A Novel Model for Predicting Stock Return with Text Information in SEC Filings

Jieyu Yao, Jun Guo, Qing Ma, Yue Chang

Abstract

The Management's Discussion and Analysis of Financial Condition and Results of Operations (MD&A) section of the SEC filings of 10-X family has attracted much attention in recent years, as latest technological developments in both hardwares and algorithms have enabled batch processing of text information. Results from various sources have indicated MD&A text harbours information that, if processed properly, can help investors better predict the filing company's future return. In this report, we present a natural language processing (NLP) approach that is based on a conventional methodology but uses novel components, together with a selection of machine learning algorithms. Together, the NLP process and machine learning estimators form a novel model that can predict the polarity of a company's return 3 months into the future, only using information in the MD&A section of its 10-X SEC filings, with accuracy significantly above chance level.

Introduction

At the end of the every January, April, July and October, markets are anticipating the earnings of big banks and leading companies from industries. Financial statements from these companies would reflect their performances of the previous quarter. What is often valued less significantly by some investors is the Management's Discussion and Analysis of Financial Condition and Results of Operations (MD&A), since this section is required as to provide a narrative explanation of the financial statements from the management's perspective. The SEC required firms to discuss and analyze the results of operations, financial and non-financial indicators, liquidity and capital resources, and any known trends, demands, commitments and uncertainties (SEC 2003). Given the prospective aspect of a company's MD&A, we can apply sentiment analysis on the MD&A and use machine learning algorithm to predict future stock return of the firm.

There have been a great many papers on applying sentiment analysis to the texts in SEC filings, especially to the MD&A section[5,6,12,13], and even the most basic "bag-of-words"

approach is still being investigated in variety and depth[17]. Furthermore, machine learning approaches have become one of the most important steps in conjunction with natural language processing (NLP). In this project, we seek to explore more possibilities based on the basic dictionary-based approach while experimenting with proper machine learning methods. Among all prior arts we found, the "sentiment analysis" mainly focuses on positivity/negativity scales of a text, while other analyses like "opinion mining" slightly extends it by including more entities[4-17,20-22]. What we do differently from those approaches is that we expand the categories beyond simple "positive" and "negative" distinctions; in addition, we do not assign the weights to each word or word category a priori, but let machine learning algorithms to find the weights that yield the best performance. It is worth noting that, instead of considering the "weighted sentiment" of each word as suggested by some papers, we instead consider only the weights of categories. Our logic behind this deviation is that we believe the formal business language used in MD&A tends to curb the intensity of sentiment-related words, and thus the variance of sentiment weight of the words appearing in the section may be quite small.

For our analysis, we obtained information of public companies from NASDAQ company list, all of the public filings from EDGAR, all stock returns from Compstat-IQ, industry SIC code and all ETFs returns from CRSP and the Loughran and McDonald (LM) Word List (LMWL), and the Harvard's General Inquirer (HGI) for textual analysis. The scope of our analysis is all the companies publicly listed in NASDAQ, AMEX and NYSE as of February 2017. The range of our study focuses on their financial activities from January 2005 to December 2016. The use of sentiment analysis in articles is gaining popularity over years. For example, text analysis on newspaper articles and editorials can be a faster economic indicator than official data. As these articles reflect how journalists and experts feel about macroeconomic, and influence general public's perception. Another well known example, the mood analysis based on Twitter content, shows that the West coast is happier than the east coast, the happiest time of a week is Sunday morning, and President Trump's mood becomes more negative after his first 100 days in office. Financial analytics company such as Bloomberg has a social velocity alert to monitor market reactions on Twitter. Our analysis takes on a similar analysis on a company's MD&A section in public filings to predict future returns of the company.

For our analysis, we obtained information of public companies from NASDAQ company list, all of the public filings from EDGAR, all stock returns from Compstat-IQ, industry SIC code and all ETFs returns from CRSP and the Loughran and McDonald (LM) Word List (LMWL), and the Harvard's General Inquirer (HGI) for textual analysis. The scope of our analysis is all the companies publicly listed in NASDAQ, AMEX and NYSE as of February 2017. The range of our study focuses on their financial activities from January 2005 to December 2016. The most fundamental and time consuming step of our analysis was the text cleaning process on 10-Q and 10-K filings we crawled from EDGAR database. After doing sentiment analysis

on text data cleaned from the previous step, we matched the result with the stock returns of at most three future months. For example, a company with fiscal year ends at Dec. 31st of each year, the analyzed results of first quarter 10-Q would be matched with aggregate return without dividend starting from April to end of June. The combined data would then be fitted into our machine learning algorithm, such as random forest to train a model predicting whether the stock would go up, down, or relatively unchanged in the future.

Methods

Data Preprocessing and Extraction

To achieve the targeted financial text data prepared for the NLP, we crawled the EDGAR database to download the financial statements and extract the managerial text from the filings. First, we requested the site book from CPSR database, which contains information including companies' names, tickers, the types of form and the links to all of the EDGAR forms from January 2005 to December 2016. We wrote a web crawler to gather all the 10-Q and 10-K forms on the site book and removed their html format using 'BeautifulSoup' package. We obtained more than 5GB of over 2000 EDGAR forms. For each 10-Q (K) form, we extracted the part of Item 2, 'Management's Discussion and Analysis of Financial Condition and Results' (MD&A) based on the depth first search and the Regex package on the key-words. In total, we get a sample of 1453 text files viable for the NLP.

For the market related data, we need to calculate the quarter return without dividend for each company at each month. First, we downloaded all the monthly-return-without-dividend data from Compstat-IQ In addition, we also found each company's corresponding industry indicator, the SIC code, mentioned earlier. For our data, the stock return of a given month is the asking price difference between the beginning of the month and the end of month divided by price at the beginning of the month. The three-month return is simply calculated through multiplication of the three consecutive months' returns plus one, and subtract one from the products. If the return is between –0.1% and 0.1% after three months, then the data is labeled as zero for our classifiers. Positive returns larger than the threshold are labeled as one, and vice versa.

As mentioned earlier, for every company we obtained the Standard Industrial Classification (SIC) label of the industry it belongs. Based on such information, we would like to benchmark the company's performance with that of its industry using the sub-industry tracking ETF. As the financial returns of companies in a sunrise industry would be much steeper than those of other more mature industries, we have to take into consideration of the overall industry performance. The dependent variable of dataset is the aggregated future return without dividend as described earlier. The independent variables for the sample dataset are the NLP features in the appendix and the industry benchmark.

Natural Language Processing (NLP)

Per the scope of this project, we decided to use the results of document-level analysis as the independent variable for subsequent machine learning stages. The specific NLP algorithms for sub-stages are carried out mainly using the Python NLTK package, or partially relying on the package, though the overall process is a custom construct.

The "Management's Discussion & Analysis" (MD&A) texts parsed from the 10-X files first undergo a process that breaks the texts down to words and then marks each word with the part-of-speech tags from the Penn Treebank Project. The lexicon-based analysis is applied to words with tags indicating an adjective, an adverb, a verb, a modal, or a noun. In another word, words with tags like "Determiner (DT)", "Wh-determiner/pronoun/adverb (WDT/WP/WRB)", and all other tags not in the four categories mentioned above. Table 1 contains the full selection list with both the tag names and descriptions. The reason for such operation is to simply reduce the input size to subsequent analysis.

The remaining word-POS pairs are then looked-up against two dictionaries: the Loughran and McDonald (LM) Word List (LMWL), and the Harvard's General Inquirer (HGI). While all word categories from the LMWL are considered, only a selected range of those from the HGI are so. The reason is that some categories in HGI are rarely relevant to the content of MD&A section or its indication of the business aspect. For example, the words describing subjective sensations (e.g., pleasure, pain, arousal, etc.) are not expected to appear in formal business documents (not considering the cases of, for example, companies involved with painkillers; on that case though, there is also a good reason not to include those words, for they may interfere with the actual sentiment of the business aspect). In addition, the reflection of the language of particular "institutions" by words should not be a major concern, as such aspects are expected to be more objective and definitive, and leaving very little freedom for the user to moderate the sentiment. Table 2 lists the selected categories from HGI and a brief explanation for why they are selected.

An important aspect of such analysis is properly processing the negation statements. We think that the negation markers generated by the sentiment analysis modules of NLTK do not properly capture the exact occurrences of negation, and the its algorithm is slow, thus we come up with our custom algorithm. With the assumption that the abbreviated forms of negation (e.g., hasn't, isn't, doesn't, etc.) do not appear often in formal documents, we narrow down the negation word to "not" only; there are other forms of negation like words "hardly", "barely", "little", but they should not appear often and thus are neglected in favor of shorter processing time. Then, instead of marking all subsequent words in the rest of sentence with negation, we only mark the first meaningful word or word pair with negation and then reset the marker; if a sentence's end is reached or another negation is encountered, the marker is also reset.

For each document, the final output is a vector of attribute values. The attributes correspond to the selected word categories, and the scores are essentially the number of words in the document that fall in these categories. For the HGI dictionary, though, as there are entries of the same word but marked with different numbers (usually corresponding to different situations, but require extra inference), the scores corresponding to one word count is the averaged values of such word across all its subtypes. For words with negation marks, the scores simply have their signs flipped.

Because the parsing algorithm does not guarantee a full extraction of the MD&A session, a normalization process is necessary. After all successfully parsed text documents are processed, with each document now corresponding to one entry, the attributes from LMWL and HGI are normalized to the sum of scores from each dictionary, respectively. Here we do not simply use the document total word count as the divisor because some documents contain format strings that contribute to word count (but will not contribute to the attribute scores).

Machine Learning Methods

Cluster Analysis: K-Means

Clustering is the task of grouping a set of objects into a cluster such that objects in the same cluster are more similar than objects in the other cluster. Clustering is an unsupervised learning technique that is a good choice to observe the interrelationship between data points. We use K-Means clustering to classify the clusters by Euclidean Distance. For the purpose of interpreting the data under more relevant content, we use the processed NLP data after normalized by total score of each document. The number of clusters is determined by the effect of clustering separation.

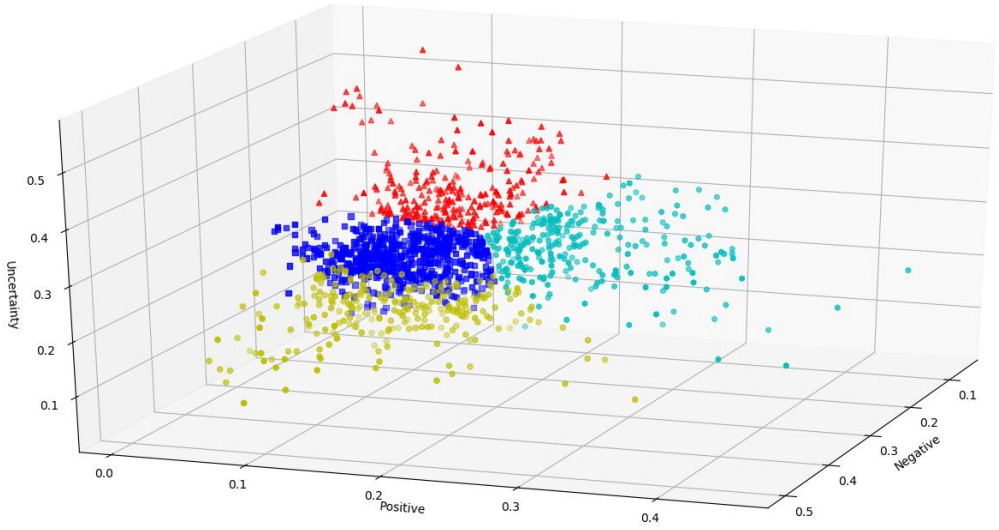

Figure 2.a, K-Means clustering on Positive, Negative and Uncertainty

Figure 2a shows four clusters of normalized score of Positive Outlook, Negative Outlook and Uncertain. The blue cluster centralizes at the scores low in all three dimensions. And the other three clusters centralizes at the scores high in one dimension and low in the other two dimensions. Thus from the figure we could figure out that one data point has low chance to be clustered at the center of high in all dimensions. These three dimensions may have negative correlation, which implies that the overall attitude of the financial statement might be written in one 'tone'.

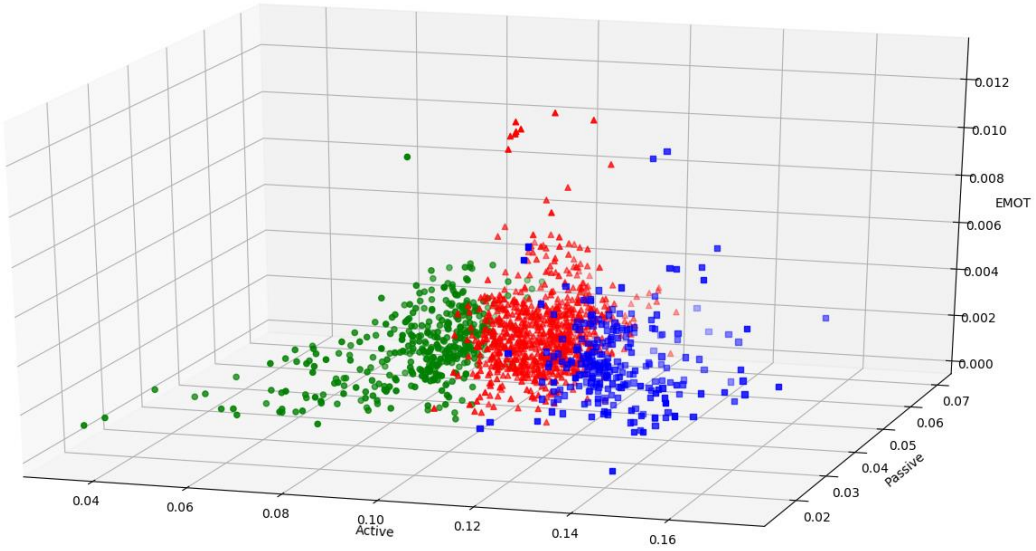

Figure 2.b K-Means clustering on Active, Passive and EMOT

Figure 2b shows three clusters of normalized score of Active Orientation, Passive Orientation and Related to Emotional. We can see all three clusters centered closely and showing low emotional level. The passive orientation is more favored but the activity level varies.

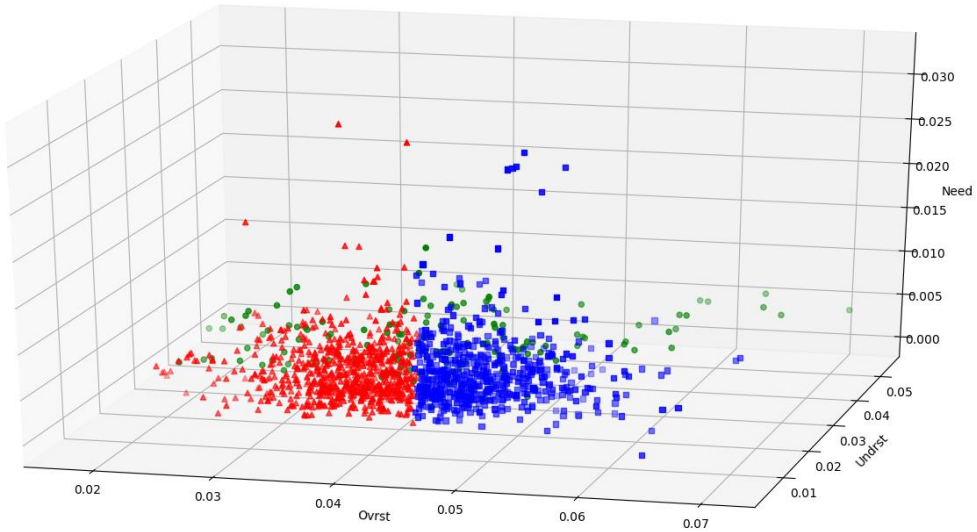

Figure 2.c K-Means clustering on Ovrst, Undrst and Need

Figure 2c shows three clusters of normalized score of Overstated, Understated and Need/Intent. From red cluster and blue cluster we can see that the need score could be high with both low and high overstatement level. And compare to the green cluster, higher understatement score are more likely to go with lower need score, which may imply that the understated sentiment would induce low needs.

Random Forests

Moving forward from the normalized output from text documents, we combined each vector of attribute values with industrial benchmark and the corresponding dependent variable. After taking into consideration of multiple regression and classification methods, our principle machine learning model is the random forest classifier. Random forest is very effective and accurate for predicting the results of model. The ensemble, or divide-and-conquer, approach improve the performance of the training model as it combines the many "weak" classifiers to get a "strong" classifiers. Each "weak" classifier in a random forest is a tree predictor trained through randomly selected features. Based on Brieiman's random forests paper, several advantages of random feature forest are low correlation while the strength of classifier maintained, high accuracy, robustness to outliers and noise, faster than bagging or boosting, useful estimation of error, strength, correlation and variable importance. The random features forest is more suited for classification than for regression. During the training process, we were also interested in selecting the best combination of parameters: number of trees in the forest and the minimum number of samples required to be at a leaf node. In order to do so we used the gridsearchcv package from sklearn.model selection to search over combinations of parameter values.

SVM

In addition, we also applied Support Vector Machine classifier on our dataset. SVM classifier prediction model works by finding the hyperplane planes that can best divide a dataset. It's greatest advantage is its accuracy, however the method might not work as well if given a noisy dataset with overlapping classes.

Multi-layer Perceptron

To be advanced, we applied a multi-layer neural network model to train and test our dataset. Multi-layer Perceptron (MLP) is a supervised learning algorithm that could learn a non-linear function approximator for our classification. Such feedforward multi-layer network contains multiple intermediate layers and hidden layers.

Adaboost

Adaptive Boosting is another choice of machine learning model that could be used in conjunction with many other models.

Result Analysis

In order to better implement our models on future dataset without overfitting our current dataset, we evaluated these models through a ten-fold cross validation method. For that reason, the training set of each prediction would be 90% of the entire sample. We trained and tested more than 10 machine learning models and chose the top five best performance list below:

	Cross Validation	**Mean Score**
Random Forest	**0.5969** 0.5954 0.5846 0.5615 0.5573 0.542 0.5231 0.5191 0.5154 0.5038	0.5499
SVC	**0.5615** 0.5581 0.5573 0.5573 0.5538 0.5496 0.5462 0.5462 0.542 0.542	0.5514
Linear SVC	**0.5891** 0.5725 0.5615 0.5538 0.5344 0.5267 0.5231 0.5077 0.5038 0.4962	0.5369
Multi-layer Perceptron	**0.5923** 0.5581 0.5496 0.5496 0.5462 0.5462 0.5344 0.5267 0.5267 0.5231	0.5453
AdaBoost	**0.5923** 0.5769 0.5581 0.5573 0.5496 0.5344 0.5308 0.5191 0.5154 0.4885	0.5422

From the result above, the overall mean scores of ten-fold cross validation are higher than 0.5, which is better than the theoretical result of random guess in two labels. Indeed, since we run the machine learning classifiers under three labels (1, 0, -1), we achieve a hopeful result through the methods above. On the other hand, highest scores approaching 0.6 are also prospective, which are comparable to the result of the text analysis[14] literature that we have referred.

A statistical analysis on the cross validation score can provide us with more insights into each machine learning methods. With 95% confidence, random forest has a confidence interval of [0.5299,1], SVC's CI is [0.5472,1], linear SVC's CI is [0.5188,1], multi-layer perception's CI is [0.5335,1] and ADAB's VI is [0.5243,1].

Discussion & Future Work

In this project we experimented on a new construct of NLP approach on the relatively subjective text component in public SEC filings, and tested the validity of the model that uses only its results to predict a company's return in the near future (after the filing). In the NLP stage, the emphasis is on retaining more information with a relatively simple analysis structure; the machine learning stage is then supposed to find the most relative features while

eliminating the less relative ones. However, because the final sample size is relatively small compared to the number of potential dimensions, only some of the most powerful ensemble methods are able to achieve this. Moreover, regression methods that attempt to predict continuous values, as many may expect, fail in such setting.

Still, our result show that even by using the subjective text from the financial document can achieve a positive result. Although the overall performance is not glaringly high, it is a significant increase above chance level (33% for 3-category random guess). Considering the fact that we had limited experience in advanced text parsing and processing before the project, we have strong reasons to believe that the current quality of the data that we input to the machine learning algorithms are suboptimal. In another word, for future work, we will be able to further improve the data processing result by using more specialized parsers (so that the full text of the MD&A section can be retrieved without error, and increase the size of valid samples) and more refined NLP algorithms (e.g., better negation marking, weighted word scores for various categories, modification of dictionaries, etc.). We can also take the liberty to explore other numerical financial data, such as the stock price volatility, to modify the stock return data and observe whether the resulted dependent variable can be better predicted using the same input features.

Furthermore, the resulted model can be combined, as an ensemble component, together with other stock return predictor models, to form a multi-layered stacking model. If formulated properly, the combined stack model should be able to see further improvement on prediction accuracy.

Acknowledgements

The authors are grateful for the inspiration and essential advice provided by Professor Eugene Wu, without whose support this project could not have come to be.

The authors also thank Mr. Weijie Steven Shi for his information on the content importance of SEC filings.

Reference

1. Asmi, Amna, and Tanko Ishaya. "Negation identification and calculation in sentiment analysis." The Second International Conference on Advances in Information Mining and Management. 2012.
2. Bird, Steven, Edward Loper and Ewan Klein (2009), Natural Language Processing with Python. O'Reilly Media Inc.
3. Breiman, Leo. Machine Learning 45.1 (2001): 5-32. Web.
4. Chen, Chien-Liang, et al. "Opinion mining for relating subjective expressions and annual earnings in US financial statements." arXiv preprint arXiv:1210.3865 (2012)
5. Cohen, L., C. Malloy, and Q. Ngyuen. Lazy Prices. Harvard Business School. working paper, 2015.
6. Cohen, Lauren, Christopher J. Malloy, and Quoc H. Nguyen. "Lazy prices." (2016).
7. Ding, Xiaowen, Bing Liu, and Philip S. Yu. "A holistic lexicon-based approach to opinion mining." Proceedings of the 2008 international conference on web search and data mining. ACM, 2008.
8. Feldman, Ronen. "Techniques and applications for sentiment analysis." Communications of the ACM 56.4 (2013): 82-89.
9. Godbole, Namrata, Manja Srinivasaiah, and Steven Skiena. "Large-Scale Sentiment Analysis for News and Blogs." ICWSM 7.21 (2007): 219-222.
10. Guerini, Marco, Lorenzo Gatti, and Marco Turchi. "Sentiment analysis: How to derive prior polarities from SentiWordNet." arXiv preprint arXiv:1309.5843 (2013).
11. Guzman, Emitza, and Walid Maalej. "How do users like this feature? a fine grained sentiment analysis of app reviews." Requirements Engineering Conference (RE), 2014 IEEE 22nd International. IEEE, 2014.
12. Heston, Steven L., and Nitish Ranjan Sinha. "News versus sentiment: Comparing textual processing approaches for predicting stock returns." Robert H. Smith School Research Paper (2014).
13. Kearney, Colm, and Sha Liu. "Textual sentiment in finance: A survey of methods and models." International Review of Financial Analysis 33 (2014): 171-185.
14. Lee, Heeyoung, et al. "On the Importance of Text Analysis for Stock Price Prediction." LREC. 2014.
15. Li, Nan, and Desheng Dash Wu. "Using text mining and sentiment analysis for online forums hotspot detection and forecast." Decision support systems 48.2 (2010): 354-368.
16. Liu, Bing. "Sentiment analysis and opinion mining." Synthesis lectures on human language technologies 5.1 (2012): 1-167.
17. Loughran, Tim, and Bill McDonald. "The use of word lists in textual analysis." Journal of Behavioral Finance 16.1 (2015): 1-11.
18. Loughran, Tim, and Bill McDonald. "When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks." The Journal of Finance 66.1 (2011): 35-65.
19. Marcus, Mitchell P., Mary Ann Marcinkiewicz, and Beatrice Santorini. "Building a large annotated corpus of English: The Penn Treebank." Computational linguistics 19.2 (1993): 313-330.
20. Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006. Generating Typed Dependency Parses from Phrase Structure Parses. In LREC 2006.

21. Mei, Qiaozhu, et al. "Topic sentiment mixture: modeling facets and opinions in weblogs." Proceedings of the 16th international conference on World Wide Web. ACM, 2007.
22. Mudinas, Andrius, Dell Zhang, and Mark Levene. "Combining lexicon and learning based approaches for concept-level sentiment analysis." Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining. ACM, 2012.
23. Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." Journal of Machine Learning Research 12.Oct (2011): 2825-2830.
24. Richard Socher, John Bauer, Christopher D. Manning and Andrew Y. Ng. 2013. Parsing With Compositional Vector Grammars. Proceedings of ACL 2013
25. Sebastian Schuster and Christopher D. Manning. 2016. Enhanced English Universal Dependencies: An Improved Representation for Natural Language Understanding Tasks. In LREC 2016.
26. Staiano, Jacopo, and Marco Guerini. "DepecheMood: A lexicon for emotion analysis from crowd-annotated news." arXiv preprint arXiv:1405.1605 (2014).
27. Stone, Philip J., and Earl B. Hunt. "A computer approach to content analysis: studies using the general inquirer system." Proceedings of the May 21-23, 1963, spring joint computer conference. ACM, 1963.
28. Tetlock, Paul C., Maytal Saar-Tsechansky, and Sofus Macskassy. "More than words: Quantifying language to measure firms' fundamentals." The Journal of Finance 63.3 (2008): 1437-1467.
29. Ting, Kai Ming, and Ian H. Witten. "Issues in stacked generalization." J. Artif. Intell. Res.(JAIR) 10 (1999): 271-289.
30. Wolpert, David H. "Stacked generalization." Neural networks 5.2 (1992): 241-259.

Appendix

Table 1.

JJ	Adjective
JJR	Adjective, comparative
JJS	Adjective, superlative
MD	Modal
NN	Noun, singular or mass
NNS	Noun, plural
NNP	Proper noun, singular
NNPS	Proper noun, plural
RB	Adverb
RBR	Adverb, comparative
RBS	Adverb, superlative
VB	Verb, base form
VBD	Verb, past tense
VBG	Verb, gerund or present participle
VBN	Verb, past participle
VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present

Source: The Penn Treebank

Table 2.

	Description	Brief Reason for Inclusion
Positiv	Positive outlook	Essential
Negativ	Negative outlook	
Pstv	Positive outlook	More selective version
Ngtv	Negative outlook	
Strong	Implying strength	Can be used to describe performance
Weak	Implying weakness	
Active	Active orientation	Reflects Tone
Passive	Passive orientation	
EMOT	Related to emotion	Reflects stability
Virtue	Moral approval	Similar to Positive
Vice	Moral disapproval	Similar to Negative
Ovrst	Overstated	Some words imply subtle grandiosity
Undrst	Understated	De-emphasis and caution
Role	Profession related	Frequency that personnel are mentioned
Need	Need or intent	Words associated with the company's future projections and the status of achievements; the frequency of occurance might reflect a company's stance.
Goal	Desirable end-state	
Try	Goal-oriented activities	
Means	Goal-oriented approaches	
Persist	Endurance related	
Complet	Achieving goals	
Fail	Not achieving goals	
Think	Thought process	
Know	Awareness related	
Causal	Occurance explanation	
Ought	Moral imperative	
Perceiv	Perception/sense	
Compare	Comparison	Reflects tendency to show results by comparison.
EVAL	Judgement and evaluation	Occurrence frequency might be related to assertiveness.
Solve	Problem solving	Related to obstacles.
ABS	Tendency of using abstract vocabulary	Reflects general concreteness.

Source: Harvard General Inquirier

For processing code, please refer to the github repository:

https://github.com/CYUlysses/W4121_SECNLP