Prediction of Stock Market with Sentiment Analysis

Lin Zeng(lz447), Qin Lu(ql224), Sizhang Zhao(sz459)

December 4, 2016

1 Idea

Stock market prediction is the core research area in trading and investment. Stock price is determined by the behavior of human investors, and the investers determine stock prices by using publicly available information to predict the market future trend. Financial news can thus play a significant role in influencing the stock trend as human react to the information. Previous reasearch have suggested that there is some lag between the time news article was released and the time the market absorbed and reflected these information. So our main goal here is to classify the news as our sentiment factor and use the sentiment factor to determine the impact of the news on the stock price.

In the first part of our research, we focus on sentiment analysis with Google news in the whole finance division, and build up classification models to measure the relationship between news sentiment and stock index trend over time. Second, we add the sentiment factor to classify the single stock up and down performances in SVM and neural network models. Finally, we try to design a tradable portfolio based on our sentiment factor and test its application in investments.

2 Data Scraping

Headers of financial news usually contain the impact of a certain event to the whole market. For example, *warn, volatility, decline, plunge* are common phrases that are conceived as negative news to market, and *hike, allow, optimism, gain* are phrases that are considered as positive ones. So in order to capture as much sentiment information as possible for news each day, we use the headers of financial news as our data set instead of the whole article.

We designe a web scraper to scratch the Google news headers in finance division from January - October 2008 in daily basis.

By locating the headers in source code from Google news, we use *BeautifulSoup* package in Python to scratch the headers in the search page by changing the time zone in advanced search. We store news headers scratched from Google news each day in separate *.txt* file as the raw data for sentiment analysis.

We also download the *Dow Jones Industrial Average* (DJIA) as our response variable data from Yahoo Finance.

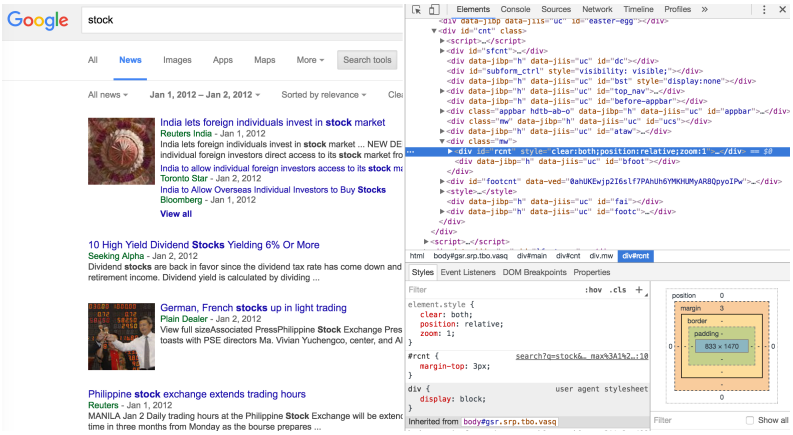

Figure 1: Source code and the search page of Google news

Figure 2: News headers stored in .txt file by our web scraper

3 Sentiment Analysis with Stock News

Our goal here is to develop a system capable of providing information about the polarity in our scratched news documents. We use two types of sentiment scoring machanism to ensure the robustness of our sentiment factor.

First, we use *OpinionFinder*(OF), which is a publicly available software package that processes documents and automatically identifies subjective sentences and sentiment expresssions. The subjectivity analysis provides us three polarity level: *positive*, *neutral*, *negative*.

The polarity classifier takes clues consisting of words with a prior polarity of *positive*, *negative* or *neutral* (for example, *hike*, *drop*, *indicate*, repectively) and thus uses a modified version of the classifier described in Wilson et al. (2005) to determine the contextual polarity of the clues. Heuristics were used to improve the speed of the classifier so it no longer needs the dependency parse output. When evaluated on the MPQA opinion corpus, the overall accuracy is 73.4%.

After obtaining the polarity word list, we mark positive words as 1, neutral as 0 and negative as -1, and take the average of them as our sentiment factor. The sample output of OF is demonstrated as follow.

Second, we count the negative and positive number in each of our word file using *Loughran and McDonald Sentiment Word Lists*, which focuses on financial reports keywords. And construct our sentiment factor by the ratio of negative words to positive words.

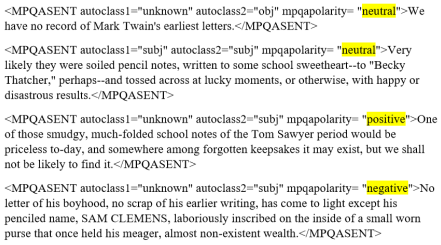

Figure 3: Sample output of OpinionFinder

4 Granger Causality Analysis

We are concerned with the question whether the changes in our sentiment factor is correlated with changes in the stock market, in particular DJIA closing values. To answer this question, we apply the econometric technique of Granger causality analysis to the daily time series produced by sentiment factor and the DJIA. Granger causality analysis rests on the assumption that if a variable X causes Y then changes in X will systematically occur before changes in Y. We will thus find that the lagged values of X will exhibit a statistically significant correlation with Y. We are not testing actual causation but whether one time series has predictive information about the other or not.

Our DJIA time series, denoted as D_t, is the adjusted close price of DIJA at time t. To test whether our sentiment factor time series predicts the changes in stock market value, we compare the variance explained by two linear models as show below. The first model (L_1) uses only the lagged value D_{t-1}, while the second model (L_2) uses the lagged value D_{t-1} and the 3 lagged values of sentiment factor $X_{t-1}, X_{t-2}, X_{t-3}$.

$$L_1 : D_t = \alpha + \sum_{i=1}^{3} \beta_i D_{t-i} + \varepsilon_t$$

$$L_2 : D_t = \alpha + \sum_{i=1}^{3} \beta_i D_{t-i} + \sum_{i=1}^{3} \gamma_i D_{t-i} + \varepsilon_t$$

The results are shown below indicating the adjusted R^2 of L_2 is larger than L_1, this means Granger causality analysis suggests a predictive relation between certain sentiment dimensions and DJIA.

L1 regression report

Dep. Variable:	y	R-squared:	0.819
Model:	OLS	Adj. R-squared:	0.805
Method:	Least Squares	F-statistic:	55.87
Date:	Thu, 03 Nov 2016	Prob (F-statistic):	8.18e-14
Time:	16:18:15	Log-Likelihood:	-23.117
No. Observations:	41	AIC:	54.23
Df Residuals:	37	BIC:	61.09
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[95.0% Conf.	Int.]
const	4.06e-15	0.070	5.81e-14	1.000	-0.142	0.142
x1	0.9803	0.166	5.923	0.000	0.645	1.316
x2	0.0267	0.233	0.115	0.909	-0.446	0.499
x3	-0.1248	0.170	-0.732	0.469	-0.470	0.221

Omnibus:	7.485	Durbin-Watson:	2.024
Prob(Omnibus):	0.024	Jarque-Bera (JB):	6.227
Skew:	-0.825	Prob(JB):	0.0445
Kurtosis:	3.961	Cond. No.	6.83

Figure 4: L_1 regression summary

L2 regression report

Dep. Variable:	y	R-squared:	0.871
Model:	OLS	Adj. R-squared:	0.848
Method:	Least Squares	F-statistic:	38.20
Date:	Thu, 03 Nov 2016	Prob (F-statistic):	1.02e-13
Time:	16:18:15	Log-Likelihood:	-16.223
No. Observations:	41	AIC:	46.45
Df Residuals:	34	BIC:	58.44
Df Model:	6		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[95.0% Conf.	Int.]
const	1.915e-15	0.062	3.11e-14	1.000	-0.125	0.125
x1	0.8412	0.153	5.499	0.000	0.530	1.152
x2	0.1656	0.210	0.791	0.435	-0.260	0.591
x3	-0.1983	0.152	-1.307	0.200	-0.507	0.110
x4	0.2311	0.072	3.195	0.003	0.084	0.378
x5	-0.0541	0.071	-0.757	0.455	-0.199	0.091
x6	0.2135	0.075	2.844	0.007	0.061	0.366

Omnibus:	1.493	Durbin-Watson:	2.027
Prob(Omnibus):	0.474	Jarque-Bera (JB):	1.300
Skew:	-0.422	Prob(JB):	0.522
Kurtosis:	2.778	Cond. No.	7.23

Figure 5: L_2 regression summary

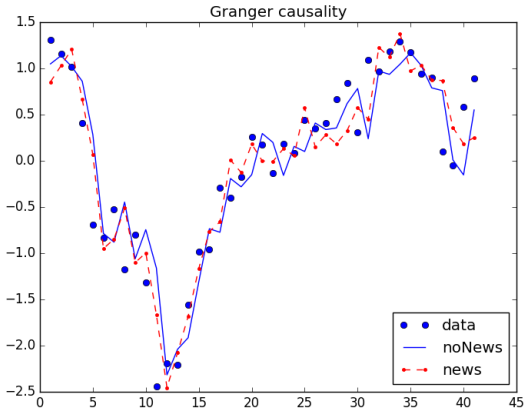

Figure 6: L_1, L_2 prediction of DJIA and true DJIA (normalized)

5 Index Prediction - Classification Approaches

For the purpose of construting a market-tracking portfolio, we care about the up or down of tomorrow's stock index. If we could predict tomorrow's stock marketing is going down, we could go short ETF tracking the Dow *DIA*, and long otherwise.

Before we introduce our classification model, we would like to visualize our sentiment factor.

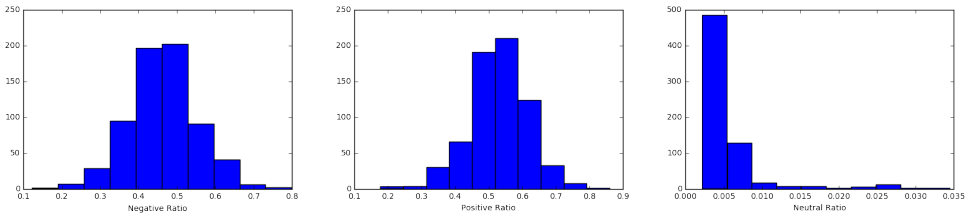

Figure7: LM Sentiment Histogram

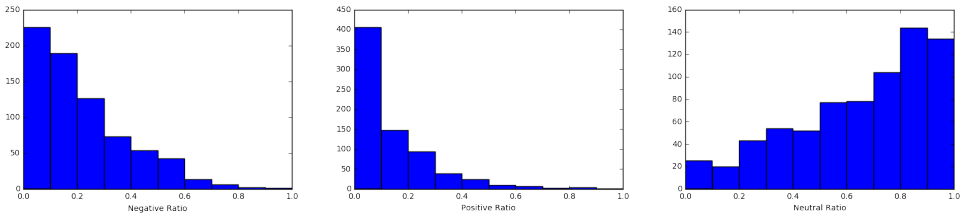

Figure 8: OF Sentiment Histogram

From the histogram, we could see that LM give more sentimental judgement than OF. This might cause by LM is a financial dictionary, which give more accurate sentiment judgement in our news dataset. So in the following prediction, we use LM ratio as our sentiment factor, which is number of negative words divided by number of positive words.

Then we visualize our sentiment factor time-series plot with DJIA daily return to see the pattern.

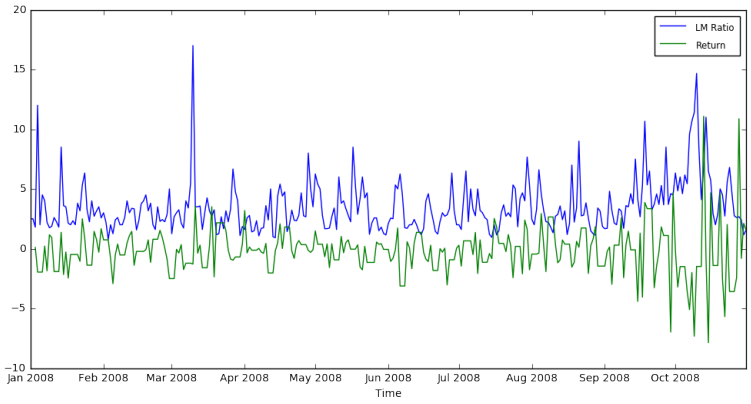

Figure 10: LM ratio and DJIA returns time-series

From this graph, we could clearly see that if our LM ratio increase, which means the stock market sentiment is bearish, we could see the following DJIA return will decrease and be negative. On the other hand, if our LM ratio decrease, indicating the stock market becomes bullish, the following DJIA return will increase and be positive. And we should notice the lag is reasonably small in time, basically around 5 days.

So we use 5-day and 10-day lagged LM ratio to predict the up and down of DJIA by using 30-days floating windows. We fit four different models: logistic loss, SVM, Latent Dirichlet Allocation(LDA) and Quadratic Discriminant Analysis(QDA) by using *ScikitLearn* in Python. The latter two models are popular neural network classification models.

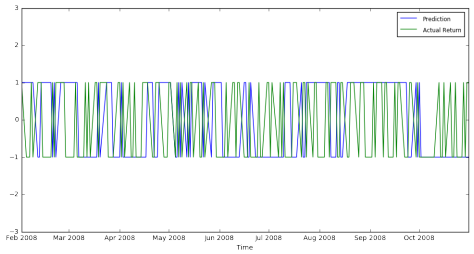

Figure 11: Logistic regression prediction

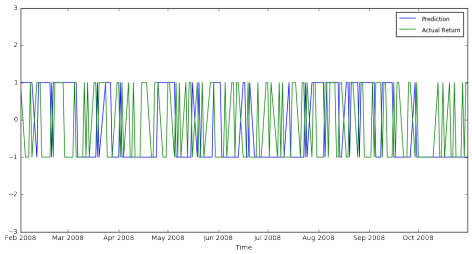

Figure 12: SVM prediction

As we could see from the plots, the prediction results are not as accurate. However, what we find interesting is that our predictor seems like a smoother for the fluctuation of the stock market, which means we predict the overall trend of market well but could not capture daily fluctuation of stock market.

Notice that here we use only lagged sentiment factors to predict the stock index, which misses important information like the historical performance of the index. Also, we find that using the whole finance sector news as sentiment to predict the index trend is of difficulty since we need to assign each news with accurate weight for its impact on stock market. This is something we think makes our prediction not as satisfatory.

Therefore, in the next part, we narrow down our research to predict single stock performance using sentiment factor. We set up a basis prediction model using famous Fama-French 3 factor model, then we add our sentiment factor to see if it improves the prediction accuracy.

6 Single Stock Prediction

In this part, we focus on predicting single stock GE up and down by using data from *Quandl*. The dataset contains average sentiment and impact scores computed on all articles related to GE, published in last 24 hours. Article sentiment(AS) measures the average sentiment of all the articles related to GE, taking values between -1 and 1. Impact score(IS) gives us the average predicted impact of articles on the stock price of GE, taking values between 0 and 100.

We choose the Fama-French 3 factor model as our basis model.

$$r = R_f + \beta_1(r_m - r_f) + \beta_2 SMB + \beta_3 HML + \varepsilon$$

Here r_m stands for market portfolio return, and r_f is the risk-free return rate. SMB stands for "Small (market capitalization) Minus Big" and HML for "High (book-to-market ratio) Minus Low". They measure the historic excess return of small caps over big caps and of value stocks over growth stocks. These factors are calculated with combinations of portfolios composed by ranked stocks and available historical data. Historical values are accessed on *Kenneth French's web page*.

Adding our sentiment factor, our model is formulated as

$$y_t = f\left((\alpha_i(r_m - r_f)_{t-i})_{i=1}^k, (SMB_{t-i})_{i=1}^k, (HML_{t-i})_{i=1}^k, IS_{i-m}, AS_{i-m}\right)$$

where k stands for the lagged period of Fama-French factors, and m is the lagged period for impact score and article sentiment.

For classifier y, we first try binary classification, that is, 1 for positive return and -1 for negative return. Then we reflect that three-class may be a more reasonable choice since there are quiet time in stock market. So in the multiclass classification setting,

$$y_t = \begin{cases} 1 & r_t > c \\ 0 & |r_t| \leq c \\ -1 & r_t < -c \end{cases}$$

where c is the threshold value for distinguish fluctuation and quiet time. $c = 0$ is binary classification.

We use SVM to fitted this model, i.e solving coefficients w by

$$minimize \sum_{i=1}^{n} l_{hinge}(x_i, y_i; w) + \lambda r(w)$$

We use three types of kernel here: linear, polynomial and Gaussian radial basis function. We use l_1 and l_2 as our regularizers $r(w)$.

In addition, we try the multilayer perceptron (MLP), which is a modification of the standard linear perceptron and can distinguish data that are not linearly separable.

Specifically, we use expanding window for fitting the model in this part to use as much information till now as possible. That is, we use data up to the first test day as our training

dataset.

For different threshold value c, we have the following results (in correctness days):

- $c = 0$

Test period	SVM (linear)	SVM (poly)	SVM (Gaussian)	MLP	Basis FF 3 factors (SVM poly)
Aug 1 - 10	6	6	6	5	3
Aug 11 - 20	5	3	4	3	4
Aug 21 - 30	4	6	6	5	4
Sep 1- 10	7	7	6	5	5
Sep 11 - 20	4	6	3	6	3
Sep 21 - 30	8	8	9	6	5
Oct 1 - 10	6	3	5	3	5
Average	5.714	5.571	5.571	4.71	4.14

- $c = 0.1$

Test period	SVM (linear)	SVM (poly)	SVM (Gaussian)	MLP	Basis FF 3 factors (SVM poly)
Aug 1 - 10	6	7	5	4	2
Aug 11 - 20	4	4	3	4	4
Aug 21 - 30	3	4	4	5	4
Sep 1- 10	7	9	7	2	5
Sep 11 - 20	5	5	7	6	4
Sep 21 - 30	7	6	7	6	5
Oct 1 - 10	6	3	5	4	4
Average	5.428	5.285	5.857	4.43	4.00

In the above model, we use the past data to predict the following 10 trading days GE stock's ups and downs. The overall correctness of our prediction is around 58%, which is quite reasonable since we found in previous literature the best prediction is nearly 56%. Comparing to our basis Fama-French 3 factor model, our prediction improve the accuracy of prediction by 37.9%.

We find that MLP does not give us a good result may because that we use nearly 200 trading days data as our training dataset, which might be too small to fit a neural network model, so it might cause overfitting problem. The SVM with linear kernel produce the best result in our test set.

Finally, we construct a simple trading strategy based on our prediction. Here we also use expanding window to fit the model. If we predit GE will go up tomorrow, we long at the close price today. If we predit GE will go down tomorrow, we short the GE at the close price today. And if there are two consecutive rise or fall, we do not change position at the end of today. We plot our cumulative profits as follows with comparison with basis Fama-French 3 factor prediction model.

In this simple trading strategy, we find that adding sentiment could greatly improve our cumulative profits in two-month trading day, compared to the famous Fama-French 3 factor model.

7 Conclusion and Future Work

The goal of our research is to unearth the relationship between the sentiment of stock market and its trend.

In the first part, we generate our sentiment factor by web scraping google news of the whole finance sector. Then we tested the Granger causality to prove there is predictability of our sentiment factor to stock index prices. Then we use logistic loss, SVM, Latent Dirichlet Allocation(LDA) and Quadratic Discriminant Analysis(QDA) to build up four classfication models by 30-days rolling windows. In these model, the only explanatory variables are 5-days and 10-days lagged sentiment factors. As a result, what we find interesting is that our predictor seems like a smoother for the fluctuation of the stock market, which means we predict the overall trend of market well but could not capture daily fluctuation of stock market. So we conjecture that there might be two reasons for the result being not as accurate. First, we use too broad information as our sentiment training set, which might includes redundant information. Also, we might need other explanatory variables as our basis model and add sentiment to see if it improves the prediction.

So in the second part of our research, we narrow down our research to predict single stock performance using sentiment factor. We set up a basis prediction model using famous Fama-French 3 factor model, then we add our sentiment factor to see if it improves the prediction accuracy. We train SVM and MLP (neural network model) model by expanding windows. In the out-of-sample test, our prediction improve the accuracy of prediction by 37.9% comparing to our basis Fama-French 3 factor model.

Finally, we build up a simple trading strategy based on our prediction results. We greatly improve the cumulative profits compared to Fama-French 3 factor model over the 60 days.

In the end, its worth mentioning that our analysis could extend in many different ways. First, we could train more financial specific classfier by assigning the keywords in our sentiment dictionary with larger weights. Second, google news headers might not be map the real financial investor sentiment exactly, we could consider expend our text data scope to news

content and financial blogs for a better sentiment factor. Third, we might include more fluctuating period such as financial crisis to test the robustness of our factor. All these remains as area of future research.

8 References

[1] J. Bollen and H. Mao. Twitter mood as a stock market predictor. IEEE Computer, 44(10):91–94.

[2] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1–27:27, 2011.

[3] G. P. Gang Leng and T. M. McGinnity. An on-line algorithm for creating self-organizing fuzzy neural networks. Neural Networks, 17(10):1477–1493.

[4] A. Lapedes and R. Farber. Nonlinear signal processing using neural network: Prediction and system modeling. In Los Alamos National Lab Technical Report.

[5] A. E. Stefano Baccianella and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In LREC. LREC.