Prediction of Patterns in Financial Time Series in Response to External Events

Bachelor Thesis

Alexei Pankov

Faculty of Economics
National Research University
Higher School of Economics

Supervisor: Alexey Artemov, Ph.D.,
Associate Professor, Faculty of Computer Science
Researcher, Skolkovo Institute of Science and Technology

This is a work-in-progress version of the thesis. Some pieces are subject to change. Please ensure you have the most recent edition:
https://github.com/apnkv/hse/blob/master/thesis.pdf

Contents

1 Introduction

1.1 Background

Stock markets

Stock markets are considered to represent current economic conditions and react lively to their changes. It is observed that stock prices respond immediately to key economic and political events as well as company reporting, whereas it can take weeks to months for them to be reflected in conventional measures of economic activity, such as macroeconomic statistics, business charts, investment ratings, etc.

For various agents, it is desirable to know how exactly various events affect financial markets. Such information can be used to gain extra profits in trading, to evade excess risk in portfolio management, to plan investment strategies against various hypothetical scenarios, to design optimal economic policies, and in many other applications.

Typical measurable stock market reaction to events include immediate price changes, periods of elevated returns or volatility, and others. There can be different reasons of such reaction. Many studies have emphasised the impact of macroeconomic news releases on stock market movements, a survey of such examples can be seen in (Funke and Matsuda 2006).

Such news include changes in interest rates, publication of official GDP growth estimates, unemployment numbers, and others.

Non-numeric information

Many different approaches have been devised to quantify inflows of non-numeric information to the market. For a particular company stocks, a rather simple approach is using the number of mentions of the company in the news. For example, (Alanyali, Moat, and Preis 2013) show that the count of mentions of a company in the daily news is correlated with trading volumes of the corresponding stock on that day and the day that follows (and also suggest that a correlation with absolute returns may exist).

Most recent studies make use of the more advanced data mining and natural language processing (NLP) techniques. Algorithms purposed for sentiment analysis, topic modelling, named entity recognition etc. are used to extract structured data for problems of returns time series regression, categorical prediction (e. g. whether the stock price is going to rise/fall/remain steady), and econometric analysis involving hypothesis testing about distribution parameters (see Nassirtoussi et al. 2014 for a survey on text mining in market prediction).

1.2 Problem statement

This paper addresses a particular task of categorical prediction: forecast the direction of daily closing stock price, given that a company has recently published its earnings figures. The research is built upon a framework suggested by H. Lee et al. 2014. It is shown that models utilizing textual features, such as the content of 8-K forms, or recent company-related news, improve the accuracy of the 3-way prediction (UP/STEADY/DOWN relatively to the market trend) by around 4 %, and this advancement is statistically significant. An attempt is made to apply the method to Russian stocks.

The rest of the paper is set as follows. In Section 2, a literature review is presented. In Section 3, details are given on data preparation. In Section 4, an experimental setup from the original paper is reproduced, with detailed description of data retrieval and preprocessing. It is verified by the reproduction of original results. An evaluation of the model is performed using updated dataset versions. In Section 5, an attempt is made to apply the model to the Russian market, specifically, to the constituents of the MOEX index. Section 6 describes key definitions, methodological tools and algorithms that are used throughout the work. Section 7 lists some implications and limitations and suggests possible research directions.

2 Literature review

Explanation and prediction of stock markets behaviour has been a popular topic of economic analysis for a long time. Initially, it was widely believed that financial time series follow a random walk pattern and thus cannot be predicted fundamentally. In 1965, the efficient market hypothesis was introduced by Eugene Fama, stating that current stock prices incorporate all information about previous events and change only in response to unexpected events as information becomes available to the market. The key point behind the efficient market hypothesis is that information is a very, if not the most, important component in prices. Though this statement implies that it is very difficult to outperform the market (because once someone has the relevant information, he or she immediately applies this knowledge to take advantage in the market), it does not rule out the theoretical possibility that relevant information might already be out there, concealed in immense streams of data that nobody has handled yet.

Correct use of existing and incoming information is discussed in the context of trading strategy planning (see Hong, Lim, and Stein 2000) and considered in hedging models, policy planning and many other cases (see Funke and Matsuda 2006).

Another direction of research of stock prices behaviour stems from the

very process of investment and trading. Traders buy and sell in response to information they get from news regarding their assets of interest. Thus, in principle, the news should bear necessary information to predict market movement to some extent. This role of news has been emphasised many times in the literature, for instance, in (Pearce and Roley 1984; Cutler, Poterba, and Summers 1988) in the context of returns and in (King and Wadhwani 1990) in the context of volatility. However, there is no complete understanding, let alone consensus in how to embed news into computations. Many different approaches have been devised to quantify information inflows to the market, including construction of numerical proxies such as amount of mentions of the company's name in daily news (Alanyali, Moat, and Preis 2013), trading volumes and others. While these news-derived indicators are helpful in discovering some useful correlations and some prediction tasks (for example, trading volumes are significant in volatility forecasting [Lamoureux and Lastrapes 1990]), they lack consideration of probably the most important features of news: their semantic content and described relations between economic actors.

Advances in this direction were made with the emergence and development of natural language processing methods and tools. Many solutions have been introduced: for instance, representing documents as count distribution of encountered words (or a so-called bag of words) and extracting event information (Ding et al. 2016). One particular useful NLP approach is to create word- and document embeddings: their representations in

vector spaces such that their proximity can be compared according to some metric corresponding to such space. This way, one can, for example, tag news as positive or negative, or know the topic of a piece of news: say, acquisition or expansion to a foreign market. For example, sentiment analysis has been widely used, yielding a stable accuracy improvement in the task of forecasting immediate stock price direction (Bollen, Mao, and Zeng 2011). Such NLP-based predictive techniques have been applied to various forecasting problems, but attained the greatest success within the problem of stock price movement direction (see Nassirtoussi et al. 2014 for reference).

3 Data preparation

Several datasets of texts are used throughout the work for different cases. One is the collection of 8-K filings, provided by H. Lee et al. 2014. In this dataset, 8-K texts are annotated with timestamps of their publication and the form fulfillment reason, making it comfortable to align them with price changes and generate valuable categorical features. However, these texts needed to be preprocessed.

The texts were cleaned from tables and numeric values since these cannot be used efficiently without sophisticated knowledge mining systems. All entries were also stripped off SEC administrative preambles, remainders

of the form fulfillment user interface, and other financially unimportant content, e. g.:

> `Check the appropriate box below if ... any of the following provisions:`
>
> - `Written communications pursuant to Rule 425 under the Securities Act (17 CFR 230.425)`
> - `Soliciting material ... (17 CFR 240.14a-12) ...`

Tables, numbers and unrelated content were removed using Python programming language. The total number of 8-K filings in the dataset is approximately 180 000, of which 14 607 entries belong to the period 2002–2012 and have corresponding EPS and price features. In total, 8-K filings of this period contain 42 787 290 words.

For Russian case, there also exist forms that are completed in case of significant corporate event. They are published continuously on (*e-disclosure.ru* 2018) along with timestamps and event categories. The number of such forms available for each company ranges from tens to thousands. Their texts are usually not as informative as typical 8-K filings, because frequently they consist of rather bureaucratic language amid a lot of insignificant information, and a large part of it is purely organisational. See an example:

```
1. Общие сведения 1.1. Полное фирменное наименование
эмитента (для некоммерческой организации -
наименование): Акционерное общество "ННК-Хабаровский
нефтеперерабатывающий завод" 1.2. Сокращенное
фирменное наименование эмитента: АО "ННК-Хабаровский
НПЗ" 1.3. Место нахождения эмитента: 680011,
Российская Федерация, г. Хабаровск, ул. Металлистов,
17. 1.4. ОГРН эмитента: 1022701129032 1.5. ИНН
эмитента: 2722010040 ...
```

Another text dataset used in the case of Moscow Exchange is a collection of headlines relevant to the constituents of the MOEX index, obtained from Finam site, where they have been pre-selected to appear on the site as somewhat relevant to the target stock. The text features were constructed by joining a Finam title with the same day's disclosure form.

4 S&P 500

4.1 Model selection

First, of all 8-K forms, only those corresponding to companies belonging to S&P 500 were left. Data on prices and earnings were subsampled so that the dataset is complete, i. e. each observation has a price record, an 8-K text and an EPS surprise figure.

The resulting dataset consists of 14 607 samples. The samples were divided into three classes as follows. First, the series of returns were calculated for each stock in the data. The returns were normalized to the S&P 500 index, as described in the methodology section. Depending on normalized returns, the following classes were assigned to each observation: UP if normalized returns $\tilde{r}_t > 0.01$, DOWN if $\tilde{r}_t < -0.01$ and STEADY otherwise. Such division yielded an approximately uniform distribution of the classes, which can be seen in the following table.

	UP	STEADY	DOWN
count	5142	4674	4791
percentage	35.2 %	32.0 %	32.8 %

Table 1: Normalized returns class distribution

Further, the data were separated chronologically. Years 2011–2012 were put out for testing, 2009–2010 for hyperparameter tuning, and 2002–2008 were used as a training set. Among the subsets, class distribution is very similar to the general sample.

4.1.1 Deterministic EPS surprise baseline

To ensure the following experiments, a very simple yet powerful baseline was applied: predict UP if EPS surprise was positive, DOWN if negative and STEADY if the consensus earnings forecast was at par with actual

called earnings. This baseline turned out to show accuracy around 47.2 % on the testing set and around 44.3 % on the whole sample. It is worth commenting on class recall—how many samples of a class were labelled as belonging to this class: A rather high recall of the UP class tells that

	UP	STEADY	DOWN
Recall	78.4 %	13.9 %	29.8 %

Table 2: Class recall by pure-EPS baseline

when a firm exceeds its earnings estimations, it is often a positive market signal: in such situations, returns on its stocks tend to outperform the index. Another thing to be learned is from low recall on the DOWN class: unmet earnings expectations are not the only reason for a stock price to fall.

4.1.2 Financial features model

Several financial features were generated and collected prior to text processing to test another purely financial, yet advanced baseline, including as features cumulative (normalized) returns over a month, a quarter and a year, 8-K event category labels and returns on the VIX ticker, a proxy index for volatility expectation.

Since not all of 8-K fulfillment categories are directly connected to in-

vestors' expectations, not all of them are relevant to price direction. A feature selection was performed to choose the category indicators connected the most to the target variable. It was done by calculating so-called Pointwise Mutual Information for each of the category features with the class vector.

This measure reflects codependency between samples of two random variables. A more detailed explanation can be found in the Methodology section. Having chosen categories with the greatest PMI, we filter our feature set to contain features more connected to the target variable. Fifteen category indicators were selected and included in the model as dummy feature columns (see table 3).

Out of all event category columns, only these fifteen were kept. Eventually, the dataset for the second model contained the EPS surprise column, 15 category dummy columns, 3 columns of cumulative returns over a month, a quarter and a year, and returns on the VIX index.

To address the classification task in a non-trivial case, several algorithms were assessed, including a logistic regression, decision trees, nearest neighbors classifiers, SVM and random forests. The highest accuracy on the hyperparameter tuning set was achieved by a random forest with 2050 individual tree estimators, assigning not fewer than 15 samples to each leaf node, making splits of not fewer than 250 samples on each step. A detailed explanation of these parameters is given in the methodology

Regulation FD Disclosure
Material Impairments
Cost Associated with Exit or Disposal Activities
Entry into a Material Definitive Agreement
Termination of a Material Definitive Agreement
Non-Reliance on Previously Issued Financial Statements or a Related Audit Report or Completed Interim Review
Material Modifications to Rights of Security Holders
Changes in Registrant's Certifying Accountant
Submission of Matters to a Vote of Security Holders
Unregistered Sales of Equity Securities
Change in fiscal year
Amendments to Articles of Incorporation or Bylaws
Temporary Suspension of Trading Under Registrant's Employee Benefit Plans
Creation of a Direct Financial Obligation or an Obligation under an Off-Balance Sheet Arrangement of a Registrant
Results of Operations and Financial Condition

Table 3: Most relevant 8-K categories

section. On the test set, the random forest classifier with optimised hyperparameters achieved accuracy of 49.9 %.

4.1.3 Models with textual features

To employ text features from 8-K forms, a text processing pipeline was constructed. Firstly, the texts were cleaned from metadata, timestamps

and unrelated service pieces as described in the previous section. All words were cut down to their stems to count various forms of one word as one term, for instance, 'acquired' → 'acquir', 'acquiring' → 'acquir'. All words that encountered in all texts fewer than ten times were removed, leaving around 5000 features out of 21 000. It is still a huge number with respect to computation cost and relatively small sample size, so, feature selection and factorization techniques were applied. All auxiliary and frequently occurring words with little meaning were removed according to the following list:

```
stopwords = ('the', 'to', 'of', 'by', 'on', 'this', 'be',
'it', 'report', 'and', 'as', 'in', 'a', 'has', 'with', 'or',
'from', 'that', 'are', 'at', 'not', 'will', 'an', 'were',
'more', 'herein', 'hereunto', 'all', 'over', 'no', 'hereto',
'do', 'exchang', 'secur', 'act', 'for', 'is', 'behalf',
'registr', 'financi', 'statement', 'may', 'was', 'million',
'per', 'our', 'year', 'these', 'requir', 'caus', 'pursuant',
'includ', 'result', 'compani', 'author', 'sign', 'undersign',
'duli', 'quarter', 'oper', 'other', 'file', 'exhibit', 'inform',
'releas', 'share', 'which', 'materi', 'end', 'forward-look',
'we', 'busi', 'relat', 'use', 'billion', 'if', 'two', 'under',
'should', 'can', 'but', 'both', 'also', 'such', 'into', 'same',
'their', 'said', 'through', 'have', 'those', 'than', 'then')
```

After that, for each text, a vector was constructed in a way that each coordinate corresponded to a unique term among all encountered in texts,

and value in that coordinate was the number of occurrences of this unique term in the text. All such vectors were written as rows of a matrix to obtain a term count matrix representation of texts in the dataset.

Simple word counts can be unreliable to compare texts, since words are not uniformly distributed in the language (i. e. auxiliary words are more frequent than scientific or business terms). To build a predictive model upon textual features, one would desire that domain-identifying features had greater weights. To achieve this, a so-called TF-IDF transformation was applied. It is a mapping that assigns higher weights to more 'unique' features, i. e. those occurring frequently in a document, but not often in the whole corpus. This transformation is discussed in detail in the methodology section.

Having a lot of features in the model still poses a potential threat of overfitting and inability of the model to generalize. To cope with this threat while retaining information provided by the features, a factorization technique called NMF was used. It is an approximate decomposition of a non-negative matrix into a product of two non-negative matrices, minimizing the difference between the product and the original matrix. One of matrix multipliers can be viewed as containing 'latent features' related to the original ones—in this case, linear combinations with non-negative coefficients. A positive side effect of such decomposition is the natural clustering property—latent features 'consist' (are weighted sums)

of frequently co-occurring source features and thus, in a language-related task, act as proxies of n-grams, collocations and even topics. This property is covered in the methodology section.

After all text transformations, 50 features were obtained. After attaching them to the data and training a new classifier, an accuracy of 53.7 % was achieved.

The model with textual features turned out to be significantly better than both baselines, as suggested by approximate randomization test (Hoeffding 1952; Noreen 1989). Let B_1 be the deterministic baseline model, B_2 the model with only financial features, T_1 the random forest with financial features and all frequently occurring terms, and T_2 the random forest with financial features and text-derived NMF factors. Also, let acc(M) be the accuracy of model M. Let H_0 be the hypothesis that subsequent (more advanced) model does not differ from the preceding one. For such setup, the p-values are shown in table 4.

H_0	$\mathrm{acc}(\mathrm{B}_2) = \mathrm{acc}(B_1)$	$\mathrm{acc}(\mathrm{T}_1) = \mathrm{acc}(B_2)$	$\mathrm{acc}(\mathrm{T}_2) = \mathrm{acc}(T_1)$
p-value	0.013*	0.0252*	0.045*

Table 4: p-values for the hypotheses of model equivalence

All p-values are less than 0.05, indicating that:

- Using previous returns, 8-K event category and volatility expecta-

tion index along with EPS surprise feature is better than forecasting price direction judging by EPS surprise alone;

- Using text features extracted from disclosure texts further improves the accuracy;
- Factorization helps when handling text features.

4.2 Model testing on new data

To further prove the decent power of the constructed model, another series of experiments were conducted. An extension of the dataset was collected from 2013 until 2018. For each of companies in S&P 500 index, all 8-K filings issued within this period were fetched from SEC database (*Security Exchange Commission EDGAR* 2018) and processed in a way described in the previous subsection.

As of today, Yahoo Finance API are deprecated for free use, so price and earnings data had to be obtained from other sources.

Earnings surprise feature was gathered from Zacks, a renowned investment research agency (*Zacks.com Earnings Calendar* 2018). Zacks Research can be deemed the most competent source of such data. Initially, the company was founded for this very purpose—to provide a wide range of earnings estimates. Its founder, Len Zacks, received his PhD for the

development of the idea that earnings estimate revisions are the most powerful force impacting stock prices.

Additional stock prices data were fetched from IEX 5-year free archive of daily ticks (*The Investors Exchange* 2018). However, GSPC ticker, the S&P 500 index data, was missing from this archive. Since it is crucially required at the returns normalization step, the series were downloaded separately from a Kaggle-hosted dataset (*Kaggle SP500 Stock Data* 2018). Kaggle is a famous platform for hosting data analysis and machine learning competitions.

In what follows in this subsection, the data over 2013–2018 were used as a testing set, and the data over 2002–2012—as a training set. The series of experiments was the same as in the previous subsection, so it makes sense to briefly present and interpret the results.

	Random	Maj. class	B_1	B_2	T_1	T_2
Accuracy	33.3 %	41.8 %	45.1 %	47 %	49.3 %	49.7 %

Table 5: Accuracy on the second dataset

An important thing to notice is that class distribution on the test dataset is no longer uniform. Majority class is STEADY, and in particular it causes the EPS deterministic baseline to deteriorate compared to the first case. This baseline is biased to predict UP and DOWN, because it will predict steady only if reported EPS totally equal consensus EPS, which

is rarely the case. Other gains are as expected, except for text features eventually adding only 2.7 % compared to 3.7 % in the first case.

Similarly to the first case, hypothesis testing was conducted to compare the models. Results can be seen in table 10.

H_0	$\text{acc}(B_2) = \text{acc}(B_1)$	$\text{acc}(T_1) = \text{acc}(B_2)$	$\text{acc}(T_2) = \text{acc}(T_1)$
p-value	0.012*	0.039*	0.048*

Table 6: p-values for the hypotheses of model equivalence

A positive result here is that all gains remain statistically significant, so, these experiments prove that using text features in price direction prediction is justified.

5 Moscow Exchange index

In an attempt to conduct a similar research in the context of Russian markets, a smaller dataset was collected in the fashion of the original S&P 500 analysis. It is important to say that the described research cannot be conducted in its full generality on Russian markets, since the features are very sparse. For example, there is no agency that would have consistently published earnings estimates and realizations. For the companies which constitute the MOEX index, consensus EPS forecasts

were obtained using Bloomberg terminal. In the Russian case, this feature is very rare, yielding only 366 samples for 30 firms.

In case if Bloomberg estimates did not provide a quarterly EPS consensus, it was derived from the yearly estimate, subtracting from it the EPS already gained and dividing by quarters left. Simply put, in these cases, an assumption was made that a firm would make earnings in equal parts among quarters left.

After joining the data on prices and EPS surprises, the following class distribution was achieved. Classes appear highly inbalanced, so the

	UP	STEADY	DOWN
percentage	26.9 %	46.5 %	26.6 %

Table 7: Normalized returns class distribution

accuracy metrics alone will not serve well in this case.

5.1 Deterministic baseline

The EPS surprise predictor shows only about 31.6 % accuracy on MOEX dataset. Its low performance is explained by the highly non-uniform class distribution. Class STEADY is dominant in Russian data, yet this deterministic estimator cannot predict 'STEADY' unless reported and

forecasted EPS are totally equal. With MOEX data, the latter is very unlikely, as can be seen in the table of class recalls.

	UP	STEADY	DOWN
Recall	43.5 %	0.7 %	73.8 %

Table 8: Class recall by pure-EPS baseline (MOEX)

5.2 Other experiments

As in the first two cases, accuracy and p-value tables were constructed. The overall accuracy turned out to be poorer than on S&P 500 samples, since all classifiers internally relied heavily on the earnings surprise feature, which in this case does not predict well if used alone (model B_1).

	Random	Maj. class	B_1	B_2	T_1	T_2
Accuracy	33.3 %	46.5 %	31.6 %	33.5 %	35.4 %	36.1 %

Table 9: Accuracy on the third dataset

However, the impact of textual features (disclosure form + news headline) remain pronounced, yet adds only about 1.9 % to the accuracy of the model with financial features.

It can be seen from the table of p-values that adding textual features shows as insignificant ($p = 0.052 > 0.05$). This could possibly be attributed to

H_0	$\text{acc}(B_2) = \text{acc}(B_1)$	$\text{acc}(T_1) = \text{acc}(B_2)$	$\text{acc}(T_2) = \text{acc}(T_1)$
p-value	0.015*	0.052	0.018*

Table 10: p-values for the hypotheses of model equivalence

a small sample, but is still close to significance at 5 %.

6 Methodology

This research is built upon H. Lee et al. 2014. In the paper, authors predict the direction of price movement after an earnings call using several financially motivated features along with text features extracted from 8-K reports, which are filled out by US firms on significant corporate events.

6.1 Key definitions

First, we will define some key financial terms. Let P_t be the price of an asset at time t. In the case of stocks, let P_t be the closing price. Then, *returns* on this asset are defined as $r_t = \dfrac{P_t - P_{t-1}}{P_{t-1}}$. Let us call *normalized returns* the value $\tilde{r}_t = r_t^{\text{stock}} - r_t^{\text{index}}$. This normalization is done to factor out the overall market trend and reflect only dynamics specific to the target company.

Earnings per share, or EPS, can be defined as the difference between net income and dividends on preferred stock divided by the average number of outstanding shares during the period.

Earnings surprise is the difference between earnings called by the company and mean consensus forecast of this value. Higher earnings surprises are believed to bring optimistic news, as it means that a company performed better than expected.

6.2 8-K forms

An 8-K form is a means of reporting which US companies are obliged to fill out and submit to the Security Exchange Commission on current events that can be significant to stakeholders. Table 11 presents some of such event types. Besides obligatory categorical description, 8-K forms usually include a written explanation and interpretation of the facts, for example:

```
IHOP Corp.  (NYSE: IHP) today announced a new strategic
growth plan designed to increase long term shareholder
value.  In 2003, IHOP plans to transition from
company-financed development of new restaurants to
a traditional franchise development model.
```

Thanks to the rather detailed structure of such reports, they can be used

completions of the acquisition of assets
changes in financial conditions
results of operation
change in control board
election or department of directors
change in fiscal year
SEC investigations and checks
equity sales
modifications to shareholder rights
material direct financial obligations
change in accountant

Table 11: 8-K fulfillment reasons

as a source of linguistic features for prediction.

6.3 Tools and algorithms

6.3.1 Random forest

Random forest is a classification algorithm which implements the voting of different decision trees trained on random subsets of features and samples. Samples can be bootstrapped to propagate more data variance into individual estimators. The voting mechanism here is known as Breiman bagging (see Breiman 2001) and helps to smooth out errors of individual decision trees and yield a sensible prediction.

A decision tree is an algorithm which, given an observation, assigns it to a class via a series of 'questions' learned on the training stage. The 'questions' are defined as follows. Given a labeled dataset, a decision tree iteratively splits it by choosing a feature f_i and setting a threshold t such that each sample that has $f_i < t$ goes to one partition and all others go to the other partition. The threshold is chosen so that the resulting split optimizes some determined criterion. The procedure is repeated until in each partition there is not more than maximum allowed observations.

6.3.2 TF-IDF transformation

TF-IDF is a shorthand for term frequency–inverse document frequency, a statistic that reflects the importance of a word to an individual document in the corpus.

Denote t a term in a document d. Also, define two values:

$$\mathrm{TF} = \frac{f(t,d)}{\sum\limits_{t' \in d} f(t',d)}$$

$$\mathrm{IDF} = -\log \frac{n_t}{N}$$

A TF-IDF transformation is a mapping of terms in a document $d_i \mapsto \mathrm{TF}(d_i) \cdot \mathrm{IDF}(d_i)$, where d_i corresponds to the i-th term in document d. This transformation can be viewed as a measure which takes into account the specificity of a word, i. e. it is greater for words occurring in smaller

number of documents. Thus, 'topic'-defining words receive higher values of this measure than auxiliary words, frequent words of language or the whole corpus, even if they occur the same number of times.

6.3.3 Feature selection and dimensionality reduction

Using features bound to individual terms as predictors can arise different problems. Words can be ambiguous in different context and carry little meaning when used without their usual collocations. Moreover, keeping too many features in the model can lead to the curse of dimensionality.

To achieve a more stable model, vectors obtained from texts are mapped to a smaller dimension. This is done using the technique called *non-negative matrix factorization*.

Non-negative matrix factorization is a feature extraction approach rather similar to PCA. It maps vectors to vectors of smaller dimensions via linear combinations, but all coefficients are kept non-negative.

The problem that is solved by NMF is, for a given non-negative matrix V, to find a decomposition $V \approx WH$, where W and H are also non-negative matrices. Quality of approximation is introduced by choosing a cost function. One of widely used examples is the Euclidean distance:

$$||A - B||^2 = \sum_{ij} (A_{ij} - B_{ij})^2,$$

which in this case becomes:

$$||V - WH||^2 = \sum_{ij} (V_{ij} - (WH)_{ij})^2 .$$

Euclidean distance vanishes if and only if $A = B$ and is lower bounded by zero. To find its local minima, the following iterative procedure can be used:

$$H_{a\mu} \leftarrow H_{a\mu} \frac{(W^T V)_{a\mu}}{(W^T W H)_{a\mu}} \quad W_{ia} \leftarrow W_{ia} \frac{(V H^T)_{ia}}{(W H H^T)_{ia}}$$

Under this update rule, the Euclidean distance is non-increasing, thus, repeating until convergence, we will eventually arrive at a local minimum with a predetermined precision. An extensive proof and other choices for a cost function can be found in D. D. Lee and Seung 2001.

It is believed that the non-negativity restriction is the reason why NMF performs better than PCA in tasks related to language and image processing.

Linear combinations of such features from words can be thought of as 'topics'—frequently co-occurring word clusters, and we usually model such topics as additive. For example, we would say that a sense conveyed by topic `'appointment of directors'` is rather defined by presence of words `'appointment' + 'director'` than `'appointment' + 'staff' - 'worker'`. The sense of the latter expression just cannot be comfortably perceived. The mapping achieved in such a way is smoother than that of

a PCA. The paper cited two paragraphs above provides more details and interesting applications.

6.3.4 Approximate randomization test

One can build arbitrarily complex models that score great values of accuracy, recall, F-score, ROC-AUC and other metrics, but one value is usually not enough to claim the supremacy of one model against another. If we compare two models, one can perform better than the other out of sheer luck—a suitable dataset, better convergence with limited number of samples, or even a finely-tuned random seed. To correctly determine the best model, statistical tests must be conducted. However, with complex models, there usually do not exist simple statistics to evaluate against. In this case, one can employ the so called approximate randomization test.

It is done as follows. Assume we have two models, A and B, that produce outputs $o_A = \{o_A^1, \ldots, o_A^n\}$ and $o_B = \{o_B^1, \ldots, o_B^n\}$ and want to show that B is better than A under a measure e. Define a 'test statistic' $t = |e(o_1) - e(o_2)|$. Call the hypothesis that there is no difference between A and B the *null hypothesis*. Further,

- start with $X = o_A$, $Y = o_B$
- repeat R times: randomly flip each o_A^i, o_B^j between X and Y with

probability $\frac{1}{2}$. Each time, calculate $t(X, Y)$

- let r be the number of times $t(X, Y) \geq t(o_A, o_B)$
- as $R \to \infty$, the ratio $\dfrac{r+1}{R+1}$ approaches the significance level

This scheme was described in a presentation by (Morgan 2018).

7 Conclusion

As expected, textual features improve the accuracy of models for price direction forecasting on top of conventional financial features.

However, this paper only considers the dates near earnings calls, when very specific, almost structured data, become available. Indeed, it is easier to filter relevant information from already relevant documents (8-K and Russian disclosure filings) than from general news. While it is also proved to be possible, more complex techniques should be used to forecast on a daily basis using news streams.

Limitations

First, the described approach is not suitable for high frequency trading. Making predictions using the proposed steps takes a lot of computing

resources, which can be unattainable or undesirable.

Second, it serves rather as a proof of concept than a means of making trading decisions. No costs were considered, though there may exist many, like transaction costs, borrowing costs to take short positions, and possibly more.

Third, the method implies the existence of reliable and frequent regulatory and news sources. If it is not the case, fewer samples become available to train on, and performance of the model deteriorates.

References

Alanyali, Merve, Helen Susannah Moat, and Tobias Preis (2013). "Quantifying the relationship between financial news and the stock market". In: *Scientific reports* 3, p. 3578.

Bollen, Johan, Huina Mao, and Xiaojun Zeng (2011). "Twitter mood predicts the stock market". In: *Journal of computational science* 2.1, pp. 1–8.

Breiman, Leo (2001). "Random forests". In: *Machine learning* 45.1, pp. 5–32.

Cutler, David M, James M Poterba, and Lawrence H Summers (1988). *What moves stock prices?*

Ding, Xiao et al. (2016). "Knowledge-driven event embedding for stock prediction". In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 2133–2142.

e-disclosure.ru (2018). URL: http://e-disclosure.ru (visited on 2018).

Funke, Norbert and Akimi Matsuda (2006). "Macroeconomic news and stock returns in the United States and Germany". In: *German Economic Review* 7.2, pp. 189–210.

Hoeffding, Wassily (1952). "The large-sample power of tests based on permutations of observations". In: *The Annals of Mathematical Statistics*, pp. 169–192.

Hong, Harrison, Terence Lim, and Jeremy C Stein (2000). "Bad news travels slowly: Size, analyst coverage, and the profitability of momentum strategies". In: *The Journal of Finance* 55.1, pp. 265–295.

Kaggle SP500 Stock Data (2018). URL: https://www.kaggle.com/camnugent/sandp500 (visited on 2018).

King, Mervyn A and Sushil Wadhwani (1990). "Transmission of volatility between stock markets". In: *The Review of Financial Studies* 3.1, pp. 5–33.

Lamoureux, Christopher G and William D Lastrapes (1990). "Heteroskedasticity in stock return data: Volume versus GARCH effects". In: *The journal of finance* 45.1, pp. 221–229.

Lee, Daniel D and H Sebastian Seung (2001). "Algorithms for non-negative matrix factorization". In: *Advances in neural information processing systems*, pp. 556–562.

Lee, Heeyoung et al. (2014). "On the Importance of Text Analysis for Stock Price Prediction." In: *LREC*, pp. 1170–1175.

Morgan, W. (2018). *Statistical Hypothesis Testing for NLP*. URL: https://cs.stanford.edu/people/wmorgan/sigtest.pdf (visited on 2018).

Nassirtoussi, Arman Khadjeh et al. (2014). "Text mining for market prediction: A systematic review". In: *Expert Systems with Applications* 41.16, pp. 7653–7670.

Noreen, Eric W (1989). *Computer-intensive methods for testing hypotheses*. Wiley New York.

Pearce, Douglas K and V Vance Roley (1984). *Stock prices and economic news*.

Security Exchange Commission EDGAR (2018). URL: https://www.sec.gov/edgar/searchedgar/companysearch.html (visited on 2018).

The Investors Exchange (2018). URL: https://iextrading.com (visited on 2018).

Zacks.com Earnings Calendar (2018). URL: https://www.zacks.com/earnings/earnings-calendar (visited on 2018).