How Much Can Machines Learn Finance From Chinese Text Data? *

Jianqing Fan
Department of ORFE
Princeton University
NJ 08544, USA
jqfan@princeton.edu

Lirong Xue
Department of ORFE
Princeton University
NJ 08544, USA
lirongx@princeton.edu

Yang Zhou
School of Data Science
Fudan University
Shanghai 200433, China
yang.zhou@princeton.edu

January 11, 2021

*The authors are grateful to Professor Wei Xiong for various comments and suggestions. Comments and suggestions are greatly appreciated.

Abstract

Most studies on equity markets using text data focus on English-based specified sentiment dictionaries or topic modeling. However, can we predict the impact of news directly from the text data? How much can we learn from such a direct approach? We present here a new framework for learning text data based on the factor model and sparsity regularization, called *FarmPredict*, to let machines learn financial returns automatically. Unlike other dictionary-based or topic models that have stringent pre-screening processes, our framework allows the model to extract information more fully from the whole article. We demonstrate our study on the Chinese stock market, as Chinese text has no natural spaces between words and phrases and the Chinese market has a very large proportion of retail investors. These two specific features of our study differ significantly from the previous literature that focuses on English-text and the U.S. market. We validate our method using the literature on the Chinese stock market with several existing approaches. We show that positive sentiments scored by our FarmPredict approach generate on average 83 bps stock daily excess returns, while negative news has an adverse impact of 26 bps on the days of news announcements, where both effects can last for a few days. This asymmetric effect aligns well with the short-sale constraints in the Chinese equity market. As a result, we show that the machine-learned sentiments do provide sizeable predictive power with an annualized return of 116% with a simple investment strategy and the portfolios based on our model significantly outperform other models. This lends further support that our FarmPredict can learn the sentiments embedded in financial news. Our study also demonstrates the far-reaching potential of using machines to learn text data.

Key Words: Machine Learning, Factor Model, Sparse Regression, Textual Analysis, Sentiment Scores, Event Studies, Financial Returns.

1 Introduction

Text data, as the most common tool for records and communications, plays a critical role in social science studies as a complement to traditional structured data. Since text data from media, news, and reports can reflect the attitudes of agents in the economy such as their comments, perspectives, objectives, and sentiments, it is useful to apply text data to financial studies. However, to extract accurate meaning and information from unstructured complex text data, we need to face the statistical obstacles of its high-dimensional features. A common method for this unstructured text data is to transform it into a structured index, by conducting analytical processes such as word screening, semantics learning, and "sentiment" measuring[1]. This "sentiment" measure can be used to predict asset prices or returns in equity markets, as an effective instrument for portfolio choice or asset pricing analysis (Gao et al., 2020; Sun et al., 2016). With developments of data science and modern computation power, now it is even possible to extract such information from encoded text data by statistical machine learning methodologies.

Most studies on text data follow the aforementioned steps while differing in details on how to use "machines" to "learn" text data. The common process of textual analysis ignores the sequence and structure of words and represents the document as a high-dimensional "bag-of-words" vector. This can be further extended to phrases composed by n consecutive words, thus called "n-gram words". Traditional studies typically count the number of particular words in the overlap of the document and a predefined dictionary and further scale them by the length of the document. The resulting measure is further treated as the "sentiment" and used for estimation or prediction in stock markets. Loughran and Mcdonald (2016) introduced the most widely used dictionaries in a review including dictionaries proposed in Henry (2008), Harvard's General Inquirer Word List, Diction Optimism and Pessimism Word Lists, and wide-applied list from Loughran and Mcdonald (2011). Previous studies have shown that the dictionary approach can provide a significant correlation between sentiments and stock returns[2], but it can also be a double-edged sword. Researchers can easily replicate

[1]Early in 1933, Cowles (1933) manually clustered the sentiment of *The Wall Street Journal* for analysis in the stock market.

[2]For more examples of studies using dictionary-based method, see Calomiris and Mamaysky (2019); Da

the analysis based on these public dictionaries and the computational process is simple without subjective influence. However, the results of the dictionary method rely heavily on the word-selection and word-score of each dictionary while it only uses the overlapping and sentiment-related information. Therefore, the dictionary-based methodology may not extract all the hidden features or capture biased information from text data. Hence, many recent studies tried to use machine learning methods to construct their own word lists.

As summarized in Gentzkow et al. (2019a), with the rapid development of machine learning in financial studies (Gu et al., 2020), another vein of financial textual analysis is to screen ad hoc words, either by text regression with penalty and variable selection methodology, or generative (topic) models based on the path of generating languages via machine learning algorithms. As an early pioneering study, Antweiler and Frank (2004) collected information of 45 companies on the internet and used a Naive Bayes model to predict their stock prices and returns. Jegadeesh and Wu (2013) conducted a text regression to assign weights to words based on market returns. With a similar research framework, Manela and Moreira (2017) used supported vector machines, a nonlinear penalized regression approach to screen useful words for volatility prediction in the financial market. However, these approaches treat text data only as a plain combination of words not considering the structure of language. Hence, from the perspective of the language generation process, generative topic models were proposed, mainly based on the latent Dirichlet allocation (LDA) (Blei et al., 2003). The LDA analysis not only focuses on the weight or coefficient of a single word but regards the document as the result of the generative process of a certain topic. Following this spirit, Gentzkow et al. (2019b) measured trends in the partisanship of congressional speech and Ke et al. (2019) proposed a supervised sentiment model to predict returns in stock markets. However, models in these papers still rely heavily on prior knowledge and statistical assumptions, though not using predefined dictionaries. For the initial step of word screening, most studies, ignoring the language structure, reduce the dimension based on relative returns or labels by ad hoc assumptions and prior experience. This close reliance limits the adaptiveness of the textual model, as it may only provide ad hoc results that cannot be replicated or achieve the same accuracy in other sectors or markets. Moreover, seman-

et al. (2015); García (2013); Glasserman and Mamaysky (2019); Tetlock (2007); Tetlock et al. (2008)

tic information is not the only dimension of a document (Calomiris and Mamaysky, 2019), and the holistic application to a document would provide more information on forecasting and prediction. Therefore, even previous models have demonstrated fair predictive capacity and returns in the stock market, it is still unclear how much machines can learn from this comprehensive text data.

Literature has verified that news articles are effective in return and risk predictions around the world. However, most studies are conducted under a language environment in English, in relatively developed financial markets, while very few studies focus on other languages and developing or emerging markets (Calomiris and Mamaysky, 2019). In this paper, we extended the research variety by using text data to Chinese text and the Chinese equity market. Unlike alphabet-based languages (phonograms) such as English, Chinese is a character-based language (logogram). Similar to other logograms in East Asia, Chinese uses each character to represent a word or morpheme (part of a word) and has a huge dictionary of characters. A word might be represented in one or multiple characters and form sentences without clear punctuation (no spaces between words). This can leads to great difficulties in text segmentation[3], especially so as each word or phrase can take on multiple meaning (Deng et al., 2016). As the second-largest economy in the world, the equity market in China is too big to ignore. Compared to the structure of market participants in the US, there are significantly more individual retailers rather than institutional wholesalers in China, leading to higher uncertainty and irrationality. Moreover, as a developing market, Chinese financial supervisions impose stricter restrictions to regulate tradings and stabilize financial markets, such as imposing limits on daily equity price movements and short-sales (Chen et al., 2019). It is still unclear how text data will perform in such conditions and should not be neglected

Due to all these research motivations, this paper introduces a novel Factor-Augmented Regularized Model for Prediction (*FarmPredict*) on stock returns by extracting the hidden topics (factors) from all words with consideration of structure and interactions of phrases or words. Since FarmPredict does not apply a marginal screening process on words in the initial

[3]The Chinese language is constructed by standalone Chinese characters with clear meanings on their own. The "words" in Chinese can be based on one or multiple characters. Compared to English, words in Chinese are more flexible and vocabulary can grow quickly over time. As almost every single character is meaningful on their own, a correct segmentation depends highly on the context of each sentence.

step, it is a more general analytical framework, with more potential applications, providing a highly adaptive modeling process for the study of text data.

FarmPredict consists of three steps. The first step is to learn hidden features from high-dimensional articles without supervision. To do this, we convert articles into vectors of hidden components consisting of multiple factors and idiosyncratic components via principal component analysis (PCA). The number of hidden factors is learned by the Adjusted Eigenvalue Thresholding method (Fan et al., 2020a). It is a pure unsupervised learning process without forced intervention from prior assumptions, and all information is learned from the article itself. We also explain the necessity of using unsupervised methodologies in text data as it can avoid the potential bias from subjective assumptions and limited data usage. We then screen the idiosyncratic variables by their correlations with our learning target, the corresponding beta-adjusted returns[4] in this paper, conditional on factors. This step is optional but helps us reduce dimensionality to a more manageable level. Finally, we apply a simple LASSO method to predict asset price using hidden factors and screened idiosyncratic components. FarmPredict also provides high flexibility in each analytical step.

Our study gathers financial news from *Sina Finance* in China, which is one of the major news hubs for Chinese equity markets. The website publishes over 500 news daily and offers timely and comprehensive coverage of all the popular financial news in Chinese. We used crawling to download publicly available news webpages from its website and extracted related time, text, and stock information for our data. The text is segmented with a hidden Markov model and paired with returns with corresponding code and time. Each article is paired with its effective beta-adjusted returns for model training. We fitted FarmPredict on a dichotomized bag-of-words vector of our data and evaluated the model's estimated sentiment scores and corresponding returns from 2015 to 2019.

We then validated the sentiment scores from FarmPredict via multiple approaches. First, we examined the meanings of major sentiment-charged words selected by our model. By comparing to the words from the ad hoc topic model, we demonstrated that FarmPredict was able to capture more interactive information which can be neglected by marginal screening.

[4]Beta-adjusted return for stock i on day t is defined as r^*_{it} = Raw Return$(r_{it})-\beta_i\cdot$Market Return(r_t^{market}), where β_i describes the linear relationship between market risk and individual asset returns.

The panel regression also illustrated that FarmPredict can learn specific information about target stocks, resulting in a significant correlation with the beta-adjusted returns of targeted stocks. We also treated the news in this paper as "events" and estimated the pattern of stock returns based on an event study. It revealed the potential mechanism of how unexpected news happens and how they can affect the financial markets in China. The results showed that about 7 days before the occurrence of positive news, the beta-adjusted returns already started to increase, while no such result was observed for negative news. This asymmetric effect of impact aligns well with the short-sale constraints and supervisions in Chinese equity market, which make the leak or anticipation of negative news harder to react to (Chen et al., 2019; Nagel, 2005). After impact peaking on the news-arrival day, with an average of 83 bps on positive news sentiments and 26 bps on the negative ones, the (positive/negative) impact of news arrivals would last for further a few days. A placebo test lends further support to this result, thus this leads to investment opportunities.

We also tested our machine-learned sentiment scores in terms of financial investments. We built daily portfolios based on predicted sentiment scores and recorded their returns. The portfolio showed robust and positive returns, as the annualized percent return (APR) reached 116% (Sharpe ratio: 9.37) for equally-weighted portfolio and 48% (Sharpe ratio: 3.34) for value-weighted one[5] in the test period of 2015-2019, significantly exceeding other models. The results also verified the effect of news and momentum in Chinese equity market. We further analyzed the portfolio's risk and return from alpha (beta-adjusted return) or different components. The alpha APR stays as high as 115% with a Sharpe Ratio of 9.37. Realistic details regarding Chinese equity specifics like transaction costs and daily price limits are also considered in the tests. To further verify the robustness of FarmPredict, we tested the model's sensitivity in terms of various transformations of input and output, choice of factors, number of stocks in constructed portfolios, and the amount of news inputs. The results stay stable, thus demonstrating the robustness of FarmPredict.

Our model has important implications for understanding how much financial information

[5]Each day, the portfolio longs 50 stocks with highest predicted sentiment scores and shorts 50 of those with lowest scores. Weights of stocks in the value-weighted portfolio are proportional to their total market capitalization.

machines can learn from text data, as well as the return prediction and realization by text-based sentiment studied by a rich set of papers. First, our FarmPredict starts with an unsupervised factor extraction of all words from each article. Instead of a supervised process on word screening, we do not rely on any prior assumptions or experiences but only conducted a data-driven textual dimension reduction. This choice provides a significant benefit to text modeling: let machines learn the meaning of the key components of text without supervision by human experience. The sole data-driven process also leads to high flexibility, suitability, and robustness of our model on text data analysis since hidden factors and features can be revealed by machine learning without any intervention from subjective or prior knowledge.

Second, the FarmPredict we demonstrated is not only a model but an analytical framework of machine learning for high-dimensional data, which is text data in this paper. By transforming the original data into the latent factors and idiosyncratic components, FarmPredict effectively converts high-dimensional data with highly correlated covariates into weakly correlated ones in an unsupervised way. Hence, FarmPredict could solve the statistical obstacle of multi-collinearity and simultaneously extract information from the whole data. The subsequent marginal screening performs an efficient dimensional reduction and selects the most related and predictive words. It is worth noting that the screening process in FarmPredict is conditional on hidden factors being learned from all elements (words) in the data, resulting in the use of all information without supervision. Thanks to all these features, the framework of FarmPredict is very flexible in learning factors and idiosyncratic components, methods for screening, and the selection of linear or nonlinear models for prediction.

Finally, this paper showed the possibility of applications of machine learning techniques such as our FarmPredict and topic model in languages other than English and developing markets. Simply by longing the high-score stocks and shorting the low-score ones, our portfolio-building strategy based on machine-learned sentiment scores can achieve significantly outperformed returns. By comparing the returns before and after news occurs, this paper also provides information transmission particularity in the financial sector in China. Our research on Chinese equity markets would complete the vein of literature on textual analysis, expand the depth of statistical machine learning techniques in financial studies, and shed light on the rich application of machine learning to social science topics.

The remainder of the paper is organized as follows. Section 2 introduces the FarmPredict method and ad hoc topic model. Section 3 describes our data and the detailed analysis process. Section 4 provides empirical results to validate our model and tests the sensitivity and robustness. Section 5 concludes this paper.

2 Methods

This section discusses the framework of using machines to learn text data. We first summarize the framework and notations shared across different models and then introduce a novel regression method (FarmPredict) using factor augmentation. Variations of the Farm-Predict framework then follows. We also briefly introduce the (ad hoc) topic model and its extension for comparisons.

2.1 Problem Setup

We use the word level statistics as a summary of each of the n articles (bag of words). Let $\mathbf{D}$ be the set of all possible Chinese words in our data of n articles and $\mathbf{d}_i \in \mathbb{N}^{|\mathbf{D}|}$ be the vector of word counts of every word in the i-th article, with $d_{i,k}$ being the number of times the k-th word appears in the article. Each article composes of several underlying topics where each topic has its own preferred vocabulary. Therefore, we assume that an article's word count $\mathbf{d}_i$ is influenced by a small number of latent factors or topics. The factors or topics can be as simple as positive versus negative, or more complex factors that contains aspects like the article's attitude, related industry sector, author's own word preference, etc.

Article i is associated with a target outcome or response Y_i. In this paper, Y_i will be the beta-adjusted return of the associated stock in the article i on the day when the news is published. The target responses $\{Y_i\}$ are mainly affected by a relatively small subset of words. We call this set of words sentiment-charged words. This assumption also helps reduce dimensionality to a reasonable level. Bag of words data is very high-dimensional and appears sparsely in each article, especially in Chinese. In our dataset of 914K articles, there

are $1,181$K distinctive words[6] in the entire set $\mathbf{D}$, while only 71K words appear in at least 50 articles in the data.

All the words are divided into two disjoint categories: the set of sentiment-charged words $\mathbf{S}$ and the set of sentiment-neutral words $\mathbf{N}$ so that $\mathbf{D} = \mathbf{S} \cup \mathbf{N}$. The sentiment score of an article is mainly associated with its sentiment-charged words. $\mathbf{d}_{i,\mathbf{S}}$ denotes the part of word counts $\mathbf{d}_i$ restricted to $\mathbf{S}$.

2.2 FarmPredict

In most traditional textual analysis, like topic models or dictionary based methods, there are a number of restrictions on the model, resulting in inflexibility and a possible inaccurate estimation of sentiments. A natural question is then if we can learn the sentiments directly from high-dimensional regression, as sentiment prediction in finance is fundamentally a regression problem. Here we propose a direct regression framework, called Factor-Augmented Regularized Model for Prediction (FarmPredict).

Selecting frequently-used words. In over 1.1 million distinct words (and phrases) in our data collection, most of them rarely occurs. Their semantics are difficult to learn by machines. As such, we begin by filtering out these infrequent words that only appear in a small fraction of articles. These words are also hardly useful as they are unlikely to appear in new articles to be scored. The screening also helps us narrow our focus to a reasonably comprehensive set of words $\mathbf{D}^{\text{freq}}$, around $10,000$ or so.

Let k_j be the number of articles that contain word j. For a threshold κ, we keep the vocabulary

$$\mathbf{D}^{\text{freq}} = \{j\text{-th word in } \mathbf{D} : k_j \geq \kappa\}. \tag{2.1}$$

The threshold κ will be tuned as hyper-parameter to strike a balance between the comprehensiveness of $\mathbf{D}^{\text{freq}}$ and the noises introduced by infrequent words.

[6]Here we refer collectively both words and phrases as words for simplicity. The median length of articles has 309 words, 209 of them distinctive.

Factor modeling. Let $\mathbf{X}_i$ be the feature vector in which $X_{i,j}$ is the feature of word $j \in \mathbf{D}^{\text{freq}}$ in the i-th article. It can be the original word counts or simply $\{0, 1\}$ indicating the absence or presence of the word j in the i-th article. The dependence among words is assumed to be driven by some latent factors. Namely, $\mathbf{X}_i$ follows an approximate factor model

$$\mathbf{X}_i = \mathbf{B}\mathbf{f}_i + \mathbf{u}_i, \qquad i = 1, \cdots, n, \tag{2.2}$$

where $\mathbf{f}_i \in \mathbb{R}^k$ is the vector of k latent factors, $\mathbf{B}$ is the factor loading matrix, and $\mathbf{u}_i \in \mathbb{R}^{|\mathbf{D}^{\text{freq}}|}$ is a vector of idiosyncratic components that can not be explained by (uncorrelated with) $\mathbf{f}_i$. Putting the factor model in the matrix form, we have

$$\mathbf{X} = \mathbf{F}\mathbf{B}^T + \mathbf{U}$$

where $\mathbf{X}$ and $\mathbf{U}$ are $n \times |\mathbf{D}^{\text{freq}}|$ matrices of data and idiosyncratic components and $\mathbf{F}$ is $n \times k$ of latent factors. Here, only $\mathbf{X}$ is observable and $\mathbf{F}, \mathbf{B}, \mathbf{U}$ will be estimated by principal component analysis.

The factors can be understood similarly to topic scores and the factor loading $\mathbf{B}$ gives a different mix to these factors (topics). For example, short market briefings and equity research articles might each have their own distinct vocabularies, and thus are influenced by different factors and their loadings.

The factor model disentangles correlated features in $\mathbf{X}_i$ by decomposing them into factors $\mathbf{f}_i$ and idiosyncratic components $\mathbf{u}_i$. Suppose that we would like to use $\mathbf{X}_i$ to predict the associated return outcome Y_i. Following a similar idea in Fan et al. (2020b), we use latent $\mathbf{f}_i$ and $\mathbf{u}_i$ as predictor and build the model

$$Y_i = a + \mathbf{b}^T\mathbf{f}_i + \boldsymbol{\beta}^T\mathbf{u}_i + \epsilon_i, \tag{2.3}$$

where ϵ_i is the idiosyncratic noise. This model is broader than linear model in $\mathbf{X}_i$ and the variables in (2.3) are less correlated. We will additionally impose a sparsity constraint on $\boldsymbol{\beta}$, as most words do not carry signals on an article's sentiments or stock returns.

Learning factors and idiosyncratic components. For a given number of factors k, we fit the approximate factor model (2.2) via least-squares, resulting in principal component

analysis. The solution is that estimated latent factor $\widehat{\mathbf{F}} = \sqrt{n}$ times the eigenvectors of the largest k eigenvalues of matrix $\mathbf{X}\mathbf{X}^T$, $\widehat{\mathbf{B}} = \mathbf{X}^T\widehat{\mathbf{F}}/n$, and $\widehat{\mathbf{U}} = \mathbf{X} - \widehat{\mathbf{F}}\widehat{\mathbf{B}}^T$. See Bai and Ng (2002); Fan et al. (2020c); Stock and Watson (2002).

There are a number of data-driven methods for selecting the number of factors k. See Fan et al. (2020c) and references therein. Here, we use the adjusted eigenvalue thresholding (Fan et al., 2020a). The method takes into account the heterogeneous scales of observed variables and estimates the number of factors via thresholding on bias-corrected estimators of eigenvalues of correlation matrix. Specifically, k is estimated as the number of corrected values that are statistically larger than one:

$$\widehat{k} = \max\{j < |\mathbf{D}^{\text{freq}}| : \widehat{\lambda}_j^C > 1 + C\sqrt{|\mathbf{D}^{\text{freq}}|/(n-1)}\}, \tag{2.4}$$

where $\widehat{\lambda}_j^C$ be the bias-corrected estimator of the j-th largest eigenvalue of the correlation matrix of the data matrix $\mathbf{X}$. [7]

Learning conditional sentiment-charged words S. With learned factors in place, we can further screen down the predictive words (sentiment-charged words) using conditional correlation screening. Let $\widehat{\mathbf{Y}}_u$ be the residual vector of $\mathbf{Y}$ after fitting a linear regression of $\mathbf{Y}$ on $\widehat{\mathbf{F}}$ with intercepts. This takes out the part of $\mathbf{Y}$ that can be explained by the factors. We seek components of $\widehat{\mathbf{U}}$ to further predict $\widehat{\mathbf{Y}}_u$.

Conditional screening is to seek words that have high correlation with $\mathbf{Y}_u$ (Fan and Lv, 2008), more precisely, the correlation between $\widehat{\mathbf{Y}}_u$ and the idiosyncratic component $\widehat{\mathbf{U}}_j$ for word j, which is the j^{th} column of $\widehat{\mathbf{U}}$. This correlation is the partial correlation between $\mathbf{Y}$ and the feature vector associated with word j, conditioning on the latent factors $\mathbf{F}$. Given

[7] Fan et al. (2020c) suggests to take $C = 1$, but this is too small for our application. It is well known that largest eigenvalues are biased upwards. The correction is as follows (Bai and Ding, 2012): Let $\widehat{\lambda}_j$ be empirical eigenvalues and $p = |\mathbf{D}^{\text{freq}}|$ be the dimension. For a given j , define

$$\begin{aligned} m_{n,j}(z) &= (p-j)^{-1}[\sum_{\ell=j+1}^{p}(\widehat{\lambda}_\ell - z)^{-1} + ((3\widehat{\lambda}_j + \widehat{\lambda}_{j+1})/4 - z)^{-1}], \\ \underline{m}_{n,j}(z) &= -(1-\rho_{j,n-1})z^{-1} + \rho_{j,n-1}m_{n,j}(z), \end{aligned}$$

with $\rho_{j,n-1} = (p-j)/(n-1)$. The corrected eigenvalue of $\widehat{\lambda}_j$ is defined as $\widehat{\lambda}_j^C = -\frac{1}{\underline{m}_{n,j}(\widehat{\lambda}_j)}$. In our application, since n is much larger than p, this step of correction is very small and can be ignored.

a threshold α, the conditional sentiment-charged words are defined by

$$\widehat{\mathbf{S}} = \left\{ j : |\text{corr}(\widehat{\mathbf{U}}_j, \mathbf{Y}_u)| > \alpha \right\} \cap \{ j : k_j \geq \kappa \} \tag{2.5}$$

The threshold α will be tuned to select around 1000 words. This step is optional (corresponding to $\alpha = 0$), but helps us speed up computation.

FarmPredict fitting. With every estimated variables in place, we can train our regression model. Among the conditional sentiment-charged words, FarmPredict solves the penalized least squares:

$$\widehat{a}, \widehat{\mathbf{b}}, \widehat{\boldsymbol{\beta}} = \text{argmin}_{a,\mathbf{b},\boldsymbol{\beta}} \left\{ \frac{1}{n} \sum_i \left(Y_i - a - \mathbf{b}^T \mathbf{f}_i - \boldsymbol{\beta}^T \mathbf{u}_{i,\widehat{\mathbf{S}}} \right)^2 + \lambda \|\boldsymbol{\beta}\|_1 \right\}, \tag{2.6}$$

where $\mathbf{u}_{i,\widehat{\mathbf{S}}}$ is the components of $\mathbf{u}_i$ restricted to the sentiment-charged words $\widehat{\mathbf{S}}$. The penalty λ, which will be chosen by the cross-validation, controls the models' bias-variance trade-off and also the sparsity of $\widehat{\boldsymbol{\beta}}$. This reduces further the sentiment-charged words.

The Lasso penalty in (2.6) can also be changed to other functions such as SCAD and elastic net, among others (Fan et al., 2020c; Nagel, 2021).

Scoring new articles. Scoring a new article consists of two steps. For a given new feature vector $\mathbf{X}_{\text{new}}$, let us decompose it into factors and idiosyncratic components. With given $\widehat{\mathbf{B}}$, applying the least-squares to model (2.2), we obtain the latent factor $\mathbf{f}_{\text{new}}$ as well as the idiosyncratic component $\mathbf{u}_{\text{new}}$ associated with the feature $\mathbf{X}_{\text{new}}$ as follows: [8]

$$\mathbf{f}_{\text{new}} = (\widehat{\mathbf{B}}^T \widehat{\mathbf{B}})^{-1} \widehat{\mathbf{B}}^T \mathbf{X}_{\text{new}}, \qquad \mathbf{u}_{\text{new}} = \mathbf{X}_{\text{new}} - \widehat{\mathbf{B}} \widehat{\mathbf{f}}_{\text{new}}. \tag{2.7}$$

Therefore, its sentiment score is predicted as

$$\widehat{Y}_{\text{new}} = \widehat{a} + \widehat{\mathbf{b}}^T \mathbf{f}_{\text{new}} + \widehat{\boldsymbol{\beta}}^T \mathbf{u}_{\text{new},\widehat{\mathbf{S}}}. \tag{2.8}$$

[8] The computation can be done expidiously, since $\widehat{\mathbf{B}}^T \widehat{\mathbf{B}}$ is a diagonal matrix, with the diagonal elements being the k largest eigenvalues of the matrix $\mathbf{X}\mathbf{X}^T/n$.

2.3 Variations on FarmPredict

FarmPredict is highly adaptive and flexible in the context of financial text analysis. This makes our approaches more versatile and adaptive to different tasks. First of all, the response variable Y can be excess returns or dichotomized returns (positive or negative). In the latter case, one can use penalized least-squares as in (2.6) or penalized logistic-regression, similar to (2.14) below. In the case of applying logistic-regression technique, conditional screening (2.5) and conditional prediction (2.7) can also be modified to accommodate the logistic-regression model; see Fan et al. (2020c).

Secondly, the feature vector can be the original counts or their modified version such as the dichotomized ones (absence and presence). In the latter case, an alternative extraction of latent factors can also be obtained from the original counts and the factor loadings on the dichotomized features can be learned from least-squares or logistic regression.

Thirdly, sentiment-charged words can be obtained from other approaches such as marginal screening to be introduced in the next subsection. They can also be augmented by these marginally screened words.

Finally, the linear prediction model (2.3) can be replaced by nonlinear models

$$Y_i = g\left(\mathbf{f}_i,\ \mathbf{u}_{i,\mathbf{S}}\right) + \epsilon_i$$

such as neural network models (Horel and Giesecke, 2020) or structured nonparametric models (Fan et al., 2020c).

In summary, FarmPredict is designed in a highly customizable way to allow many ad hoc modifications on inputs, words screening, and techniques for fitting regression functions, etc.

2.4 Ad Hoc Topic Model

SESTM, introduced by Ke et al. (2019), is an ad hoc two-topics model for learning sentiments of new articles based on stock returns. It assumes that each article is a mixture of just two topics – positive and negative, and uses the mixture probability p_i to indicate the

positive sentiment on the i-th article. Thus, p_i represents the degree of positive sentiment of the i-th article, with 1 being most positive and 0 most negative. Naturally, p_i is expected to be positively associated with return Y_i.

Assume sentiment-neutral vocabulary $\mathbf{N}$ is independent of either score p_i or return Y_i given the sentiment-charged words $\mathbf{S}$ so that we can focus on $\mathbf{S}$. Let s_i be the number of sentiment charged words in article i. We assume the word counts $\mathbf{d}_{i,\mathbf{S}}$ follows a multinomial distribution:

$$\mathbf{d}_{i,\mathbf{S}} \sim \text{Multinomial}\left(s_i,\ p_i\boldsymbol{\theta}_+ + (1-p_i)\boldsymbol{\theta}_-\right), \tag{2.9}$$

where $\boldsymbol{\theta}_+$ and $\boldsymbol{\theta}_-$ are two parameters vectors of dimension $|\mathbf{S}|$, inditing the probabilities of occurrences of sentiment-charged words $\mathbf{S}$ in a purely positive or negative article.

Learning sentiments from a set of training data $\{\mathbf{d}_i, Y_i\}_{i=1}^n$ consists of two main steps: learning the sentiment charged vocabulary $\mathbf{S}$ and learning semantics of these words $\boldsymbol{\theta}_+$ and $\boldsymbol{\theta}_-$. The former uses the sure (marginal) screening techniques in Fan and Lv (2008) and the latter uses supervised learning with the assistance of the percentile ranking of the return Y_i in the training set. Once the sentiment charged words and their semantics are learned, a new article's sentiment score p_i can be estimated using the maximum likelihood estimator (MLE) based on model (2.9).

Learn sentiment charged words. A word is selected in $\mathbf{S}$ based on two conditions. First, it needs to appear frequently enough. Second, the word needs to correlate enough with Y_i, which is measured by its marginal correlation (Fan and Lv, 2008) with the sign of returns. This correlation in the current context is [9]

$$f_j = \frac{\#\text{ articles with word } j \text{ AND return} > 0}{\#\text{ articles with word } j}.$$

We now select the sentiment changed words as

$$\widehat{\mathbf{S}}^{\text{Screen}} = \{j : f_j \geq 0.5 + \alpha_+ \text{ or } f_j \leq 0.5 - \alpha_-\} \cap \{j : k_j \geq \kappa\}. \tag{2.10}$$

[9] This computes the proportion of word j associated with positive returns when the word j appears in an article. Since returns are either positive or negative, rarely exactly 0, $1 - f_j$ is the proportion of the word j associated with the negative returns. Therefore, only one of these two numbers is informative.

Thresholds α_+ and α_- are hyper-parameters chosen a priori at tuning. In SESTM, the sentiment changed words are taken as $\widehat{\mathbf{S}}^{\text{Topic}} = \widehat{\mathbf{S}}^{\text{Screen}}$. In the next subsection, we will augment this with a few other proposals.

Learn semantics of words. Let $\mathbf{P} \in \mathbb{R}^{n\times 2}$ with the i-th row $(p_i, 1-p_i)$ and $\mathbf{\Theta} = (\boldsymbol{\theta}_+, \boldsymbol{\theta}_-)$ be a $|\mathbf{S}| \times 2$ matrix. Set $\mathbf{D_S}$ an $n \times |\mathbf{S}|$ matrix, whose i-th row is the proportions of sentiment charged words in the i-th article. Then from the multinomial assumption, we have in expectation $\mathbb{E}\mathbf{D_S} = \mathbf{P}\mathbf{\Theta}^T$. This implies that $\mathbf{D_S}$ can be approximately represented as the product of two rank-2 matrices. Given $\mathbf{P}$, $\mathbf{\Theta}$ can be estimated by the least-squares regression as

$$\widehat{\mathbf{\Theta}} = \left(\mathbf{P}^T\mathbf{P}\right)^{-1}\mathbf{P}^T\mathbf{D_S}. \tag{2.11}$$

Given $\mathbf{\Theta}$, $\mathbf{P}$ can be similarly obtained by the least-square regression. Iterating this way with proper identifiability constraints leads to a solution to the problem with the best rank-2 approximation.

Technically, $\mathbf{\Theta}$ and $\mathbf{P}$ can be learnt from the data matrix $\mathbf{D_S}$ via best rank-2 approximation and identifiability conditions. In this ad hoc topic model, SESTM uses information Y to guide the learning of $\mathbf{\Theta}$ via (2.11). For each article i, assign the value of p_i as the normalized rank

$$\widehat{p}_i = \left(\text{rank of } Y_i \text{ in } \{Y_j\}_{j=1}^n\right)/n. \tag{2.12}$$

This estimate is intuitively reasonable but such assignment may contains many errors.[10] Replacing p_i with $\widehat{p}_i$, the semantics are estimated as $\widehat{\mathbf{\Theta}} = (\widehat{\mathbf{P}}^T\widehat{\mathbf{P}})^{-1}\widehat{\mathbf{P}}^T\mathbf{D}_{\widehat{\mathbf{S}}^{\text{Topic}}}$.

Score news articles. With estimates $\widehat{\mathbf{S}}^{\text{Topic}}$ defined by (2.10), $\widehat{\boldsymbol{\theta}}_+$ and $\widehat{\boldsymbol{\theta}}_-$, we are ready to assign sentiments to new articles. For a new article with word counts $\mathbf{d}^{\text{new}}$, its sentiment score $\widehat{p}^{\text{new}}$ is estimated by penalized maximum likelihood (PMLE). SESTM uses

$$\widehat{p}^{\text{new}} = \text{argmax}_p \sum_{j\in\widehat{\mathbf{S}}^{\text{Topic}}} \log\left(p\widehat{\theta}_{j+} + (1-p)\widehat{\theta}_{j-}\right)^{d_j^{\text{new}}} + \lambda_{\text{PMLE}}\log\left(p(1-p)\right), \tag{2.13}$$

[10]The sentiment assignments depend too much on the random outcomes of the trading results. One article has higher daily return does not necessarily imply that its associated article has a higher sentiment. In addition, since the returns are compared across multiple years, all market risk factors influence the returns and hence are used to assign the sentiment of an article, which is not reasonable.

for a given tuning parameter λ_{PMLE}.

2.5 Comparing FarmSelect with SESTM

Both FarmSelect and SESTM use the word features $\mathbf{X}_i$ and its associated outcome Y_i to learn the sentiment scores. FarmSelect takes into considerations of the dependence and interactions among words in sentiment assignments, whereas SESTM predominantly uses individual words to come up with sentiment scores. When selecting sentiment-changed words, FarmSelect begins with a comprehensive set of vocabulary, choosing a subset of words to best predict the outcome via (2.6). In contrast, SESTM relies on marginal screening (2.10) to select, which ignores the interactions among words. This step can be improved by using penalized logistic regression (2.14) at the expense of a higher computation cost, which makes it more similar to FarmSelect.

Let $\widetilde{Y}_i \in \{0, 1\}$ indicate whether the return is negative or positive and $\widetilde{\mathbf{X}}_i$ be the vector of binary features, indicating whether a word is in article i. We only restrict to the words that appear at least in κ articles and then fit the penalized logistic regression:

$$\min_{\mathbf{w},c} \sum_i \left[\widetilde{Y}_i(\widetilde{\mathbf{X}}_i^T\mathbf{w} + c) - \log\left(1 + \exp(\widetilde{\mathbf{X}}_i^T\mathbf{w} + c)\right) \right] + \lambda_{\text{Logistic}}\|\mathbf{w}\|_1. \tag{2.14}$$

Sentiment-charged words can be chosen as the words corresponding to non-zero entries in $\mathbf{w}$. Denote the resulting set of words as $\widehat{\mathbf{S}}^{\text{Logistic}}$. The penalty $\lambda_{\text{Logistic}}$ can be chosen along the lasso path to control the number of words selected. This leads to three possibilities for choosing sentiment-charged words $\widehat{\mathbf{S}}^{\text{Topic}}$ for topic modeling:

1. Words selected by marginal correlation screening $\widehat{\mathbf{S}}^{\text{Screen}}$,
2. Words selected by penalized logistic regression $\widehat{\mathbf{S}}^{\text{Logistic}}$,
3. Union of words selected by both methods $\widehat{\mathbf{S}}^{\text{Screen}} \cup \widehat{\mathbf{S}}^{\text{Logistic}}$.

In the empirical study, we will test each of them.

Both FarmPredict and SESTM rely on some model assumptions. FarmPredict intends to find a set of factors and sentiment charged words to directly predict the outcome. Yet,

SESTM relies critically two models (2.11) and (2.13) and sentiment score assignments (2.12). Therefore, it is not as robust to the model assumption as FarmPredict. As noted in (2.11), while the estimation of $\mathbf{P}$ there is intuitively reasonable, there are also several problems. That is another reason behind our development of the new model.

3 Data and Analysis

3.1 Data Collection

We used the news data downloaded from *Sina Finance* website. *Sina Finance* is one of the largest Chinese Financial news websites. It publishes over a thousand Chinese-stock related news every day and covers all stocks in the market. News articles are downloaded by crawling through the website. Crawling is a widely used strategy by programmers to systematically and efficiently download webpages.

The downloading process can be viewed as searching on a net. We started from the root of the net (main page). Nodes (webpage) in the net are connected if one has a link that points to the other and we visit them sequentially by its distance to the root. Technically, we scrawled in a breadth-first fashion. Starting with the main page of *Sina Finance* and *Sina Caijing*, we downloaded the html file of the webpage and saved it to disk, then analyzed the html contents to get all links to other webpages, screened the links to only keep the ones inside domain *finance.sina.com.cn* and domain *cj.sina.com.cn*, and finally, pushed the obtained links to a queue to visit later. We iteratively looped through this process for each link in the queue and let our crawler program run for several months from the end of 2019 through 2020.

After all of this, our crawler visited 6.3 million unique website links, among which 5.8 million are valid news articles. Due to the net-like search structure of our crawling, the number of news articles we downloaded is a little random and not exactly uniform across years.

For each webpage downloaded, the publish time and title are extracted from corresponding html headers. The main articles are extracted from corresponding html sections with ID

as 'article'. For webpages without an ID, we analyzed their html structure and applied case specific article extractors using combination of html structures and regex expressions.

3.2 Preprocessing

We went through a series of data preprocessing to clean, select and prepare downloaded data for model fitting. First, we removed duplicated and very similar articles. If two articles have the same title after removing special characters and are published in the same day, then only the first one will be kept in our dataset. The remaining articles are then cleaned as follows. First, all contents and titles are trimmed to Chinese characters only so all the html digits, punctuation marks, special characters and remaining html codes are stripped away.

Then the articles are matched with stocks. We used a combination of html and article content to find the matching stocks. We searched for the website's special stock specifier ID by regex on entire html file to see if the page is tagged with some stocks officially by Sina. For pages without such a tag, we scan the article title and content matching for stock names and symbols. We removed articles attached to zero or more than one stock.

Each remaining article is then matched with the return of its associated stock. We used beta-adjusted returns, which are calculated as the stock's own returns minus its market-induced returns as follows:

$$\text{Beta-adjusted Return}_{it} = \text{Dividend Adjusted Return}_{it} - \beta_i \cdot \text{SSEC Return}_t,$$

in which the beta (β_i) of stock i is calculated by regressing its own daily return on the daily returns of Shanghai Stock Composite Index (SSEC, market return[11]) using data from 2005 to 2014.

There are several options on what kind of return and what time range of return should be used. We used the time range of *effective return* which reflects the news' impact on the stock, which covers the article's publish time. The time range is chosen carefully so it can reflect the immediate price impact of the article. We used the close-to-close return covering the article's publish time. For example, if an article is published at 1pm inside trading hours

[11]SSEC is a market-value-weighted index of all stocks in Shanghai Stock Exchange.

on Tuesday, then the return from Monday market closest to Tuesday market close is used. If an article is published at 6pm after market closest on Friday, then return from current Friday market close to next Monday market close will be used. Dividend payments and stock splits are also merged into returns to correctly reflect the stocks' actual value changes. Some stocks might not be matched with a valid return on certain time for reasons like trading halt, etc. We dropped the articles without a matching return.

Table 1: Number of Sina Finance articles after each stage of preprocessing.

Sina Finance articles	# articles
all htmls downloaded	6,343,491
removed non-articles	5,880,943
removed very similar articles	4,195,741
removed missing date/time	4,195,726
matched with at least one stock	2,465,127
matched with exactly one stock	1,985,781
matched with an effective return	1,791,364
downsampled ($\leq$ 300/d)	914,070

Finally, we used Jieba[12] (Sun, 2017) to divide an article's title and main text to lists of words (and phrases). Jieba uses a Hidden Markov Model based method to divide words. The method works on single Chinese character level and label each character as one of the four states: B(begin), M(middle), E(end) and S(single). With their existing emission and transition probability on every state and every single Chinese characters, Viterbi algorithm is used to find the most likely sequence of hidden states. Then the text can be divided into words and phrases using the estimated hidden states. We chose the algorithm for its ability to deal with unknown phrases and fast speed (linear time complexity with respect to number of characters). The number of article after each operations is listed in Table 1.

In the final step, we downsampled our training data to balance the number of samples each day. Due to our crawling strategy, the number of articles downloaded from each day is unbalanced. There are over 700K articles downloaded in 2019 while only 10K in 2012.

[12]Jieba is an open source Python package for Chinese word segmentation. It is available on Github at github.com/fxsjy/jieba/commit/cb0de2973b2fafaa67a0245a14206d8be70db515

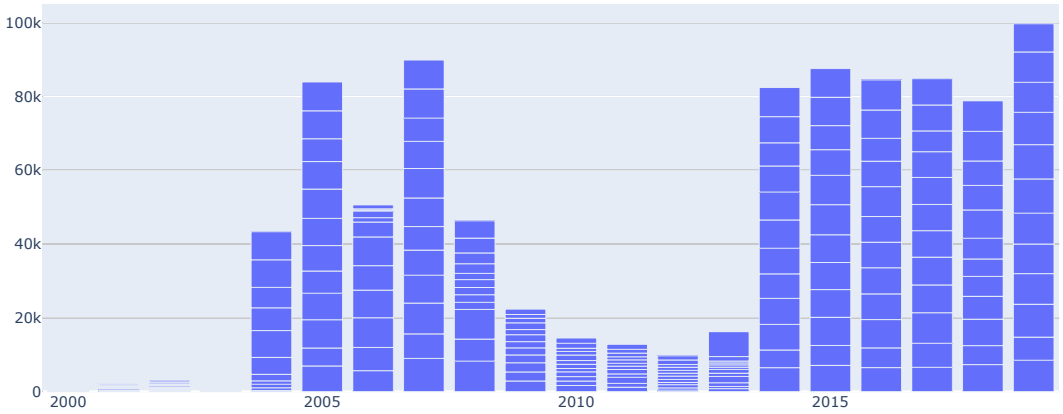


Figure 1: Number of data for each year in our final dataset. The dataset was downsampled so that each day has at most 300 articles. The thin white lines inside each year's bar divides data by months. Data from 2000 to 2014 are used for tuning and training and only data from 2015 - 2019 are used for testing.

To balance the amount of data, as well as to lower computational burden, we randomly down-sampled the data to at most 300 articles each day. The amount of data is reduced to 914K in total and much more evenly distributed among the days[13]. The number of data each year after downsampling is plotted in Figure 1. The majority of removed data are from second half year of 2019, which is never used in training.

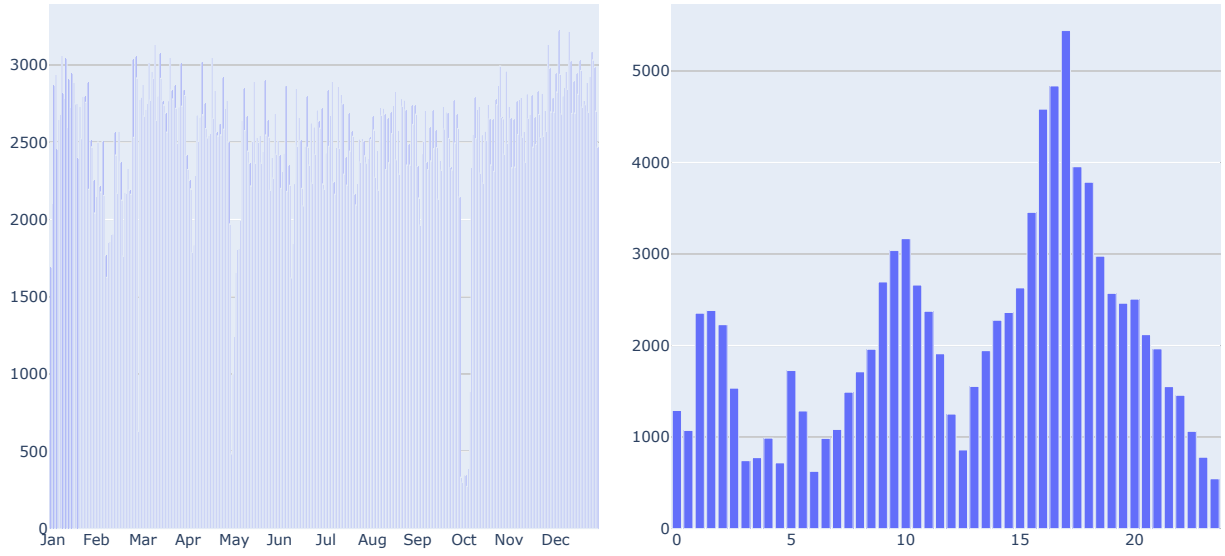
3.3 Basic Statistics

In our dataset of 914K articles, there are 1181K words in the entire set **D**, out of which 71K words appear in at least 50 articles (0.004% of all articles). In all models, we began with these 71K words and their corresponding word counts in each article. The word count matrix is highly sparse with each article having in median 309 words and 209 distinctive words. So a median article has only 0.29% non-zero entries, among 71K dimensional word-count vectors.

We presented the amount of articles collected for every day of year in Figure 2a. The number of data is roughly and evenly distributed across each day except a couple of holidays.

[13]We also checked the results of using the full data without downsampling. They do not change very much and do not alter our conclusion. See sensitivity test in section 4.3.4.

There is less data in February and the first weeks of May and October, which corresponds to the three largest holidays in China. The dates for Chinese Spring festivals are based on Traditional Chinese Calendar and can happen from late January to late February. Labor day golden week and National day golden week take places at the first days of May and October, respectively, and each lasts for a whole week. The number of data aggregated along each half-hour window of a day is also plotted in Figure 2b. Most news are published from market open time around 9am to end of day. There are also some news published after mid-nights that are mostly auto-generated news or overseas news.



(a) #data by day of year

(b) #data by time of day in 2019

Figure 2: Number of data distributed on each day of year from 2000 to 2019 (a) and number of data by time of day in 2019, divided into 30-minute intervals. Data are largely evenly distributed across days except for the three major holidays in China. Most news are published around market open and market close.

More details on the datasets are presented in Table 2. We report the word counts and associated returns of every single news article. In addition, we grouped data from the nearest five years (2015-2019) by the date associated with their effective returns. The number of articles, number of distinct stocks covered and SSEC returns of each group are reported. We also reported the percentage of news that are associated with a positive return for every group.

Table 2: Summary statistics of collected data

Data	basis	#data	mean	std	skewness	kurtosis	10%	25%	50%	75%	90%
# words	all	914,070	680	1077	6.5	120.6	77	152	376	781	1440
# distinct words	all	(articles)	278	255	2.5	12.6	54	99	209	373	578
returns	all		0.4%	5.3%	68.3	9903.5	-3.3%	-0.9%	0.0%	1.5%	4.7%
beta-adjusted returns	all		0.3%	5.1%	75.7	11365.0	-2.9%	-1.1%	0.0%	1.4%	4.2%
# articles	daily 2015-19	1,220	356	139	2.2	5.5	268	289	308	349	549
# distinct stocks	daily 2015-19	(days)	250	81	1.7	3.4	184	206	231	261	370
% positive returns	daily 2015-19		47%	19%	0.1	-0.5	23%	34%	46%	61%	73%
SSEC returns	daily 2015-19		0.0%	1.5%	-1.0	6.4	-1.4%	-0.5%	0.1%	0.6%	1.6%

Note: We summarize our data on two bases. The 'all' basis looks at the entire dataset, views each article as a data point. The summary statistics of each article's number of words, number of distinct words, associated effective raw returns and beta-adjusted returns are displayed. On the 'daily 15-19' level, we group and summarize articles from 2015 to 2019 by their publish date. For each day, we calculate its number of articles, number of reported distinctive stocks, SSEC return and the proportion of articles associated with an positive return in that day.

3.4 Tuning and Testing

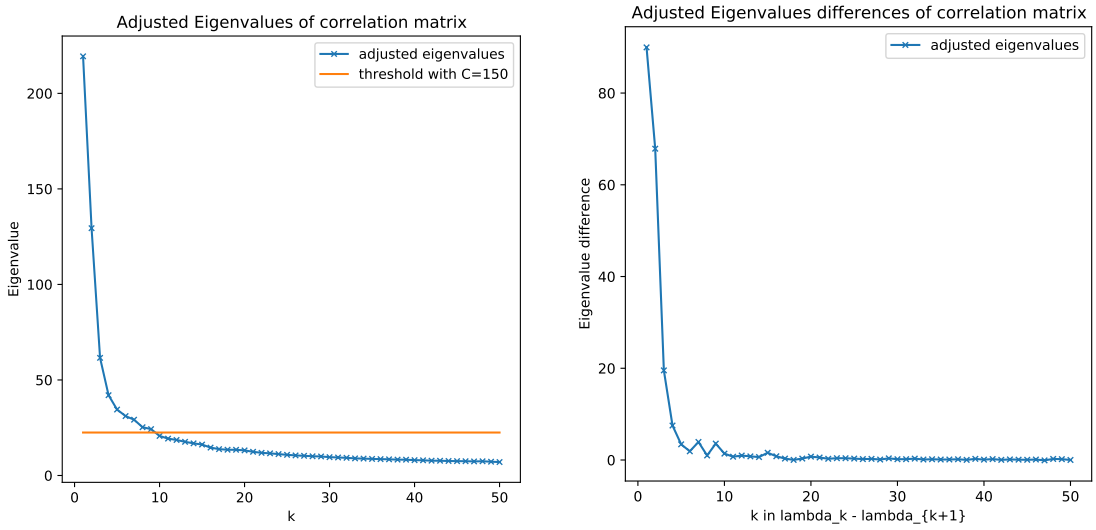
Tuning. Tuning are conducted with data from 2000 to 2014. More specifically, we used data from 2000 to 2010 as the training set and the 2011 to 2014 as the validation set for selecting tuning parameters. For every model and every combination of hyper-parameters, the model is fitted with training set and then used for prediction in the validation set. Then an equally-weighted portfolio[14] is built and tested daily based on predicted scores. The combination of hyper-parameters with the highest cumulative return in the validation set are then fixed and used in all subsequent tests.

In FarmPredict, tuning starts with finding C in equation (2.4), which controls the amount of underlying factors. Figure 3 shows the scree plot and eigen differences plot of data from 2000 to 2014, using the adjusted eigenvalues of the correlation matrix of binary word counts. The figure shows that there are 2 stronger factors and 7 relatively weaker factors. Inspired by this, we choose $C = 150$, which gives $\widehat{k} = 9$ factors in adjusted eigenvalue thresholding method (2.4). We then, fix $C = 150$ through the study.[15]

With C fixed in FarmPredict, we need only to tune κ for screening frequently-used words in $\mathbf{D}^{\text{freq}}$ and α for screening sentiment charged words in $\mathbf{S}$. The tuning parameter κ is chosen from the 80% to 94% quantiles of k_j's of all words, with increments of 2%. There are around 70K words in each 10-year training period and the range of κ corresponds to the range of 4500 - 15000 words from $\mathbf{D}^{\text{freq}}$. The tuning parameter α is the threshold in conditional correlation screening, for controlling the number of words selected into $\widehat{\mathbf{S}}$. It is chosen to ensure the number of remaining words $|\widehat{\mathbf{S}}|$ being exactly 500, 1000 or 2000. Further selection of sentiment-charged words is done via penalized logistic regression (2.6) with cross-validation.

[14]The portfolio longs the stocks with top 50 predicted scores and short with the 50 lowest with 1% capital each. More details can be found in Section 4.2. Remaining capital will be kept as cash if less than 50 stocks are selected.

[15]We also tested the choices of $C = 30$ and $C = 1$ (suggested by Fan et al. (2020c)). They result in 80 and 1043 weak factors respectively. Since our sample size is very large, over estimation of k is not a serious problem and the results are similar. More details regarding choices of the number of factors can be found in Section 4.3.2.



(a) Top adjusted eigenvalues

(b) Top adjusted eigenvalue differences

Figure 3: Top adjusted eigenvalues and eigen differences of the correlation matrix of binary word counts, using data from 2000 to 2014. There are 2 major factors and 7 relatively weaker factors, corresponding to $C = 150$ in adjusted eigenvalue thresholding (2.4).

There are four hyper-parameters $(\alpha_+, \alpha_-, \kappa, \lambda)$ in tuning our topic model. Similarly, κ is chosen from the 80% to 98% quantiles of k_j's of all words, with increment of 2%. Thresholds (α_+, α_-) in sure screening are set so there are exactly 10/25/50/100 positive words and same number of negative words remaining in $\widehat{\mathbf{S}}$. Similarly $\lambda_{\text{Logistic}}$ is set appropriately so there are exactly 20/50/100/200 words with non-zero regression coefficients. Penalty λ_{PMLE} in predicting sentiment score is chosen from $(1, 3, 10)$.

Rolling windows test. All methods are trained and tested via rolling windows on the basis of six months. The optimal hyper-parameters for each model selected in tuning are kept fixed for all training windows. For each window, 10 years of data are used for training models and the subsequent 6-months' data are then used for testing. Predicted scores for every article in testing are recorded. After the training and testing for a window are complete, we roll forward the entire window by 6 months and redo training and testing, and repeat. The first window uses data from 2005-2014 for training and data from 2015.1-6 for testing. The last window uses 2019.7-12 as testing period. In total 10 windows are examined and we

recorded the predicted sentiment scores on every trading day from 2015 to 2019.

The training and testing windows in our rolling window test are carefully chosen based on the distribution of our data. The amount of training and testing data is stable across windows. Among the ten windows, the number of training articles range from 428K to 529K with input words ranging from 761K to 863K. Only words appeared at least 50 times among all articles are regarded as valid words as our model input, so the input dimension of $\mathbf{X}$ ranges from 66K to 71K each window.

4 Results

4.1 Validation of Sentiment Scores

4.1.1 Sentiment-charged words

To verify our sentiment indices extracted from the context of news, we first report the top sentiment-charged words and provide a comparison between our FarmPredict and the marginal screening from the ad hoc topic model. Figure 4 separately presents the top positive and negative words selected by the models. We adopted the Chinese style of coloring where red indicates positive sentiments and green for negative sentiments. Figure 4(a) shows the result from FarmPredict and (b) shows the result from marginal screening (2.10). The font size of each word is proportional to its sentiment strength in the models. In FarmPredict (Figure 4(a)), we selected only words in $\widehat{\mathbf{S}}$ and used their regression coefficients in $\widehat{\boldsymbol{\beta}}$ as the sentiment strength, while for marginal screening (Figure 4(b)), we used word i's marginal correlation f_i as its sentiment strength.

(a) Sentiment-Charged Words from FarmPredict

(b) Sentiment-Charged Words from Marginal Screening

Figure 4: Top sentiment words estimated by FarmPredict and Marginal Screening. The top 50 words by their sentiment strength in corresponding methods are selected and their font sizes are proportional to their sentiment strengths. FarmPredict(a) selected only words in $\widehat{\mathbf{S}}$ and used their regression coefficients in $\widehat{\boldsymbol{\beta}}$ as sentiment strength. Marginal Screening(b) used word i's correlation f_i from sure screening as its sentiment strength.

Table 3: Top sentiment-charged words chosen by FarmPredict and their corresponding Pinyin and English meanings.

Rank	Positive Words			Negative Words		
	Chinese	Pinyin	English	Chinese	Pinyin	English
1	涨停	Zhang Ting	reach daily upper limit	跌停	Die Ting	reach daily lower limit
2	走强	Zou Qiang	trending high	敢死队	Gan Si Dui	suicide squad
3	十只	Shi Zhi	ten stocks	准确率	Zhun Que Lv	accuracy
4	涨	Zhang	rise	日盘	Ri Pan	open hours market
5	抢反弹	Qiang Fan Tan	trade before revert	跌	Die	drop
6	拉升	La Sheng	push up	不超	Bu Chao	less than
7	发稿	Fa Gao	report	全网	Quan Wang	all over the internet
8	早盘	Zao Pan	morning market	十档	Shi Dang	level ten
9	面上	Mian Shang	on the surface	净流入	Jing Liu Ru	net inflow
10	日复盘	Ri Fu Pan	daily market review	送股	Song Gu	bonus share
11	首日	Shou Ri	first day	高频	Gao Ping	high frequency
12	快讯	Kuai Xun	breaking news	全线	Quan Xian	everywhere
13	起复盘	Qi Fu Pan	market review	最低价	Zui Di Jia	lowest price
14	首个	Shou Ge	first	减持	Jian Chi	selling stock
15	股票交易	Gu Piao Jiao Yi	stock trading	汇总	Hui Zong	summary
16	预增	Yu Zeng	rise before earning report	跌幅	Die Fu	decline
17	举牌	Ju Pai	IPO	弱	Ruo	weak
18	上证指数	Shang Zheng Zhi Shu	SSEC Index	大跌	Da Die	fall sharply
19	差额	Cha E	difference	涉嫌	She Xian	involved in
20	大阳线	Da Yang Xian	rise $\geq 7\%$ intraday	终止	Zhong Zhi	terminate

Note: These words are selected as a group to best augment the prediction by latent factors.

Table 4: Top sentiment-charged words in SESTM (ad hoc topic model) and their corresponding Pinyin and English meanings.

Rank	Positive Words			Negative Words		
	Chinese	Pinyin	English	Chinese	Pinyin	English
1	走强	Zou Qiang	trending high	跌停	Die Ting	reach daily lower limit
2	拉升	La Sheng	push up	造假	Zao Jia	fraud
3	早盘	Zao Pan	morning market	涉嫌	She Xian	involved in
4	涨	Zhang	rise	因涉嫌	Yin She Xian	because was involved in
5	面上	Mian Shang	on the surface	大跌	Da Die	fall sharply
6	发稿	Fa Gao	report	立案	Li An	initiate investigation
7	午后	Wu Hou	afternoon	违法行为	Wei Fa Xing Wei	illegal activity
8	居前	Ju Qian	heading	不超	Bu Chao	less than
9	概念股	Gai Nian Gu	concept stock	跌	Die	drop
10	快讯	Kuai Xun	breaking news	弱	Ruo	weak
11	涨停	Zhang Ting	reach daily upper limit	主管人员	Zhu Guan Ren Yuan	people in charge
12	抢反弹	Qiang Fan Tan	buy before rebound	警示	Jing Shi	warning
13	客观性	Ke Guan Xing	objectivity	通报	Tong Bao	announce
14	个人观点	Ge Ren Guan Dian	personal opinion	下挫	Xia Cuo	plummet
15	解套	Jie Tao	get out of the position	一事	Yi Shi	the incident
16	新闻报道	Xin Wen Bao Dao	news report	证监局	Zheng Jian Ju	SEC
17	点评	Dian Ping	comment	走低	Zou Di	drop
18	该股	Gai Gu	the stock	起诉	Qi Su	sue
19	异动	Yi Dong	unnatural movement.	股民	Gu Ming	retail investor
20	成交额	Cheng Jiao E	traded volume	案件	An Jian	case

Note: These words are chosen to have best correlation with the sign of returns. While each individual word appears intuitive, these words together are not necessarily the best predictors for the returns.

Since our study focused on Chinese text data, we also present the *pinyin* for pronunciation and the meaning of top positive and negative words in Table 3 and Table 4. The words are ranked by their sentiment level, namely, the correlation between returns in Table 3 and Table 4. The top 5 sentiment-charged words for positive returns are:

FarmPredict: 涨停 (Reached daily upper limit), 走强 (Trending high), 十只(Ten stocks), 涨 (Rise), 抢反弹 (Trade before revert)

Marginal Screening: 走强 (Trending high), 拉升(Push up), 早盘 (Morning market), 涨 (Rise), 面上 (on the surface)

and the following words are the top 5 sentiment-charged words for negative returns:

FarmPredict: 跌停 (Drop to the lower limit), 敢死队 (Suicide squad), 准确率 (Accuracy), 日盘 (Open hours), 跌 (Drop)

Marginal Screening: 跌停 (Drop to the lower limit), 造假 (Fraud), 涉嫌 (Suspected), 大跌 (Fall sharply), 立案 (Initiate investigation)

Comparisons between the sentiment-charged words between FarmPredict and the ad hoc topic model reveal that ad hoc topic model chooses words that are marginally highly correlated with sentiments, while FarmPredict applies all information of the article to select coordinated words, resulting in more "non-sentiment" words such as "十只 (ten stocks)" and "敢死队 (suicide squad)". Since there are particular language and writing mannerisms of each human being, not only general sentiment-charged words but also fixed collation and metaphor may be used to express and state comments and opinion in news. For instance, we barely find the word "敢死队 (suicide squad)" in any sentiment dictionary from previous studies, but when writing articles, the reporter and editor usually analogize the monetary inflow in a depressed stock market as "suicide squad". Hence, we found it has a strong predictive power of negative returns. Moreover, when focusing on the negative words, it can be revealed that topic model detects more law-violation words while words estimated from FarmPredict related more to asset pricing and trading.

Another interesting finding is that top positive sentiment-charged words in Chinese stock markets are more "trading-related" while previous literature about the US market concluded a more"value-related" result. It also matches the current condition in Chinese stock markets that individual investors play a more critical role in market trading and are more likely to be influenced by trading related news. Therefore, instead of a value-related signal, positive trading-related news of stocks will be more effective in explaining asset price changes in Chinese stock markets, known as the "herding effect". This result also demonstrates a relatively lower efficiency in Chinese stock markets. Unlike positive words, there is a more "value-related" phenomenon in the negative part, with more legal-related words such as "involved in" and "fraud", which are traditional influencing factors on asset pricing. This result illustrates more rational behavior and implies greater similarity to the US market[16]. Since short sale is constrained and strictly supervised in China, which is usually conducted by institution investors or professionals, asset pricing information provides a stronger signal on driving market trading.

4.1.2 Do sentiments predict returns?

Even though we have tested the consistency of our sentiment-charged words and the sentiments, it is still critical to directly validate whether our calculated sentiment scores have any prediction power on the returns. Based on our training target, we would expect that our sentiment scores can predict the beta-adjusted returns of their associated stocks. However, this process should not capture the information of the whole market, thus expecting a much weak prediction power on market returns.

We first conducted the regression by forming a panel data for the beta-adjusted returns of stocks from Jan 2005 to Dec 2019. The multiple regression is the following, in which we suppress the regression coefficients:

$$Return_{it} = Sentiment_{i,t-1} + Return_{i,t-1} + Return_{i,t-2} + Return_{i,t-3} + \mu_t + \epsilon_{it}$$

where $Return_{it}$ is the beta-adjusted return of stock i in day t; $Sentiment_{i,t-1}$ is the corre-

[16]The positive and negative words in the US market are cited from Ke et al. (2019) using the topic model. The positive words include *undervalue, repurchase, surpass, upgrade, rally* and negative words are *shortfall, downgrade, disappointing, tumble, blame*

sponding sentiment score of stock i in day t computed for the news article; μ_t is time (day) fixed effects, capturing the time-related daily effect such as market conditions and economic growth. Besides the fixed effects, since the beta-adjusted returns might be correlated with their past data, we also added the lagged returns as the control variables. As our sentiment score is trained with the stock-related news, there is a possible endogeneity issue that the news we use is driven by the beta-adjusted returns, i.e. the news might be reported after the extremely high/low beta-adjusted return occurred. Our use of lagged returns mitigates this endogeneity issue between returns and the sentiment scores.

Table 5 presents the results of sentiment scores estimated by different models separately. We gradually added the control variables into the model to test the robustness of the correlation. As shown in Table 5, there is a significant positive correlation between beta-adjusted return and the sentiment score according to Column 1, 4 and 7. This positive correlation stays robustly significant, only with coefficient turning smaller after controlling the lagged terms of beta-adjusted returns, shown in other Columns in Table 5. It can be seen from Table 5 that our sentiment scores are highly correlated with the beta-adjusted returns of each corresponding stock with strong multiple-R^2, thus can be applied to build portfolios with high beta-adjusted returns. [17]

[17]The sentiments for the three presented models have different scales. Hence, their estimated regression coefficients can only be compared within the model, but not across the model. Statistical significance and multiple R^2 are comparable across different models.

Table 5: Correlation between sentiment score and stock beta-adjusted return

	FarmPredict			SESTM-Logistic			SESTM		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
$sentiment_{i,t-1}$	0.347***	0.220***	0.206***	0.844***	0.457***	0.417***	4.240***	2.360***	2.165***
	(0.0011)	(0.036)	(0.036)	(0.032)	(0.093)	(0.090)	(0.168)	(0.461)	(0.447)
$Return_{i,t-1}$		0.107***	0.104***		0.120***	0.116***		0.118***	0.114***
		(0.027)	(0.027)		(0.028)	(0.027)		(0.028)	(0.027)
$Return_{i,t-2}$			0.017***			0.020***			0.020***
			(0.004)			(0.003)			(0.003)
$Return_{i,t-3}$			0.029***			0.033***			0.033***
			(0.004)			(0.004)			(0.004)
Time fixed effect	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Adjusted R^2	0.076	0.091	0.092	0.071	0.089	0.090	0.071	0.089	0.090

Note: This table presents the estimation results of equation $Return_{it} = Sentiment_{i,t-1} + Return_{i,t-1} + Return_{i,t-2} + Return_{i,t-3} + \mu_t + \epsilon_{it}$. The outcome variable is the beta-adjusted return of stock i in day t. Columns 1 to 3, Columns 4 to 6 and Columns 7 to 9 show the results in terms of different method on sentiment score estimation. All standard errors are clustered by stocks. SESTEM-Logistic refers to selecting words from logistic regression and then applying SESTEM to score sentiments. All scores are normalized and centered at 50. The range of score estimated SESTEM is smaller than the others, leading to the difference in coefficient scale. For statistical significance, $^*p < 0.1$, $^{**}p < 0.05$, and $^{***}p < 0.01$.

Even though Table 5 provides strong evidence on the prediction power of our sentiment scores on their associated stock returns, it is still essential to check if we captured the genuine specific features of stocks in that day but not the global attributes and information of the whole market. With this goal, we then conducted a similar regression analysis between daily market returns and daily average sentiment scores and their dispersions, which are calculated based on all sentiment scores for the articles published in that day. We took daily returns of market indices in Shanghai and Shenzhen stock markets to form time-series data from Jan 2005 to Dec 2019 and fit the following regression model:

$$Return_t = AveSentiment_{t-1} + \mathrm{DISP}_{t-1} + Return_{t-1} + Return_{t-2} + Return_{t-3} + D_{year} + D_{month} + \epsilon_t$$

where $Return_t$ is the return of SSEC index; $AveSentiment_{t-1}$ is the daily average sentiment score; DISP_{t-1} is the dispersion variable of the score to control the variation[18] represented by the standard deviation, and D_{year} and D_{month} are year and month fixed effects to control yearly and seasonally related trends. We also used the lagged terms to mitigate the endogeneity issue and controlled the lagged terms of market returns in our models to provide a robust estimation.

The results are shown in Table 6. Columns 1 to 3, Columns 4 to 6 and Columns 7 to 9 depict the regression results based on the scores estimated by different models. We studied the correlation between the sentiment scores and market returns by sequentially adding the lagged terms. The results in Table 6 reveal that, unlike Table 5, none of the results could provide evidence on the predictability of sentiment scores on the market returns. These non-significant results in Table 6 meet our expectation: since the sentiment scores are trained based on the beta-adjusted returns of individual stocks, a well-tuned model will only capture information of individual stock but not the entire market. Both of the results in Tables 5 and 6 validate the performance of our model on extracting stock-level information from news and neglect the global information of the market.

[18]The average and standard deviation are used for a quick summary of the distribution of the sentiments of daily news articles. They can be replaced by quintiles or deciles for a more informative summary.

Table 6: Correlation between sentiment score and market return

	FarmPredict			SESTEM-Logistic			SESTEM		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
$sentiment_{t-1}$	-0.001	-0.005	0.049	-0.093	-0.139	-0.688	-1.999	-2.331	-4.619
	(0.121)	(0.124)	(0.131)	(0.967)	(0.975)	(1.091)	(4.223)	(4.317)	(4.649)
$DISP_{t-1}$			-0.287			-0.843			-2.552
			(0.233)			(0.741)			(2.597)
$Return_{t-1}$		0.042	0.045		0.048*	0.046		0.050*	0.047
		(0.029)	(0.029)		(0.029)	(0.029)		(0.029)	(0.029)
$Return_{t-2}$		-0.071**	-0.071**		-0.066**	-0.068**		-0.065**	-0.068**
		(0.029)	(0.029)		(0.029)	(0.029)		(0.029)	(0.029)
$Return_{t-3}$		0.023	0.024		0.028	0.025		0.029	0.025
		(0.029)	(0.029)		(0.029)	(0.029)		(0.029)	(0.029)
Year fixed effect	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Month fixed effect	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Adjusted R^2	-0.005	-0.001	-0.0004	-0.002	0.002	-0.001	-0.002	0.003	-0.0004

Note: This table presents the estimation results of the model $Return_t = AveSentiment_{t-1} + DISP_{t-1} + Return_{t-1} + Return_{t-2} + Return_{t-3} + D_{year} + \epsilon_t$. The outcome variable is the market return of Shanghai stock index in day t. The dispersion variable is represented by the standard deviation of the daily sentiment scores we calculated. For statistical significance, $^*p < 0.1$, $^{**}p < 0.05$, and $^{***}p < 0.01$.

4.1.3 Event-study on sentiment scores

In this subsection, we conducted an event-study to see whether there is a significant reaction of individual stocks to sentiment scores. To do this analysis, we treated the occurrence of sentiment score as an "event" and took a subsample which covered 14 days before and after the occurring date. Therefore, we can observe the pattern of beta-adjusted return change caused by the news and stock sentiment in this panel data. Then we conducted the regression as follows:

$$Return_{it} = \sum_{p=-13}^{14} \beta_p Day_{ip} + \delta_i + \mu_t + \epsilon_{it} \tag{4.1}$$

where $Return_{it}$ is the beta-adjusted return of stock i in day t; Day_{ip} are indicators of days before and after the sentiment occurs, of which the range is -13 to 14[19]; δ_i and μ_t are stock individual and day fixed effects, to control the heterogeneity in stock and date, respectively.

Model (4.1) provides straightforward results on how the markets and stocks anticipate (before) and react (after) to the sentiment scores. Figure 5 depicts the results of fitting model (4.1). The results show significant heterogeneous mechanisms between positive and negative news. For the positive sentiments, the beta-adjusted returns start to increase and reach a relatively high level about seven days before the sentiment occurs, indicating a market-driven sentiment. The highest impact is on the day of news arrivals, with an average of 83 bps. Consistent with the discussion on the words we extracted from the news, positive news in the Chinese stock markets mainly covers the trading related reports. Another possible reason is that the information is leaked to market participants, leading to an increase in return before the news occurs. However, for negative news, we didn't observe this phenomenon that returns decrease before a news announcement, and the beta-adjusted return is only negative while news occurred, which has an average impact of 26 bps. This aligns well with the short-sale restrictions in the Chinese markets: Even if the news are leaked or anticipated, transactions are hard to take place. It is also consistent with the result in Figure 5 that the positive news has bigger impact on stock returns than the negative ones, contrarily to the

[19]We assume that the latest news will have a higher power to affect beta-adjusted returns of stocks. Hence if another news occurred within the 14 days range of former news, we will recalculate and renew the periods of the day indicator.

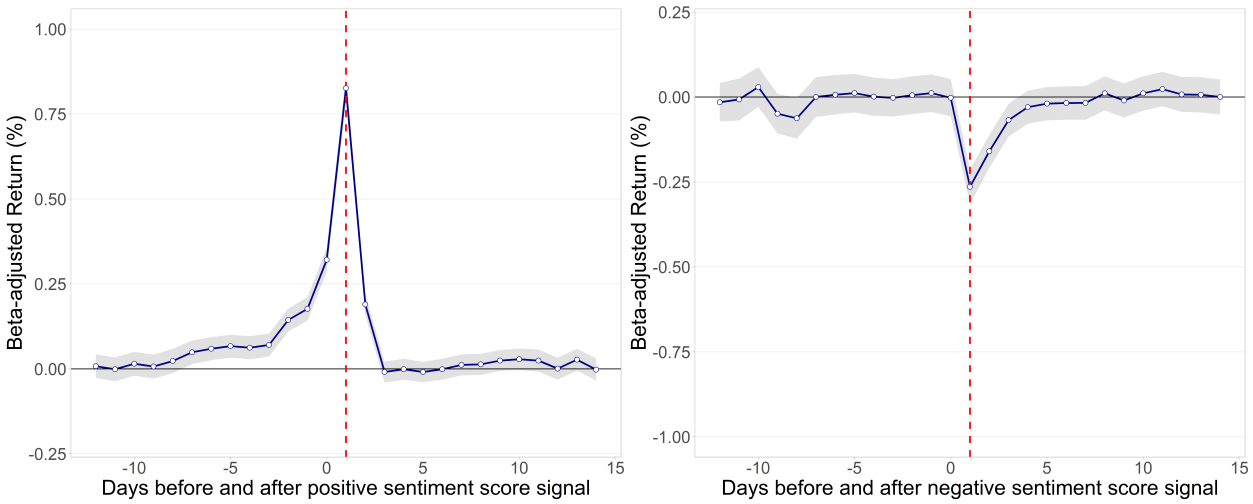


Figure 5: Event study on beta-adjusted return before and after the new announcement. The horizontal axis represents the days before and after news announcements, and the vertical axis is the beta-adjusted return during that day. We set day 1 as the day of event (news) occurring. The white circles in this figure are the point estimates of the mean beta-adjusted return (estimated β_p in model (4.1)) and the bands around the circles indicate the 95% confidence interval of the point estimates. This figure illustrates the trend of beta-adjusted returns before and after news announcements.

behavior in the US equity market.

For the beta-adjusted returns after the news announcement, we found similar patterns for both positive and negative news, with an existence period of 2 days and 3 days, respectively. Beta-adjusted returns after this period are statistically insignificant from zero for both groups. The results show an arbitrage opportunity for portfolios built at the day after news announcements. Therefore, the results of this event study also provide a mechanism on why our constructed portfolios in the next subsection can achieve high beta-adjusted returns based on the sentiment scores.

4.1.4 Placebo test

In this subsection, we conducted a placebo test for our event study to test if this specific trend of beta-adjusted returns is caused by the event, as measured by the sentiment scores in this paper. To evaluate this, we randomly pick a subsample with continuous 28 days

from each stock, the same length as that in the previous event study, from the data not overlapping with the event period. Then we reran the event study regression on this new random sample and replicated it 200 times to see if the significant out-performed returns will occur. This results in 200 curves, depicted in Figure 6. The gray area is the accumulated estimation results of each replication, showing a distribution with mean of zero. The results in Figure 5 are superimposed in Figure 6 for comparisons. This result boosts our confidence that the results in Figure 5 are robust and genuine, specifically caused by the news and reports. Moreover, we can observe that beta-adjusted returns after the initial day of new announcements still stand out the placebo returns, providing a tradable portfolio building strategy, which will be introduced in the next section.

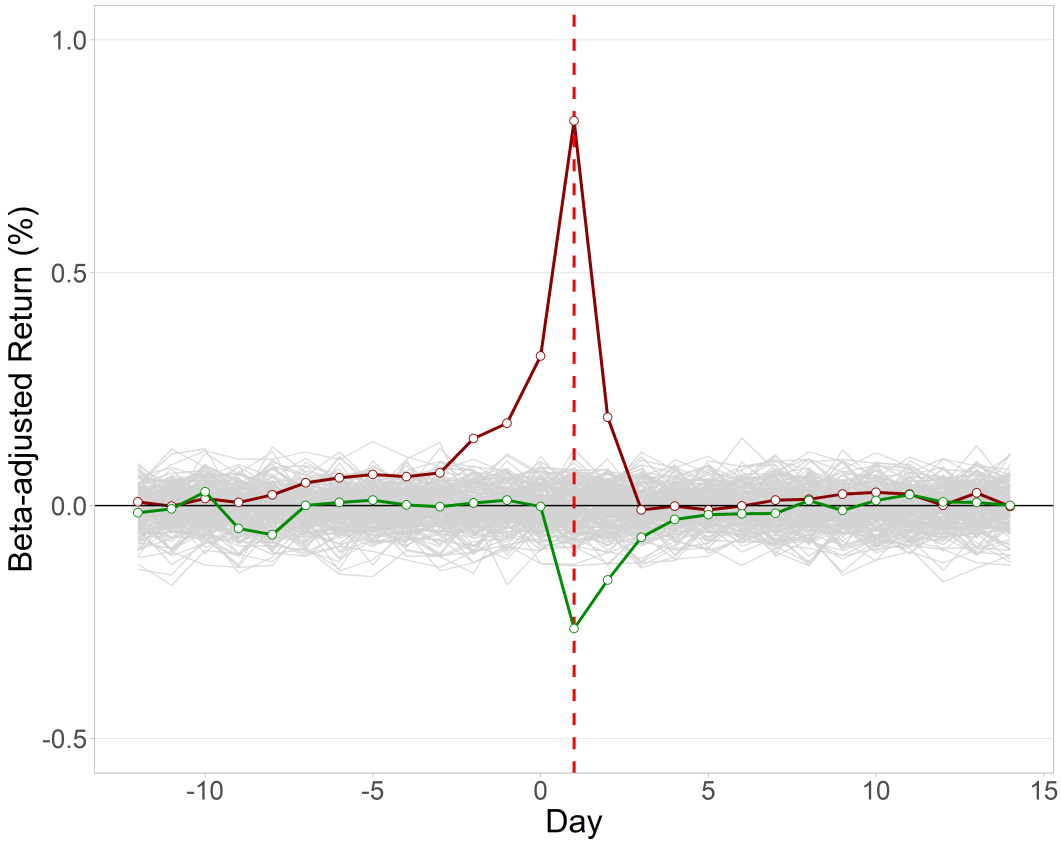


Figure 6: Placebo test of sentiment score. The light gray lines are the point estimates, based on 200 experiments, from fitting the model $Return_{it} = \sum_{p=-13}^{14} \beta_p Day_p + \delta_i + \mu_t + \epsilon_{it}$ on the subsample by removing observations of stocks affected by the news (sentiment). All others are the same as those in Figure 5.

4.2 Portfolio Performance

We also tested the models by building stock portfolios based on their predicted scores. Portfolios are built and tested in each rolling window as follow. For each day before market close (15:00 HKT), the model looked at all the articles since the previous market close and calculate their scores. Then invest by longing 50 stocks with the highest scores and shorting 50 stocks with the lowest scores. We long and short each stock with the same fixed 1% total capital exposure each day. If there are less than 50 stocks with positive / negative signals, then the un-allocated capital will be kept as cash (with no interest), so the portfolio's total capital exposure is never greater than 100%. We form our position at the day's closing auction and close it at the second trading day's closing auction. We call this an equally-weighted portfolio (EW).

Value-weighted portfolios (VW) are also tested for comparison. With the same set of stocks to long or short as EW, the weights in a value-weighted portfolio is set to be proportional to stocks' total market capitalization on prior day. Such portfolio would put larger weights on large-cap stocks compared to small-cap ones. Usually there are more informed investors trading large-cap stocks so their prices are typically more efficient. So value-weighted portfolios typically have better liquidity in trading and less returns. We anticipated that it is less affected by new sentiments as in Ke et al. (2019).

4.2.1 Performance of Portfolios

The portfolio returns for the EW strategy are computed based on $100 invested each day, investing $1 on each of long or short positions and adding up the total daily gains divided by 100. Since this is a long-short portfolio, the actual capital expenditure is much lower than 100, yielding even better performances. A similar computation of portfolio returns is applied to the VW strategy.

As a result of tuning, the optimal screening κ is chosen as 14% of total number of words

[20]Let the returns on long and short leg on day d be $r_{d,\text{Long}}$ and $r_{d,\text{Short}}$. Then, the yearly return is compounded through $1 + \text{APR} = \prod_d (1 + r_{d,\text{Long}} + r_{d,\text{Short}}) \approx \prod_d (1 + r_{d,\text{Long}})(1 + r_{d,\text{Short}}) = (1 + \text{Long}) * (1 + \text{Short})$.

Table 7: Portfolio performances from 2015 to 2019 of FarmPredict and Ad Hoc Topic Models (AHTM).

		Equally-weighted				Value-weighted			
Method	#Words	SR	APR	Long	Short	SR	APR	Long	Short
FarmPredict	217.1	9.37	116%	80%	18%	3.34	48%	47%	1%
AHTM w/ $\widehat{\mathbf{S}}^{\text{Screen}}$	50	7.00	104%	58%	27%	2.69	41%	31%	5%
AHTM w/ $\widehat{\mathbf{S}}^{\text{Logistic}}$	20	8.58	84%	68%	9%	3.21	43%	43%	-1%
AHTM w/ $\widehat{\mathbf{S}}^{\text{Screen}} \cup \widehat{\mathbf{S}}^{\text{Logistic}}$	37.1	8.17	80%	64%	9%	3.01	40%	41%	-1%
Averaged portfolio	N/A	9.83	95%	67%	16%	3.87	43%	41%	1%

Note: The averaged portfolio invests 1/4 capital in each of the 4 daily portfolios. For each model, the corresponding columns show its Sharpe Ratio (SR), Annualized Percent Return (APR) as well as its annualized return on the portfolio's Long-side (Long) and Short-side (Short) respectively. Hyper-parameters, including those affecting the number of sentiment-charged words, are tuned using data from 2005 to 2014. Due to daily compounding, APR is related to the returns in the long and short lags via $\text{APR} \approx (1 + \text{Long}) * (1 + \text{Short}) - 1$. [20]

for FarmPredict and 10% for Ad Hoc topic models. For example in the last 10-year window of 2009.7 - 2019.6, there are 71K words as input and FarmPredict selected 9948 most frequent words to be in $\mathbf{D}^{\text{freq}}$. Results for different methods are shown in Table 7 and the cumulative log 2 returns for equally-weighted portfolios are plotted in Figure 7. #Words indicates the average sizes of $\widehat{\mathbf{S}}^{\text{Topic}}$ in ad hoc topic models and is the average sizes of non-zero entries of $\widehat{\boldsymbol{\beta}}$ in FarmPredict. In addition to FarmPredict and the three variants of the Ad Hoc Topic Model, we presented also the performance of their averaged portfolio as a meta-portfolio. The averaged portfolio invests 1/4 of its capital on each of the four portfolios, so its daily returns are the average of those four portfolios. The average portfolio benefits from the reduced risk of diversification and stability of model averaging.

All methods performed very well in equally-weighted portfolios, which strongly indicates that there are stock price related signals residing in Chinese news texts. Among the methods, FarmPredict performed the best with 116% annualized return and estimated Sharpe Ratio of 9.37. Value-weighted portfolios also showed good returns of 48% annually, while not as good as that of equally-weighted portfolios. This suggests that large-cap stocks' are more

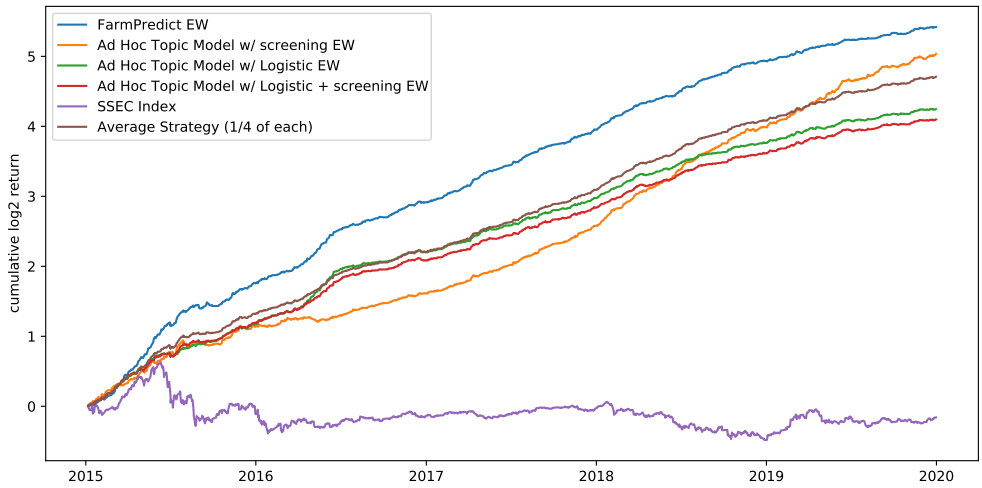


Figure 7: Cumulative log2 returns of each strategy over time from 2015 to 2019. All methods performed reasonably well and FarmPredict attains the best final return.

popular and better studied, so their prices are less affected by the arrivals of financial news. This result for the Chinese market is consistent with those in the U.S. market obtained by (Ke et al., 2019).

4.2.2 Return compositions and market risks

To better understand the strategy, we studied the detailed components of its returns and risks. We introduced measures to decompose and evaluate a portfolios' idiosyncratic return and market risk exposure, and used them to analyze FarmPredict's returns and risks from its long leg, short leg and market movements.

Financial returns are noisy and affected by various market conditions. Stock short-term movements induced by market conditions are usually thought as orthogonal to the stock's own fundamentals or stock-specific signals. To evaluate alphas, we uses beta-adjusted returns, rather than raw returns, to isolate the part of returns induced by its own signal. Here, beta represents the stock's exposure to market movements. Beta-adjusted returns can also be approximately viewed as the returns of longing the stock while shorting market index future or ETFs.

To reduce noise from market movement and better evaluate our portfolios, we calculated

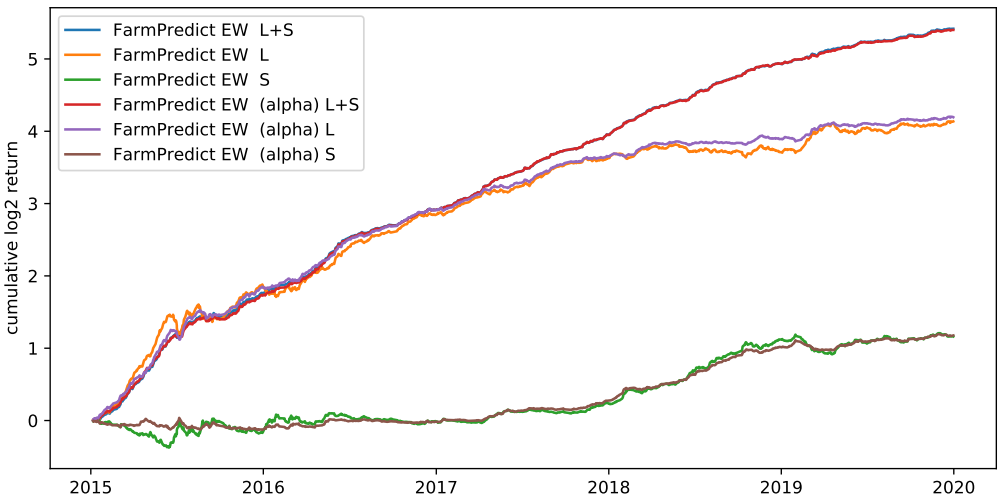


Figure 8: Cumulative log2 returns of Long-Short, Long-only and Short-only strategy and their associated beta-adjusted returns from 2015 to 2019. The long-short equally-weighted portfolio has little correlation with the market, with the beta-adjusted return curves almost perfectly overlapping with the raw return curves. Having investments on both long and short side greatly helped smooth out market volatility. The long leg of the portfolio contributes most returns prior to 2018 while the short leg picks up after that.

the beta-adjusted returns of stocks by hedging them against SSEC. This hedging is also tradable in the market as there are several liquid ETFs and futures tracking this market index. The betas are calculated by linearly regressing each individual stock' returns against SSEC returns, using all daily returns from 2005 to 2014:

$$\text{Return}(r_{it}) = \alpha_i + \beta_i \cdot \text{SSEC Return}(r_t^{market}) + \varepsilon_{it}$$

Let $\mathbf{w}_t$ be a vector of portfolio weights on each stocks as of time t. Coefficients of $\mathbf{w}_t$ are positive for long and negative for short positions. We can calculate the portfolio's beta (exposure to market) as the weighted sum of betas from each individual stock, namely compute $\beta(\mathbf{w}_t) = \sum_i w_{it}\beta_i$. Let $r_t(\mathbf{w}_t)$ be the portfolio's return at time t. We can accordingly define its beta-adjusted return (alpha) $r_t^*(\mathbf{w}_t)$ as:

$$r_t^*(\mathbf{w}_t) = r_t(\mathbf{w}_t) - \beta(\mathbf{w}_t) \cdot r_t^{market} = \sum_i w_{it}(r_{it} - \beta_i \cdot r_t^{market}).$$

The performances of portfolios in either regular returns or beta-adjusted returns from 2015 to 2019 is shown in Figure 8. The models are also tuned and fitted using beta-adjusted

returns. The curves of cumulative raw returns and beta-adjusted returns (alpha) are almost the same in the figure, with beta-adjusted returns outperforming after 2018, indicating that the portfolio is minimally exposed to market risks. The long and short legs cancel out each other's short-term variations and contributed to the overall portfolios in different period of times.

Table 8: Characteristics of EW and VW portfolios based on FarmPredict.

		Sharpe Ratio	APR	Alpha APR	R^2_{market}	Daily Return
EW	L + S	9.37	116%	115%	6.3%	31 bps
	L	4.06	80%	82%	44.7%	24 bps
	S	1.24	18%	18%	48.4%	7 bps
VW	L + S	3.34	48%	48%	5.2%	16 bps
	L	2.55	47%	47%	54.0%	16 bps
	S	0.01	1%	1%	56.8%	0 bps

Note: The testing period ranges from 2015 to 2019. Sharpe ratios, daily and annualized average returns, alpha returns and market exposures are reported. Separate returns of the short (S) and long (L) leg of both portfolios are reported as well.

To further quantify the relationship of our returns to the market. We proposed the following R^2 measure to account for the amount of variance in portfolio returns that are related to the market. Let $\bar{r} = T^{-1}\sum_t r_t(\mathbf{w}_t)$ be the average portfolio return across time. Based on the decomposition of the beta-adjusted returns $r^*_t(\mathbf{w}_t)$ (alpha) and the market-related returns, we define R^2_{market} as the proportion of variance in returns from market as

$$R^2_{market} = \frac{\sum_t \left[r_t(\mathbf{w}_t) - r^*_t(\mathbf{w}_t)\right]^2}{\sum_t \left[r_t(\mathbf{w}_t) - \bar{r}\right]^2}$$

Note that the R^2_{market} is not a result from OLS regression. It is borrowed from OLS' definitions to illustrate the amount of market movements in our portfolio.

Results on the portfolios based on FarmPredict model are shown in Table 8. Only 6.3% of the overall variance is related to the market, since market exposures from longs and shorts cancel out when combined. The long and short legs themselves, as expected, assume large market exposures of around 45%. Another observation is that returns from the long-leg is

larger than those from the short-leg. Such a finding is also in line with those presented in Figure 5.

4.2.3 Transaction cost

Many challenges and costs come with real trading with a mid-frequency strategy like ours. Our models and portfolios are built on news data that change on a daily basis and our portfolios are reconstructed everyday. There are also significant transaction costs and taxes charged by the exchanges or stock retailers for our high turnover portfolios. Furthermore, there are difficulties in building a desired position due to liquidity constraints for various reasons. We designed our portfolio so it only trades at closing auctions where liquidity is at its maximum to minimize market impacts. The short sale restrictions in Chinese stock market might also significantly increase the cost in shorting.

The average turnover ratio (4.2) of a portfolio is defined as the daily average of the total changed proportion of portfolios. The total portfolio weight $\mathbf{w}_t$ is no greater than 1 by construction (typically equal to 1), so $\|\mathbf{w}_t\|_1 \leq 1$ and the case $\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_1 = 2$ implies the portfolios are totally different between day t and day $t+1$. The portfolios between each two adjacent days are compared and only their differences are traded. For simplicity, we ignored the changes of weights from t to $t+1$ due to stock price changes in turnover calculations. Turnover ratios for each portfolio's components are shown in Table 9, in which

$$\text{Average Turnover Ratio} := \frac{1}{2(T-1)} \sum_{t=1}^{T-1} \|\mathbf{w}_t - \mathbf{w}_{t+1}\|_1 \tag{4.2}$$

Transaction costs of trading in Chinese stock markets are made up of the following three main components:

1. **Stamp Duty** is 0.1% of total capital transaction amount. Only sellers are charged. Equivalent to 10 bps cost in our portfolio if all positions are liquidated next day.

2. **Transfer Fee** is 1 CNY for each 1000 shares traded and is charged to both buyer and seller. It is only charged on stocks traded on Shanghai Stock Exchange, not on Shenzhen. Thus, there is 1 bps combined cost (buy and then sell in Shanghai Stock Exchange) for a stock with a price of 20 CNY per share.

3. **Trade Commission** ranges from 0.01% to 0.3% each transaction and is charged by stock retailers on both sides of a trade. Typical rates are around 2.5 bps.

We can see, on an typical case with stock market capitalization above 20 CNY (most stocks are above this price), each trade we made (buy and sell combined) incurs 10 bps in Stamp Duty, 1 bps in Transfer fee and 5 bps in trade commission. So only trades with a positive expected return of over 16 bps daily is profitable under this conditions.

Table 9: Comparison of FarmPredict portfolios with and without transaction costs.

			no Transaction Cost		w/ Transaction Cost	
		Turnover Ratio	APR	Sharpe Ratio	APR	Sharpe Ratio
EW	L + S	91.6%	116%	9.37	45%	4.46
	L	90.0%	80%	4.06	47%	2.62
	S	92.2%	18%	1.24	-4%	-0.25
VW	L + S	91.8%	48%	3.28	0%	0.04
	L	90.8%	47%	2.55	20%	1.18
	S	92.8%	1%	0.01	-19%	-1.44

Note: The transaction cost is placed daily when components of the portfolio are changed. Transaction cost includes stamp duty, transfer fee and trade commission in China. We assumed a 16bps transaction cost each buy & sell trade combined for the 'w/ Transac Cost' column.

We tested our portfolios with transaction costs and their annualized cumulative returns after transaction costs are shown in Table 9. For simplicity, we assumed transfer fee being fixed 1 bps in total and thus total transaction cost being 16 bps. We ignored transaction costs caused by price impacts or bid-ask spread here. The equally-weighted strategy still has a positive profit after transaction costs, while the scale of profit reduced by more than half.

4.2.4 Daily price limit in Chinese equity markets

China imposes a 10% price limit in its equity market[21], serving as a market stabilization tool. On each trading day, no order can be placed or traded at prices outside the ±10% range of its previous closing price. This restriction might affect our strategy by making stocks at

[21]For special treatment stocks, the limit is 5%.

limits more difficult to trade, as more aggressive orders cannot be placed any more. Since all trades happen at the same limit prices in a long queue, only a fraction of orders in the stock might be eventually executed. In addition, such a mechanism might affect price discovery in several ways (Chen et al., 2019). On one hand, stocks prices failing to reach their fair values due to the limit might continue to move in the same direction on the next day. On the other hand, it is widely believed by Chinese media and investors that some limits are artificially hit by speculators for price manipulation purposes to lure people to buy and prices will revert the next day.

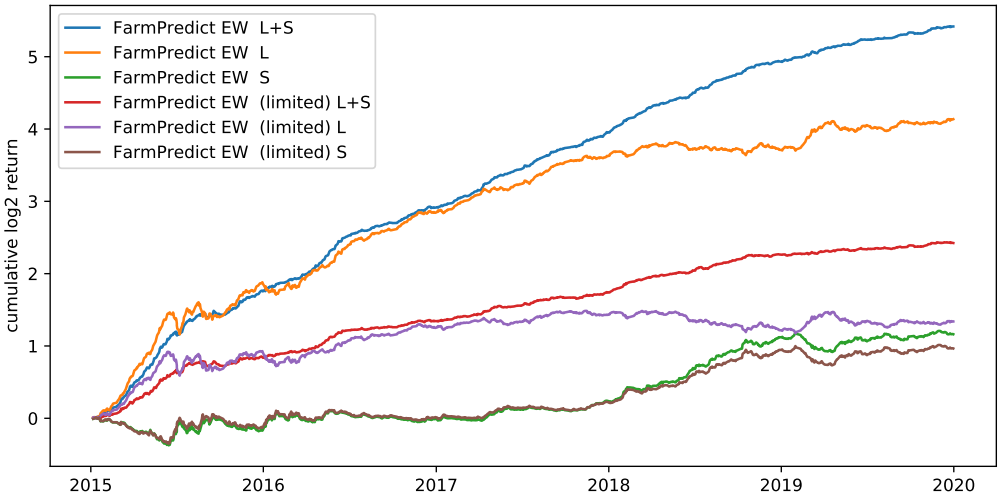


Figure 9: Comparison of cumulative log2 returns with and without daily price limit in place. Liquidity of a stock is usually lower if its price reaches daily price limit and here we assumed the extreme case where liquidity becomes strictly zero.

To verify our sentiment signals under such constraints, we tested the equally-weighted portfolio with daily price limit in place. We used the most restrictive version, assuming absolutely no liquidity if a stock reaches its daily limit. More specifically, each day we first filter out stocks that ended up at limit price, then build equally-weighted portfolios with the top 50 and bottom 50 of the remaining stocks.

Results are shown in Figure 9, where we compared cumulative returns with and without the constraints in place. While a bit lower than the unconstrained case, portfolio under limit constraints still attained a fairly nice performance of 41.2% APR with 4.74 Sharpe ratio from 2015 to 2019. From the figure we can see that the largest decrease in returns come

from the long leg, which results in 21.0% APR and is only 26% of the APR of the leg in the unconstrained case. The short leg performed equally as well as the unconstrained except for 2019. Short-term variations in both legs still cancel out with each other nicely.

4.3 Sensitivity Tests

Some alternative details of FarmPredict model are studied here as sensitivity tests. We started with the input of the model and tried some variations of $\mathbf{X}$ and Y other than dichotomized word counts and beta-adjusted return. We then studied the impacts of different choices on the number of factors in terms of portfolio returns and concluded by examining the influence of the choice of number stocks in portfolio constructions on the investment results.

4.3.1 Input forms of X and Y

In FarmPredict we used the dichotomized word counts as $\mathbf{X}$ and beta-adjusted equity returns as $\mathbf{Y}$ in fitting the model. Thanks to the flexible design of the FarmPredict model, other forms of inputs are supported. Here we present results on other variations of feature extraction of data.

For $\mathbf{X}$ we tested either its normalized word count version or dichotomized version. In a normalized word count $\mathbf{X}$, each article's word counts are scaled by their sum to ensure they add up to the same amount in each article. In $\mathbf{Y}$, dichotomized beta-adjusted return takes value in $\{0, 1\}$ depending on whether beta-adjusted return is positive or negative. The raw returns and its dichotomized version are also tested as $\mathbf{Y}$ for completeness. Dichotomized word counts and beta-adjusted returns are the choice of combination presented in FarmPredict so far.

Annualized returns and Sharpe ratios of each combination of $(\mathbf{X}, \mathbf{Y})$ are summarized in Table 10. All methods showed very nice returns. In general, models trained using dichotomized returns tend to perform worse than their counterparts. One explanation might be that by resorting to binary encoding of Y, some information on the strength of text signals is lost. On the other hand, models fitted with raw returns tend to have similar APRs

Table 10: APR and Sharpe ratios of portfolios with different combinations of input forms.

Form of $\mathbf{Y}$	raw $\mathbf{X}$ (normalized)	dichotomized $\mathbf{X}$
beta-adjusted return	105% (6.9)	116% (9.3)
beta-adjusted return (dichotomized)	71% (3.1)	78% (3.7)
raw return	111% (6.4)	112% (7.5)
raw return (dichotomized)	72% (3.0)	91% (3.7)

Note: Values in percentages are APR and in brackets are the portfolio's corresponding Sharpe ratio. Here we use $2 \times 2 \times 2$ combinations of $\mathbf{X}$ and $\mathbf{Y}$. The input $\mathbf{X}$ can either be normalized word counts or dichotomized count. The outcome $\mathbf{Y}$ can either be beta-adjusted returns or raw returns, as well as their dichotomized versions. In dichotomized Y version, FarmPredict is fitted by Logistic Regression with l_1 penalty instead of least-squares LASSO. In normalized $\mathbf{X}$, each row is scaled so the row sum of each article is 1. In general, models with dichotomized returns or raw (normalized) word counts tend to perform worse than their counterparts. Models fitted with raw returns tend to have slightly better returns and worse Sharpe ratios compared to their counterparts with beta-adjusted returns.

and worse Sharpe ratios compared to their counterparts with beta-adjusted returns. This is in line with our expectation as beta-adjusted returns serve as a less noisy representation of our signals, with market noise removed.

4.3.2 Number of factors

An important parameter in FarmPredict model is its number of factors k. We used adjusted eigenvalue thresholding for its estimation which resulted in an estimated $\widehat{k}$ of around 10. Instead of choosing the number of factors with adjusted eigenvalue thresholding, we also tested fitting the model with fixed a handpicked number of factors. Other hyper-parameters are kept at the same for each test. Their resulted cumulative returns in each year is plotted in Figure 10. We can see that, except for year 2018, no year showed an obvious trend of returns with respect to the number of factors selected.

In addition to our method, eigenvalue ratio test (Ahn and Horenstein, 2013) is another widely used method for choosing k in factor models. Let $\widetilde{\lambda}_i$ be the i-th largest eigenvalue of the sample covariance matrix of data $\mathbf{X}$. The method estimates k as $\widehat{k}^{\mathrm{ER}} = \mathrm{argmax}_{k \leq [p/2]} \widetilde{\lambda}_k / \widetilde{\lambda}_{k+1}$.

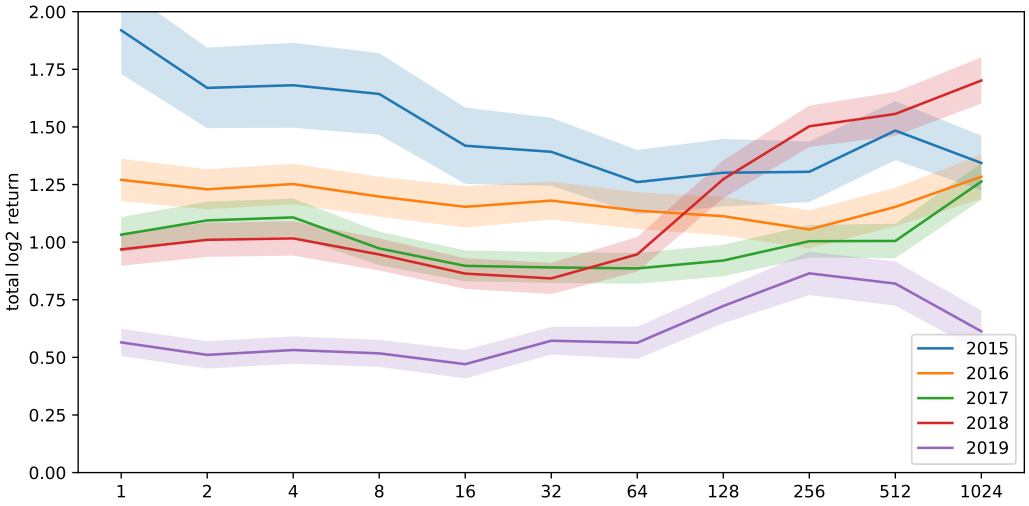
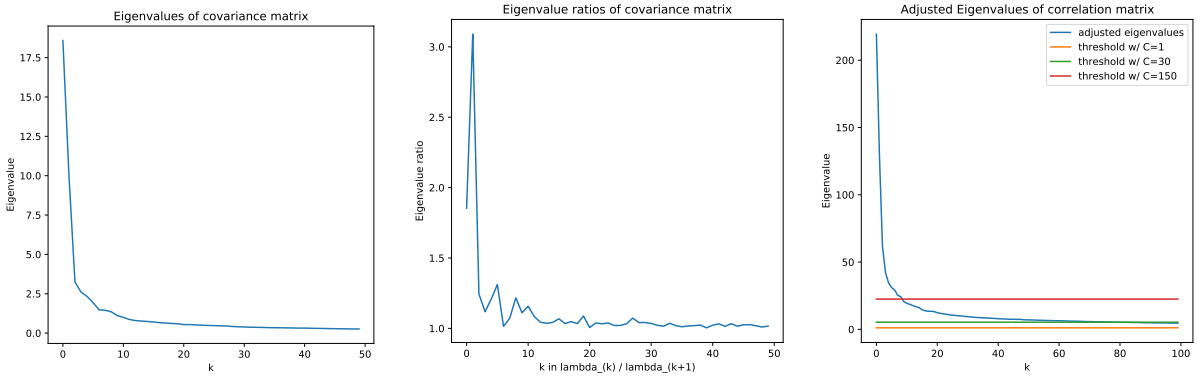


Figure 10: Portfolio's log2 returns in each year with respect to the choice of number of factors (ranging from 1 to 1024) using FarmPredict. The width of shades indicate the annualized standard deviation of returns each year.



(a) Top eigenvalues of covariance matrix

(b) Ratios of eigenvalues of covariance matrix

(c) Top adjusted eigenvalues of correlation matrix

Figure 11: Top eigenvalues of the covariance matrix, their ratios, and top adjusted eigenvalues of the correlation matrix based on dichotomized word counts in the last training window (July 2009 to June 2019). The ratio at $k = 2$ is significantly higher than others, so eigenvalue ratio test would choose 2 as the number of factors. Adjusted eigenvalue thresholding with $C = 1, 30, 150$ chooses 1043, 78 and 11 factors respectively.

The eigenvalue ratios for our last testing window is depicted in Figure 11. The ratio at $k = 2$ is significantly higher than other choices and the method chooses 2 as the number of

factors. Indeed, 2 is constantly chosen as the number of factors by eigenvalue ratio method in every window of our experiments. Their corresponding returns can thus be found in Figure 10 at the number of factor being 2. From the figure we see that the choice of 2 factors in ER shows no particular advantage or disadvantage in yearly returns compared to others. In comparison, adjusted eigenvalue thresholding in Figure 11 selects 11, 78 and 1043 factors with $C = 150, 30, 1$.

4.3.3 Number of stocks in each portfolio

The number of stocks in each portfolio plays an import role in our strategy, which involves trade-offs between signals and diversification, or returns and risks. For a portfolio with a larger number of stocks, the portfolio is more diversified and less risky. At the same time, since more weak predicted signals are selected and invested, portfolio returns are usually lower. In reality, rather than a plain equally-weighted portfolio, investors might as well decide each stock's weights based on their signal strength and liquidity.

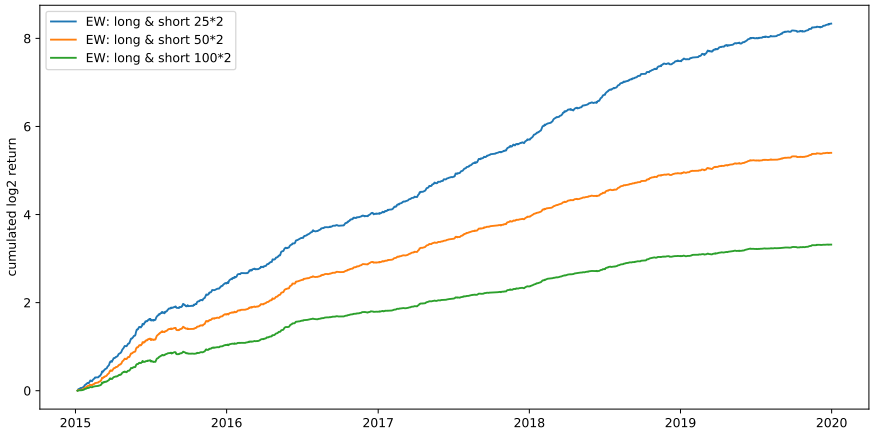


Figure 12: Cumulative portfolio returns of FarmPredict model with different number of stocks invested in each portfolio: 25, 50 and 100 on each long and short side. Each line in this figure represents the cumulative log2 returns over time with a given number of stocks on long and short legs.

In the main results, we fixed the portfolio size at 50 stocks each side, which is 1% total capital for each stock in equally-weighted portfolio. We have also tested the same model with different portfolio sizes. For the smaller portfolio, 25 stocks on each side is used,

corresponding to 2% capital exposure to each stock. The larger portfolio is set to 100 stocks each side and 0.5% capital exposure to each stock. Results are shown in Figure 12. The result is in line with our expectations that larger portfolios have weaker returns while smaller portfolios have a seemingly more bumpy but higher cumulative returns.

4.3.4 Full data

For the purpose of balancing data distribution across months and years, we down-sampled our dataset to at most 300 news articles a day. Such an operation reduced the total amount of data from 1791K to 914K. Most of the discarded data come from the year of 2019 (53%), year of 2007 (12%) and 2005 (7%). The distribution of data across each year is shown in Figure 13. The amount of data from 2019 is significantly larger than other years due to the randomness in data collection. The entire data collection is done in 2019, so the news from that year are more likely to be available (links to many old news are lost) and easier to be found (links being more likely to appear in other webpages).

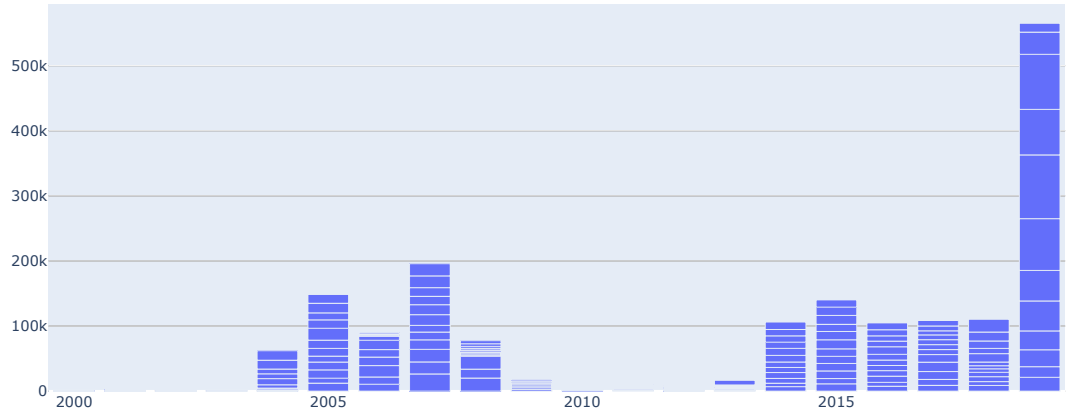


Figure 13: Amount of valid data each year in our dataset before subsampling. The amount of articles downloaded in 2019 is disproportionately larger than other years.

Differences between subsampled and full data are similar to standard bias variance trade-offs. On one hand, less balanced data might lead to biased models and thus biased predictions. On the other hand, a larger amount of data in training can reduce a model's variance, and more predictive news can be picked for investments. For completeness of the research, we also tested FarmPredict with the entire dataset and compared it with

previous models. Comparisons are plotted in Figure 14. Model fitted with the entire dataset outperformed in equally-weighted portfolio especially in 2019, likely because some profitable signals are omitted at the subsampling stage. Both models performed similarly well in value-weighted portfolios.

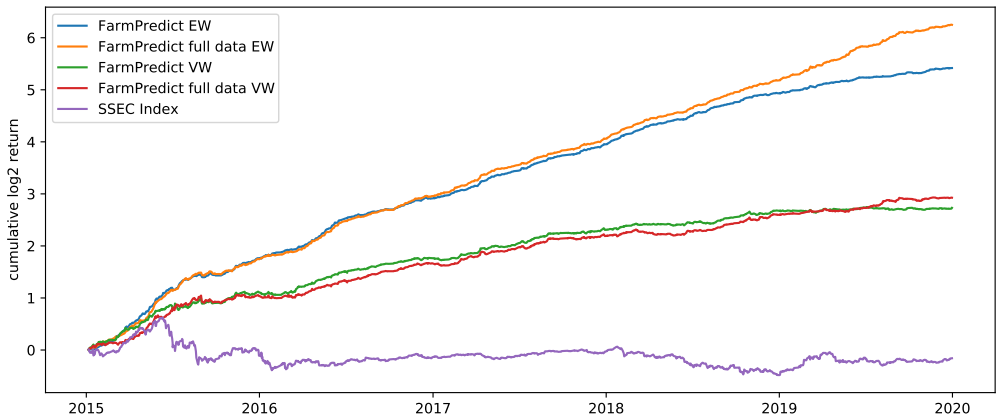


Figure 14: Cumulative log2 return of FarmPredict fitted with full data and subsampled data.

5 Conclusion

Previous studies on text data usually rely on a pre-defined dictionary and human's prior experience, resulting in a non-adaptive and incomplete capture of information. In contrast to these models, we proposed a novel analytical framework on a textual study that conducts unsupervised information extraction: FarmPredict first isolates the hidden factors and idiosyncratic components as a vector from high-dimensional text data via unsupervised learning, without the reliance on prior knowledge. Then we screen the idiosyncratic components, which are mainly constructed by sentiment-related words according to their correlations with corresponding beta-adjusted return conditional on hidden factors. This step is optional but helps reduces computational cost. Even though only a part of the words are selected, all information is used for screening due to embedded factors. In other words, FarmPredict transforms the high-dimensional data into important factors and useful idiosyncratic components; then uses them as the input for further penalized regression or other prediction models.

To demonstrate its applicability, we applied FarmPredict on news data to Chinese stock

market to verify our novel framework's effectiveness in several ways. These include analysis of selected words, the correlation between machine-learned sentiments and financial returns, and the returns of sentiment-based portfolios. The results prove that FarmPredict can extract useful information from an article, as exemplified by rarely selected words and phrases in previous studies. The empirical results emphasized that the sentiment scores from our model is a powerful predictor in asset pricing and revealed the mechanism of market response to related news. Finally, we used a simple trading strategy on portfolio construction to realize our model's advantage in textual analysis and prediction power, where our accumulated return outperforms other models.

FarmPredict can extract all information from text data by converting correlated high-dimensional data into weakly correlated ones in an unsupervised manner. Therefore, not only is it a novel model in the financial analysis, but our FarmPredict is also a general and adaptive supervised-learning framework for high-dimensional data, like text analysis in this paper, with flexibility on the choice of method in each process.

Acknowledgement

This study was supported by the key project of Chinese National Natural Science Foundation (No.71991471 and 71991470) and the China Postdoctoral Science Foundation funded project (No. 2019M650076, No. 2020T130107).

References

Seung C Ahn and Alex R Horenstein. Eigenvalue Ratio Test for the Number of Factors. *Econometrica*, 81(3):1203–1227, 2013. ISSN 0012-9682. doi: 10.3982/ECTA8968.

Werner Antweiler and Murray Z. Frank. Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards. *The Journal of Finance*, 59(3):1259–1294, 2004. ISSN 1540-6261. doi: 10.1111/j.1540-6261.2004.00662.x.

Jushan Bai and Serena Ng. Determining the number of factors in approximate factor models. *Econometrica*, 70(1):191–221, 2002.

Zhidong Bai and Xue Ding. Estimation of spiked eigenvalues in spiked models. *Random Matrices: Theory and Applications*, 1(02):1–21, 2012.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003. ISSN ISSN 1533-7928.

Charles W. Calomiris and Harry Mamaysky. How news and its context drive risk and returns around the world. *Journal of Financial Economics*, 133(2):299–336, August 2019. ISSN 0304-405X. doi: 10.1016/j.jfineco.2018.11.009.

Ting Chen, Zhenyu Gao, Jibao He, Wenxi Jiang, and Wei Xiong. Daily price limits and destructive market behavior. *Journal of Econometrics*, 208(1):249–264, January 2019. ISSN 03044076. doi: 10.1016/j.jeconom.2018.09.014.

Alfred Cowles. Can Stock Market Forecasters Forecast? *Econometrica*, 1(3):309–324, 1933. ISSN 0012-9682. doi: 10.2307/1907042.

Zhi Da, Joseph Engelberg, and Pengjie Gao. The Sum of All FEARS Investor Sentiment and Asset Prices. *Review of Financial Studies*, 28(1):1–32, January 2015. ISSN 0893-9454, 1465-7368. doi: 10.1093/rfs/hhu072.

Ke Deng, Peter K. Bol, Kate J. Li, and Jun S. Liu. On the unsupervised analysis of domain-specific Chinese texts. *Proceedings of the National Academy of Sciences*, 113(22): 6154–6159, May 2016. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1516510113.

Jianqing Fan and Jinchi Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5): 849–911, 2008. ISSN 1467-9868. doi: 10.1111/j.1467-9868.2008.00674.x.

Jianqing Fan, Jianhua Guo, and Shurong Zheng. Estimating Number of Factors by Adjusted Eigenvalues Thresholding. *Journal of the American Statistical Association*, pages 1–33, September 2020a. ISSN 0162-1459, 1537-274X. doi: 10.1080/01621459.2020.1825448.

Jianqing Fan, Yuan Ke, and Kaizheng Wang. Factor-adjusted regularized model selection. *Journal of Econometrics*, 216(1):71–85, May 2020b. ISSN 03044076. doi: 10.1016/j.jeconom.2020.01.006.

Jianqing Fan, Runze Li, Cun-Hui Zhang, and Hui Zou. *Statistical foundations of data science*. CRC press, 2020c.

Zhenyu Gao, Haohan Ren, and Bohui Zhang. Googling Investor Sentiment around the World. *Journal of Financial and Quantitative Analysis*, 55(2):549–580, March 2020. ISSN 0022-1090, 1756-6916. doi: 10.1017/S0022109019000061.

Diego García. Sentiment during Recessions. *The Journal of Finance*, 68(3):1267–1300, 2013. ISSN 1540-6261. doi: 10.1111/jofi.12027.

Matthew Gentzkow, Bryan Kelly, and Matt Taddy. Text as Data. *Journal of Economic Literature*, 57(3):535–574, September 2019a. ISSN 0022-0515. doi: 10.1257/jel.20181020.

Matthew Gentzkow, Jesse M. Shapiro, and Matt Taddy. Measuring Group Differences in High-Dimensional Choices: Method and Application to Congressional Speech. *Econometrica*, 87(4):1307–1340, 2019b. ISSN 1468-0262. doi: 10.3982/ECTA16566.

Paul Glasserman and Harry Mamaysky. Does Unusual News Forecast Market Stress? *Journal of Financial and Quantitative Analysis*, 54(5):1937–1974, October 2019. ISSN 0022-1090, 1756-6916. doi: 10.1017/S0022109019000127.

Shihao Gu, Bryan Kelly, and Dacheng Xiu. Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies*, 33(5):2223–2273, May 2020. ISSN 0893-9454, 1465-7368. doi: 10.1093/rfs/hhaa009.

Elaine Henry. Are Investors Influenced By How Earnings Press Releases Are Written? *The Journal of Business Communication (1973)*, 45(4):363–407, October 2008. ISSN 0021-9436. doi: 10.1177/0021943608319388.

Enguerrand Horel and Kay Giesecke. Significance tests for neural networks. *Journal of Machine Learning Research*, 21(227):1–29, 2020.

Narasimhan Jegadeesh and Di Wu. Word power: A new approach for content analysis. *Journal of Financial Economics*, 110(3):712–729, December 2013. ISSN 0304-405X. doi: 10.1016/j.jfineco.2013.08.018.

Zheng Tracy Ke, Bryan T. Kelly, and Dacheng Xiu. Predicting Returns with Text Data. SSRN Scholarly Paper ID 3389884, Social Science Research Network, Rochester, NY, July 2019.

Tim Loughran and Bill Mcdonald. When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, 66(1):35–65, 2011. ISSN 1540-6261. doi: 10.1111/j.1540-6261.2010.01625.x.

Tim Loughran and Bill Mcdonald. Textual Analysis in Accounting and Finance: A Survey. *Journal of Accounting Research*, 54(4):1187–1230, 2016. ISSN 1475-679X. doi: 10.1111/1475-679X.12123.

Asaf Manela and Alan Moreira. News implied volatility and disaster concerns. *Journal of Financial Economics*, 123(1):137–162, January 2017. ISSN 0304-405X. doi: 10.1016/j.jfineco.2016.01.032.

Stefan Nagel. Short sales, institutional investors and the cross-section of stock returns. *Journal of financial economics*, 78(2):277–309, 2005.

Stefan Nagel. *Machine Learning in Asset Pricing*. Princeton University press, 2021.

James H Stock and Mark W Watson. Forecasting using principal components from a large number of predictors. *Journal of the American statistical association*, 97(460):1167–1179, 2002.

Junyi Sun. jieba v0.39. `https://github.com/fxsjy/jieba`, 2017.

Licheng Sun, Mohammad Najand, and Jiancheng Shen. Stock return predictability and investor sentiment: A high-frequency perspective. *Journal of Banking & Finance*, 73:147–164, December 2016. ISSN 0378-4266. doi: 10.1016/j.jbankfin.2016.09.010.

Paul C. Tetlock. Giving Content to Investor Sentiment: The Role of Media in the Stock Market. *The Journal of Finance*, 62(3):1139–1168, 2007. ISSN 1540-6261. doi: 10.1111/j.1540-6261.2007.01232.x.

Paul C. Tetlock, Maytal Saar-Tsechansky, and Sofus Macskassy. More Than Words: Quantifying Language to Measure Firms' Fundamentals. *The Journal of Finance*, 63(3):1437–1467, 2008. ISSN 1540-6261. doi: 10.1111/j.1540-6261.2008.01362.x.