THE UNIVERSITY OF CHICAGO

Company Announcements and Stock Returns

By

Songrun He

June 2021

A paper submitted in partial fulfillment of the requirements for
the Master of Arts degree in the Master of Arts in
Computational Social Science

Faculty Advisor: Dacheng Xiu
Preceptor: Pedro Alberto Arroyo

Company Announcements and Stock Returns

Songrun He*

May 8, 2021

Abstract

Employing a novel text mining method proposed by Ke et al. (2019), I carry out a comprehensive textual analysis of Chinese company announcements for stock return prediction. The long-short portfolio based on article sentiment scores yields 78% cumulative returns with a Sharpe ratio of 0.86 from 2016 to 2019. The model well captures the contemporaneous association between announcement sentiment and stock returns. There is also evidence for the information leakage before the announcements are made public. Careful examination of the traded articles finds that they are longer than average, suggesting complex information may not be fully digested by the investors as they are released. Company annual reports and financing activities reports play important roles in explaining the portfolio's performance.

Keywords: Text Mining, Machine Learning, Financial Returns, Topic Modeling, Penalized Likelihood

*M.A. Student in Computational Social Science, The University of Chicago
I would like to thank Dacheng Xiu, Christopher Roark, Shilin Jia, Michael Galperin, as well as seminar participants at the University of Chicago for helpful comments and suggestions.
Email: hesongrun@outlook.com

1 Introduction

> *"... One of the most promising use of relatively new AI techniques may be processing unstructured natural language data in the form of news articles, company reports and social media posts, in an effort to gain insights into the future performance of companies, currencies, commodities, or financial instruments. ..."*

– Knight (2016), MIT Technology

With the rapid development of technology, an increasing share of human interactions is recorded digitally in data, among which texts contain the most important information about the underlying latent state for social sciences study. The advancement in NLP and machine learning have also opened the gate for modeling and analyzing the unstructured textual data, which are ultra-high dimensional in their nature.

In this study, I will apply a novel text-mining approach proposed in Ke et al. (2019) to extract the sentiment from Chinese company announcements to predict stock returns. I am interested in the question: whether the stock prices fully reflect all the 'textual' information contained in the announcements. Even in a market where participants are fully rational, cognitive limitations can still cause the agents to overlook certain information in the announcements. A trading strategy using machines to digest textual information from thousands of companies should be able to tap into such inefficiencies and gain abnormal returns. Before introducing the methodology, I will briefly review the textual analysis in finance.

An early application goes back to Cowles (1933). The author subjectively categorizes the texts in Wall Street Journals as 'bullish', 'bearish', or 'doubtful'. He then uses the classification to predict the Dow Jones Industrial Average.

Fast forward into the modern financial natural language processing realm, with access to a large amount of data, it is mainly computationally driven. There are three major sources for modern financial textual analysis: financial news, company announcements, and stock trading forums. In Gentzkow et al. (2019), the authors provide a general procedure with three steps researchers typically apply to carry out NLP on these datasets:

- Step 1: Convert the raw text into a numerical array
- Step 2: Fit a model to map the numerical array to the variable of interest
- Step 3: Use the trained model to perform in-sample or out-of-sample test for descriptive as well as causal analysis

In general, in steps 1 and 2, the textual analysis poses a high-dimensional challenge for researchers. For a text having a length of l, if there are w possible words in each position, the total number of possible combinations will be w^l. If we use very flexible representation in step 1, due to the curse of dimensionality, it will require a large amount of training data to deliver good performance. In asset pricing, the data is very weak in information. In efficient markets, we have a very low signal-to-noise ratio as a result of competition. Therefore, to learn the asset pricing implications of textual information, researchers usually use bag-of-words models where each article is treated as a set of words, and the sequence of words does not matter. The semantic information is left out here. With the bag of words, the researchers then use a dictionary, which encompasses prior information of 'positive' and 'negative' wordlists to count words. The counting of the positive versus negative words will determine the sentiment content of an article.

The earliest paper adopting this methodology is Tetlock (2007). The author used the Harvard General Inquirer Word List with optimism and pessimism word lists to quantify the media sentiment in Wall Street Journals 'Abreast of the Market' column. They have found that the pessimism sentiment can predict the downward movement of the market.

Later on, Loughran and McDonald (2011) proposed a financial sentiment dictionary. They have discovered that the Harvard General Inquirer Word List cannot distinguish the positive versus negative words in the financial context. 73.8% of negative words from the Harvard dictionary do not have negative meaning under a financial context(e.g., liability, etc.). To overcome this drawback, the authors screened for appropriate words for financial context to form the Loughran and MacDonald financial dictionary, which has widespread use in financial textual analysis. Using the new financial dictionary, the authors find a greater correlation of the negative textual sentiment with stock returns.

There are many papers following this approach. Jegadeesh and Wu (2013) finds that word weighting plays an important role in extracting sentiment that is predictive of stock returns. Jiang et al. (2019) constructs a manager sentiment index using the 10-K and earnings call transcript. Cohen et al. (2020) measured the similarity between 10-K documents across years using word count and associate such similarity with stock returns.

However, this dictionary-based approach has two major drawbacks: (1) The choice of wordlist is ad-hoc and may subject to the bias of researchers; (2) Every word plays the same role in the final aggregation. It is very likely that words have different informational content. Word weighting would be necessary. The tradeoff can be visualized in figure 1. The counting methods have great economic interpretability and computational scalability. However, it ignores the linguistic complexity in the underlying language.

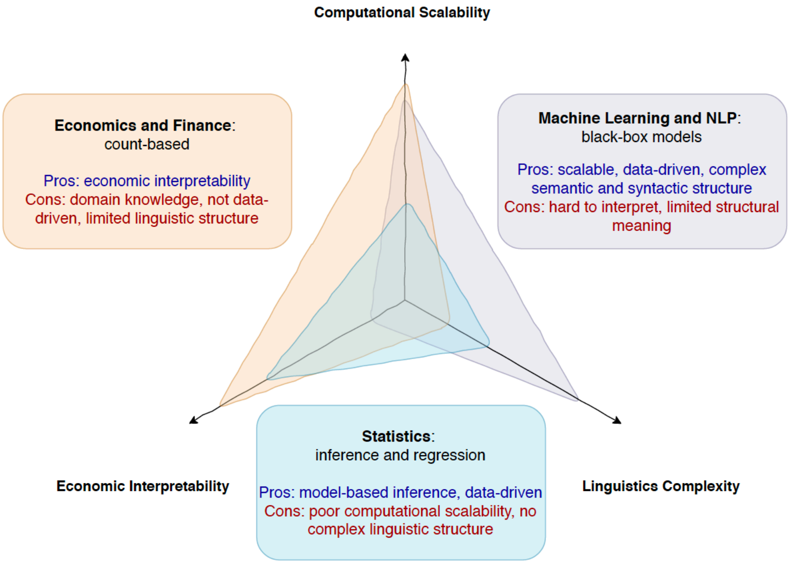
Computational Scalability
Economics and Finance: count-based
Pros: economic interpretability
Cons: domain knowledge, not data-driven, limited linguistic structure
Machine Learning and NLP: black-box models
Pros: scalable, data-driven, complex semantic and syntactic structure
Cons: hard to interpret, limited structural meaning
Statistics: inference and regression
Pros: model-based inference, data-driven
Cons: poor computational scalability, no complex linguistic structure
Economic Interpretability
Linguistics Complexity

Figure 1: Tradeoffs in Various Approaches

This image comes from Cong et al. (2019)

There are many recent pieces of research adopting new algorithms and approaches that allow for greater linguistic complexity. Bybee et al. (2020) uses an LDA model to extract topics from the Wall Street Journals. The authors demonstrate the topic attention can offer a transparent predictive relation from texts to many economic variables. Lopez-Lira (2020) used the LDA model to analyze the 'risk factor' section of the company 10-K report. They found that some textual factors represent the systemic risks faced by companies and are significantly priced.

The focus of my paper is on return prediction. In this research direction, Ke et al. (2019) proposed a novel tax-mining approach that addresses the concerns of ad-hocness of word counting and sentiment dictionaries. The authors proposed a model where latent sentiment generates text and stock returns jointly. In this model, both the wordlist and term weights are parameters to be estimated. While it still uses the bag-of-the-words representation of texts, this approach is flexible enough to extract more information from the observable texts and returns compared to the dictionary-based approach. Fan et al. (2021) extend the framework by allowing for more general factor and topic structures. They used the Chinese news to predict stock returns.

This framework has three steps: (1) screen for sentiment-charged words based on their co-occurrence with positive stock returns; (2) assign optimal topic weights for each word based on the magnitude of return response; (3) apply the learned sentiment dictionary and the optimal topic weights to score new articles based on penalized MLE. The benefit of the model is to introduce linguistic complexity by taking a probabilistic approach to model sentiment topics while still allowing transparency and interpretability. As I will show in the paper, one can always trace the trading decision back to the word counting and distribution. The sentiment-charged words list also makes it easy to interpret what information matters for the portfolio decision.

My analysis focuses on the Chinese company announcement from the Juchao database, which is the official information disclosure platform designated by the China Securities Regulatory Commission. Chinese text brings additional challenges as there are no natural separations between words. I follow the text cleaning and preprocessing procedure laid out in Fan et al. (2021) and cut the text into words using the 'Jieba' package (Chinese for 'to stutter'). After the marginal screening step, I find the negative word list I get is more meaningful compared to positive ones. One hypothesis would be this reflects the euphemism effects: it is more common for managers to use negation before positive words to convey negative meanings than using negative words directly. As a result, if a negative word is used, it truly has negative meanings. A comparative study on the US 10-K MD&A finds similar patterns.

Furthermore, I implement the text-based trading strategy from the year 2016 to 2019 and found significantly positive out-of-sample portfolio returns based on the sentiment score. The cumulative portfolio return is 78% in this four-year period with a Sharpe Ratio of 0.86 while the market only rises by less than 10%. The portfolio has significant alpha after controlling for the Fama-French factors (Chinese version). Besides this day $t + 1$ portfolio, I also experiment with the day t portfolio and day $t - 1$ portfolio. The day t portfolio shows extremely high returns suggesting the model captures the contemporaneous stock return with the texts very well. The day $t - 1$ portfolio also exhibits significant positive returns meaning that there is some information leakage even before the announcement is made.

I looked into articles being traded and found they are longer than the average articles in my sample. This suggests that due to cognitive limitations, market participants may not fully react to the information contained in many complex, long company announcements when they are announced. As a result, trading strategies based on these documents can yield abnormal returns in the short term. Last but not least, I find that company regular reports such as annual announcements or semi-annual announcements and reports related to company financing activities play an important role in explaining the performance of the trading strategy. This suggests further research can be done to study the response of investors to these announcements.

The rest of the paper is organized as follows. Section 2 reviews the detail of the methodology proposed by Ke et al. (2019). Section 3 introduces my data, the procedures I applied to clean them, and detailed training, tuning, and testing specifications. Section 4 focuses on the empirical analysis of the trading strategy. Section 5 concludes.

2 Methodology

In this section, I focus on the methodology used to extract sentiment from company announcements. Instead of using the ad hoc pre-defined sentiment dictionaries and word counting to quantify the sentiment, I will rely on a supervised probabilistic model to find the sentiment-charged words and the corresponding topic distributions on these words simultaneously from the data. This novel method was first introduced by Ke et al. (2019) to predict stock returns using company news articles from the Dow Jones Newswire in the US stock market. I will briefly discuss the setup of the problem and go through the specific procedures of the algorithm.

2.1 Problem Setup

For all articles, we model them as bags of words, i.e., we only keep track of the number of times certain words occurring in an article while discarding their positional information. Although this approach gets rid of the rich semantic meanings of words, it keeps the first-order information contained in the text: the word choices. To put it formally, let **D** denotes all the words in Chinese. Each article will be represented by a $|\mathbf{D}|$ dimensional vector $\mathbf{d_i}$ for $i = 1, ..., n$. The $k - th$ element of the vector, $d_{i,k}$, will be the number of times word k occurring in this article. It is worth noting that for the Chinese textual data, the dimension of **D** can be very large. For the company announcement sample, I have around 2 million articles with over 5 million distinctive words in total.

Each article will be associated with a response variable Y_i, which measures the influence of the information contained in the article. In our example, the response variable is the contemporaneous return on the day the announcement is made. We assume that both the response variable and a subset of all the words in the article are driven by the latent 'sentiment'. This sentiment does not necessarily correspond to the underlying latent deep states managers have when producing these documents. We use this word to summarize all the information related to stock prediction. In this way, we divide our total word set into two parts: (1) sentiment charged words: **S** and (2) words that are not related to stock returns: **U**. In this way, $\mathbf{D} = \mathbf{S} \cup \mathbf{U}$. For all the articles, their corresponding sentiment score, denoted as p_i, will be determined by the corresponding word distribution on this sentiment charged word set **S** as well as the sentiment topic weights on each word. We can denote these word counts for article i as $d_{i,\mathbf{S}}$ and the total count of sentiment charged words for article i as s_i.

With this setup, there are three goals in total: (1) screening for sentiment-charged words; (2) estimate sentiment topic weights on all the words; (3) given (1) and (2), obtain an algorithm to score new articles. This brings us to the next section on the details of the supervised sentiment extraction algorithm.

2.2 Supervised Sentiment Extraction Algorithm

Firstly, following Ke et al. (2019), I assume the bag of word for each article is drawn from a mixture of multinomial distribution on two sentiment topics: one corresponding to the positive sentiment and the other corresponding to the negative sentiment:

$$d_{i,\mathbf{S}} \sim Multinomial(s_i, p_i\mathbf{O}_+ + (1 - p_i)\mathbf{O}_-)$$

where O_+ corresponds to the word distribution for articles with the most positive sentiments (corresponding to high stock returns) while O_- corresponds to the word distribution for articles with the most negative sentiment (corresponding to low stock returns). $p_i \in [0, 1]$, the sentiment score, is like a sliding bar between these two topics. In this setup, the three estimation corresponds to: (1) estimate $\mathbf{S}$; (2) estimate $\mathbf{O}_+$ and $\mathbf{O}_-$; (3) given $\mathbf{S}$, $\mathbf{O}_+$ and $\mathbf{O}_-$, obtain a new p_i for article i.

2.2.1 Screening for Sentiment Charged Words

The first step is to estimate the dictionary for sentiment-charged words $\mathbf{S}$. I apply the marginal screening technique used in Ke et al. (2019). To be specific, only words that co-occur very often with positive or negative returns are retained. Define such frequency as f_k:

$$f_k = \frac{\text{count of articles with word } k \text{ and with } syn(y) > 0}{\text{count of articles with word } k} \tag{1}$$

Sentiment neutral words are expected to have f value very close to 0.5 or the average count of articles associated with positive returns, which we denote as $\hat{\pi}$. Therefore, words with very large or small f values will be retained. Furthermore, for some special words occurring very infrequently, their f values will be estimated with great noise. Therefore, I get rid of these words. In all, I use three parameters to determine the final sentiment-charged word set:

$$\hat{\mathbf{S}} = \{k : f_k \geq \hat{\pi} + \alpha_+ \text{ or } f_k \leq \hat{\pi} - \alpha_-\} \cup \{k : c_k \geq \kappa\}$$

where c_k denotes the count of word k in all articles.

2.2.2 Estimating the Sentiment Topics

With the sentiment charged words, the next step is to estimate the topic loadings on these words. Use $\mathbf{P}$ to denote the $n \times 2$ matrix of sentiment scores for the n articles in our training set. The $i - th$ row of $\mathbf{P}$ will be $(p_i, 1 - p_i)$, which are article i's loadings on the positive sentiment topic and the negative sentiment topic. Furthermore, use $\mathbf{O}$ to denote the $|\mathbf{S}| \times 2$ matrix of the topic weights on words: $\mathbf{O} = (\mathbf{O}_+, \mathbf{O}_-)$. Use $\mathbf{H}$ to denote the $n \times |\mathbf{S}|$ matrix of the sentiment charged word frequency for all the articles. Its $i - th$ row is defined as: $h_i = \frac{d_{i,\mathbf{S}}}{s_i}$, which is a $1 \times |\mathbf{S}|$ vector. By the multinomial assumption:

$$\mathbb{E}[h_i] = p_i \mathbf{O}_+ + (1 - p_i)\mathbf{O}_-$$

Put all the elements together in the matrix form:

$$\mathbb{E}[\mathbf{H}] = \mathbf{P}\mathbf{O}^\top$$

Given $\mathbf{H}$ and $\mathbf{P}$, the $\mathbf{O}$ can be directly estimated via the least square estimator:

$$\hat{\mathbf{O}}^\top = (\mathbf{P}^\top \mathbf{P})^{-1} \mathbf{P}^\top \mathbf{H}$$

To implement the above estimator, we can directly plug in the $\hat{\mathbf{H}}$ using $\hat{\mathbf{S}}$ from our first marginal screening step. As for the $\mathbf{P}$, since it is also reflected in stock returns, I use the sample ranking of the article contemporaneous stock returns to get their corresponding sentiment scores:

$$\hat{p_i} = \frac{\text{rank of } Y_i \text{ in } \{Y_j\}_{j=1}^n}{n}$$

The final feasible estimator of $\mathbf{O}$ is:

$$\hat{\mathbf{O}} = (\hat{\mathbf{P}}^\top \hat{\mathbf{P}})^{-1} \hat{\mathbf{P}}^\top \hat{\mathbf{H}}$$

Given the fact that I have millions of observations (large n) while only need to estimate hundreds of parameters (small $|\mathbf{S}|$), there is no need for penalization in this step. The parameters are estimated with great precision.

2.2.3 Scoring New Articles

To complete the final piece, I need a procedure to score new articles given $\hat{\mathbf{S}}$ and $\hat{\mathbf{O}}$. To do this, I will perform a penalized maximum likelihood estimation to get the estimates for the sentiment score p_{new} for the new article given the article's word counts of the sentiment charged words, denoted as $d_{new,\mathbf{S}}$. With the multinomial distribution assumption, I can write the optimization problem as:

$$\hat{p}_{new} = argmax_p \sum_{j \in \hat{\mathbf{S}}} \log \left(p\hat{O}_{j+} + (1-p)\hat{O}_{j-} \right)^{d_{new,j}} + \lambda \log(p(1-p))$$

The final penalization term drags the estimates of p toward 0.5, which is the sentiment score for sentiment-neutral articles. Note that the penalization is needed here because the total counts of sentiment charged words, $|d_{new,\mathbf{S}}|$, can be small. In this case, the sample frequency may be a noisy approximate to the true word distributions.

In all, there are four hyperparameters in this procedure: α_+, α_-, κ, and λ, three controlling the screening of the sentiment charged words and the remaining one controlling the strength of penalty in the scoring new article step. I will specify how I tune these parameters in detail in the empirical analysis part.

3 Data

In this part, I will first provide a detailed description of the data collection and cleaning for the Chinese company announcements. Next, I will focus on the merge between announcement articles and corresponding open-to-open returns. Finally, I will specify the empirical analysis setup with detailed information on training, tuning, and testing schemes.

3.1 Data Collection

I used the Chinese company announcements downloaded from the Juchao Information (http://www.cninfo.com.cn/new/index), which is the official information designated by the China Securities Regulatory Commission. According to Juchao, this platform is the place where market participants can get the earliest release of the company announcements.

My raw data has two components: (1) the company announcements in PDF format, which are downloaded using a scraper directly from Juchao; (2) the metadata for the company announcements containing the detailed information of announcement id, company stock id, announcement release time, categories and the reporting entity. I get the metadata from an official API created by Juchao.

The time range for my data is from the year 2012.01.01 to 2019.12.31. I start my analysis in the year 2012 because this is the time when the API starts. For data before 2012, I cannot know the precise release time of those documents for market participants. There are 3,498,921 documents over this sample period. On average, there are 1800 documents per day. Figure 2 shows the number of articles over time. As can be seen from the figure, there is an upward trend for company announcements. In the year 2012, I have 240,924 documents in total, while toward the end of the sample in the year 2019, the total number goes up to 706,990. I remove all the announcements with 0 words.

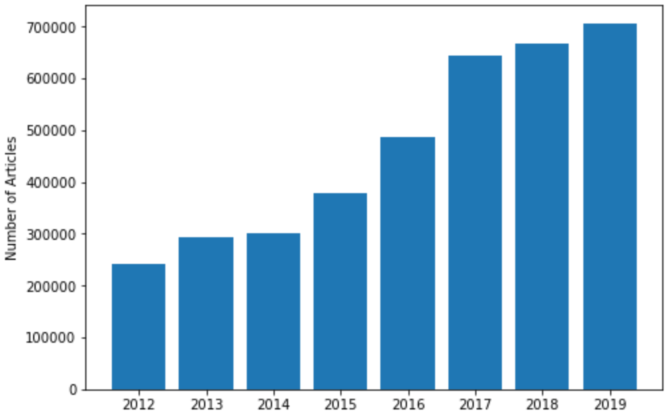

Figure 2: Number of articles over time

As for the announcement categories, they cover nearly all important firm financial activities. The categories include: regular reports such as annual or quarterly reports, IPO announcements, announcements on the general meetings of shareholders, company financing and investment decisions, merge and acquisition, stock price abnormal movement announcements, special treatment and many others. I produce a wordcloud plot in figure 3 to show the categories. The font size corresponds to the document counts of the corresponding category. As can be seen from the figure, this dataset covers much different textual information about Chinese companies.

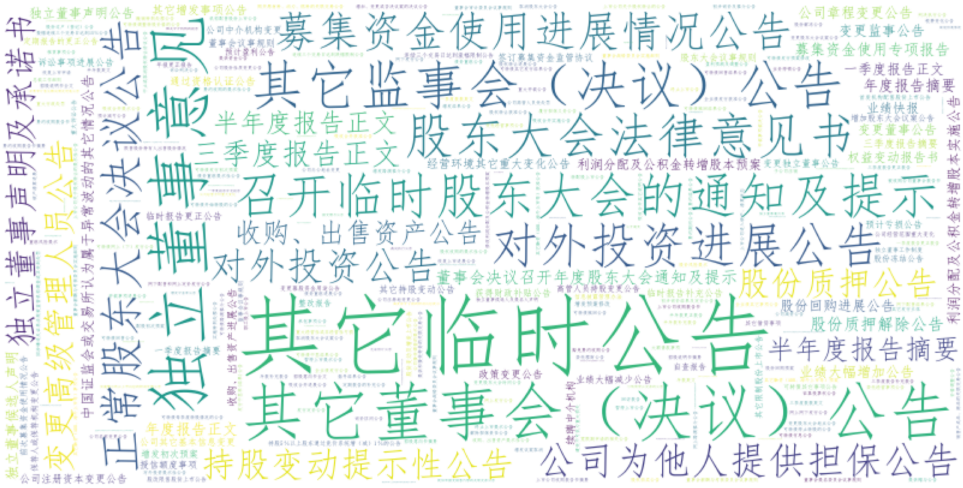

Figure 3: Announcement Categories

Among all the announcements, I remove the announcements related to the abnormal trading activities. Due to the daily price limits and the liquidity concerns, it is very difficult to trade on this type of announcement.

3.2 Data Cleaning

For data cleaning, I followed the procedure used in Fan et al. (2021), who applied this textual analysis approach to the Chinese news data scraped from Sina news. In the first step, I convert the PDF documents into texts. It is worth noting that many company announcements contain tables that contain many noise words for my analysis. However, there is no reliable algorithm that can remove tables when parsing PDFs. To accommodate this issue, I directly add the noisy words that have no meaning resulting from parsing PDF tables to the stopwords list in order to get rid of them.

Next, for all the documents, I retain only Chinese characters. This means getting rid of all the numbers, special symbols, punctuation marks, as well as English characters.

With these Chinese-only texts, I further used Jieba to cut them into wordlist. Note that, unlike English, there is no natural space between words for the Chinese language. This poses a major challenge for Chinese NLP. Jieba (Chinese for ‘to stutter') is a package for Chinese text segmentation. It adopts a hidden Markov approach where each Chinese character is linked to one of the four hidden states: B(begin), M(middle), E(end) or S(single). The Viterbi algorithm is used to find the most likely hidden state given the observed character sequence. The text segmentation task will be done based on the inferred hidden states.

Finally, I remove the Chinese stopwords as well as the noisy, meaningless words resulting from parsing tables in PDFs. After this step, I get a dictionary of size 5,605,738.

3.3 Return Merging

The stock returns data I use come from the CSMAR database, which offers detailed daily stock trading data in the Chinese stock market. However, the daily returns data at time t from CSMAR are returns from the time $t-1$ market close at 15:00 to time t market close at 15:00. If I directly merge the stock returns with the announcements, since the news at date t has a time range from 0:00 to 23:59, the $t+1$ trading strategy using news published after 15:00 will not be feasible. To accommodate this, I construct the open-to-open return using CSMAR data. To be specific, I divide all the daily open prices by the corresponding adjusting factor for the cash dividend. The time t open-to-open daily return is constructed by dividing time $t+1$ adjusted open price at 9:30 against the time t one at 9:30.

Next, we merge the daily open-to-open returns with all the company announcements. After the merging step, I finally get a dataset consisting of 1,859,364 articles. Table 1 presents the summary statistics at the article level for returns, number of words, and number of unique words. It is worth noting that the number of words is highly skewed across documents. The mean is 1890 per document, while the median is only 536 per documents.

Data	mean	std	skew	10%	25%	50%	75%	90%
returns	0.20%	6.34%	161.99	-3.50%	-1.48%	0.09%	1.69%	3.89%
# words	1890	5918	11.36	145	228	536	1116	2942
unique words	370	595	4.86	78	111	199	335	702

Table 1: Summary Statistics at Article Level ($N = 1,859,364$)

I also present the summary statistics for the number of articles, number of stocks, and proportion of positive returns across time in table 2. There are 1943 days in total from the year 2012 to 2019. Both the number of articles and the number of stocks have positive skewness. For the proportion of positive returns, its mean and median are very close to 50%.

Data	mean	std	skew	10%	25%	50%	75%	90%
# articles	957	724	2.64	303	474	824	1188	1681
# stocks	374	187	0.93	163	224	351	492	606
% +ret	53.60%	25.04%	-0.05	18.83%	33.69%	53.75%	74.05%	88.01%

Table 2: Summary Statistics across Time ($T = 1943$)

3.4 Training, Tuning, and Testing

In this part, I specify the details of the training, tuning, and testing schemes. Firstly, the model requires 4 hyperparameters: α_+, α_-, κ, and λ. To tune the model, I train it on the data from 2012 to 2014 and test its performance in the year 2015. In the training step, I link time t announcements with time t returns. During the validation step, I link time t announcements with time $t+1$ returns and check the portfolio performance of longing the articles with the top 50 sentiment scores while shorting the articles with bottom 50 sentiment scores. The final hyperparameters are chosen to maximize the time $t+1$, $50-50$ long-short portfolio performance in the validation set.

I used a grid search method to find the optimal hyperparameters. $\alpha_+(\alpha_-)$ is chosen so that there are exactly $[25, 50, 100]$ words in the positive(negative) words dictionary during the marginal screening step. For κ, my original dictionary has 5,605,738 words while only 165,909 of them appear in at least 50 documents. In my later analysis, I focus on these 166K words that appear more than 50 times. κ will be chosen from the $[92, 94, 96, 98]$ percentiles of the word occurrence counts. Last but not least, the penalization parameter λ will be chosen from $[1, 5, 10]$. This gives us $3 \times 3 \times 4 \times 3 = 81$ different model specifications.

On our validation set, the model that has the highest $50-50$ long-short return is the one with $\alpha_+ = 25$, $\alpha_- = 50$, $\kappa = 94$, and $\lambda = 5$, meaning: (1) there are 25 words in the positive sentiment dictionary, 50 in negative sentiment dictionary; (2) the cutting threshold will be 94 percentile of all words' occurrence counts; (3) the

MLE penalty parameter is set to be 5. In my out-of-sample test, this specification will be fixed, and I do not re-tune the model.

My out-of-sample test begins in 2016 and ends in 2019. For this data, I run a 6-month rolling estimation, e.g., 2016.01.01 - 2016.06.30 portfolio strategy is based on model estimated from 2012.01.01 to 2015.12.31. The window is expanded forward with new data. The last window uses data from 2012.01.01 to 2019.06.30 to train the model, and the portfolios are formed from 2019.07.01 to 2019.12.31. There are eight six-month windows in total.

4 Empirical Analysis

In this part, I focus on the main empirical results. Firstly, I will present the sentiment-charged words from marginal screening.

4.1 Sentiment-charged Words from Marginal Screening

I first report the positive and negative word list from the marginal screening step. In this part, I follow the tuning parameters and consider the word occurring greater than the 94 percentile of all word occurrence counts (in our sample, this corresponds to words occurring in at least 4757 articles).

Figure 4 and figure 5 shows the wordcloud for the positive and negative words. The font size is proportional to their corresponding f value.

I also provide a list of top 20 positive and negative words in table 3 and table 4, with pinyin, English translations, total article occurrence counts and corresponding f values.

Firstly, compare table 3 and table 4, I find that the negative words list seems to be more intuitive. There are many negative-sentiment words: negative news, incompatible, weakness, inconvenient and etc. For the positive word list, there are also some positive words like excess surplus and rational, but the rest of them are not as intuitive as the negative list. In Loughran and McDonald (2011), the authors find similar patterns in the US stock market. They argue that this is

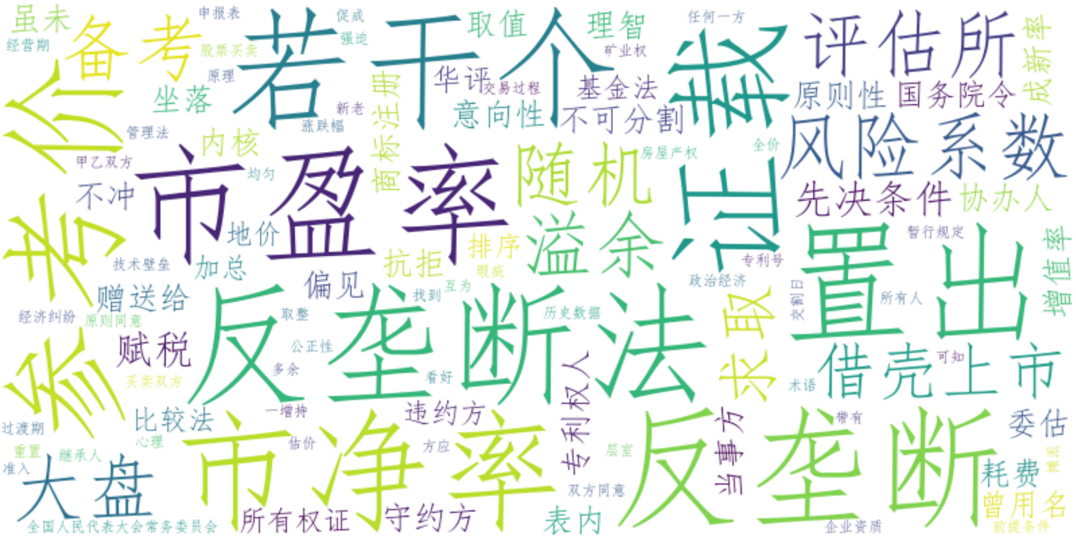

Figure 4: Positive Sentiment-Charged Words from Marginal Screening

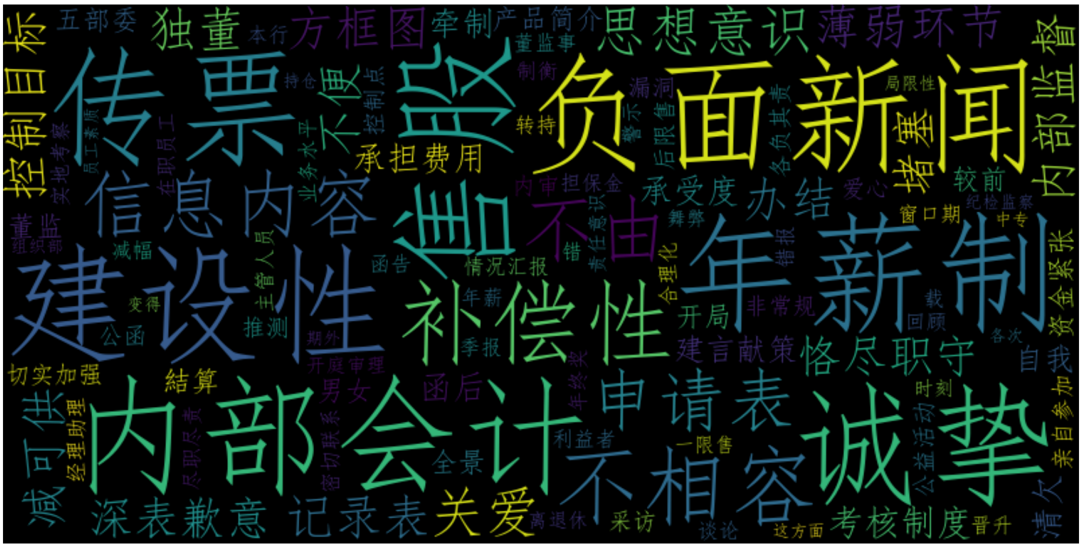

Figure 5: Positive Sentiment-Charged Words from Marginal Screening

due to the negation for positive words. It is far more common for managers to use negation before positive words to convey negative meanings rather than use negative words directly. This confounds the association between positive occurrence and outcome. While for negative words, when it occurs, it is indeed having negative meanings.

Furthermore, the algothrithm captures some interesting financial activities for the companies. In table 4, I find the sale of shares as a negative word for stock price. This is intuitive as it may send out the signal that firms believe the share price is very high. In table 3, the word backdoor listing is considered as

Word	Pinyin	English	# Count	f_k
反垄断法	fan long duan fa	antitrust law	5167	0.619
证载	zheng zai	evidence	4974	0.616
置出	zhi chu	sell out	7783	0.608
市盈率	shi ying lv	P/E ratio	9914	0.606
若干个	ruo gan ge	several	5025	0.602
反垄断	fan long duan	antitrust	11183	0.602
市净率	shi jing lv	P/B ratio	6165	0.602
参考价	can kao jia	reference price	9711	0.599
风险系数	feng xian xi shu	risk factor	8811	0.599
随机	sui ji	random	7059	0.599
评估所	ping gu suo	evaluation office	10240	0.597
大盘	da pan	market	6878	0.596
备考	bei kao	for reference	23992	0.595
溢余	yi yu	excess surplus	9794	0.595
借壳上市	jie ke shang shi	backdoor listing	10672	0.594
求取	qiu qu	seek	4782	0.594
先决条件	xian jue tiao jian	prerequisites	10423	0.594
赋税	fu shui	tax	4846	0.592
不可分割	bu ke fen ge	indivisible	7879	0.592
理智	li zhi	rational	6004	0.592

Table 3: Top Positive Sentiment-charged Words from Marginal Screening

a positive word. This occurs because of the Chinese special listing system. It is run by an approval system instead of registration system. As a result, many firms choose backdoor listing. When the news is announced, the ‘shell’ company value will typically rise significantly. This effect is also captured by the algorithm.

4.2 A Comparison Study with US Data

In this part, I compare the results from Chinese company announcements with the US counterpart. I collected the management discussion and analysis part from the US company’s 10-K reports over the year 2010 - 2020 from EDGAR. For this sample, I performed the same marginal screening procedure to get the sentiment-charged word lists. I will first briefly introduce my US data sample. Next I will analyze and compare the empirical results with Chinese data.

Word	Pinyin	English	# Count	f_k
诚挚	cheng zhi	sincere	4887	0.460
售股	shou gu	sale of shares	6700	0.469
内部会计	nei bu hui ji	internal accounting	4934	0.472
负面新闻	fu mian xin wen	negative news	5848	0.472
年薪制	nian xin zhi	annual salary system	4963	0.475
传票	chuan piao	summons	5013	0.475
建设性	jian she xing	constructive	13725	0.476
补偿性	bu chang xing	compensatory	9086	0.477
不相容	bu xiang rong	incompatible	12223	0.477
信息内容	xin xi nei rong	information content	16193	0.479
不由	bu you	not by	35700	0.479
申请表	shen qing biao	application form	9428	0.479
思想意识	si xiang yi shi	ideology	9346	0.480
关爱	guan ai	caring	6862	0.481
薄弱环节	bo ruo huan jie	weakness	7372	0.482
控制目标	kong zhi mu biao	control objectives	13263	0.482
独董	du dong	independent director	6921	0.482
内部监督	nei bu jian du	internal supervision	18729	0.482
不便	bu bian	inconvenient	16873	0.482
减可供	jian ke gong	less available	6237	0.483

Table 4: Top Negative Sentiment-charged Words from Marginal Screening

4.2.1 US MD&A Data

I downloaded all 10-Ks from the EDGAR website (www.sec.gov) over the year 2010 to 2020 with the python API sec-edgar. There are over 80,000 documents in my sample period. I then merge the 10-K text with CRSP daily returns, requiring that the company is traded publicly in the stock market and reported on CRSP as an ordinary common equity firm. After the merge, I get 37,720 documents in total. The drop in number is not surprising as there are many real estate, nonoperating, or asset-backed partnerships/trusts that are not publicly traded but are required to file with SEC. My final corpus has 5,405 companies with 242,263,144 words in total. Figure 6 shows the number of articles over time.

To carry out an efficient comparative analysis, I mainly focus on the Management Discussion and Analysis subsection (item 7) of the 10-K. This is a separate section in the company annual report that provides insightful information on the business status and financial situations of the company in view of the various macro-economic barriers under which it operates. This section contains the most

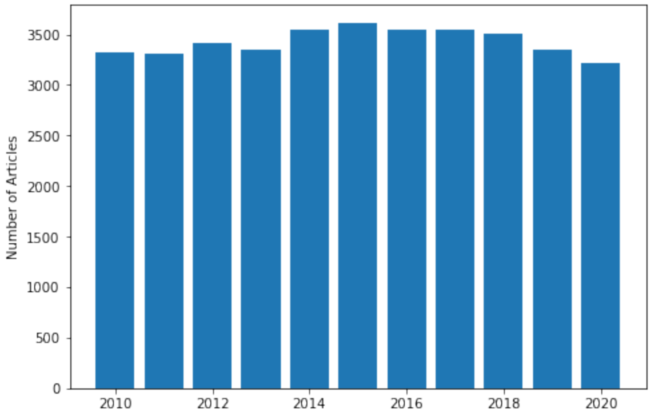

Figure 6: Number of Articles over Time for the US Sample

important textual information in 10-K reports.

To parse the 10-K text and extract the MD&A section (item 7), I first apply beautiful soup to the html files downloaded. After that, we look for item 7 in the html. We start read the management discussion and analysis from item 7 until we see item 8. We exclude extracted text with less than 1000 characters because there are some cases the MD&A section information is 'incorporated by reference'(typically deferring to the shareholders annual report). At text level, we also remove special characters in the text.

After these steps, I performed data cleaning at the word level. I first tokenize the text. Next I remove the stopwords. Finally, I performed lemmatization to all the words.

With all the preparations, I get the bag-of-word representation of the company MD&A linked to the corresponding returns on the day the annual report is made public.

4.2.2 Sentiment-charged Word Lists from US Sample

I calculated the f-value using the same formula in equation 1 for the words in the US MD&A sample. I set κ such that the words occurring in at least $2,000$ documents. Figure 7 and figure 8 show the wordclouds for the positive sentiment-charged words and the negative sentiment-charged words. I also report the top 20 negative and positive words with their corresponding f value and document occurrence counts in table 5.

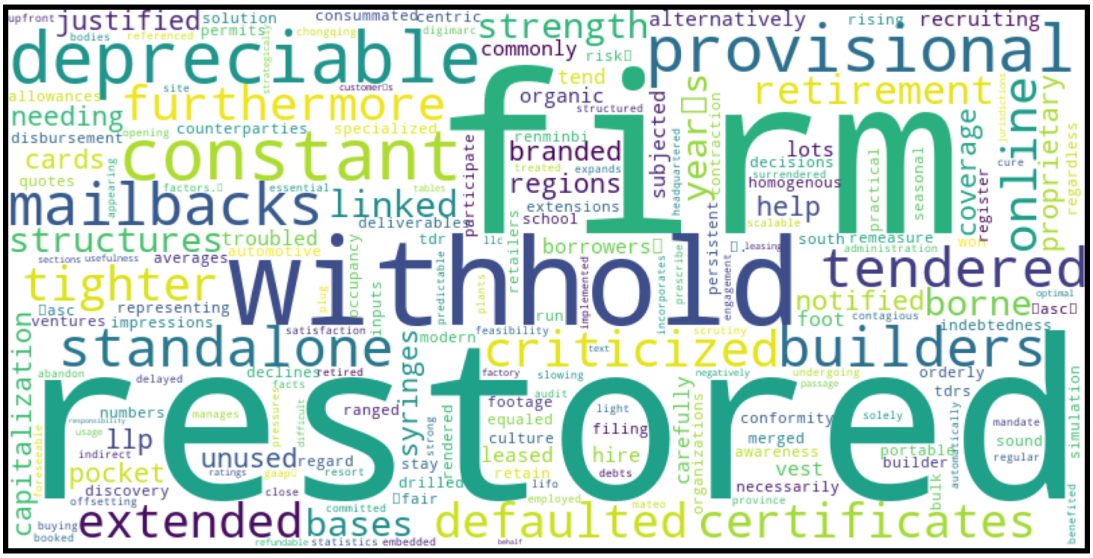

Figure 7: Positive Sentiment-charged Words for the US Sample

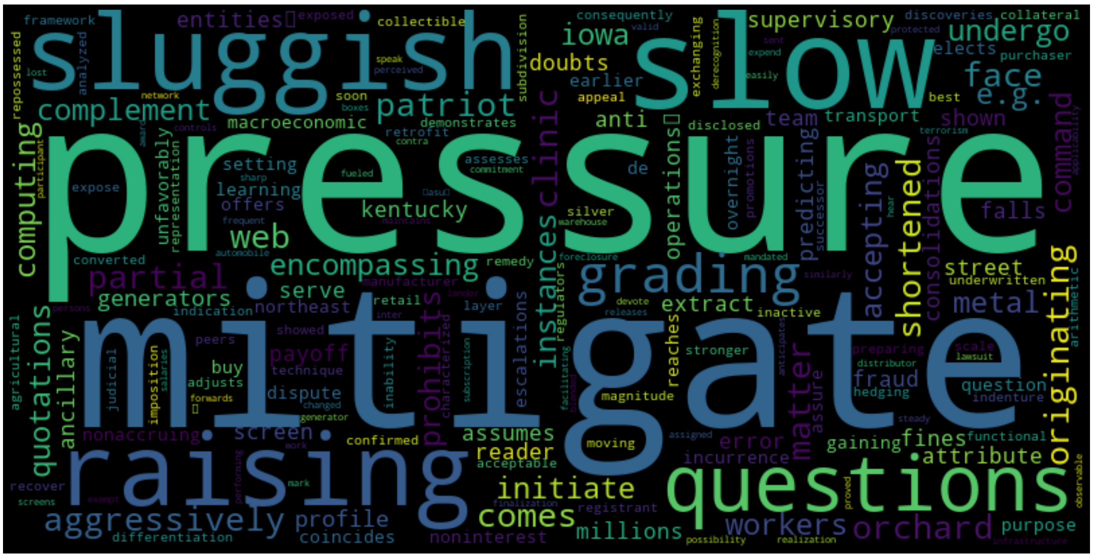

Figure 8: Negative Sentiment-charged Words for the US Sample

Firstly, as can be seen from figure 8 and table 5, the negative words list is very intuitive. I find many words that are related to negative meanings in this sample, such as pressure, slow, sluggish, raising, questions, and etc. On the other hand, the positive words contain much noise compared to the negative ones, which can be seen from the figure 7. In this sense, the US sample has much similarity with Chinese ones. In both scenarios, I find it is the negative word list is more intuitive and understandable. The similar results are found in Loughran and McDonald (2011). In some sense, the negative words carry much heavier weights than the positive ones. When managers use negative words in the announcements,

Negative Words			Positive Words		
Word	f_k	# Count	Word	f_k	# Count
pressure	0.436	3574	restored	0.519	2283
mitigate	0.445	2128	firm	0.512	2679
slow	0.447	8707	withhold	0.511	3985
sluggish	0.450	2655	depreciable	0.511	3258
raising	0.450	4090	constant	0.511	2188
questions	0.450	5105	provisional	0.508	2270
grading	0.451	2439	mailbacks	0.506	3001
aggressively	0.451	2479	standalone	0.506	5028
partial	0.452	3237	tendered	0.506	2844
face	0.452	2538	builders	0.506	3011
matter	0.453	2302	criticized	0.506	2320
shortened	0.453	2514	furthermore	0.506	4247
clinic	0.453	2443	online	0.506	5619
orchard	0.455	2365	defaulted	0.505	2461
comes	0.455	2822	extended	0.505	6246
originating	0.455	2216	certificates	0.505	2321
patriot	0.455	8261	tighter	0.505	5431
web	0.455	2378	retirement	0.505	2676
encompassing	0.456	3024	strength	0.505	2764
complement	0.456	2929	borne	0.504	4375

Table 5: Top 20 Negative and Positive Sentiment-charged Words from Marginal Screening for US 10-K MD&A Data

they mean it. On the other hand, the positive words can be confounded by the negations: it is more common to see the framing of ‘not benefit’ than ‘harm’ for company announcements. Furthermore, from table 5, I find the f value of negative words are more different from 0.5 than the positive words. This lends additional support to the fact that negative words play a more important role for understanding market sentiment and reactions toward announcements.

Since I only have 30K documents for the US sample and the 10-K announcements have great seasonality pattern (most companies will have their 10-K published around March or April), it is not possible to build a trading strategy. Therefore, for the portfolio performance evaluation, I focus on the Chinese Company Announcements.

4.3 Portfolio Performance

In this part, I focus on the out-of-sample portfolio performance of the text-based trading strategy pioneered by Ke et al. (2019). In this out-of-sample analysis, I split the 2016-2019 sample period into intervals of 6 month and run expanding rolling estimation for the model. The estimated model will be fixed within the 6-month period. The strategy I considered is to long 50 stocks with the highest sentiment score and to short the 50 stocks with the lowest sentiment score.

I run three separate experiments to examine the relation between the news sentiment and the stock returns on the day $t-1$, t, and $t+1$, which I abbreviated as day $t-1$, t, and $t+1$ strategy. It is worth noting that the day $t-1$ and day t strategies are not implementable in reality. The day $t-1$ strategy is meant to examine whether there is information leakage before the announcements are made public. If this is the case, there will be significant portfolio returns for linking news sentiment with the previous one day stock returns. The day t strategy is used to examine whether the algorithm picks up important return-related information in the text. This can be interpreted as a prediction task from text to contemporaneous market response.

The most important one will be the day $t+1$ strategy. This is a feasible strategy in reality. For this strategy, at 9:30 when the market opens, it trades stocks based on the announcement information from the previous day and hold the portfolio for one day. The entire portfolio will be rebalanced every day based on the daily updating announcement information. This strategy examines whether the algorithm I employed can pick up the news sentiment momentum contained in announcements. Consider the case where market participants have rational limited attention. When thousands of announcements come to the market on one single day, it is highly likely that the market participants cannot fully digest the information content as well as complex interactions among all of them. In this sense, the market will underreact to the information contained in the announcements. If this is true, it is likely that the algorithm can pick up some leftover pricing signals contained in the announcements.

I present the cumulative returns for day $t+1$, t, and $t-1$ portfolios in figure 9, figure 10, and figure 11. The summary statistics of these portfolios are reported in the table 6.

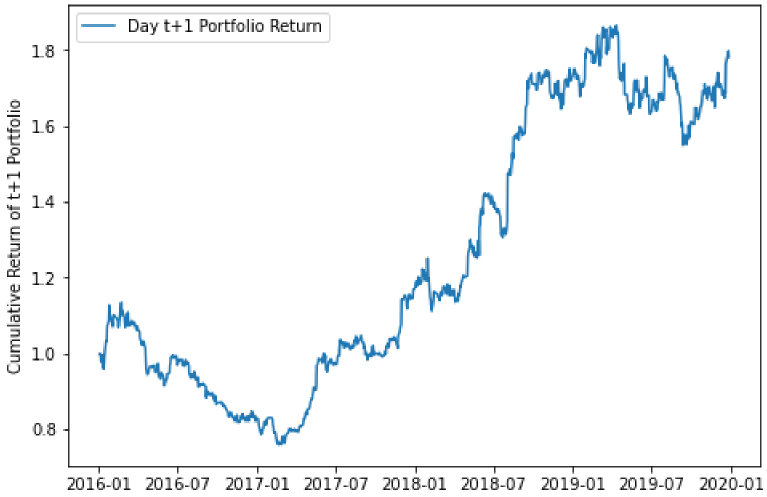

Figure 9: Cumulative Return of Day $t + 1$ Portfolio

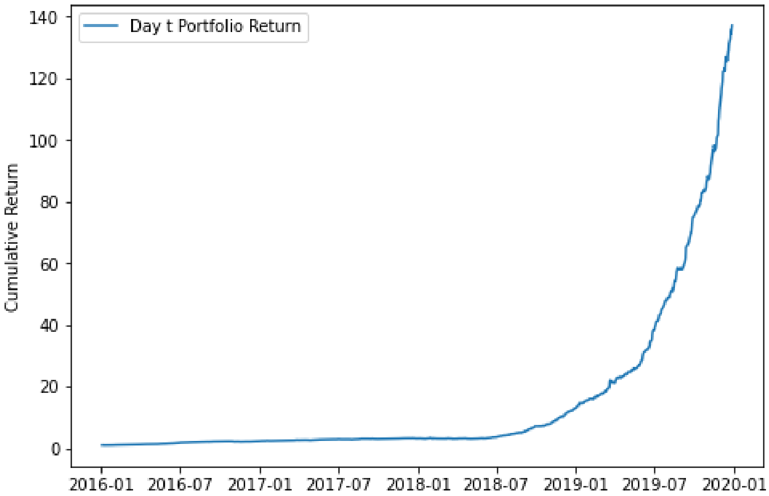

Figure 10: Cumulative Return of Day t Portfolio

Firstly, as can be seen from the figure 9, the day $t + 1$ generate a cumulative return of 78% during this 4-year period. The long-short portfolio has a Sharpe ratio of 0.86. During this time period, the CSI 300 index, the Chinese counterpart to the S&P 500 index only increased from 3731.00 to 4096.58 with a 9.8% percent rise.

Using the Chinese Fama-French factors constructed by China Asset Management Center from the CUFE, I regress the day $t + 1$ portfolio returns on the Fama-French factors. I considered the CAPM, Fama-French 3 and 5 factor models as well as the Fama-French 5 factor with momentum. The results are shown in the table 7. From the table, I find that the regression α is significant at 10% level

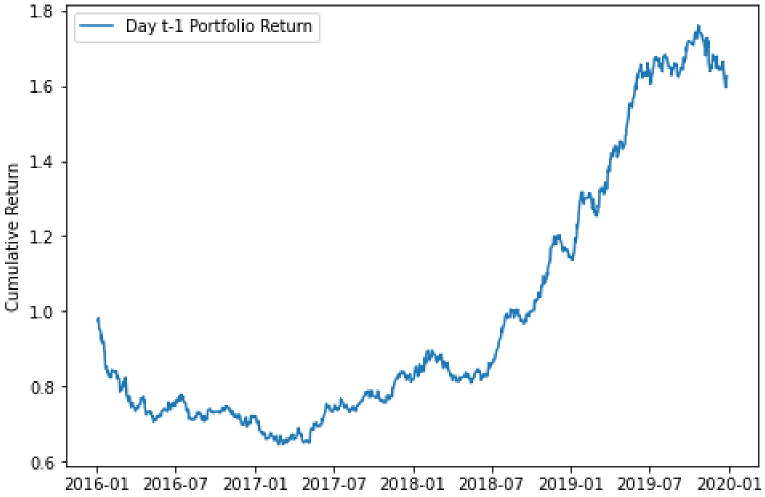

Figure 11: Cumulative Return of Day $t - 1$ Portfolio

across these different model specifications.

In general, the day $t+1$ text-based trading strategy is able to generate returns far above the market returns. This suggests that the algorithm is able to extract the leftover return-predictive signals from the company announcements.

Furthermore, for the day t infeasible portfolio, I find that the algorithm is able to generate very high returns. Table 10 shows that this infeasible long-short portfolio attains a Sharpe ratio of 6.79. The portfolio cumulative return is shown in figure 10. This reflects the fast response the market has toward the new information. When the announcement is released, its effect will be reflected in prices very quickly. This experiment also suggests that the algorithm is able to detect the return-related information content in company announcements. Given an announcement, the model performs very well in judging whether its contemporaneous return is positive or negative.

Last but not least, for the day $t - 1$ portfolio, its cumulative return is shown in figure 11. There is also significant positive return for this day $t - 1$ portfolio. As can be seen from the table 6, the long-short portfolio has a Sharpe ratio of 0.85. In this sense, there is some evidence for the information leakage in the Chinese market in that the prices have already started to change prior to the announcement. However, its magnitude is much smaller compared to the day t portfolio.

In the final section of my portfolio analysis, I ask the question what kind

Portfolio	SR	mean	std	min	25%	50%	75%	max
t+1 L-S	0.86	0.07%	1.14%	-4.61%	-0.54%	0.00%	0.58%	9.35%
t+1 L	-0.63	-0.07%	1.85%	-9.77%	-0.97%	0.01%	0.89%	7.75%
t+1 S	-1.29	-0.14%	1.72%	-6.83%	-1.03%	-0.02%	0.83%	6.97%
t L-S	6.79	0.51%	1.19%	-5.18%	-0.21%	0.42%	1.15%	8.11%
t L	4.28	0.55%	2.03%	-15.83%	-0.37%	0.60%	1.55%	8.36%
t S	0.29	0.04%	1.89%	-13.51%	-0.84%	0.20%	0.98%	7.40%
t-1 L-S	0.85	0.05%	0.94%	-4.06%	-0.50%	0.05%	0.61%	4.42%
t-1 L	1.13	0.14%	1.89%	-14.93%	-0.75%	0.27%	1.17%	7.46%
t-1 S	0.69	0.08%	1.84%	-14.37%	-0.77%	0.17%	1.04%	7.08%

Table 6: Summary Statistics for Portfolio Returns

	CAPM		FF3		FF5		FF5+MOM	
α	0.0007*	(1.79)	0.0007*	(1.87)	0.0007*	(1.91)	0.0007*	(1.92)
MKT	-0.0498*	(-1.88)	-0.0669**	(-2.29)	-0.0490	(-1.49)	-0.0529	(-1.60)
SMB			0.0107	(0.18)	0.0169	(0.13)	0.0229	(0.18)
HML			-0.1183*	(-1.80)	-0.2039**	(-2.20)	-0.1944**	(-2.08)
RMW					0.0472	(0.44)	0.0234	(0.21)
CMA					0.1370	(1.30)	0.1399	(1.33)
MOM							0.0455	(0.86)

Table 7: Day $t+1$ Portfolio Fama-French Factor Regression

of announcements matter for stock returns. To examine this, I look into the categories of the articles being traded. Table 8 reports the document counts, proportion of all the articles traded, mean word count and median word count for the top 20 categories of articles used for trading.

From table 8, the traded articles are much longer compared to average articles. There are 27,431 words on average in traded articles while there are only 1,890 on average in all articles. Firstly, the longer announcements mechanically contain more information. Thus, they are more likely to have more significant impact on stock returns. Moreover, due to cognitive limitations, market participants may not fully digest all the information contained in the articles. This rational limited attention view is also consistent with the fact that my algorithm picks up return-related signals contained in these long articles.

Furthermore, I find that company's regular announcement plays a very significant role in asset pricing. As can be seen from the table 8, annual report, with a document count of 7,799, ranked second among all articles traded, followed by the

Category	Count	Prop	Mean	Median
Other Temporary Report	16524	0.17	14601	4630
Annual Report	7799	0.08	94873	91686
Semi-annual Report	5856	0.06	71503	69231
Shareholding Change Report	5294	0.055	1663	1383
Asset Purchase, Sale Report	5152	0.053	63032	12799
Investment Progress Report	4807	0.05	3663	2744
Board of Directors Announcement	3510	0.036	5098	3199
Foreign Investment Announcement	2505	0.026	5689	3512
Third Quarter Report	2222	0.023	11881	10540
Equity Change Report	2023	0.021	10455	7534
Annual Report Correction	2008	0.021	87002	93320
Board of Supervisors Announcement	1937	0.02	4638	2412
Additional Issuance Announcement	1919	0.02	48281	15640
Progress Report on Raised Funds	1861	0.019	6817	2230
Additional Listing Announcement	1694	0.017	17194	15528
Independent Directors' Opinions	1615	0.017	1292	941
Notice of Extraordinary General Meeting	1349	0.014	4233	4055
Performance Bulletin	1249	0.013	878	842
Announcement on Profit Distribution	1224	0.013	1529	1450
Legal Opinions of the Shareholders' Meeting	1094	0.011	5060	3982
All Traded Articles	97300	.	27431	7399
All Articles	1859364	.	1890	536

Table 8: Summary Statistics (Document Count, Proportion of All Articles Traded, Mean Word Count and Median Word Count) for Traded Articles Group by Categories

semi-annual report with a document count of 5,856. The third-quarter report is also ranked in the top 10 categories. The regular reports of companies contain important quantitative information about a company's financial performance. This study suggests that the textual information in these reports is also very informative about the stock market response. Additionally, I find reports related to firm's financing decisions also play important roles in affecting the stock prices. Reports shareholding changes, equity composition change, additional issuance and listing are also among the top 20 traded categories.

5 Conclusion

Employing a novel algorithm pioneered by Ke et al. (2019), I carry out a comprehensive analysis of stock return prediction based purely on the textual information in Chinese company announcements.

The technique follows three steps: (1) screen for sentiment charged words dictionary; (2) learn sentiment charged topics; (3) score new articles based on the dictionary and word weight learned with penalized MLE. The power of this approach lies in its data-driven characteristics. There are two major advantages. Firstly, unlike traditional pre-defined ad hoc dictionaries, it let the data determine the words that are significantly associated with the variable of interest, i.e., stock returns. Secondly, by adopting a probabilistic approach, the model can find the optimal weights for words based on the response strength of the outcome variable.

I demonstrate the usefulness of the text-mining algorithm by applying the algorithm to the Chinese company announcements in many different ways. Firstly, I use the marginal screening technique to extract the sentiment-related word dictionary from the text directly. I find that the negative words are very meaningful compared to positive ones. In a comparative study using the US management discussion and analysis text, I find similar patterns: the negative word list extracted from the marginal screening step contain many intuitively negative words about the future performance of the company.

Furthermore, I separately investigated the out-of-sample association between the announcement sentiment and stock returns on the day $t-1$, t, and $t+1$. For each scenario, I examine the performance of the portfolios that long the top 50 high-sentiment articles and short the bottom 50 ones. Firstly, the implementable day $t+1$ trading strategy offers a cumulative return of 78% over the four-year test sample with a Sharpe ratio of 0.86. This suggests that there exists some news-sentiment momentum in the announcement texts. Even one day after the publication of the news, there is still some information left over. From the day t portfolio, I find a significant association between article sentiment scores and portfolio returns, which suggests the algorithm can find the information in the announcements that indicate high contemporaneous returns. The day $t-1$ portfolio also generates positive returns, meaning some information leakage even before the announcements are made public.

After careful examination of the articles traded, I find they are, on average, having much higher length than average articles. Company regular announcements and announcements related to financing activities are important compositions of the traded articles. All these facts are consistent with the hypothesis that market participants have limited attention. As a result, textual information in company announcements has some predictive power for future asset prices over a short horizon.

References

Antweiler, W., Frank, M.Z., 2004. Is all that talk just noise? the information content of internet stock message boards. The Journal of finance 59, 1259–1294.

Baker, S., Bloom, N., Davis, S., 2016. Measuring economic policy uncertainty. The Quarterly Journal of Economics 131, 1593–1636. URL: `https://EconPapers.repec.org/RePEc:oup:qjecon:v:131:y:2016:i:4:p:1593-1636`.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. the Journal of machine Learning research 3, 993–1022.

Bybee, L., Kelly, B.T., Manela, A., Xiu, D., 2020. The structure of economic news. Technical Report. National Bureau of Economic Research.

Calomiris, C.W., Mamaysky, H., 2019. How news and its context drive risk and returns around the world. Journal of Financial Economics 133, 299–336.

Cohen, L., Malloy, C., Nguyen, Q., 2020. Lazy prices. The Journal of Finance 75, 1371–1415.

Cong, L.W., Liang, T., Zhang, X., 2019. Textual factors: A scalable, interpretable, and data-driven approach to analyzing unstructured information. SSRN (September 1, 2019) .

Cowles, A., 1933. Can stock market forecasters forecast? Econometrica: Journal of the Econometric Society , 309–324.

Da, Z., Engelberg, J., Gao, P., 2015. The sum of all fears investor sentiment and asset prices. The Review of Financial Studies 28, 1–32.

Deng, K., Bol, P.K., Li, K.J., Liu, J.S., 2016. On the unsupervised analysis of domain-specific chinese texts. Proceedings of the National Academy of Sciences 113, 6154–6159.

Evans, J.A., Aceves, P., 2016. Machine translation: Mining text for social theory. Annual Review of Sociology 42, 21–50.

Fan, J., Lv, J., 2008. Sure independence screening for ultrahigh dimensional feature space. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 70, 849–911.

Fan, J., Xue, L., Zhou, Y., 2021. How much can machines learn finance from chinese text data? Available at SSRN .

Gentzkow, M., Kelly, B., Taddy, M., 2019. Text as data. Journal of Economic Literature 57, 535–74.

Gu, S., Kelly, B., Xiu, D., 2020. Empirical asset pricing via machine learning. The Review of Financial Studies 33, 2223–2273.

Gu, S., Kelly, B., Xiu, D., 2021. Autoencoder asset pricing models. Journal of Econometrics 222, 429–450.

Jegadeesh, N., Wu, D., 2013. Word power: A new approach for content analysis. Journal of financial economics 110, 712–729.

Jiang, F., Lee, J., Martin, X., Zhou, G., 2019. Manager sentiment and stock returns. Journal of Financial Economics 132, 126–149.

Ke, Z.T., Kelly, B.T., Xiu, D., 2019. Predicting returns with text data. Technical Report. National Bureau of Economic Research.

Knight, W., 2016. Will ai-powered hedge funds outsmart the market.

Lopez-Lira, A., 2020. Risk factors that matter: Textual analysis of risk disclosures for the cross-section of returns. Jacobs Levy Equity Management Center for Quantitative Financial Research Paper .

Loughran, T., McDonald, B., 2011. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. The Journal of finance 66, 35–65.

Loughran, T., McDonald, B., 2016. Textual analysis in accounting and finance: A survey. Journal of Accounting Research 54, 1187–1230.

Manela, A., Moreira, A., 2017. News implied volatility and disaster concerns. Journal of Financial Economics 123, 137–162.

Nagel, S., 2021. Machine Learning in Asset Pricing. volume 1. Princeton University Press.

Tetlock, P.C., 2007. Giving content to investor sentiment: The role of media in the stock market. The Journal of finance 62, 1139–1168.

Tetlock, P.C., 2014. Information transmission in finance. Annu. Rev. Financ. Econ. 6, 365–384.

Tetlock, P.C., Saar-Tsechansky, M., Macskassy, S., 2008. More than words: Quantifying language to measure firms' fundamentals. The Journal of Finance 63, 1437–1467.