the prediction results we report are based only on information derived from the news source and no other input is used. Hence the accuracy results cannot be directly compared with accuracies reported in the literature on models that make predictions based on time series (predictions from past values). It is striking that news-derived information alone contains a level of signal strong enough to predict the direction of movement of market volatility. Clearly, a model that includes even just previous volatility values is likely to display higher levels of accuracy. Future work, using either $\mathbf{Model_N}$ or $\mathbf{Model_{TA}}$ as part of a trading strategy, should consider the magnitude of volatility or asset price changes. This magnitude should subsume the bid-ask spread plus any transaction costs incurred, in order to be profitable. As directional changes in volatility may in some cases be too small to trade, one solution is to adapt the model to form a regression, or to trade securities known for large

directional changes, based on historical analysis. Also, DJI and NASDAQ Composite predictions may be traded through (volatility) index tracker funds, as they are not tradable securities, unlike GS and JPM. Tracking error between the tradable security and the index may affect profitability, unless predictions were made on tracker fund prices instead.

Aside from the novelty of the empirical study and results, the discrete decay function for dealing with non-stationary news data will be of use to researchers or practitioners developing similar models. Although, a smoother continuous function, chosen based on historical data, may model the phenomena more accurately and could yield more significant performance improvements.

The freely available news source we used was not pre-filtered in any way, lacks any accompanying structured metadata, and is likely to contain a significant amount of noise, contributing to the higher standard deviation in accuracies for $\mathbf{Model_N}$ compared to $\mathbf{Model_{TA}}$. In the financial setting, it is possible to purchase pre-filtered and targetted news (from Reuters itself, for example). Such news is likely to improve prediction accuracies further, leading to a stronger input signal for a trading strategy. In addition to this, our future work will be focused on systematic optimization of the many model parameters (*e.g.* topic model and cut-off parameters in the vocabulary section). The feature selection method, based on the Chi-squared criterion, considers features one at a time. More elaborate methods such as greedy forward selection (known as wrapper methods) and their joint optimization with the performance metrics (*e.g.*[59]) also forms part of our current research. Moreover, the TA benchmark classifier features were selected manually, based on literature usage. More thoughtful feature selection methods, such as genetic algorithm optimisation of a wider feature set, may improve predictive accuracy.

In terms of computational model performance of $\mathbf{Model_N}$, to maintain reasonable running time on a single 2.5 GHz Intel Core i7 machine, a compromise was taken on topic model perplexity. Model accuracy improvement (lower perplexity) is possible by making use of the distributed computing capabilities of the gensim topic modelling library, allowing for more iterations under the same time constraints, although these gains would likely be outweighed by acquiring cleaner news data, model selection and hyperparameter optimisation. $\mathbf{Model_N}$ predicts market movements using patterns learned from macro news activity, with no company/stock/sector specific filtering involved. The advantage of this approach is that the large sample of news articles produces a dense topic distribution over each 60 min prediction window. Subsequent issues arising from highly sparse data are avoided, and don't need to be considered by the predictive model. High-level filtering (*e.g.* at a sector-level) may still lead to appropriately dense topic distributions, so can be considered for future developments.

As noted in the introduction, our results are based on a wide range of assets, split into temporal windows to deal with non-stationarity in the data, which is in contrast to several publications in recent literature which quote empirical results on selective periods in time (*e.g.*[49]). Hence we believe the hypothesis we advance about news-derived information being a better predictor of volatility than of market price changes is robustly demonstrated in this work.

Conflicts of interest

None.

Acknowledgment

The authors would like to thank Professor Akiko Takeda, University of Tokyo, for helpful discussions during the course of this research.

Appendix A. Detailed results

Here, we give detailed results on model and benchmark prediction accuracies. The news-based model ($\mathbf{Model_N}$) is evaluated across four three-month periods, and the technical analysis model ($\mathbf{Model_{TA}}$) is evaluated across a single one year period. In each period, the first half of data is used for training and the second for testing. The results below are the predictive accuracy of the model and benchmarks on the test set.

Appendix A.1. News-based model results

Appendix A.1.1. Predicting changes in volatility

Table A.4
Predicting change in volatility: 09/09/2011–09/12/2011.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	56.6	46.7	49.2	50.8	56.6	58.2	54.8	0.150	122
NASDAQ	61.5	44.3	50.0	50.0	61.5	63.0	60.3	0.244	122
Goldman Sachs	50.4	46.2	47.9	52.1	50.4	51.7	48.3	0.028	119
J.P. Morgan	62.5	40.0	45.0	55.0	62.5	62.2	62.1	0.235	120

Table A.5
Predicting change in volatility: 09/12/2011–09/03/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	52.8	38.7	47.2	52.8	52.8	52.5	52.4	0.046	106
NASDAQ	54.8	40.4	48.1	51.9	54.8	54.7	54.7	0.093	104
Goldman Sachs	59.4	42.6	47.5	52.5	59.4	59.3	58.9	0.182	101
J.P. Morgan	56.7	40.4	45.2	54.8	56.7	56.4	56.4	0.119	104

Table A.6
Predicting change in volatility: 09/03/2012–09/06/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	57.4	37.0	44.4	55.6	57.4	57.2	57.3	0.134	108
NASDAQ	48.1	40.7	45.4	54.6	48.1	48.1	48.1	−0.046	108
Goldman Sachs	53.0	42.0	35.0	65.0	53.0	52.0	52.5	−0.054	100
J.P. Morgan	59.8	44.9	40.2	59.8	59.8	60.0	59.9	0.167	107

Table A.7
Predicting change in volatility: 09/06/2012–07/09/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	54.6	44.3	43.3	56.7	54.6	58.2	54.0	0.137	97
NASDAQ	57.7	39.2	45.4	54.6	57.7	57.5	57.6	0.143	97
Goldman Sachs	51.7	34.5	48.3	51.7	51.7	52.5	50.7	0.046	87
J.P. Morgan	49.5	35.5	47.3	52.7	49.5	51.2	46.4	0.017	93

Appendix A.1.2. Predicting changes in asset price

Table A.8
Predicting change in close price: 09/09/2011–09/12/2011.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	49.2	53.3	50.0	50.0	49.2	49.0	47.1	−0.018	122
NASDAQ	53.3	59.8	46.7	53.3	53.3	54.3	52.9	0.079	122
Goldman Sachs	48.7	58.0	52.9	47.1	48.7	46.8	45.6	−0.061	119
J.P. Morgan	51.7	45.8	48.3	51.7	51.7	51.1	48.4	0.019	120

Table A.9
Predicting change in close price: 09/12/2011–09/03/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	47.2	45.3	59.4	40.6	47.2	45.4	46.0	−0.131	106
NASDAQ	52.9	46.2	53.8	46.2	52.9	52.7	52.7	0.048	104
Goldman Sachs	42.6	53.5	43.6	56.4	42.6	44.1	42.5	−0.132	101
J.P. Morgan	43.3	44.2	49.0	51.0	43.3	43.3	43.1	−0.133	104

Table A.10:
Predicting change in close price: 09/03/2012−09/06/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	47.2	58.3	53.7	46.3	47.2	48.2	46.5	−0.039	108
NASDAQ	50.9	56.5	51.9	48.1	50.9	50.8	50.8	0.015	108
Goldman Sachs	45.0	49.0	42.0	58.0	45.0	46.7	45.3	−0.092	100
J.P. Morgan	47.7	46.7	56.1	43.9	47.7	49.6	47.3	−0.021	107

Table A.11
Predicting change in close price: 09/06/2012−07/09/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	57.7	42.3	52.6	47.4	57.7	59.2	57.1	0.174	97
NASDAQ	51.5	48.5	49.5	50.5	51.5	52.2	49.3	0.039	97
Goldman Sachs	48.3	48.3	54.0	46.0	48.3	46.3	45.9	−0.076	87
J.P. Morgan	52.7	45.2	46.2	53.8	52.7	53.0	52.8	0.055	93

Appendix A.2. Technical analysis model results

Appendix A.2.1. Predicting changes in volatility

Table A.12
Predicting change in volatility: 22/09/2011−07/09/2012.

Security	$Model_N$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	58.3	37.7	47.8	52.3	58.3	58.8	56.5	0.162	575
NASDAQ	53.7	39.8	47.5	52.5	53.7	53.4	53.0	0.064	570
Goldman Sachs	58.4	40.4	48.0	52.0	58.4	59.3	58.0	0.180	565
J.P. Morgan	51.7	41.9	49.4	50.6	51.7	51.8	51.6	0.035	565

Appendix A.2.2. Predicting changes in asset price

Table A.13
Predicting change in close price: 22/09/2011−07/09/2012.

Security	$Model_{TA}$	*RW*	*SD Up*	*SD Down*	*Recall*	*Precision*	*F1*	*MCC*	N
DJI	57.0	50.6	51.3	48.7	57.0	57.1	56.4	0.139	575
NASDAQ	54.9	52.3	50.2	49.8	54.9	55.2	54.1	0.101	570
Goldman Sachs	56.6	47.3	47.4	52.6	56.6	56.6	56.6	0.129	565
J.P. Morgan	57.0	49.4	48.5	51.5	57.0	57.0	57.0	0.140	565

Appendix B. Example topic model

Table B.14
Example LDA topic model output, for one topic. Tokens are multiplied by their respective weight/probability within the topic. The topic as a whole is the sum of all tokens occurring in this topic, multiplied by their respective weight.

Weight	Token		Weight	Token		Weight	Token	
0.005	Govern	+	0.005	State	+	0.004	Would	+
0.004	Al_Qaeda	+	0.004	Last	+	0.004	Countri	+
0.004	Oil	+	0.004	Iran	+	0.004	Peopl	+
0.003	One	+	0.003	Forc	+	0.003	Secur	+
0.003	Two	+	0.003	Offici	+	0.003	Syria	+
0.003	Kill	+	0.003	Unit	+	0.003	Also	+
0.003	Presid	+	0.003	Minist	+	0.003	Month	+

(*continued on next page*)

Table B.14 (*continued*)

Weight	Token		Weight	Token		Weight	Token	
0.003	Told	+	0.003	Group	+	0.002	Nation	+
0.002	New	+	0.002	Syrian	+	0.002	Could	+
0.002	Power	+	0.002	Assad	+	0.002	Attack	+
0.002	Foreign	+	0.002	Militari	+	0.002	Polit	+
0.002	Includ	+	0.002	Intern	+	0.002	Plan	+
0.002	China	+	0.002	May	+	0.002	Week	+
0.002	Day	+	0.002	Back	+	0.002	Parti	+
0.002	Time	+	0.002	Sinc	+	0.002	Million	+
0.002	South	+	0.002	Protest	+	0.002	Call	+
0.002	Elect	+	0.002	Rule				

Appendix C. Financial terms

Table C.15
List of allowed financial terms, for article filtering.

Acquisition	Disaster	Gdp	Liabilities	Shareholders
Assets	Dollar	Germany	Liquidity	Shares
Australia	Domestic	Gold	Luxembourg	Singapore
Austria	Dow	Goldman	Market	Spain
Balance	Earnings	Greece	Markets	Spread
Bank	Earthquake	Gross	Merger	Stability
Bankrupt	Economic	Growth	Morgan	Stanley
Bankruptcy	Economist	Hedge	Mortgage	States
Belgium	Economy	Hong	Nasdaq	Sterling
Blackrock	Egypt	Hsi	Netherlands	Stock
Bloomberg	England	Hurricane	Norway	Stocks
Bonds	Eur/Usd	Index	Oil	Suisse
Budget	Euro	Inflation	Options	Sweden
Canada	Europe	Interest	Peace	Switzerland
Capital	European	Invest	Portugal	Syria
China	Exchange	Investment	Pound	Terrorism
Collapse	Fall	Investors	Product	Turkey
Cost	Finance	Ireland	Profit	UK
Credit	Financial	Israel	Rate	Union
Crisis	finland	Italy	Recession	United
Csi300	Foreign	J.P.	Revolution	USA
Currency	Forex	Japan	Rise	Usd/eur
Debt	France	Jones	Russia	Usd/gbp
Default	funds	Jpm	S&p500	Volatility
Denmark	Gains	Kingdom	Sachs	War
Derivatives	Gbp/usd	Kong	Sales	Zealand

Appendix D. Technical indicators

Table D.16
Indicators used in the technical analysis time series model, on 5-min resolution data. C_t, H_t, L_t are close, high and low prices at t, respectively. t follows 5 min increments.

Indicator	Calculation	Description & Rationale
Exponential moving average (EMA)	$ema(t) = \frac{2}{n+1} \times (C_t - ema(t-1)) + ema(t-1)$ *where* $n = 6$	Greater weight is given to recent data, so the indicator reacts more quickly to price changes than Simple Moving Average. Look-back period of 30 min appropriate for hourly predictions.

Table D.16 (*continued*)

Indicator	Calculation	Description & Rationale
Commodity channel index (CCI)[60]	$cci(t) = \frac{tp(t) - sma_{period=n}(tp(t))}{D_c \times md(t)}$ $where \quad tp(t) = \frac{H_t + L_t + C_t}{3}; \quad D_c = 0.015; \quad n = 20;$ $md(t) = \frac{\sum_{i=t}^{t-n} \lvert sma_{period=n}(tp(t)) - tp(i) \rvert}{n};$ $sma_{period=n} = simple\ moving\ average,\ look - back\ n\ mins$	Measures mean 'true price' deviation to indicate trends and extreme market conditions, as well as overbought and oversold scenarios.
Average directional movement index (ADX)[61]	$adx(t) = \frac{adx(t-1)(n-1) + dx(t)}{n}$ $where \quad n = 14; \quad dx = directional\ movement\ index$ [61]	Non-directional indicator, measuring trend strength. ADX is calculated with a look-back of 14 min.
Kaufman adaptive moving average (KAMA)[62,63]	$kama(t) = kama(t-1) + sc(t) \times (C_t - kama(t-1))$ $where \quad sc(t) = (er(t) \times (p_f - ps) + p_s)^2$ $and \quad er(t) = \frac{\lvert C_t - C_{p_{er}} \rvert}{\sum_{i=t}^{t-p_{er}} \lvert C_i - C_{i-1} \rvert};$ $p_{er} = 10; \quad p_f = 2; \quad p_s = 30$	Adjusts for noise by incorporating volatility into moving average. Useful for smoothing short time periods containing wide swings followed by price re-adjustments.
Average true range (ATR)[61]	$atr(t) = \frac{tr(t-1)(n-1) + tr(t)}{n}$ $where \quad tr(t) = TrueHigh(t) - TrueLow(t); \quad n = 14$ $and \quad TrueHigh(t) = \max(H_t, C_{t-1});$ $TrueLow(t) = \min(L_t, C_{t-1})$	A measure of volatility, using a look-back period (n) of 14 min.
Momentum (MOM)[52]	$mom(t) = C_t - C_{t-4}$	Simple trend identification by quantifying the change of a security's price over time.
Williams' %R[60]	$\%R(t) = 100 \times \frac{H_t - C_t}{H_t - L_t}$	Aids in identifying trends and overbought/oversold scenarios.
Rate of change (ROC)[64]	$roc(t) = \left(\frac{C_t}{C_{t-n}} - 1\right) \times 100$	Momentum indicator that measures percentage change in price between two periods. Look-back (n) is 10mins.
Relative strength index (RSI)	See[61,60]	A momentum indicator, which attempts to capture whether a security has been overbought or oversold by comparing the magnitude of recent gains to recent losses.

References

1. Glantz M, Kissell R. *Multi-asset Risk Modeling: Techniques for a Global Economy in an Electronic and Algorithmic Trading Era*. Academic Press; 2013.
2. Reuters. *Stock Options Trading Volume Strong Even as Volatility Dips*; June 2017. https://uk.reuters.com/article/us-usa-stocks-options-idUKKBN18T2V6.
3. Settlements BFI. *Bis Statistical Bulletin*; March 2017. http://www.bis.org/statistics/bulletin1703.pdf.
4. Markowitz H. Portfolio selection. *J Finance*. 1952;7(1):77−91.
5. Brodie J, Daubechies I, De Mol C, Giannone D, Loris I. Sparse and stable markowitz portfolios. *Proc Natl Acad Sci*. 2009;106(30):12267−12272.
6. Takeda A, Niranjan M, Gotoh J-y, Kawahara Y. Simultaneous pursuit of out-of-sample performance and sparsity in index tracking portfolios. *Comput Manag Sci*. 2012;10(1):21−49.
7. Hull JC. *Options, Futures, and Other Derivatives, Pearson*. Upper Saddle River (N.J.): Prentice Hall; 2006.
8. Hamilton J. *Time Series Analysis*. Princeton, NJ: Princeton Univ. Press; 1994. http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+126800421&sourceid=fbw_bibsonomy.

9. Wilmott P. *Paul Wilmott Introduces Quantitative Finance*. 2nd ed. New York, NY, USA: Wiley-Interscience; 2007.
10. Armano G, Marchesi M, Murru A. A hybrid genetic-neural architecture for stock indexes forecasting. *Information Sciences*. 18 February 2005;170(1):3−33. https://doi.org/10.1016/j.ins.2003.03.023. Computational Intelligence in Economics and Finance. http://www.sciencedirect.com/science/article/pii/S002002550300433X.
11. Chenoweth T, Obradovic Z, Lee SS. Embedding technical analysis into neural network based trading systems. *Appl Artif Intell*. 1996;10(6):523−542.
12. Emir S, Dinçer H, Timor M. A stock selection model based on fundamental and technical analysis variables by using artificial neural networks and support vector machines. *Rev Econ Finance*. 2012;2:106−122.
13. White H. Economic prediction using neural networks: the case of ibm daily stock returns. In: *Neural Networks, 1988., IEEE International Conference on*. IEEE; 1988:451−458.
14. Zekic M. Neural network applications in stock market predictions-a methodology analysis. In: *Proceedings of the 9th International Conference on Information and Intelligent Systems*. vol. 98. Citeseer; 1998:255−263.
15. Li Y, Bontcheva K, Cunningham H. *Deterministic and Statistical Methods in Machine Learning: First International Workshop, Sheffield, UK, September 7-10, 2004. Revised Lectures*. Berlin, Heidelberg: Springer Berlin Heidelberg; 2005:319−339. https://doi.org/10.1007/11559887_19. Ch. SVM Based Learning System for Information Extraction.
16. Ferrucci DA. Introduction to "this is watson". *IBM J Res Dev*. 2012;56(3):235−249. https://doi.org/10.1147/JRD.2012.2184356.
17. Suwannaroj S, Niranjan M. Enhancing automatic construction of gene subnetworks by integrating multiple sources of information. *J Signal Process Syst*. 2008;50(3):331−340. https://doi.org/10.1007/s11265-007-0148-4.
18. Sudhahar S, De Fazio G, Franzosi R, Cristianini N. Network analysis of narrative content in large corpora. *Nat Lang Eng*. 2015;21:81−112. https://doi.org/10.1017/S1351324913000247. http://journals.cambridge.org/article_S1351324913000247.
19. Kloptchenko A, Eklund T, Karlsson J, Back B, Vanharanta H, Visa A. Combining data and text mining techniques for analysing financial reports: research articles. *Int J Intell Syst Account Finance Manag*. 2004;12(1):29−41. https://doi.org/10.1002/isaf.v12:1.
20. Schumaker RP, Chen H. Textual analysis of stock market prediction using breaking financial news: the azfintext system. *ACM Trans Inf Syst*. 2009;27(2):12:1−12:19. https://doi.org/10.1145/1462198.1462204.
21. Fung GPC, Yu JX, Lu H. The predicting power of textual information on financial markets. *IEEE Intell Inform Bull*. 2005;5(1):1−10.
22. Gidofalvi G, Elkan C. *Using News Articles to Predict Stock Price Movements*. San Diego: Tech. Rep.; 2001.
23. Moat HS, Curme C, Avakian A, Kenett DY, Stanley HE, Preis T. Quantifying wikipedia usage patterns before stock market moves. *Sci Rep*. 2013;3.
24. Loughlin E, Harnisch Chris. *The Viability of Stocktwits and Google Trends to Predict the Stock Market*; 2014. http://stocktwits.com/research/Viability-of-StockTwits-and-Google-Trends-Loughlin_Harnisch.pdf.
25. Preis T, Moat HS, Stanley HE. Quantifying trading behavior in financial markets using google trends. *Sci Rep*. 2013;3.
26. Bollen J, Mao H, Zeng X. Twitter mood predicts the stock market. *J Comput Sci*. 2011;2(1):1−8.
27. Asgharian H, Sikström S. In: *Predicting Stock Price Volatility by Analyzing Semantic Content in Media*. vol. 38. Working Paper/Department of Economics, School of Economics and Management, Lund University; 2013.
28. Franses PH, Van Dijk D. Forecasting stock market volatility using (nonlinear) garch models. *J Forecast*. 1996:229−235.
29. Hansen PR, Lunde A. A forecast comparison of volatility models: does anything beat a garch (1, 1)? *J Appl Econom*. 2005;20(7):873−889.
30. Poon S-H, Granger CW. Forecasting volatility in financial markets: a review. *J Econ Lit*. 2003;41(2):478−539.
31. Gavrishchaka VV, Ganguli SB. Volatility forecasting from multiscale and high-dimensional market data. *Neurocomputing*. 2003;55(1):285−305.
32. Allen DE, McAleer M, Singh AK. *Machine News and Volatility: The Dow Jones Industrial Average and The TRNA Sentiment Series*; January 2014. jEL: C58, G14 http://eprints.ucm.es/24356/.
33. Robertson CS, Geva S, Wolff RC. News aware volatility forecasting: Is the content of news important?. In: *Proceedings of the Sixth Australasian Conference on Data Mining and Analytics*. vol. 70. Australian Computer Society, Inc; 2007:161−170.
34. Oliveira N, Cortez P, Areal N. On the predictability of stock market behavior using stocktwits sentiment and posting volume. In: *Progress in Artificial Intelligence*. Springer; 2013:355−365.
35. Zadeh RB, Zollmann A. *Predicting Market-Volatility from Federal Reserve Board Meeting Minutes NLP for Finance*. 2009.
36. Schumaker RP. Analyzing parts of speech and their impact on stock price. *Commun IIMA*. 2014;10(3):1.
37. Dumais S. Using svms for text categorization. *IEEE Intell Syst*. 1998;13(4):21−23.
38. Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. *J Mach Learn Res*. 2003;3:993−1022.
39. Saddiki H, McAuliffe J, Flaherty P. Glad: a mixed-membership model for heterogeneous tumor subtype classification. *Bioinformatics*. 2015;31(2):225−232.
40. Porteous I, Newman D, Ihler A, Asuncion A, Smyth P, Welling M. Fast collapsed gibbs sampling for latent dirichlet allocation. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM; 2008:569−577.
41. Sontag D, Roy D. Complexity of inference in latent dirichlet allocation. In: *Advances in Neural Information Processing Systems*. 2011:1008−1016.
42. Xiao H, Stibor T. *Efficient Collapsed Gibbs Sampling for Latent Dirichlet Allocation*. 2010.
43. Zhang H. The optimality of naive bayes. *AA*. 2004;1(2):3.
44. Colas F, Brazdil P. Comparison of svm and some older classification algorithms in text classification tasks. In: *Artificial Intelligence in Theory and Practice*. Springer; 2006:169−178.
45. Zhang Y, Ji D-H, Su Y, Wu H. Joint naive bayes and lda for unsupervised sentiment analysis. In: *Advances in Knowledge Discovery and Data Mining*. Springer; 2013:402−413.
46. Georgiadis A. *A News-driven Approach to Stock Prediction*. Southampton: Tech. rep.; 2014.

47. Rehurek R, Sojka P. *Software Framework for Topic Modelling with Large Corpora*. Valletta: ELRA; 2010:45−50.
48. Hoffman MD, Blei DM. Online learning for latent dirichlet allocation. *Adv Neural Inf Process Syst*. 2010;1:858−864.
49. Schumaker RP, Chen H. A discrete stock price prediction engine based on financial news. *Computer*. 2010;1:51−56.
50. Trippi RR, Turban E. *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real World Performance*. McGraw-Hill, Inc; 1992.
51. Luhn HP. The automatic creation of literature abstracts. *IBM J Res Dev*. 1958;2:159−165.
52. Kim K-j. Financial time series forecasting using support vector machines. *Neurocomputing*. 2003;55(1):307−319.
53. Krollner B, Vanstone B, Finnie G. *Financial Time Series Forecasting with Machine Learning Techniques: A Survey*. 2010.
54. Tay FE, Cao L. Application of support vector machines in financial time series forecasting. *Omega*. 2001;29(4):309−317.
55. Kara Y, Boyacioglu MA, Baykan ÖK. Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the istanbul stock exchange. *Expert Syst Appl*. 2011;38(5):5311−5319.
56. Patel J, Shah S, Thakkar P, Kotecha K. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Syst Appl*. 2015;42(1):259−268.
57. Cortes C, Vapnik V. Support-vector networks. *Mach Learn*. 1995;20(3):273−297.
58. Huang W, Nakamori Y, Wang S-Y. Forecasting stock market movement direction with support vector machine. *Comput Oper Res*. 2005;32(10):2513−2522.
59. Scott M, Niranjan M, Melvin D, Prager R. Maximum realisable performance: a principled method for enhancing performance by using multiple classifiers. In: *Proceedings of the British Machine Vision Conference*. 1998.
60. Achelis SB. *Technical Analysis from A to Z*. New York: McGraw Hill; 2001.
61. Wilder JW. *New Concepts in Technical Trading Systems*. Trend Research; 1978.
62. Kaufman PJ. In: *Trading Systems and Methods*. vol. 591. John Wiley & Sons; 2013.
63. Kaufman PJ. *Smarter Trading*. New York: McGraw-Hill; 1995.
64. Murphy JJ. *Technical Analysis of the Futures Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance; 1999.