12-2015

The Information of Spam

Sawyer C. Anderson
University of Arkansas, Fayetteville

The Information of Spam

A thesis submitted in partial fulfillment
of the requirements for the degree of
Computer Science B.S.

by

Sawyer Anderson

December, 2015
University of Arkansas

Michael S. Gashler, Ph.D.
Thesis Director

Susan E. Gauch, Ph.D.
Committee Member

Kelly M. Sullivan, Ph.D.
Committee Member

Abstract

This paper explores the value of information contained in spam tweets as it pertains to prediction accuracy. As a case study, tweets discussing Bitcoin were collected and used to predict the rise and fall of Bitcoin value. Precision of prediction both with and without spam tweets, as identified by a naive Bayesian spam filter, were measured. Results showed a minor increase in accuracy when spam tweets were included, indicating that spam messages likely contain information valuable for prediction of market fluctuations.
Keywords: *spam, filtering, preprocessing, machine learning, bitcoin*

Contents

1 Introduction

1.1 Motivation

Spam messages are messages sent over the internet that contain information irrelevant or uninteresting to the user. Spam can be found in every communication medium on the internet: blogs, social media sites, video services, emails, and so on. Most often administrators of these sites and services attempt to remove spam, or at the very least filter it from the view of users. Yet while spam contains little to no information beneficial to the user, it could provide information that improves the accuracy of prediction or classification models.

1.2 Background

The microblogging service Twitter is a social media platform for users to share short 140-character messages publically or privately with other users. Users tweet about a variety of subjects, viewing and interacting with other users via replies and likes. Naturally these tweets contain a wealth of information from which organizations (from social media analytics companies to the US Center for Disease Control) can extract meaningful insights and useful predictions. Many tweets are spam, often tweeted and retweeted by automated bots. One estimate puts the spam rate on Twitter at nearly 10%, though this number varies drastically depending upon the subject matter of the Tweets. For example, tweets mentioning financial services corporation Visa have a spam rate of over 80% [1].

This paper uses tweets about Bitcoin, presently the leading cryptocurrency globally. The blockchain, the distributed ledger upon which Bitcoin is built, was released in January 2009, although it would be 2011 until the technology began to pick up volume and gain public interest. Perhaps due in part to the technical nature of Bitcoin and its underpinnings, online discussion about Bitcoin has increased dramatically as adoption of the currency has grown. In 2015, Twitter reported over 20 million tweets containing bitcoin, bitcoins, or btc.

1.3 Goals

This paper attempts to provide insight into the value of spam messages through a case study, in which the author constructs a system of models to predict the value of Bitcoin

based on the content of Twitter messages discussing the cryptocurrency, and then compares the precision of this prediction system when trained on a dataset including spam messages to the precision when trained using a dataset which has been filtered of spam.

The system consists of three parts: 1) a naïve Bayesian spam classifier trained to recognize and flag tweets as spam; 2) a Latent Dirichlet Allocation (LDA) model to discover conversation topics within Bitcoin tweets and reduce those tweets to a set of topic scores; and 3) a logistic regression model to learn correlations between topic scores and the movement of Bitcoin value. A difference in the logistic regression model's precision between spam-inclusive and spam-exclusive datasets would be indicative of the value of spam tweets. The necessity of spam filtering during preprocessing is highly dependent upon the problem or question being addressed spam filtering would likely prove wise if one was attempting to train a gender classifier, while such may not be the case for predicting the number of people who would see a tweet about a particular subject on a given day. The results of this project will ideally provide an intuitive base from which machine learning practitioners can answer the question "should I remove spam messages from my dataset?"

2 Related Work

In 2014, Kaminski and Gloor used a combination of sentiment and emotion extracted from Bitcoin tweets in conjunction with intraday market movement to predict Bitcoin market movement[13]. They also showed that a simplistic analysis of sentiment and emotion signals from tweets mentioning Bitcoin could be significantly correlated with both Bitcoin closing price and trading volume within a 2 day period, perhaps reflecting speculative momentum within the market. Zhang et al showed that simple analysis of tweets for "emotional outbursts of any kind gives a predictor of how the stock market will be doing the next day"[28]. Oh and Sheng further established that stock microblog sentiments do have predictive power for market returns[19]. Numerous other works have echoed these results[6][12][24][23].

2.1 Naïve Bayesian Spam Classifiers

Naïve Bayesian classifiers were first used to classify spam by Jason Rennie's 1996 ifile program [21]. Sahami et al published the first scholarly work demonstrating the principles in 1998, and Paul Graham greatly improved upon the precision of previous works by reducing the prevalence of false positives[22][10]. Androutsopoulos et al discuss and evaluate the suitability of naïve Bayesian classifiers for spam filtering, showing that other methods may provide more precise spam identification[3]. However, given the relative simplicity of a naïve Bayesian classifier, in conjunction with an impressive accuracy rate (over 97% in the former example), it was deemed sufficient for the purposes of this project.

2.2 Latent Dirichlet Allocation

LDA is a model which infers topics from a document collection, first proposed in 2004 by Blei, Ng, and Jordan[5]. Jahanbakhsh and Moon used LDA to reduce tweet dimensionality with the goal of predicting US presidential election outcomes (and succeeded)[11]. Waldhauser employed LDA to discover topics within political tweets, and to show that thematic content was more popular than episodic content within those tweets[27]. Bollen et al applied LDA to extract mood state from public tweets, from which emotive trends were modeled[6]. Kang et al developed a modified LDA, dubbed Transfer Hierarchical LDA, to create topic models of tweets and Facebook posts[14]. Finally, Diaz-Aviles et al used LDA to model topics of tweets in support of a groundbreaking method of surveilling the spread of epidemics, such as influenza, detecting outbreaks well in advance of other

widely-used systems[9].

2.3 Logistic Regression

Logistic regression is a probabilistic multiclass classifier with an extended history of use and improvement. Taddy developed multinomial inverse logistic regression as a method of dimensionality reduction on text[26]. Psomakelis et al showed that logistic regression performed mediocrely when used to perform sentiment analysis on tweets[20]. Other modified versions of logistic regression are also used for action recognition in video[15] and feature selection for support vector machines[25].

3 Background and Methods

3.1 Data Collection

This project combined two separate datasets to provide the features and labels used in prediction. A third dataset was used to train the naïve Bayesian spam classifier. Tweet content discussing Bitcoin comprised the features, and was collected through Twitter subsidiary GNIP's API service. The tweets collected span one year, from April 1, 2014 to March 31, 2015. The tweets comprised a randomly-selected 5% of tweets containing bitcoin, bitcoins, btc (a common abbreviation for Bitcoin), or any capitalization permutation of any of the three. In total, 875,847 Tweets were collected.

Bitcoin trading values were used to label the data. There exist a variety of exchanges through which users can actively buy and sell Bitcoin for another currency, though exchange prices vary little between exchange sites. This project utilized trade history from the BTC-e exchange during the given time period - data which is available publicly[2]. All trade prices were in United States Dollars.

A dataset of 52,070 web comments (including tweets), hand-labeled either spam or not spam, was used to train the naïve Bayesian classifier. Of these comments, 50,080 were not spam and 1990 were spam. These comments were provided by social listening and analytics company DataRank, who collected and labeled them.

3.2 Tooling and Techniques

Due to the scale of the dataset and the need for distributed processing, Apache Hadoop (distributed filesystem) and HBase (key-value NoSQL database) were used to store comments. Apache Spark's Machine Learning Library (mllib) provided a feature-rich, in-memory, distributed implementation of both the LDA and logistic regression models to process the comment data, as well as a few basic matrix algebra operations.

MySQL was used to store, process, and query aggregated comment topic scores and all Bitcoin trade data.

3.3 Spam Classification

Naïve Bayesian classifiers are a family of simple probabilistic classifiers based on Bayes theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{3.1}$$

For detecting spam, the classes S and H represent spam and not-spam (ham) categories, respectively. Given a document D consisting of N words, the probability that the document is spam is given by

$$P(S|D) = \prod_{n=0}^{N} \frac{P(W|S)P(S)}{P(W|S)P(S) + P(W|H)P(H)} \tag{3.2}$$

where $P(W|S)$ is the probability of the word W (the nth word in D) appearing in a spam document, $P(S)$ is the overall probability that a comment is spam, $P(W|H)$ is the probability of W appearing in ham documents, and $P(H)$ is the overall probability that the comment is ham. The probability that a document is ham can be calculated in the same manner.

A naïve Bayesian classifier was used to detect and flag spam Tweets. A Tweet in the corpus was marked as spam if the probability was above a threshold of 90%:

$$C(D) = \begin{cases} S, & P(S|D) > 0.9. \\ H, & P(S|D) \leq 0.9. \end{cases} \tag{3.3}$$

To account for words appearing in the corpus of Tweets but not appearing in the spam training dataset, and to prevent underflow during computation, an insignificant value of $P(W|C) = .01$ was substituted for both classes S and H for all $P(W|C) < .01$.

3.4 Latent Dirichlet Allocation

LDA is a natural language processing model that describes sets of observations with the assumption that they are generated by unobserved groups. For example, a tweet can be described as a mixture of a small number of topics, where each word in the tweet is attributable to one of these topics. LDA derives its name from its use of the Dirichlet distribution to model both the topic distributions of documents and the word distributions for each topic.

LDA attempts to fit the topic distributions of documents and word distributions of topics to the dataset using Bayesian inference. Bayesian inference derives posterior probability as a consequence of the prior distribution (in LDA's case, the Dirichlet distribution) and a likelihood function using Bayes' rule. Iterative simultaneous updates of the likelihood functions of topic and word distributions results in a more representative topic distribution, which subsequently results in more representative word distributions for each topic, and so on.

3.5 Logistic Regression

Logistic regression attempts to fit a standard logistic function curve of the form

$$F(x) = \frac{1}{1 + \exp -(\beta_0 + \beta_1 x)} \tag{3.4}$$

to the provided dataset X by finding β_0 and β_1 such that the error rate e, defined by

$$e = F(x) - y \tag{3.5}$$

where y is the discrete label corresponding to the features x, is minimized. This allows the prediction of a binary response. For multilabel classification problems, a multinomial logistic regression model containing $K - 1$ binary logistic regression models can be used. BFGS is a quasi-Newtonian optimization method which converges faster in many cases than gradient decent. A limited memory version of BFGS was used in this project. A more thorough description of L-BFGS is described by Nocedal[18].

4 Experiments and Results

4.1 Spam Classification

After training on the dataset of spam+ham labeled web comments, a small random sample of 239 Bitcoin tweets were hand-labeled and used to determine the precision and recall of the spam classifier, shown in 4.1 below (false positive indicating a ham comment marked as spam).

Correctly Classified	221
True Positives	79
False Positive	4
True Negatives	142
False Negatives	14
Precision	.9518
Recall	.8495

Figure 4.1: Spam classification metrics.

When used to label the entire dataset of 875,847 tweets, the classifier flagged 332,877 as spam, indicating that 38% of Bitcoin tweets are spam, which was very near the 34.7% spam rate found in hand-labeled Bitcoin comments. Given the monetary nature of Bitcoin, these figures are intuitively unsurprising.

4.2 Tweet Preprocessing

To prepare tweets for dimensionality reduction and to reduce noise from low-information words, common English stopwords were removed, as well as all single and double character tokens. A bag-of-words approach was then used to transform each tweet into a map of tokens to their frequency of occurrence within that tweet.

4.3 Tweet Dimensionality Reduction

The dataset consisted of tweets from twelve months, from April 1, 2014 through March 31, 2015. The first eleven months were used as the training corpus, and the final month of March 2015 was withheld as the test corpus.

LDA was used to discover topics within the training corpus. Expectation Maximization was used to implement a smoothed LDA model, as described in Asuncion et al[4]. Details of the full implementation can be found in the Spark 1.5.2 documentation[16]. Default values were used for the topic concentration Dirichlet prior (1.1) and document concentration Dirichlet prior (k dimensional vector with values $(50/k)+1$). 200 iterations of Expectation Maximization were used.

To determine the effect of the number of topics on the end prediction result, two topic sizes were used. First, sixteen topics were inferred over the full dataset of tweets including spam, and the topic scores for each tweet were calculated and stored. Sixteen topics were again inferred, this time over the tweet set which did not include spam, and topic scores for each tweet were again calculated and stored. The same process was repeated using 32 topics. 4.2 shows a few top terms and their weights from a sample topic.

term	weight
bitcoin	0.0418
crypto	0.0320
altcoin	0.0212
spanning	0.0196
site	0.0164
around	0.01300
app	0.0118

Figure 4.2: Sample topic terms and weights.

Note that the inferred topics used to calculate topic scores in both the training and test corpora were only those topics inferred from the training set; that is, no topics were inferred using tweets from March 2015.

Tweet topic scores were then aggregated by the hour in which the corresponding tweet was submitted: the topic scores of all tweets within a one-hour period were averaged, resulting in a table as shown in 4.3.

timestamp	topic_00	topic_01	...	topic_15
2014-03-01 00:00:00	1638.3486	553.5873	...	1084.3497
2014-03-01 01:00:00	1577.6205	512.3791	...	961.5373
2014-03-01 02:00:00	1724.7447	573.4869	...	1089.5858

Figure 4.3: Sample tweet topic scores, aggregated by hour.

4.4 Trade Signal Calculation

To provide labels for each hour of tweet topic scores, the movement of average Bitcoin trade value between each hour and the following hour was calculated:

$$\Delta avgValue = \frac{\sum tradePrice_{0,n}}{N} - \frac{\sum tradePrice_{1,m}}{M} \tag{4.1}$$

Each hour was then labeled with the trade signal indicating what action should be taken based on the above movement, using a threshold requiring movement of at least $0.25 in either direction:

$$signal(timestamp) = \begin{cases} buy & \Delta avgValue > 0.25. \\ hold & -0.25 < \Delta avgValue < 0.25. \\ sell & \Delta avgValue < -0.25 \end{cases} \tag{4.2}$$

The final datasets used in prediction were formed by combining the hourly topic scores from Section 4.3 with these hourly movement signals, producing rows such as though shown in 4.4 in the 16 topic case.

timestamp	topic_00	topic_01	...	topic_15	signal
2014-03-01 00:00:00	1638.3486	553.5873	...	1084.3497	buy
2014-03-01 01:00:00	1577.6205	512.3791	...	961.5373	hold
2014-03-01 02:00:00	1724.7447	573.4869	...	1089.5858	sell

Figure 4.4: Sample features and labels for k=16.

4.5 Prediction

A logistic regression model was trained using the topic scores and trade signals derived above. L-BFGS, an optimization algorithm in the quasi-Newton family of methods, was

used to approximate the objective function locally as a quadratic without evaluating the second partial derivatives of the objective function to construct a Hessian matrix. This removes vertical scalability issues, and tends to faster convergence than stochastic gradient descent. More information on L-BFGS can be found in the mllib documentation [17].

4.5 and 4.6 show the precision of the logistic regression model when predicting trade signals for the test corpus of topic scores.

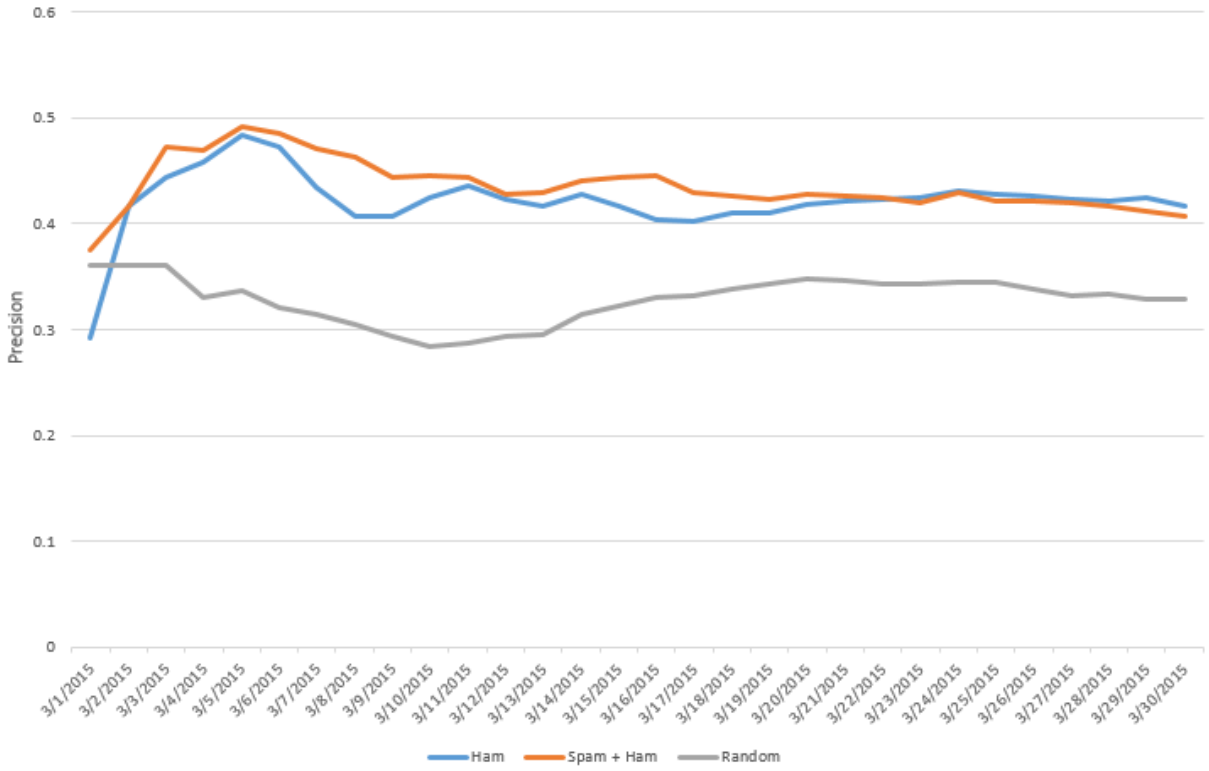

Figure 4.5: Prediction precision for 16-topic ham and spam + ham datasets from the month of March, 2015. Random indicates precision when a trade signal was chosen at random.

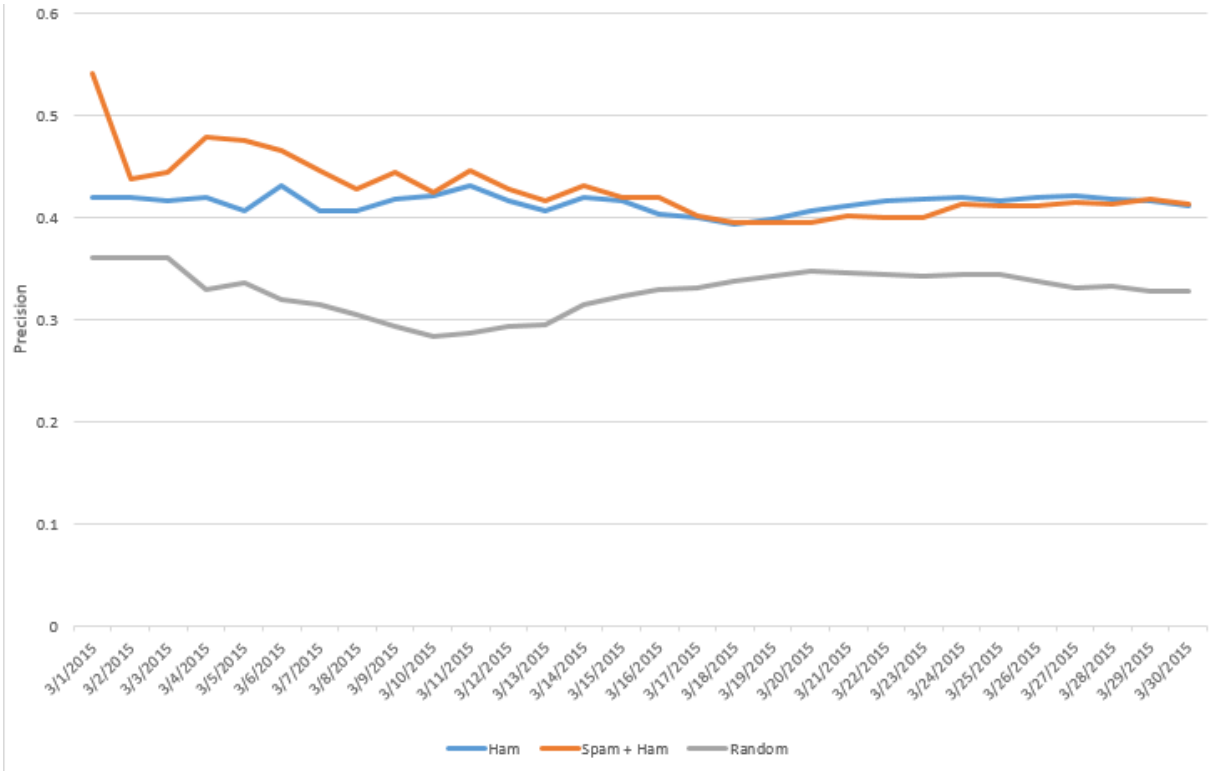

Figure 4.6: Prediction precision for 32-topic ham and spam + ham datasets from the month of March, 2015. Random indicates precision when a trade signal was chosen at random.

It is clear that in both the 16- and 32-topic cases, the exclusion of spam caused a decrease in the ability of the logistic regression model to accurately predict trade signals. Two explanations present themselves. First, that the information contained in spam comments contributed to an increase in precision. Second, that false positive spam comments, incorrectly excluded from the ham dataset, contributed to an increase in precision in the spam + ham dataset. However, given that only 1.67% of comments from the hand-labeled random sample of Bitcoin comments were false positive spam comments, it seems unlikely that false positives contributed even the smallest of increase in precision. In either case, without a perfectly accurate spam filter, it is clear that the exclusion of spam comments does not improve prediction accuracy in this case.

Finally, the difference in the derivative between 16- and 32-topic models at the beginning of the month seems to indicate that greater granularity in topics increases the immediate precision of prediction, though this effect disappears after only a few days, and would need to be verified by testing further topic count parameters in the LDA model.

5 Conclusion

Spam tweets are irrelevant to most human users and often filtered out from learning datasets. The results of this project indicate that spam tweets can contain information which improves prediction ability for complex classifications. However, these results should not be seen as definitive, since many factors affect the utility of spam documents, including the prediction goal and the distribution of spam to ham documents within the dataset. This experiment serves to show that there exist prediction tasks for which spam comments contain valuable information, and that spam should not be carelessly excluded from training datasets.

References

[1] fastcompany *Almost 10 percent Of Twitter Is Spam.* http://www.fastcompany.com/3044485/almost-10-of-twitter-is-spam

[2] BitcoinCharts *Bitcoin Charts Trade Data.* http://api.bitcoincharts.com/v1/csv/

[3] Androutsopoulos, I., Koutsias, J., Chandrinos, K., Paliouras, G., Spyropoulos, C. *An Evaluation of Naive Bayesian Anti-Spam Filtering.* 11th European Conference on Machine Learning, Barcelona, Spain, pp. 9-17, 2000.

[4] Asuncion, A., Welling, M., Smyth, P., Teh, Y. W. *On Smoothing and Inference for Topic Models.* http://arxiv.org/abs/1205.2662

[5] Blei, D. M., Ng, A. Y., Jordan, M. I. *Tweet Sentiment Analysis with Latent Dirichlet Allocation.* International Journal of Information Retrieval Research, 4(3), 0-0. doi:10.4018/ijirr.2014070105, 2004

[6] Bollen, J., Mao, H., Zeng, X. *Twitter mood predicts the stock market.* http://arxiv.org/pdf/1010.3003.pdf

[7] Bollen, J., Pepe, A., Mao, H. *Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena.* http://arxiv.org/abs/0911.1583 , 2009

[8] Deshpande, V. P., Erbacher, R. F., Harris, C. *An Evaluation of Naive Bayesian Anti-Spam Filtering Techniques.* 2007 IEEE SMC Information Assurance and Security Workshop. doi:10.1109/iaw.2007.381951, 2007

[9] Diaz-Aviles, E., Stewart, A., Velasco, E., Denecke, K., Nejdl, W. *Epidemic Intelligence for the Crowd, by the Crowd.* http://arxiv.org/abs/1203.1378 , 2012

[10] Graham, P. *A Plan for Spam.* http://www.paulgraham.com/spam.html

[11] Jahanbakhsh, K., Moon, Y. *The Predictive Power of Social Media: On the Predictability of U.S. Presidential Elections using Twitter..*

[12] Jaimes, A. *Correlating financial time series and micro blogging data.* https://labs.yahoo.com/publications/6380/correlating-financial-time-series-micro-blogging-data , 2012.

[13] Kaminski, J. *Nowcasting the Bitcoin Market with Twitter Signals.* http://arxiv.org/abs/1406.7577 , 2014.

[14] Kang, J., Ma, J., Liu, Y. *Transfer Topic Modeling with Ease and Scalability.* http://arxiv.org/abs/1301.5686 , 2013.

[15] Liu, W., Liu, H., Tao, D., Wang, Y., Lu, K. *Multiview Hessian regularized logistic regression for action recognition.* http://arxiv.org/abs/1403.0829 , 2014.

[16] Spark Docs *MLlib - Clustering*. http://spark.apache.org/docs/1.5.2/mllib-clustering.html

[17] Spark Docs *MLlib - Optimization*. http://spark.apache.org/docs/1.5.2/mllib-optimization.html

[18] Nocedal, J. *Updating quasi-Newton matrices with limited storage*. http://www.ams.org/journals/mcom/1980-35-151/S0025-5718-1980-0572855-7/home.html

[19] Oh, C., Liu Sheng, O. R. *Investigating Predictive Power of Stock Micro Blog Sentiment in Forecasting Future Stock Price Diretional Movement*. ICIS, 2011

[20] Psomakelis, E., Tserpes, K., Anagnostopoulos, D., Varvarigou, T. *Comparing methods for Twitter Sentiment Analysis*. http://arxiv.org/ftp/arxiv/papers/1505/1505.02973.pdf

[21] Rennie, J. *ifile*. http://people.csail.mit.edu/jrennie/ifile/old/README-0.1A , 1996.

[22] Sahami, M., Dumais, S., Heckerman, D., Horvitz, E. *A Bayesian Approach to Problems in Stochastic Estimation and Control. Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*. doi:10.1109/9780470544198.ch58, 2007.

[23] Si, J., Mukherjee, A., Liu, B., Li, Q., Li, H., Deng, X. *Exploiting Topic Based Twitter Sentiment for Stock Prediction*. https://www.cs.uic.edu/ liub/publications/ACL-2013-Jianfeng-stock-short.pdf

[24] Sprenger, T. O., Tumasjan, A., Sandner, P. G., Welpe, I. M. *Tweets and Trades: The Information Content of Stock Microblogs*. http://onlinelibrary.wiley.com/doi/10.1111/j.1468-036X.2013.12007.x/abstract , 2013.

[25] Stambaugh, C., Yang, H., Breuer, F. *Analytic Feature Selection for Support Vector Machines*. http://arxiv.org/abs/1304.5678 , 2013

[26] Taddy, M. *Multinomial Inverse Regression for Text Analysis*. http://arxiv.org/abs/1012.2098 , 2010.

[27] Waldhauser, C. *Public Spheres in Twitter- and Blogosphere. Evidence from the US*.

[28] Zhang, X., Fuehres, H., Gloor, P. A. *Predicting Stock Market Indicators Through Twitter I hope it is not as bad as I fear*. Procedia - Social and Behavioral Sciences, 26, 55-62. doi:10.1016/j.sbspro.2011.10.562, 2011.