Stock Returns Prediction with FarmPredict: Empirical Study on Chinese Text and Stocks

Man Yin Michael Yeung

supervised by Prof. Haifeng You

to fulfil the requirement of UROP2100F under
Undergraduate Research Opportunities Program on the project
"Investment Analysis with Machine Learning"
The Hong Kong University of Science and Technology

May 28, 2022

Abstract

The advent of machine learning facilitated the extraction of sentiment from financial text corpus. Recent literatures have introduced state-of-the-art models to make stock return predictions. In this study, replication of FarmPredict model by Fan et al. (2021) is implemented to examine its ability to mine sentiment from Chinese analyst reports and make Chinese stock returns predictions. The framework implemented with grid-search and cross-validation can reach optimal test Spearman correlation of 11.7% comparable to that found in the SESTM framework (11.8%) in previous work (Yeung, 2021). Variant of the framework with input not transformed to word embeddings have better performance. The magnitude of correlation provides evidence that FarmPredict can generate predictions useful for investment analysis but to a similarly limited degree.

Keywords: Machine Learning, Factor Model, Sentiment Scores, Sparse Regression, Textual Analysis, Regularization, Chinese Stocks

1. Introduction

In the advent of machine learning, it is becoming more and more feasible to exploit large datasets. Textural data is less structured than numerical data and is often considered high dimensional when represented numerically. The grow of machine learning especially facilitated the mining on it. However, as the numerical representation matrices of textual data are often large and sparse, it is still computationally challenging to unleash its potential.

Sentiment in financial texts refers to emotions or polarity of the author, whether the author is having a positive, negative, or sometimes neutral emotion states. Thus, if one can interpret the sentiment be financial text correctly, the person can obtain the author's view on the topic. For example, one can obtain the opinion polarity of a financial analyst by analysing a financial report. If one can instead collect a large number of similar reports, he/she can obtain the views of a large number of analysts and in different time periods. That will be useful to estimate the generally view on a certain asset universe in different time points, which may be useful for investment analysis.

Fan et al. (2021) presents Factor-Augmented Regularized Model for Prediction (FarmPredict), a framework for learning text data based on the factor model and sparsity regularization. Different from dictionary-based or topic models such as SESTM by Ke et al. (2020), FarmPredict does not have a stringent pre-screening process. Instead, it allows the model to extract information from the whole article. FarmPredict is replicated in this study to examine the ability of it to generate sentiment predictions based on Chinese analyst reports. We additionally implemented a modified version which use word embeddings instead of word frequencies as the model input for comparison.

The rest of this paper is organized as follows. Section 2 reviews the literature by Fan et al. (2021), which describes the FarmPredict in a more detailed manner. Section 3 presents methodologies, including the different models adopted in this study

- the original and modified FarmPredict. Section 4 includes the empirical results obtained by implementing the methodologies using Python. Sections 5 and 6 contains the discussions, limitations of this study, and the conclusion made.

2. Literature Review: FarmPredict

Fan et al. (2021) suggests FarmPredict framework for learning text data based on the factor model and sparsity regularization. The FarmPredict framework is summarized in the following subsections.

2.1. Problem Setup

The bag of words for each of the n articles are considered. D represents all possible Chinese words and vector $d_i \in \mathbb{N}^{|D|}$ represent the word counts of article i, with $d_{i,k}$ being the number of k-th word appeared. Each article composes of several underlying topics with own preferred vocabulary; hence an assumption is made that an article's word count vector is influenced by a small number of latent factors or topics such as simply positive versus negative.

Each article is associated with a response Y_i being the beta-adjusted return of the associated stock in article i on the day of article publishment. The response is affect by a relatively small subset of words called sentiment-charged words (set S), reducing dimensionality. The remaining words are called sentiment-neural (set N). The two sets are disjoint, with $D = S \cup N$.

2.2. Words Screening and Factor Modelling

The words are filtered according to their frequency. Only frequent words are considered. Let k_j be the number of articles containing word j, keep vocabulary with threshold κ.

$$\mathrm{D^{freq}} = \{\mathrm{j-th\ word\ in\ D: k_j} \geq \kappa\}$$

The hyperparameter κ will be tuned to balance the comprehensiveness of D^{freq} and the noise produced by infrequent words.

2.3. Learning Factors and Components

Let X_i represent the feature vector with $X_{i,j}$ being the feature of word $j \in \mathrm{D}^{\mathrm{freq}}$ in the i-th article. The features can be word counts or {0,1} representing presence of words. The dependence among words is summed to be driven by some latent factors $f_i \in \mathbb{R}^k$. B is factor loading matrix and $u_i \in \mathbb{R}^{|D^{freq}|}$ is a vector of idiosyncratic components uncorrelated with f. The approximate factor model and its matrix form:

$$\mathrm{X_i = Bf_i + u_i \quad i = 1, \ldots, n}$$

$$\mathrm{X = FB^T + U}$$

where X and U are $n \times |\mathrm{D}^{\mathrm{freq}}|$ matrices and F is $n \times k$ of latent factors. Only X is observable and F, B, U will be estimated by principal components analysis. With given k, the factor model is fit via-least squares, resulting in principle components analysis. The solution is that

$$\hat{\mathrm{F}} = \sqrt{\mathrm{n}} \times \text{eigenvectors of largest k eigenvalues of } \mathrm{XX^T}$$

$$\hat{\mathrm{B}} = \mathrm{X^T}\hat{\mathrm{F}}/\mathrm{n} \quad \text{and} \quad \hat{\mathrm{U}} = \mathrm{X} - \hat{\mathrm{F}}\hat{\mathrm{B}}^{\mathrm{T}}$$

Let $\hat{Y}_u$ be the residual vector of Y after fitting a linear regression of Y on $\hat{F}$ with intercepts. Given a threshold α, the conditional sentiment charged words are defined

$$\hat{\mathrm{S}} = \{\mathrm{j}: |\mathrm{corr}(\hat{\mathrm{U}}_{\mathrm{j}}, \mathrm{Y_u})| > \alpha\} \cap \{\mathrm{j}: \mathrm{k_j} \geq \kappa\}$$

2.4. FarmPredict Fitting and Scoring New Articles

FarmPredict solves the penalized least squares

$$\hat{\mathrm{a}}, \hat{\mathrm{b}}, \hat{\beta} = \mathrm{argmin}_{\hat{\mathrm{a}}, \hat{\mathrm{b}}, \hat{\beta}}\{\frac{1}{\mathrm{n}}\sum_{\mathrm{i}}(\mathrm{Y_i - a - b^T f_i} - \beta^{\mathrm{T}}\mathrm{u}_{\mathrm{i},\hat{\mathrm{S}}})^2 + \lambda||\beta||_1\}$$

with λ chosen by cross-validation controlling bias-variance trade-off and sparsity of β. This further reduces sentiment-charged words. Score of new articles are predicted as

$$\mathrm{f_{new}} = (\hat{\mathrm{B}}^{\mathrm{T}}\hat{\mathrm{B}})^{-1}\hat{\mathrm{B}}^{\mathrm{T}}\mathrm{X_{new}} \quad \text{and} \quad \mathrm{u_{new}} = \mathrm{X_{new}} - \hat{\mathrm{B}}\hat{\mathrm{f}}_{\mathrm{new}}$$

$$\hat{\mathrm{Y}}_{\mathrm{new}} = \hat{\mathrm{a}} + \hat{\mathrm{b}}^{\mathrm{T}}\mathrm{f_{new}} + \hat{\beta}^{\mathrm{T}}\mathrm{u}_{\mathrm{new},\hat{\mathrm{S}}}$$

3. Methods

We implement FarmPredict with two versions of modifications (sections 3.1 and 3.2 respectively). All raw Chinese texts are first cleaned and cut by Jieba into vocabularies (c.f. tokenization in English), with part-of-speech tags, prior to the subsections below. Different from Fan et al. (2021), we only include adjectives and verbs in our analysis. Other words are removed. The data is split into training set and testing set.

3.1. Model 1: FarmPredict

To start with, we implement the following for words screening

$$\mathrm{D}^{\mathrm{freq}} = \{\eta \text{ most frequent words in D}\}$$

We then perform principal components analysis (PCA) as formulated in section 2.3. The eigenvectors of XX^T can indeed done by singular value decomposition on X^TX which is less computationally costly. For the number of eigenvalues chosen, we consider a certain proportion of the number of words used (i.e., η). We denote by ε the proportion of eigenvalues used in PCA, hence having $\lfloor \varepsilon \cdot \eta \rfloor$ principal components.

The remaining parts follow the FarmPredict model. Note that the four hyperparameters, including η, ε, α (section 2.3), and λ (section 2.4)[1], are tuned via a five-fold cross validation which maximizes the validation Spearman correlation.

3.2. Model 2: FarmPredict with Word Embeddings as Input

For this version, the words in pre-processed text (adjectives and verbs) are transformed into their 100-dimensional embeddings[2] and then averaged at document level. These forms the 100 features as the input. We continue similar to section 3.1 but without the words screening step at the beginning. The 100 features are passed to PCA and all later steps. The three latter hyperparameters are tuned via cross validation similarly.

---

[1] We consider $\eta \in \{500, 1000, 5000\}$, $\varepsilon \in \{0.01, 0.05, 0.1\}$, $\alpha \in \{0.0005, 0.001, 0.005\}$, and $\lambda \in \{0, 0.000005, 0.00001, 0.00005, 0.0001, 0.0005, 0.001, 0.005\}$ in the grid search by cross validation.

[2] We use Tencent AI Lab Embedding Corpus for Chinese Words and Phrases (Song et al., 2018). Available: https://ai.tencent.com/ailab/nlp/en/embedding.html

4. Empirical Analysis

4.1. Dataset

Raw datasets of analyst reports and stock prices are pre-processed in a previous study by Yeung (2021) earlier in the same undergraduate research project. The 472,693 entries of processed text augmented with specific returns columns are used. In particular, the two-day return is used as the label in this empirical study. The training data refers to the 180,353 entries dated before 2015-01-01, and the testing data refers to the 292,340 entries after the date.

4.2. Performances

As shown in Exhibits 1 and 2, we observe that the actual and predicted returns have a rather clear positive relationship. However, the distributions of actual returns are widened as we observe Q-cuts (quantile-based discretization) of higher predict returns.

Exhibit 1 Model 1: Actual against Predicted Returns, Training & Testing Sets

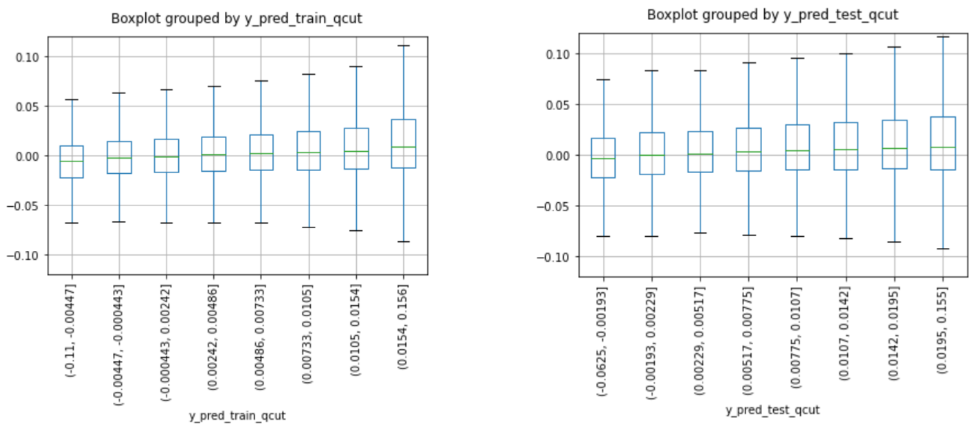


Exhibit 2 Model 2: Actual against Predicted Returns, Training & Testing Sets

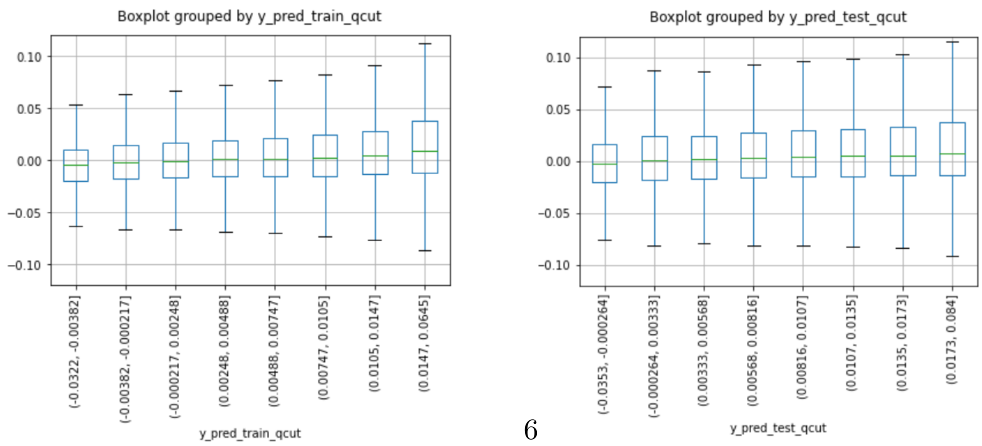
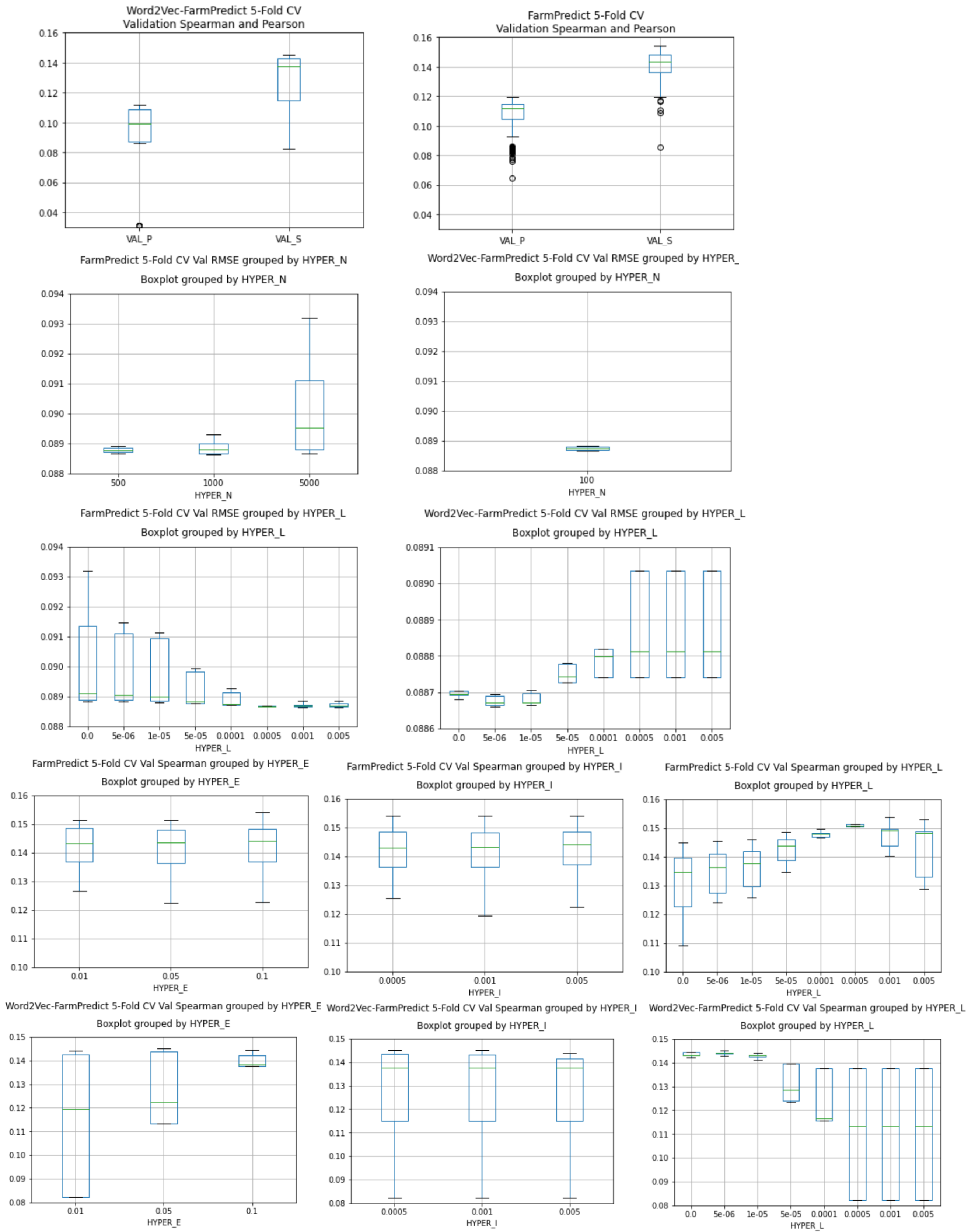


Exhibit 3 Test Correlations, and RMSE (grouped), Comparison[3]



[3] In this exhibit, FarmPredict and Word2Vec-FarmPredict refers to Models 1 and 2 respectively. Hyperparamteres (HYPER) N, E, I, and L refers to $\eta, \varepsilon, \alpha, and\ \lambda$ respectively. Validation stated on the figures is referring to testing in this exhibit.

Exhibit 3 compares the performances of the two models and in terms of their testing correlations and RMSE. Patterns and trends are observed from the boxplots implying that the change in hyperparameters do have impact on the performance.

Exhibit 4 shows the correlations between predictions and labels in the testing set with the optimal hyperparameters[4]. The upper table shows Spearman correlation, and the lower table shows Pearson correlation[5]. The result shows 11.7% and 10.1% test Spearman correlation for Models 1 and 2 respectively[6]. Model 1 performs better in general. The predictions from both models are more than 70% correlated. The systematic components are much more correlated to the label than the idiosyncratic component for Model 1, but for Model 2 the idiosyncratic components are more correlated to the label.

Exhibit 4 Spearman/Pearson Correlations between Predictions and Label

	specret_2d	y_pred	y_pred_sys	y_pred_ido	w2v_y_pred	w2v_y_pred_sys	w2v_y_pred_ido
specret_2d	1.000000	0.116604	0.118163	-0.007696	0.098404	0.059815	0.084630
y_pred	0.116604	1.000000	0.981207	-0.149254	0.721859	0.563906	0.462593
y_pred_sys	0.118163	0.981207	1.000000	-0.258330	0.730404	0.568655	0.470155
y_pred_ido	-0.007696	-0.149254	-0.258330	1.000000	-0.240178	-0.213300	-0.126411
w2v_y_pred	0.098404	0.721859	0.730404	-0.240178	1.000000	0.785381	0.613710
w2v_y_pred_sys	0.059815	0.563906	0.568655	-0.213300	0.785381	1.000000	0.040873
w2v_y_pred_ido	0.084630	0.462593	0.470155	-0.126411	0.613710	0.040873	1.000000

	specret_2d	y_pred	y_pred_sys	y_pred_ido	w2v_y_pred	w2v_y_pred_sys	w2v_y_pred_ido
specret_2d	1.000000	0.100533	0.102946	0.009791	0.090553	0.062574	0.069662
y_pred	0.100533	1.000000	0.964276	0.311633	0.697297	0.555956	0.443056
y_pred_sys	0.102946	0.964276	1.000000	0.048794	0.716001	0.567977	0.458584
y_pred_ido	0.009791	0.311633	0.048794	1.000000	0.060855	0.058894	0.025598
w2v_y_pred	0.090553	0.697297	0.716001	0.060855	1.000000	0.792940	0.640887
w2v_y_pred_sys	0.062574	0.555956	0.567977	0.058894	0.792940	1.000000	0.040465
w2v_y_pred_ido	0.069662	0.443056	0.458584	0.025598	0.640887	0.040465	1.000000

[4] Model 1: $\eta = 1000, \varepsilon = 0.1, \alpha = 0.005, \lambda = 0.0005$. Model 2: $\varepsilon = 0.05, \alpha = 0.0005, \lambda = 0.000005$.
[5] The columns are (from left to right): two-day specific return (label), predicted return by Model 1, the systematic component of predicted return by Model 1, the idiosyncratic component of predicted return by Model 1, predicted return by Model 2, the systematic component of predicted return by Model 2, and the idiosyncratic component of predicted return by Model 2.
[6] The training Spearman/Pearson correlations are 16.5%/13.3% and 15.1%/9.9% for the two models.

Exhibit 5 Idiosyncratic Components (words) of Optimal Models 1 and 2

Model 1:

beta	word
0.005655	承诺v

Model 2:

增长v 促v 组合v 调v 短a 饱满v 组建v 存v 分为v 延迟v 侵蚀v 搭建v 改进v 抢占v 提振v 延长v 固定a 认识v 举措v 跨v 成v 利息支出v 涵盖v 略降v
供v 包装v 治疗v 注册v 结转v 取决于v 停产v 加v 位居v 接受v 简单a 不错a 减排v 集成v 尚有v 宣布v 回调v 庞大a 调低v 休闲v 很好a 扩容v 趋缓v
安排v 横向v 收获v 均衡a 赢v 静待v 避免v 来临v 决心v 不可v 得出v 鼓励v 理财v 做到v 解禁v 上限v 通用v 寻找v 悲观a 吸引v 较弱a 审核v 向下v
抓住v 配股v 纯a 批复v 催化v 深a 拿到v 接待v 增多v 引起v 差a 收缩v 创造v 实行v 纳入v 受制于v 汇兑v 拉低v 持v 减持v 下沉v 趋于v 抵消v 送v
回收v 扩充v 达成v 推迟v 负责v 坚定a 变更v 高效a 切入v 聚焦v 扶持v 疲软a 计入v 类似v 滞后v 重a 验证v 表示v 小于v 包含v 构建v 强烈a 是否v
把握v 组织v 优异a 最好a 纪要v 争取v 独特a 装修v 属v 费v 注意v 搬迁v 共有v 沟通v 承压v 审议v 转化v 增至v 回报v 出货v 适当a 定制v 兼并v 相
信v 回顾v 落后a 探索v 开启v 独立v 站v 承担v 拟向v 可控v 萎缩v 修复v 前列v 反转v 较慢a 召开v 注重v 授予v 面对v 授权v 有待v 简称v 开设v 引
导v 单纯a 采v 协调v 复杂a 规避v 终止v 延期v 徘徊v 品v 诊断v 投v 团购v 冷a 起来v 录得v 饱和a 脱v 环v 更换v 增利v 摆脱v 发放v 敏感a 封装v
只能v 报道v 调高v 折v 追求v 垂直v 减v 倾向v 比高v 跟随v 贬值v 平滑a 预v 详见a 返还v 给与v 并入v 收取v 予以v 放弃v 受累v 集合v 负担v 筹备
v 呈现出v 入v 增产v 回v 遇到v 补v 契合v 分离v 位列v 已现v 接v 组成v 作v 深厚a 征收v 扎实a 脱硫v 未有v 不够v 放在v 度过v 摊v 热销v 尚待v
出租v 认同v 设v 红a 流出v 叠加v 理解v 有别于v 签v 无忧v 批发v 例v 崛起v 震荡v 便宜a 辐射v 保v 转换v 停牌v 精密a 增收v 特殊a 容易a 上网v
置换v 保费v 成交v 受让v 减轻v 吸收v 推v 简报v 夯实v 打破v 承销v 靓丽a 冶炼v 走低v 通知v 托管v 募投v 有着v 回购v 看出v 行为v 分成v 受阻v
角度看v 满a 加息v 转正v 占有v 规模较v 令v 隐含v 首选v 充沛a 衍生v 位v 带v 停止v 相结合v 审计v 做强v 退税v 稀释v 平淡a 可售v 验收v 备考v
适度a 缩短v 发达v 关闭v 过于v 最强a 微降v 共享v 广a 理顺v 纵向v 略低v 收紧v 占到v 营收v 维修v 认购v 佳a 统计v 平稳a 恢复v 保障v 能够v 支
撑v 值得v 谨慎a 集中v 复合v 增强v 巨大a 接近v 支持v 采购v 丰富a 率v 为主v 并购v 控股v 薄a 取得v 面临v 拟v 下调v 期待v 显示v 增发v 没有v
后续v 相比v 推广v 发行v 加强v 有利于v 持有v 推出v 成功a 确认v 造成v 用于v 拖累v 成熟a 广阔a 有助于v 重大a 加上v 改造v 制造v 达产v 应收v
结合v 提价v 预告v 建立v 支出v 解决v 延伸v 超出v 使用v 好转v 注入v 估计v 增厚v 应用v 保证v 合同a 完善v 募集v 有限a 要求v 成立v 回落v 覆盖
v 出v 迎来v 需v 打造v 加快v 设立v 增v 关注v 产能v 带来v 完成v 提高v 保持v 可能v 导致v 较大a 符合v 进行v 超过v 达v 处于v 扩张v 超v 受益v
推荐v 归属v 给予v 继续v 收入v 实现v 提升v 下降v 新a 提示v 增速v 需求v 高a 认为v 增加v 低于v 占v 达到v 考虑v 上升v 有望v 受v 减少v 收购v
显著a 释放v 降低v 平均a 扩大v 拥有v 相关v 整合v 形成v 重要a 表现v 上调v 计算v 放缓v 强a 最大a 高于v 推动v 开发v 投入v 拓展v 稳定a 看好v
出现v 发布v 新增v 具有v 上涨v 获得v 良好a 加大v 公布v 推进v 包括v 成长v 控制v 提供v 净a 突出v 很大a 采用v 不利a 引入v 摊销v 新高v 平安a
引进v 下属v 说明v 平衡a 复制v 等待v 结束v 拟以v 来源于v 发v 签约v 低估v 剩余v 共计v 了解v 选择v 涉及v 增资v 重新a 参考v 回款v 转移v 通道
v 应该v 上行v 得以v 交付v 坚持v 深入v 确保v 超越v 不到v 开支v 先进a 加权v 完工v 看到v 希望v 制定v 介入v 走v 排名v 紧张a 退出v 致使v 预增
v 齐升v 围绕v 逐v 确立v 导读v 升值v 彰显v 可达v 增值v 申请v 刺激v 联合v 观察v 未能v 超市v 落实a 完整a 扩建v 摘要v 分布v 过剩v 窄a 展开v
有利a 依赖v 应对v 展望v 从事v 租赁v 发挥v 奠定v 节能v 强劲a 连续a 满足v 支付v 乐观a 折合v 描述v 减值v 定v 持股v 收v 购买v 运行v 应v 发生
v 参与v 执行v 扣除v 回暖v 作用v 建成v 披露v 位于v 激烈a 反映v 截止v 合并v 准备v 不同a 在建v 全a 充足a 发现v 远a 属于v 参股v 手续v 降v
转让v 决定v 审慎a 不能v 分红v 依靠v 优秀a 健康a 折旧v 锁定v 转变v 占据v 较低a 概述v 限制v 可比v 获批v 清晰a 已有v 收到v 巩固v 出台v 处理
v 获取v 相当于v 剔除v 长a 少a 开v 保守v 表明v 提出v 简评v 估算v 依托v 强化v 难a 构成v 源于v 限购v

Model 1 with optimal hyperparameters have only one idiosyncratic component after the feature selection by LASSO. Model 2 has much more idiosyncratic components as shown in Exhibit 5.

5. Discussions

Models 1 and 2 in this study are based on the same set of words. However, Model 2 first transform words to 100-dimensional vectors and then perform PCA. We can expect a considerable amount of information loss hence lowered variance explained. Model we are considering higher dimensions. For example, Model 1 with optimization hyperparameters include 100 principal components compared to Model 2 with only 5 principal components. That may also be an explanation that Model 2 has many more idiosyncratic components as variance are less explained by the only 5 latent factors. The residual from the linear regression can still be explained by some words (idiosyncratic components) to certain considerable degrees.

The idiosyncratic words found in Model 2 as shown in Exhibit 5 are intuitive. Some words such as 增長 (meaning "growing" or "increasing") and 下沉 (meaning "sinking" or "going down") can be intuitively or by etymological meaning regarded as positive and negative sentiment-charged words.

6. Limitations and Conclusion

There are some limitations in this study.

1. This study focuses on the bag-of-words approach to deal with the phrases cut by Jieba. The intra-sentence POS relations are not investigated nor modelled. For example, some negation words such as "not" inverts the sentiment of the later words in some cases. Moreover, some parts of the article should be more important than others. Intuitively, the conclusion or summary/abstract parts of the article should have paid more emphasis on. (Yeung, 2021)
2. Only one regularized linear model (LASSO) is considered in our analysis. The study could potentially include and compare different machine learning models with regularization, including non-linear models which as more flexible.
3. We only considered one type of word embeddings. Also, the word embedding was not trained with emphasis on financial texts. Other embeddings training by financial text could replace the current one to seek enhanced performances.

Despite the limitations aforementioned, a conclusion can be made according to the empirical results of this study.

The best model of the FarmPredict framework implemented in this study is able to obtain a 11.7% Spearman correlation in the testing set having Chinese analyst reports from 2015 to mid-2021. This provides evidence to support the statement that sentiment extraction of Chinese analyst reports is useful to predict returns of Chinese stocks to a certain considerable degree. The model variation which directly use word frequencies instead of 100-dimensional word embeddings performs better in our study, and systematic component (principal components or latent factors) of predictions is more correlated to the label than the idiosyncratic component. Compared to the implementation of the SESTM framework in previous work (Yeung, 2021) which has an optimal test correlation of 11.8%, the performances of the two frameworks are similar in terms of predictive power. As financial data is generally noisy, we consider this result to be rather meaningful while making investment decisions.

References

[1] Fan, J., Xue, L., and Zhou, Y. (2021). How Much Can Machines Learn Finance from Chinese Text Data? Available at SSRN: https://ssrn.com/abstract=3765862 or http://dx.doi.org/10.2139/ssrn.3765862

[2] Fan, J., Li, R., Zhang, C.-H., and Zou. H. (2020). Statistical foundations of data science. CRC press, 2020c.

[3] Ke, Z., Kelly, B. T., and Xiu, D. (2020). Predicting Returns with Text Data. University of Chicago, Becker Friedman Institute for Economics Working Paper No. 2019-69, Yale ICF Working Paper No. 2019-10, Chicago Booth Research Paper No. 20-37, Available at SSRN: https://ssrn.com/abstract=3389884 or http://dx.doi.org/10.2139/ssrn.3389884

[4] Song, Y., Shi, S., Li, J., and Zhang, H. (2018). Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings. NAACL 2018 (Short Paper).

[5] Yeung, M. Y. M. (2021). Stock Returns Prediction with Chinese Text Using Machine Learning. Final report for UROP1100E submitted to Hong Kong University of Science and Technology.