A BERT-based Firm-level Risk Measurement on Epidemic Diseases

Zhou Xing

May 26 2020

1 Introduction

We adapt a front-end natural language pretraining model, BERT, to construct a measurement of firm-level risk on epidemic diseases, and focus on the impact of COVID-19 since Q4,2019 to Q1,2020. Based on the share of quarterly earnings reference call transcripts, we combine semantics and sentiment analysis to gauge risk faced by individual US firms, and their risk awareness. We focus on two main topics. First, the relation between risk and risk awareness of firm in different stages and their performance in the stock market. Second, the industrial difference on risk brought by epidemic diseases. We validate our measure with firm-level stock market volatity which is assumed highly indicative of risk. We may also extend to decomposition the risk into diverse categories, such as political risk, international risk.

2 Data

Our research is based on three corpus of data. The main part of our analysis relies on **quarterly shared earnings conference call transcripts** where company executives discuss their company's financial performance. The earning calls usually involves three members of the management board, most commonly the CEO (Chief Executive Officer) , CFO (Chief Financial Officer and senior analyst or CMO (Chief Market Officer). During a conference call, investors and analysts can call in over the phone or listen online to hear a company's management comment on the financial results of a recently completed quarter. The duration of earnings conference call is on avarage approximately one hour.

We obtained earnings conference call transcripts of 3597 companies of nine industries from Q3,2019 to Q2, 2020 from *seekalpha.com*. Although related literatures obtain the transcripts from Thomson One Financial and Management Database, which is of the form PDF and thus harder to parse and obtain the text, we tend to make use of the html data from websites, but the number of companies may be smaller. The industrial distribution of transcripts among industries is as follows, with classification based on the categories provided by *seekalpha.com*. [Figure 1] The main technology used in deriving transcripts is web scrapping, mostly based on selenium (webdriver) and regular expression modules on Python.

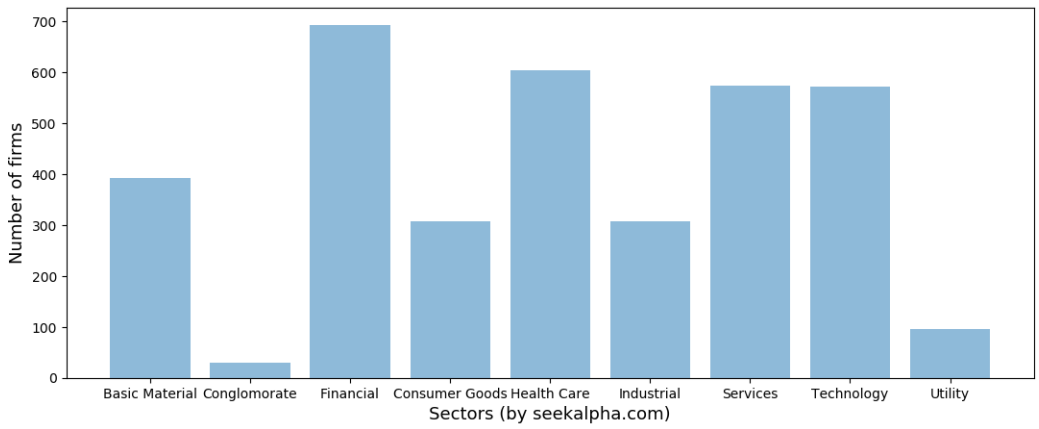

Figure 1: Distribution of firms across sectors

To classify the text into epidemic-disease-related and unrelated, we need a **training corpus for our classification model**. Here we make use of the reports from financial newspaper and label their content based on the title. If the title contains words related to epidemic disease, such as "COVID-19", "virus", "vaccine", "lockdown", we label the text in the articles as "epidemic-disease-related". For contrast, news before the outbreak of COVID-19 from same financial media sources are collected, and labeled as "not-epidemic-disease-related". The data sources is *News API*, a simple HTTP REST API for searching and retrieving live articles from all over the web. We focus on reports from two major business news media **Bloomberg** and **Financial Times.** We extracted 200 articles with disease-related title, such as *Biotech Tourists Drive Short-Lived Rallies in Covid-19 Stocks*[1], with average length 24.6 sentences. On the other hand, 600 disease-free articles from the same plateforms are used as contrary samples, such as news with title *Elon Musk declares plan to take Tesla private.* [2] with average length 35.8 sentences.

Our third data corpus is the votality of stock price of 3597 firms stated above in Q2 2020, Q1 2020, Q4 2019 and Q3 2019. The source of data is *Mergent Online* Database.

3 Model and Methods

The pipeline of the model is as follows: first, we adapt BERT pretraining model to the training data corpus, which gives us a classification model for sentences. Second, we label sentences in firms' earnings conference call transcripts as disease-related or disease-unrelated. Third, we perform sentiment analysis using pretrained Natural Language Processing models on sentences labeled as disease-related. For both the number of disease-related sentences and each sentence's sentiment score, we normalized each of these two features for all firms and take the product of two features as the measure of firm-level

[1]https://www.bloomberg.com/news/articles/2020-05-26/biotech-tourists-drive-short-lived-rallies-in-covid-19-stocks?srnd=coronavirus

[2]https://www.ft.com/content/73b700dc-9a2d-11e8-ab77-f854c65a4465

epidemic-related risk. Finally, we compare the risk measure for individual firms with the panel data of stock price volatility with simple OLS regression.

The key part of our model is the training of the sentence classifier. The advantages of BERT-based sentence classifier were threefold: (1) it allowed for a parallelization of tasks, (2) resulted in simpler operations, and (3) achieved better results overall. Their idea was to build a model based on attention mechanisms, which some of the CNNs and RNNs at that time used to connect their encoder and decoder.

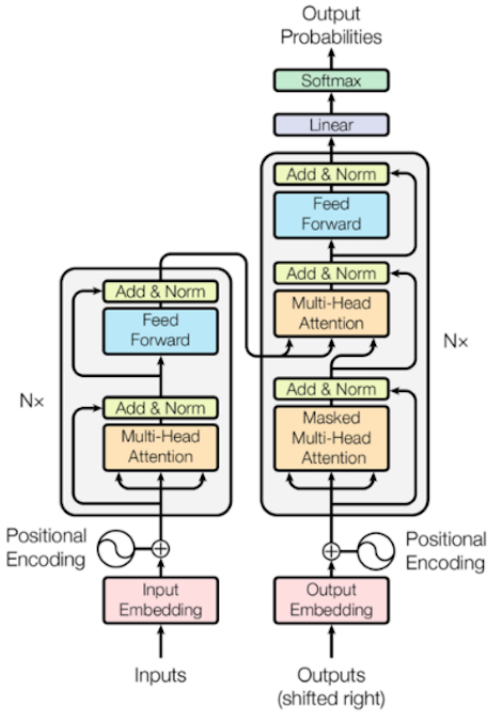

Figure 2: Transformers Architecture.

Adapted from Vaswani et al., 2017[2], Figure 2 displays a transformer. On the left side is the encoder, next to it on the right the decoder. As in preceding models, the encoder is responsible for forming continuous representations of an input sequence. The decoder in turn maps these representations back to output sequences. Both encoder and decoder consist of several layers (denoted with Nx in the graph) with two and three sub-layers each, respectively:

- **Multi-Head Attention:** here, keys, values and queries (which come from the self-

attention - in this case, the previous layer's output) are linearly projected to then perform the attention function in parallel (which Vaswani et al. call the "scaled dot-product attention"). The multi-head characteristic makes it possible to use "different representation subspaces at different positions" .

- **Masked Multi-Head Attention** (only in decoder): as the first sub-layer in the decoder, this layer performs the multi-head attention on the encoder's output. It is masked in order to prevent predictions based on information that must not be known yet at a certain position.

- **Feed Forward Network:** two linear transformations are applied to each position separately and identically.

Before information enters the layers, the positional encoding conveys information on the relative position of a token in a sequence and allows the transformer to make use of the token order. Once the layers have performed their attention function and transformations, another two transformations take place: the Linear, applying another linear transformation, and the Softmax, which transforms the output back to probabilities.

BERT (Bidirectional Encoder Representations from Transformers) was published in 2018 by Devlin et al. [1] from Google and performed so well that - within a year - it inspired a whole model-family to develop. BERT built on the original transformer idea, but used a slightly changed architecture, different training, and (as a result) increased size.

- **Architecture:** BERT's architecture is largely based on the original transformer but is larger (even in its base version) with more layers, larger feed-forward networks and more attention heads.

- **Training:** The real innovation comes in the bidirectional training that BERT performs. There are two pre-training tasks: masked language modeling (MLM) and

next sentence prediction (NSP). These are performed on the pre-training data of the BookCorpus (800 million words) and English Wikipedia (2500 million words).

After understanding the transformer-based models in theoretical context, we apply them to our classification problem: to determine if there is epidemic-disease-related uncertainty in conference call transcript with the aim of extending and improving the current classification methodologies. For this purpose, we use the package **Simple Transformers**, which was built upon the Transformers package (made by HuggingFace). Simple Transformers supports binary classification, multiclass classification and multilabel classification.

4 Model Implementation and Initial Result

At this stage we have obtained initial result for the classification model. BERT-based models come with a set of hyper-parameters which need to be adjusted properly for the task. For our binary classification approach, we replaced the hyper-parameters' default values for the most impactful ones:

- **Number of training epochs:** The epochs were increased from 1 to a number as high as 10. However, increasing the number of epochs also increased the running time of the model intensively. At the end, 4 epochs turned out to be the most optimal number, balancing between running time and giving good results.
- **Batch size:** The batch size was increased from 2 till 16, and batch size of 8 gave good results. With a batch size of 4, we observed a certain overfitting.
- **Maximum Sequence Level:** Although the model can handle a maximum of 512 tokens, we started off first with 100 tokens, and then opted for 200 tokens. The average length of the articles in the entire dataset was 534, which implies that

setting this hyper-parameter to 512 could have been an ideal choice. However, owing to certain computational difficulties it was limited to 200.

- **Learning Rate:** 4e-5. Changing this learning rate made the model run either faster or slower, results were optimal for this chosen learning rate.

We compare the performance of our classifier to two non-transformer based models as well. We have used a small bidirectional neural network and a SVM classifier. We also ran other BERT-family models (RoBERTa, DistilBERT and ALBERT) with the same hyper-parameters for comparison and further improvement in the implementation of BERT in our model.

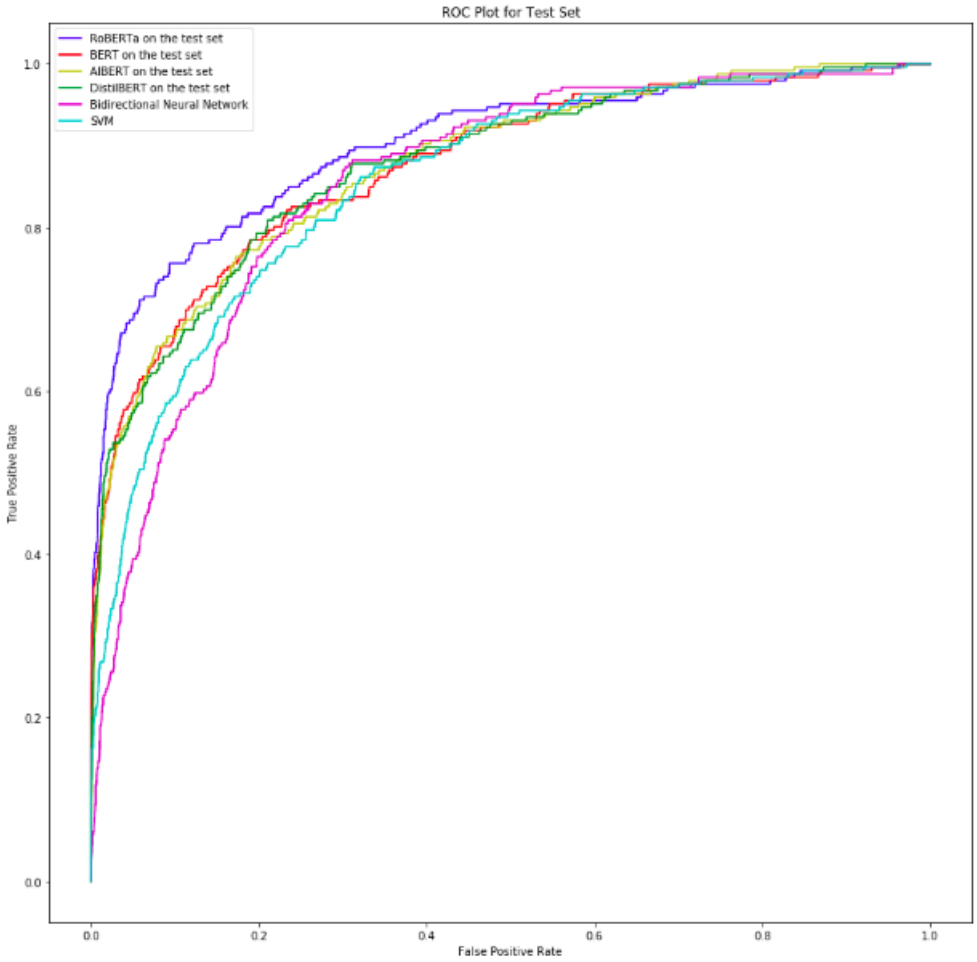

Figure 3: Comparison of ROC among models

From Figure 3 we can see that the BERT-family models all perform better than the

benchmark models. While BERT's, DistilBERT's and ALBERT's performance does not differ much from each other across all of our metrics (AUC [3] of around 0.87), RoBERTa largely outperforms them. Their runtimes were similar around 18 minutes, with DistilBERT and ALBERT being slightly faster, and the SVM taking by far the least time with only 5 minutes runtime.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.

[3] **Area Under Curve (AUC)** is a measure that takes into account the true and false positives. It measures the area under the (ROC) curve, which plots the false positive rate (specificity) against the true positive rate (sensitivity). A perfect model would have an AUC of 1, a randomly assigning model would be at 0.5.