Replication Summary Report

-- replication of paper Ke et al.(2020) 'Predicting Returns with Text Data'

-- replicated by Zhongchen WANG (20745072)

2021.09.20 - 2021.12.08

Outline

1. Introduction

The project is aimed to replicate the Ke et al.(2020)'s paper 'Predicting Returns with Text Data'. The paper developed an algorithm to get the sentiment score for given articles, and the algorithm could be concluded in three main steps: first, screening the training samples and select the positive and negative words automatically based on the return label, then the paper assign prediction weights to the selected words via topic modeling, and finally the paper get the predictive score for the testing texts by using penalized likelihood method. For the convenience, we named the papers' algorithm as SESTM method according to the notation in the paper.

The work can be concluded in the following parts: First, summarize the whole idea and data processing procedure of the paper. Second, benchmark construction using dictionary based method. Third, replicate the paper following the three main step. Fourth, algorithm extension using ngram2vec method and DSG method. Fifth, evaluate the replication result and plot the performance. Last but not least, portfolio construction and back-testing.

The dataset is composed of a large number of subtracts of Chinese financial analysis reports of stocks in Chinese stock markets. For the final cleaned dataset, there are 472,188 articles in total matched with two return labels. The publish date of articles in the final data range from 2006-04-21 to 2021-09-09, and due to the relative less articles in the first several years, our training data range from 2010-01-01 to 2014-12-31 and testing data is the rest articles dated from 2015-01-01. Hence, the training dataset is composed of 180,034 articles, and testing data is fulfilled with 290,947 articles.

2. Paper replication summary and data pre-processing

2.1 paper algorithm summary

This part of work can refer to the appendix file '*Replication method.pdf*'.

2.2 data pre-processing

Before the replication, we need to do data cleansing and data pre-processing.

The steps are stated as follows:

1) Return labels construction. (two time window)
    a) Label one: [t:t+1] (the sum of 'specret' in time t and t+1)
    b) Label two: [t+2:t+6] (the sum of 'specret' from time t+2 to t+6)
2) Divide contexts into words
    a) Since we are using the Chinese texts, 'jieba' package is applied.
    b) Stop words are removed
    c) Words are tagged with POS (Parts of Speech)
3) Words refinement
    a) Punctuation is removed
4) Bag of words construction
    a) Dived bag of words into different word length
    b) Adopt three methods for the selection of bag of words
        i. method one: only select adj. and v. words (notation e.g.: av100 means 100 words tagged with a and v)
        ii. method two: delete different POS of words for different word length (notation e.g.: ex100)
        iii. method three: select all the words versus select the words whose word length >=2 (notation e.g.: ex100_2p)

3. Benchmark construction

3.1 Two dictionary are applied as benchmark:

Benchmark 1: https://github.com/dictionaries2020/SentimentDictionaries

Benchmark 2:

https://github.com/MengLingchao/Chinese_financial_sentiment_dictionary

3.2 Define 12 scores

- Score 1 = # positive words/# of words excluding stop words and symbols etc.

- Score 2 = (-1) * # negative words/# of words excluding stop words and symbols etc.

- Score 3 = Score 1+score 2

- Score 4-6 = change the denominator to (#positive+#negative words) in score 1-3.

- Score 7-9 = use # of sentences instead of # words in Score 1-3

- Score 10-12 = use # of sentences instead of # words in Score 4-6

3.3 Correlation calculation

The results are showed below:

Benchmark 1														
		label	score1	score2	score3	score4	score5	score6*	score7	score8	score9	score10	score11	score12
t:t+1	pearson	1.0000	0.0221	0.0745	0.0471	0.0728	0.0728	0.0728	0.0287	0.0250	0.0325	0.0291	0.0291	0.0291
	spearman	1.0000	0.0423	0.0896	0.0690	0.0908	0.0908	0.0908	0.0420	0.0272	0.0439	0.0334	0.0334	0.0334
t+2:t+6	pearson	1.0000	0.0028	0.0130	0.0073	0.0128	0.0128	0.0128	0.0061	0.0032	0.0062	0.0053	0.0053	0.0053
	spearman	1.0000	0.0150	0.0008	0.0129	0.0043	0.0043	0.0043	0.0085	0.0015	0.0071	0.0025	0.0025	0.0025
Benchmark 2														
		label	score1	score2	score3	score4	score5	score6	score7	score8	score9	score10	score11	score12
t:t+1	pearson	1.0000	0.0171	0.0562	0.0402	0.0557	0.0557	0.0557	0.0321	0.0290	0.0361	0.0336	0.0336	0.0336
	spearman	1.0000	0.0322	0.0711	0.0562	0.0712	0.0712	0.0712	0.0447	0.0344	0.0481	0.0422	0.0422	0.0422
t+2:t+6	pearson	1.0000	-0.0050	0.0103	0.0011	0.0080	0.0080	0.0080	0.0039	0.0024	0.0040	0.0042	0.0042	0.0042
	spearman	1.0000	0.0036	0.0010	0.0022	0.0014	0.0014	0.0014	0.0050	0.0000	0.0036	0.0011	0.0011	0.0011

* Picked one for the benchmark representative

We can conclude that score 6 is relatively high in both two dictionaries and hence score 6 is used as the benchmark for later performance evaluation and portfolio construction, noted as ‘corr_benchmarkscore6’.

In addition, the fixed dictionary of benchmark one is also implemented with the following SESTM method for the comparison, the result is noted as ‘corr_git’ in the later performance evaluation plot.

4. Replication

4.1 Step 1 -- Screening for Sentiment-Charged words

4.2 Step 2 -- Learning Sentiment Topics

4.3 Step 3 -- Scoring New Articles

This part is mainly about the coding, and follows exactly the same as the original paper. For the hyper-parameter selection, there’s not much parameter tuning since the difference is not significant and here’s the specific value for hyper-parameters:

- kappa: 94%

- alpha+,alpha-: set the value of two alphas such that the number of words in each positive and negative bag of words group is either 100 or 500

- lambda = 5

There are several variations for the SESTM method. In general, the variations can be classified into three main domains: (1) different number of sentiment charged words; (2) words with different POS tag; (3) words with different length.

The results is shown as follows:

Method variation	Pearson correlation		Spearman correlation	
	t:t+1	t+2:t+6	t:t+1	t+2:t+6
av100	0.081291	0.011405	0.089647	-0.006809
av100_2p	0.082798	0.014310	0.088064	-0.006157
av500	0.084137	0.011824	0.091250	-0.007669
av500_2p*	0.085856	0.012898	0.092809	-0.006484
ex100	0.082819	0.010349	0.091117	-0.008460
ex100_2p	0.079949	0.009607	0.085898	-0.009695
ex500	0.079703	0.009715	0.084734	-0.010824
ex500_2p	0.081212	0.010228	0.086365	-0.009975

* Picked one for the SESTM final implementation representative

It is concluded from the above table that use 500 words whose word length is larger or equal to two for each sentiment charged bag with adj. and v. tag shows the best performance and hence is selected for the SESTM final implementation representative.

5. Algorithm extension

Despite replicating the original paper and evaluate its performance, we also try different methods to extend the algorithm and build new models.

5.1 Training dataset selection

First, we try to split the training dataset and only select those articles whose corresponding stocks are with extremely returns, quantified as return label one ([t:t+1]) outside the range of [-2%,2%]. Using those selected articles as our training sample, we implement the SESTM method again to get a new model and get the new predicted scores for the testing sample.

The results is shown as follows:

Result comparison 'av500_2p' as SESTM method	Pearson correlation		Spearman correlation	
	t:t+1	t+2:t+6	t:t+1	t+2:t+6
Whole training dataset*	0.085856	0.012898	0.092809	-0.006484
Selected articles in the training sample (return Outside range of [-2%,2%])	0.084739	0.013537	0.088556	-0.008232

* Picked one for the SESTM final implementation representative

From the table we can find that there's not much difference for this extremely return

variation and therefore ‘av500_2p’ method using the whole training dataset is still be picked for the later portfolio comparison. The correlation change as the time goes plot will be showed in the later part.

5.2 Pre-trained word embedding corpus implementation

There’s some famous methods in the NLP domain such as word2vec. For the existing pre-trained Chinese word vectors, two embedding corpus are well-known, one is trained by ngram2vec toolkit (a superset of word2vec and fasttext toolkit) and another one is trained by Directional Skip-Gram (DSG). These two embedding corpus are hereby used in this project to do the algorithm extension. Instead of using the bag of words, the pre-trained word vectors trained by these two methods are used and a weighted average vectors is calculated representing a article in our dataset.

Pre-trained word embedding corpus resourse link:

- Ngram2vec pre-trained word vectors link: 'Financial News 金融新闻 300d' in https://github.com/Embedding/Chinese-Word-Vectors). (300 dimension)

- DSG pre-trained word vectors:

https://ai.tencent.com/ailab/nlp/en/embedding.html. (200 dimension)

1) SESTM method extension

First we try to implement the original SESTM method while bag of words are replaced with article vectors. However, when still using the MLE method to calculate the predicted score, the algorithm seems to not run well as the result shows a lot of repeatable predicted scores for different articles.

Next, to solve the possible issue caused by the article vectors variables, we choose to use the difference between the article vectors and the O+ and O- vectors (extracted from the training data by implementing the SESTM method) respectively and the difference between these two difference respectively is also calculated for the third predicted score. According to the result, the difference between the article vector subtract O- and the article vector subtract O+, known as the third value (difference of difference), performs best and will be plotted in evaluation charts, noted as ‘corr_n2v_dis’ and ‘corr_DSG_dis’ respectively.

The final correlation for the third value for each method is showed below:

	Pearson Correlation		Spearman correlation	
	t:t+1	t+2:t+6	t:t+1	t+2:t+6
ngram2vec*	0.058775	0.000975	0.058420	-0.014048
DSG	0.056085	0.000704	0.055007	-0.014329

* Picked one for the SESTM extension method representative

2) Machine learning based method

Both regression and classification methods are applied, trying to find the most

powerful predictable model. The logic behind is to regard the article vector entries as the features and the return labels as predicted labels. For the classification problem, returns are labeled with 1 and 0. Two thresholds are implemented, one is 0, returns above or equal to 0 is labeled as 1 and return below 0 is labeled as 0. Second threshold is only to choose top 30% returns articles labeled with 1 and bottom 30% returns articles labeled with 0 as training data. The probability to be label 1 is set to be the predicted sentiment score.

The boundary value for the top 30% return is 0.015943040571993985 and boundary value for the bottom 30% return is -0.0119198913757139.

The results is shown as follows:

ngram2vec	Method variation	Pearson correlation		Spearman correlation	
		t:t+1	t+2:t+6	t:t+1	t+2:t+6
Regression	Linear regression	0.076263	0.018424	0.085789	0.001162
	SVR	0.082435	0.027413	0.086300	0.001689
Classification	Logistic regression (0)	0.092798	0.019045	0.117202	0.011371
	Logistic regression (30%)*	0.095060	0.019217	0.119327	0.010182
	KNN (0)	0.065391	0.006842	0.072908	-0.004894
	KNN (30%)	0.071280	0.006468	0.079083	-0.005909

* Picked one for the SESTM extension final implementation method

DSG	Method variation	Pearson correlation		Spearman correlation	
		t:t+1	t+2:t+6	t:t+1	t+2:t+6
Regression	Linear regression	0.071699	0.013689	0.078169	-0.005170
	SVR	0.081170	0.021812	0.080696	-0.003101
Classification	Logistic regression (0)	0.093262	0.015926	0.116557	0.007929
	Logistic regression (30%)	0.094236	0.016124	0.115696	0.006234
	KNN (0)	0.064300	0.004417	0.071991	-0.007270
	KNN (30%)	0.069855	0.003885	0.076918	-0.008251

From the above two tables, it is obvious that in general the ngram2vec pre-trained word vectors performs slightly better than DSG pre-trained word vectors. Within each extension, logistic regression using the extremely return for the training data performs best. Therefore, the ngram2vec method applying logistic regression with articles whose corresponding stocks has top 30% and bottom 30% return as training data is picked for the later portfolio comparison.

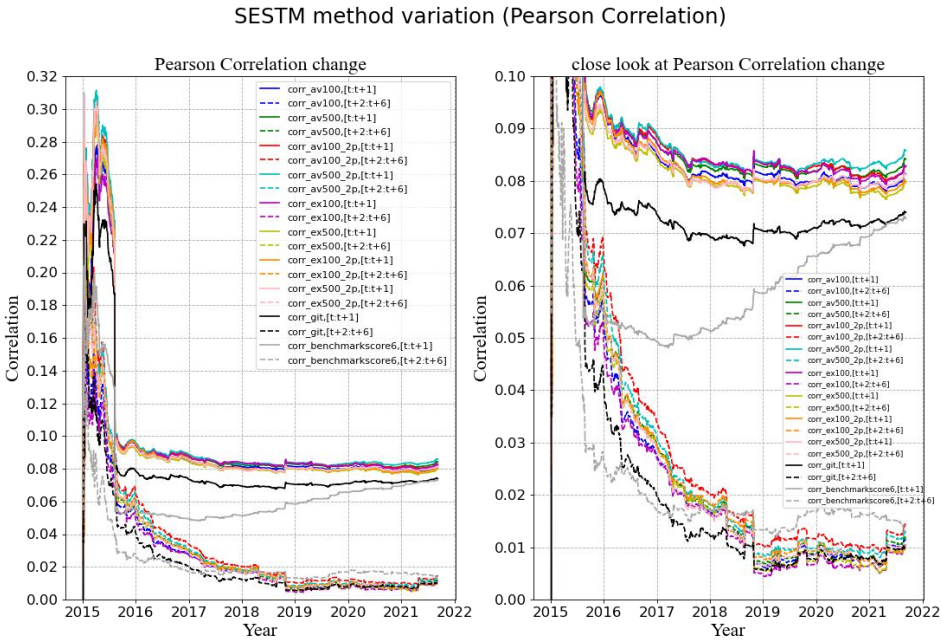
6. Performance evaluation

In this part, we evaluate the prediction results by calculating both the Pearson and Spearman correlation between the predictions and the real returns. For the dictionary based method, fixed dictionary with SESTM method is applied and fully dictionary method with score 6 is also selected for the comparison.

The results for each variation model are showed in the following plots.

6.1 SESTM method variation

1) Pearson correlation

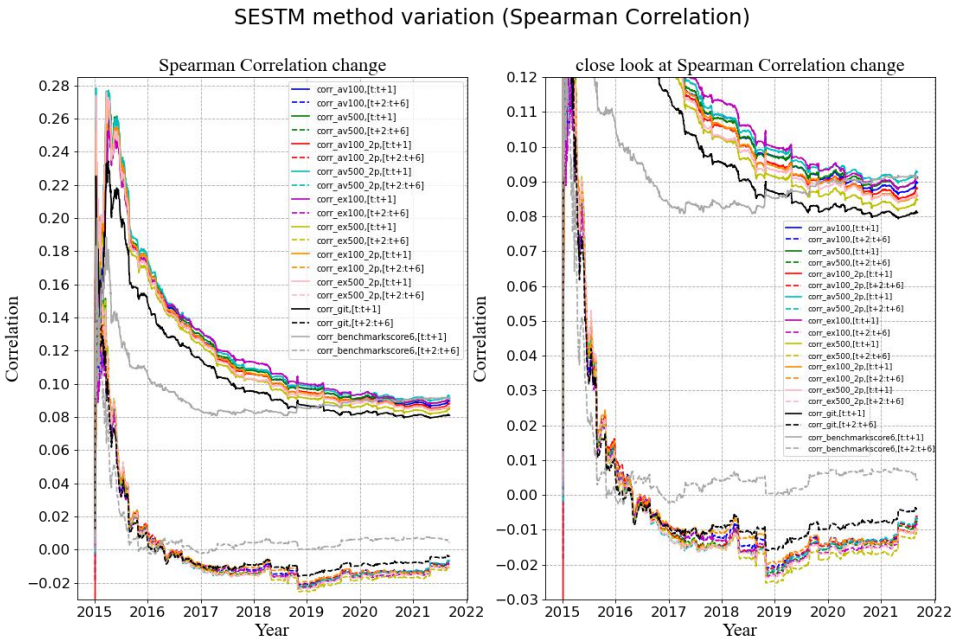


This chart shows the Pearson Correlation change as the time goes by in the testing data period.

The number 100 and 500 indicate the number of words in each sentiment bag and '_2p' suffix stands for (length two plus), means only selecting the words whose length is larger or equal to two. The 'corr_git' means the correlation result for using the words in dictionary from the github link one applying the SESTM algorithm. The 'corr_benchmarkscore6' means the correlation result for fully dictionary based method with score 6 as the predicted sentiment score.

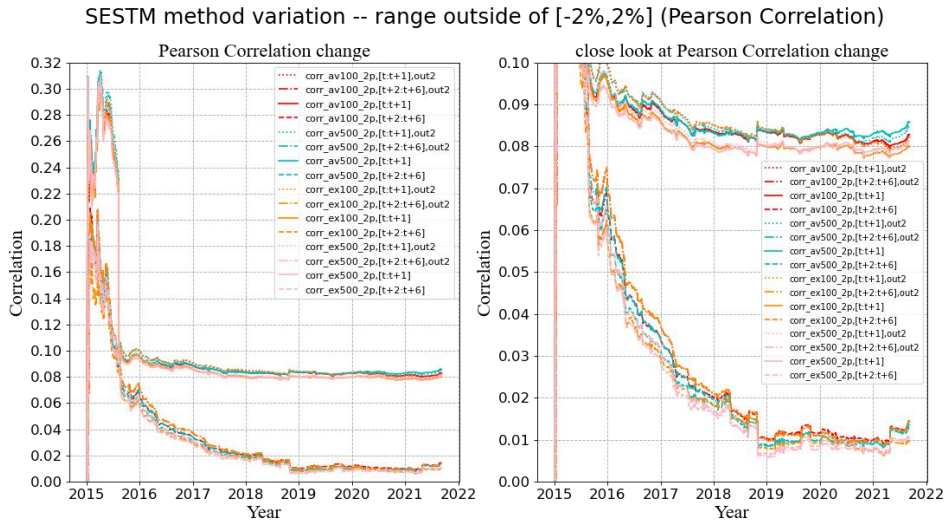
From the chart, we can conclude that the correlation for the return label one [t:t+1] is relative more significant than the correlation for the return label two [t+2:t+6]. For SESTM method, the 'corr_av500_2p,[t:t+1]' performs the best, indicate that using 500 words with a (adj.) and v (verb) POS tag each whose word length is larger or equal to two for the sentiment charged bag is the best solution. In addition, for the comparison, two dictionary based methods including applying the bag of words with SESTM method and directly dictionary based score method strongly show that applying adapted SESTM method that extract the words from the training dataset performs better and hence the algorithm is efficient and replication work is successful.

2) Spearman correlation



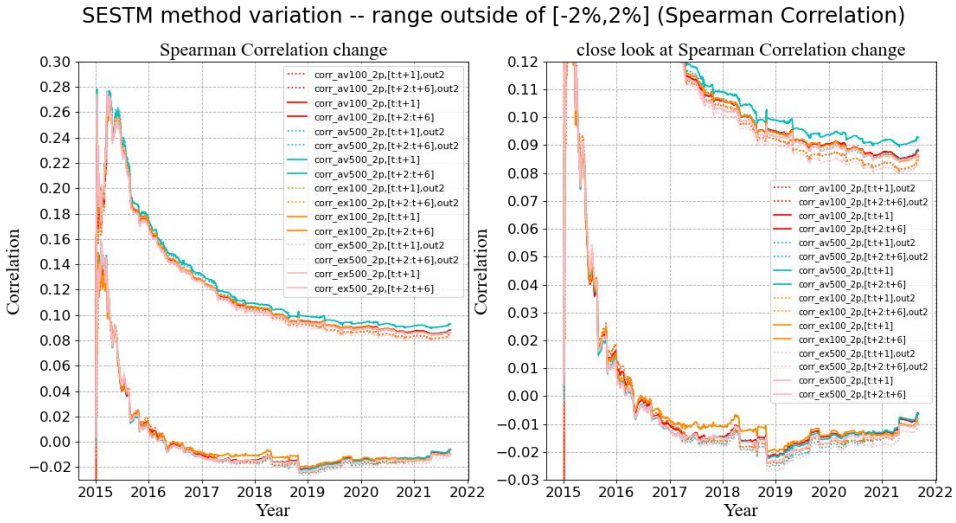
This chart shows the Spearman Correlation change as the time goes by in the testing data period, and the trend is the same as Pearson Correlation change. However, there is the negative correlation performance for the return label two [t+2:t+6] which raise the further question and needs to be explored more.

3) Pearson correlation for return outside range of [-2%,2%]



This chart is the Pearson Correlation change for the comparison between using the whole training dataset and using only those with return outside range of [-2%,2%], and the later one is noted with 'out2' suffix. There's not much difference for these two considerations as in some periods the whole training dataset one is slightly higher while in other periods, the outside range one is slightly higher for all methods and all time window.

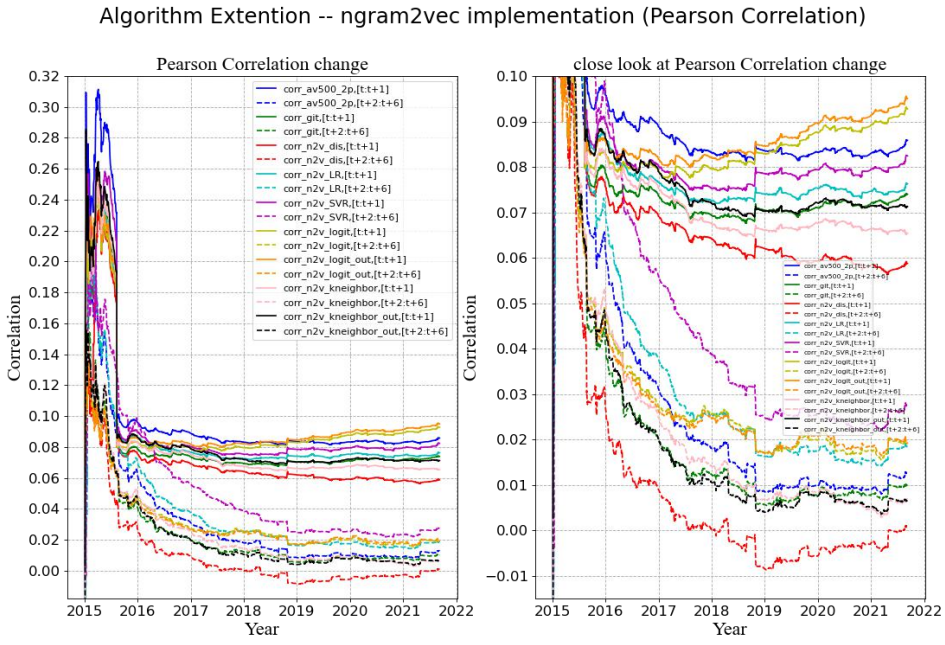
4) Pearson correlation for return outside range of [-2%,2%]



This chart is the Spearman Correlation change for the comparison between using the whole training dataset and using only those with return outside range of [-2%,2%], and the notation is the same as the previous plot. In general, the Spearman Correlation score of the outside range one is slightly lower than the whole training dataset one.

6.2 Algorithm extension

1) Pearson correlation for ngram2vec method

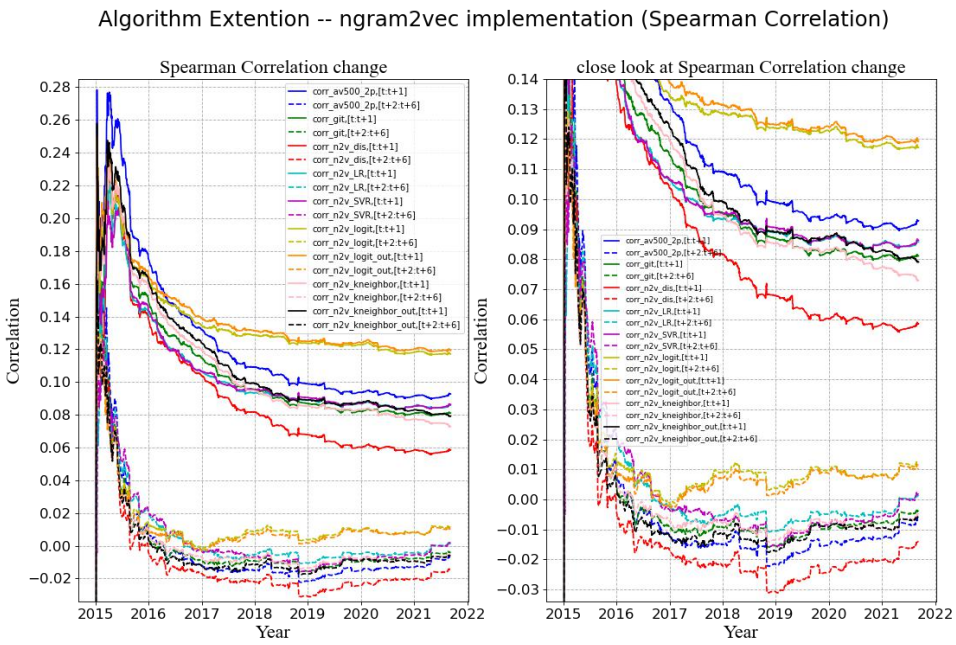


This chart shows how the algorithm extension using the ngram2vec pre-trained word vectors performs. We can find that part of the extension methods perform better than the original SESTM method noted as ‘corr_av500_sp’ and the fixed dictionary based

method applying SESTM method noted as ‘corr_git’, while some of the extension methods perform worse.

To conclude, the prediction ability for the extension method using logistic regression model is the best, and the threshold choose top 30% and bottom 30% (noted with suffix ‘out’) performs slightly better than than simple threshold 0.

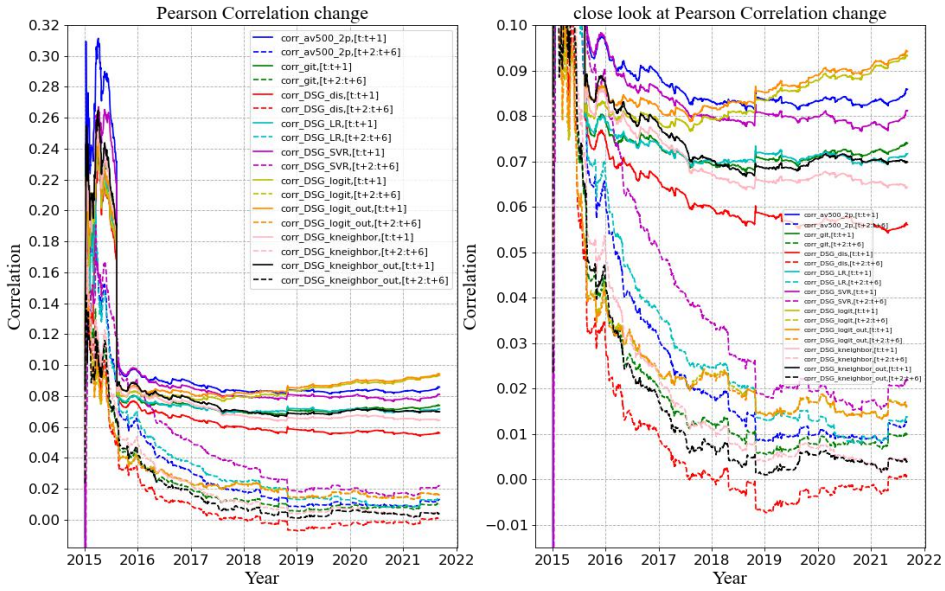
2) Spearman correlation for ngram2vec method



This chart the same, shows the Spearman Correlation change as the time goes by in the testing data period for ngram2vec method implementation. For the Spearman Correlation, the difference is even more significant than the Pearson Correlation which shows that our algorithm extension strategy is successful.

3) Pearson correlation for DSG method

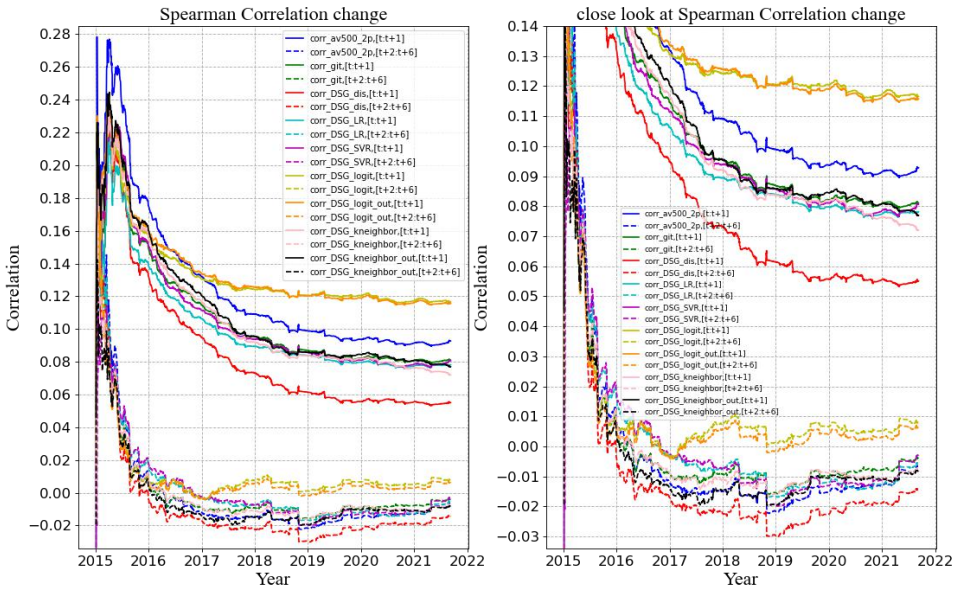
Algorithm Extention -- DSG implementation (Pearson Correlation)



This chart is how each method performs implementing the DSG pre-trained word vectors using the Pearson Correlation. It is obvious that logistic regression performs best with articles whose corresponding stock return is the top 30% and bottom 30% as training data.

4) Spearman correlation for DSG method

Algorithm Extention -- DSG implementation (Spearman Correlation)



The chart is the Spearman Correlation performance for each method implementing the DSG pre-trained word vectors. The same, logistic regression method performs best.

7. Portfolio construction and back-testing

First we calculate both the cross-sectional Pearson and Spearman correlation between previous 30, 60 and 90 days respectively and the following 30 days to see if there is any prediction power and the result is as follows:

Pearson Correlation	Pre 30 days	Pre 60 days	Pre 90 days
Benchmark 1	0.011895	0.012258	0.010908
Benchmark 2	0.006716	0.006618	0.004495
SESTM (av500_2p)	-0.009035	-0.008897	-0.008102
Extension model (n2v logistic regression threshold 30%)	0.014498	0.016344	0.016905

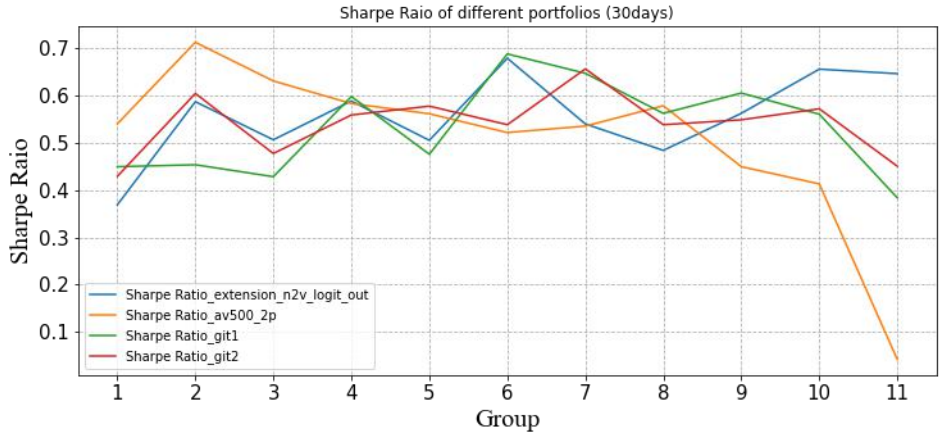
Spearman Correlation	Pre 30 days	Pre 60 days	Pre 90 days
Benchmark 1	0.011895	0.012258	0.010908
Benchmark 2	0.006716	0.006618	0.004495
SESTM (av500_2p)	-0.009035	-0.008897	-0.008102
Extension model (n2v logistic regression threshold 30%)	0.014498	0.016344	0.016905

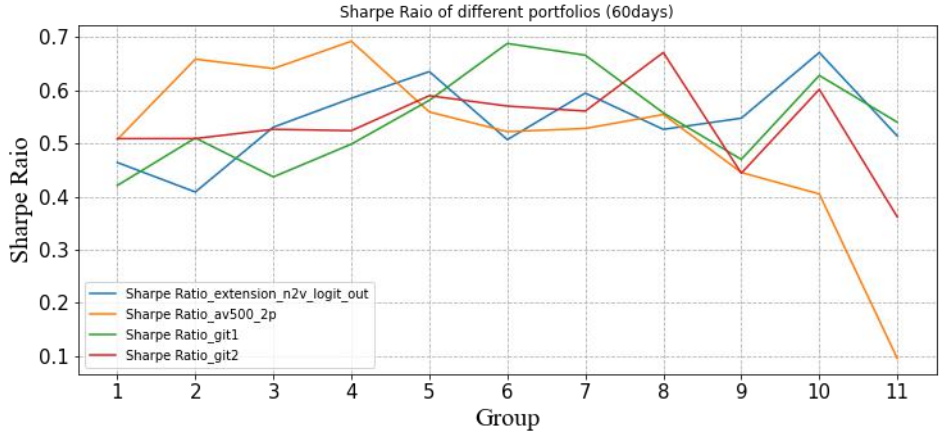
From the table, the extension method using logistic regression selecting top 30% returns and bottom 30% returns as training data performs the best while the correlation for SESTM method is always negative, which needs further exploration.

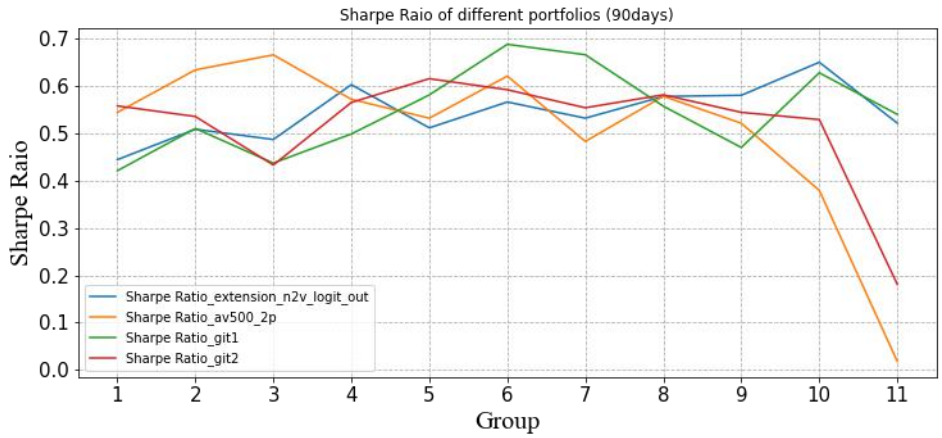
Then 11 portfolios are constructed according to the previous 30, 60 and 90 days predicted sentiment score with monthly balanced. The stocks are divided in to 10 groups, ascending sorted by their predicted sentiment score within one month. No. 10 portfolio means the top 10% predicted sentiment score stocks in the basket and No. 1 portfolio means the bottom 10% predicted sentiment score stocks in the basket. Other No. 2 to No. 9 portfolio follow the similar logic and No. 11 portfolio represents the hedged one, long the stocks in No. 10 portfolio and short the stock in No.1 portfolio.

The charts for average Sharpe Ratio, average annual return and average annual risk of 11 portfolios are showed as below.

1) Sharpe ratio

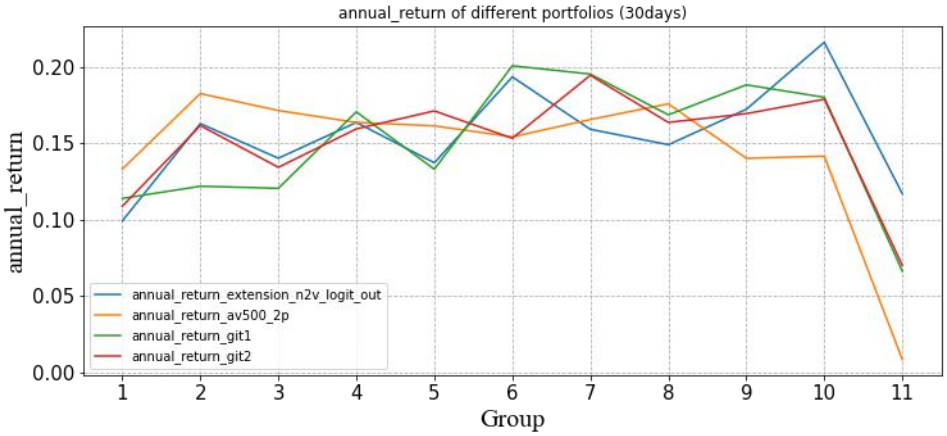


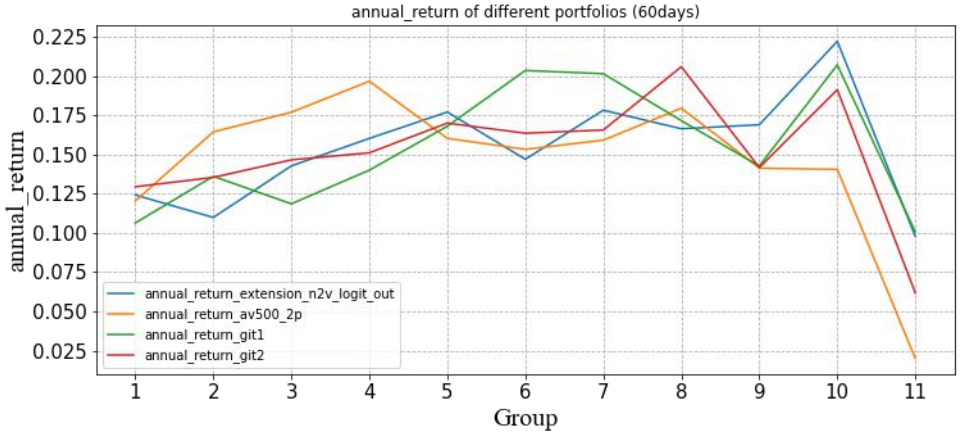


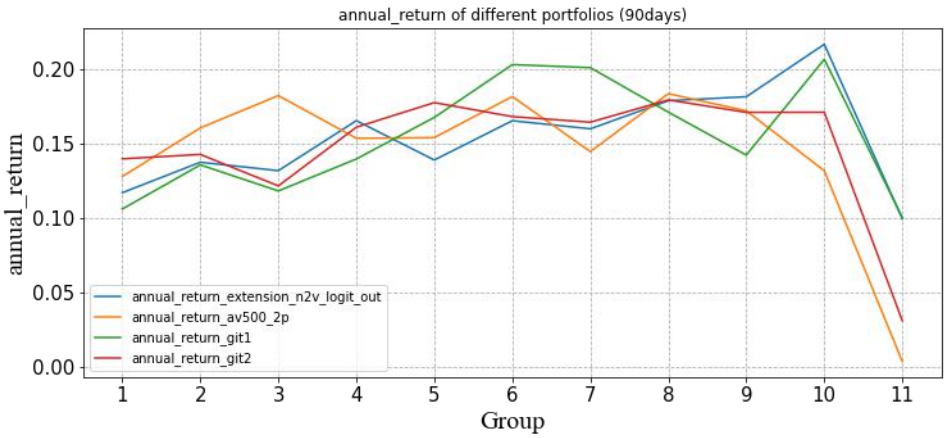


The above three charts are the corresponding Sharpe Ratio for previous 30, 60 and 90 days for different group of portfolios.

2) Annual return

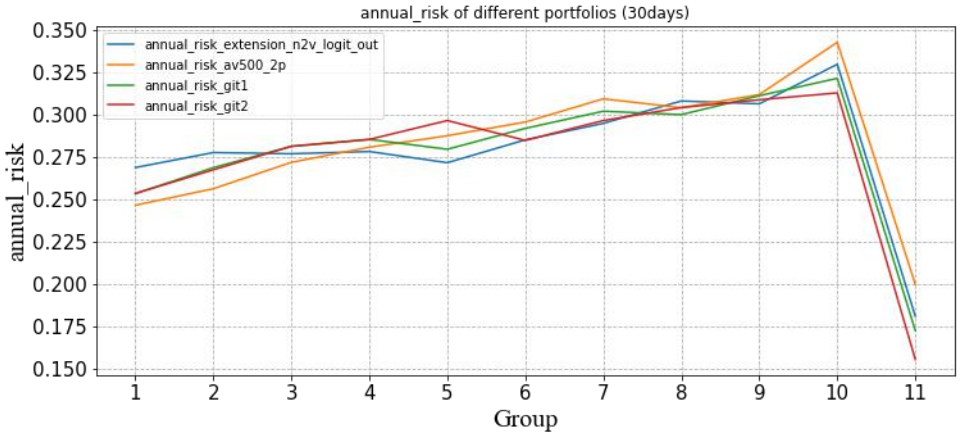


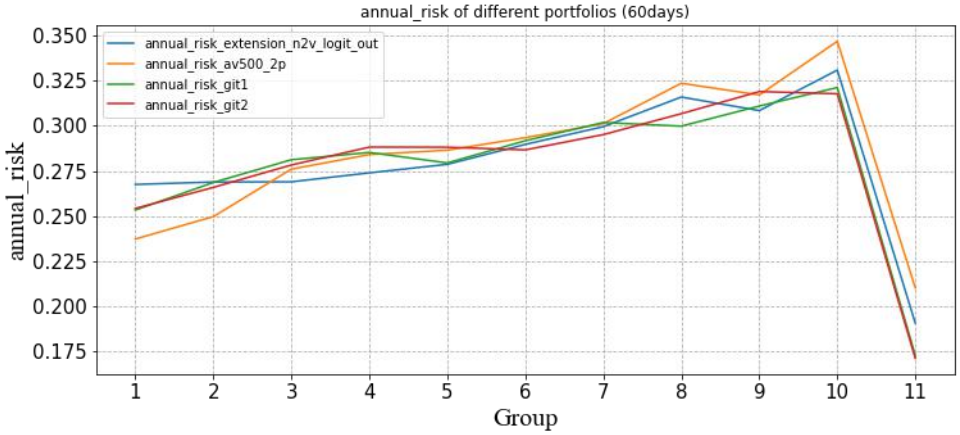


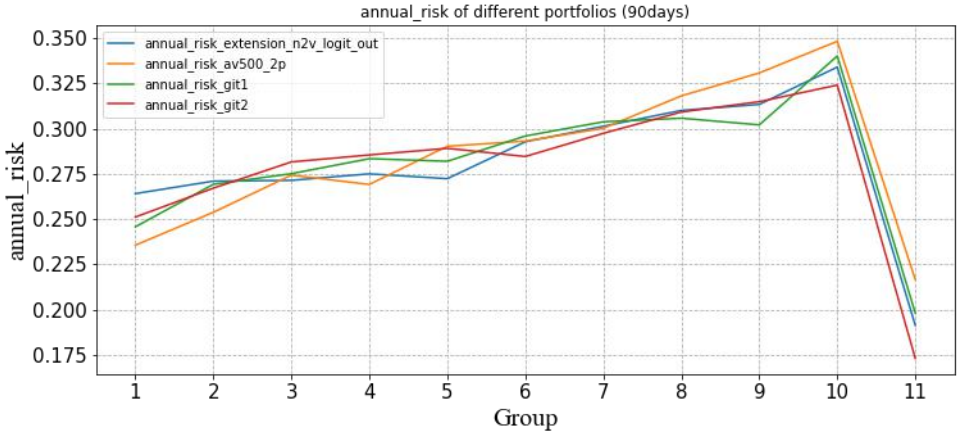


The above three charts are the corresponding annual return for previous 30, 60 and 90 days for different group of portfolios.

3) Annual risk



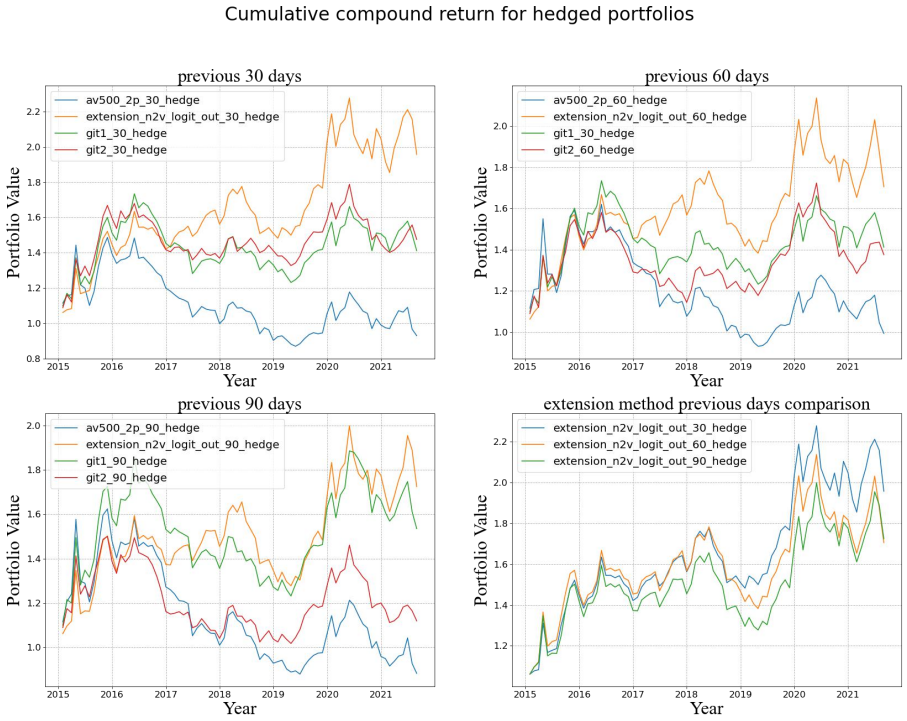




The above three charts are the corresponding annual risk for previous 30, 60 and 90 days for different group of portfolios.

4) Cumulative compound return

Lastly, to backtest our strategies, the cumulative compound return for hedged portfolios adopting four methods are also plotted and extension method performs the best.



8. Self-reflection and Future work

There are a lot more extensions we can to do.

First, the Spearman Correlation of different methods for return label two ([t+2:t+6]) shows the trend reaching negative values, which may need to be verified if there's something wrong with the replication process.

Second, the cross-sectional correlations for the selected best performance 'av_500_2p' method are all negative for previous 30, 60 and 90 days, which needs a further verification.

Lastly, we can also conduct event study to try the correlation between next 5 days return and previous 30, 60, 90 day consensus sentiment score.

9. Conclusion

9.1 method variation comparison

- 500 words for positive and negative word each is better than 100 words each

- adj + v is better than detailed exclusion

- word length >=2 is better than not selecting word length (for SESTM method only)

- 500 words each with a. and v. whose word length larger or equal to two performs best

- github fixed dictionary + SESTM method performs worst

9.2 correlation result analysis

The correlation varies as time goes by.

- for the correlation between predicted score at time t and return label time [t:t+1]

At the begining of the testing sample, the pearson correlation can be up to around 0.31.

While after around half a year, the pearson correlation sharply drop to around 0.08 in average for all methods and become stable afterwards.

The spearman correlation follows the same trend, but with a slightly lower peak at the beginning (around 0.28) and a slightly higher correlation when the trend is stable (around 0.09).

- for the correlation between predicted score at time t and return label time [t+2:t+6]

Follow the same trend as [t:t+1], but the correlation drops ealier.

At the begining of the testing sample, the pearson correlation can also be up to around 0.31.

While in the following months, the pearson correlation gradually drop to around 0.01 in average for all methods and become stable afterwards.

The spearman correlation follows the same trend, with a similar peak at the beginning (around 0.28) but a relative lower correlation when the trend is stable (around -0.01).

9.3 algorithm extension

- Comparison between using the whole training dataset and using only those with return outside range of [-2%,2%]

For pearson correlation: Not much difference as in some periods the whole training dataset one is slightly higher while in other periods, the outside range one is slightly higher for all methods and all time window.

For spearman correlation: In general, the correlation score of the outside range one is slightly lower than the whole one.

- ngram2vec method implementation / DSG method implementation

Both regression and classification methods are applied and logistic regression performs best

9.4 portfolio result analysis

- Adopt four methods in total to test the correlation between previous 1-3 month average predicted sentiment score and the following one month return for each stock.

- Four methods are as followed:

n2v logistic regression threshold 30%

av_500_2p

github link 1 with score 6 (dictionary based method)

github link 2 with score 6 (dictionary based method)

- Construct 10 portfolios and one hedged portfolio (notated as portfolio 11) based on the previous 1-3 month average predicted sentiment score rank

The portfolio is rebalanced at each month.

The higher the number of portfolio, the more positive for the sentiment score.

Therefore, the hedged portfolio 11 is constructed by (portfolio 10 - portfolio 1) at each rebalance date.

- The correlation results show that the ngram2vec method performs best among different methods and different previous time window.

- The sharpe ratio doesn't vary too much among the first 10 portfolios for different methods and different previous time window with an average score at around 0.6.

- However, the sharpe ratio all drops at the hedged portfolio for all different methods and different previous time window with varied values for different combination of method and time window.

- The trend of annual return is the same as sharpe ratio and the average annual return among the first 10 portfolios is around 0.16.

- Interestingly, the trend of annual risk is very different from the previous two. Among the first 10 portfolios, the bigger the number of the portfolio, the higher the annual risk, which means the more positive sentiment of the stocks in blasket the higher the risk of the portfolio. While for hedged portfolio, since it adopts hedged strategy, the risk is supposed to be the lowest, and the result proves the expectation. The annual risk of the hedged portfolio is the lowest among all the 11 portfolios.

9.5 Final conclusion.

1) The algorithm extension applying ngram2vec method performs best.
2) The predicted score is most powerful in the very short period of time [t:t+1], while not significant in the relative longer time window [t+2:t+6]
3) The portfolio based on the sentiment score doesn't perform well which may make sense, since we are back-testing the portfolios on a monthly scale which far beyond the predict ability of the sentiment score.

Appendix

'Replication method.pdf': the summary of the replication method of the paper

'Final coding in all.html': the coding for the whole project in 'html' version

'Final coding in all.ipynb': the coding for the whole project in 'ipynb' version