

Forecasting Firm Material Event Sequences from 8-K Reports

Shuang Zhai

Iowa State University
szhai@iastate.edu

Zhu Zhang

Iowa State University
zhuzhang@iastate.edu

Abstract

In this paper, we show deep learning models can be used to forecast firm material event sequences based on the contents in the company’s 8-K Current Reports. Specifically, we exploit state-of-the-art neural architectures, including sequence-to-sequence (Seq2Seq) architecture and attention mechanisms, in the model. Our 8K-powered deep learning model demonstrates promising performance in forecasting firm future event sequences. The model is poised to benefit various stakeholders, including management and investors, by facilitating risk management and decision making.

1 Introduction

One Corporate Event Sequence (CES) is a sequence of events that take place at one company during a period of time. A series of company events can represent corporate strategy and future plans. Therefore, CES can be used as a tool to probe corporate strategy and decision-making behaviors.

Investors can use existing CES to project a company’s future CES. For instance, an acquisition is a sign for financing, and insufficient funding is a vane for refinancing. Similarly, a failing operational decision can bring an executive personnel change, and a new senior-level appointment can be expected after that. Since CES embodies consistency (to corporate strategy) and continuity (to time), it is equipped to illuminate a maze pathway for organizational strategy evaluation.

Researches increasingly reveal the merit of textual data in financial studies. Many studies are centered on the financial market, such as a firm’s stock price, return, and volatility (Fang and Peress, 2009; Tetlock, 2010; Edmans, 2011). Meanwhile, the high dimensionality characteristic of textual data presents challenges to traditional econometric

models. Therefore, they render themselves to machine learning and deep learning models naturally. Deep learning models have become gradually popular in finance applications recently, and many of them have focused on market-related tasks as well (Ding et al., 2014, 2015). These models, however, didn’t take corporate strategy into account and didn’t focus on the sequence nature of corporate events.

U.S. Security and Exchange Commission (SEC) requires the publicly-traded company to file Form 8-K (also called ‘Material Event Report’ or ‘Current Report’) when certain types of the corporate event take place. In general, an 8-K report should be filed when the company has an event that its shareholders should be aware of. A material event is defined as *a matter if there is a substantial likelihood that a reasonable person would consider it important*¹, and a “rule of thumb” impact scale of a material event is *five to ten percent of net income*².

Although public companies are also required to file Form 10-K (annual report), and Form 10-Q (quarterly report), 10-K/Qs have apparent and significant drawbacks compared to 8-Ks. 10-K/Qs are designed to cover a mixed category of information. It is easy for them to plunge lower readability and create higher barrier for amateur readers. While the length of 10-K/Qs gets longer and longer (Cazier and Pfeiffer, 2015), not all investors have the skill to decipher the insightful message from the lengthy 10-K/Qs. When they encounter difficulties in 10K/Qs, most retail investors do not have enough resources as advanced institutional investors do. Most importantly, 10-K/Qs are re-

¹<https://www.sec.gov/interp/account/sab99.htm>

²Financial Accounting Standards Board (“FASB”), Statement of Financial Accounting Concepts No. 2, Qualitative Characteristics of Accounting Information (“Concepts Statement No. 2”), Concepts Statement No. 2, 167.

leased long time after the event and get considerably prolonged-release intervals. It means investors have to wait one quarter or longer to see the updated official release from the company.

Various stakeholders, not only investors but also management teams and regulators, can find CES useful. CES provides not only corporate strategy hints but also corporate operation patterns. Given the continuous characteristic of CES, it notches up contents for stakeholders to achieve higher profits and a better position in a timely manner.

Given the versatile benefits of 8Ks and the textual data forecasting ability of deep learning models, we propose an end-to-end sequence-to-sequence neural network to predict corporate event sequences from 8-K reports in this paper.

2 Related Techniques

2.1 Gated Recurrent Units (GRUs)

Gated Recurrent Unit (GRU) is a representative deep learning architecture, and it was first proposed by [Cho et al. \(2014\)](#). Numerous works have been done in natural language processing using GRUs, such as part of speech (POS) tagging, information extraction, syntactic parsing, speech recognition, machine translation ([Cho et al., 2014](#)), and question answering.

2.2 Sequence to Sequence (Seq2Seq) Neural Network

Sequence-to-sequence (seq2seq) model was introduced by [Sutskever et al. \(2014\)](#). It is widely used by machine translation tasks ([Bahdanau et al., 2015](#); [Luong et al., 2015](#)), i.e. translating sentences from one language to another language, such as French to English. Attention mechanism has been explored broadly in recent publications. The intuition for attention technique in natural language processing is to assign higher attention to texts where contain more information for the task on hand. [Yang et al. \(2016\)](#) employed both word-level and sentence-level attentions for document classification task. [Wang et al. \(2016\)](#) proposed an aspect-level attention to capture different sentiments for different aspects in a sentence. [Ma et al. \(2017\)](#) proposed an interactive attention architecture that models the interaction between the context and target for sentiment classification task. More recently, [Vaswani et al. \(2017\)](#) used multi-head attentions alone to solve sequence prediction problems which are traditionally handled by other

neural network techniques such as Long Short-Term Memory and Convolution Neural Networks. We are inspired by [Bahdanau et al. \(2015\)](#), [Luong et al. \(2015\)](#), [Kadlec et al. \(2016\)](#) and [Cui et al. \(2017\)](#) to compute event attention for our prediction task.

3 Material Events and Form 8-K Current Reports

3.1 Related Works

Various items ³ are required to be filed in company 8-K reports. Many studies have tried to categorize 8-Ks into different categories. [Zhao \(2016\)](#) classified 8-Ks into seven categories: 1) information about business and operations (OPR), 2) financial information (FIN), 3) matters related to the exchange or trading of the securities, 4) information related to financial accountants and financial statements, 5) corporate governance and management (GOV), 6) events related to Regulation Fair Disclosure (REG), 7) other events considered important to the firm (OTH). OPR, FIN, GOV, REG, and OTH are the five major 8-K categories which cover more than 95% of all 8-K reports in their study.

[Feuerriegel and Pröllochs \(2018\)](#) used Latent Dirichlet Allocation (LDA) method and categorized 8-K reports into topics: energy sector, insurance sector, change of trustee, real estate, corporate structure, loan payment, amendment of shareholder rights, earnings results, securities sales, stock option award, credit rating, income statements, business strategy, securities lending, management change, health care sector, tax report, stock dilution, mergers and acquisitions, and public relations. Earnings results and public relations are the top two topics in their study. [He and Plumlee \(2019\)](#) categorized voluntary items (Item 2.02, 7.01, and 8.01) into a business combination, conference presentations, dividend announcement, litigation, patents, restructuring, security offerings, share repurchase, and shareholder agreement.

3.2 Our Approach

However, none of the above studies tried to categorize 8-Ks by the event nature. Meanwhile, certain items are mandatory to be reported and other items are voluntary. Therefore, we first read

³<https://www.sec.gov/fast-answers/answersform8khtm.html>

thousands of 8-Ks by ourselves and designed taxonomies to holistically characterize 8-Ks into multiple event types, based on human understanding of the report content and nature of the event. Then, we map every report to one of our event types for analysis. We list our event types in Table 1, and they are our prediction’s *target variables*. Since some reports can be filed under different item numbers (from 1.01 to 8.01, we eliminated 9.01 Exhibits), the mapping between Report Items and Event Types is many to many. In other words, one report item number can also be seen in different event types. The bold item numbers in Table 1 are items shown in more than one event types. For instance, senior personnel change can be filed under Item 5.02 Departure of Directors or Certain Officers; Election of Directors; Appointment of Certain Officers; Compensatory Arrangements of Certain Officers, or Item 8.01 Other Events.

4 Problem Definition

4.1 Background

The goal of our work is to predict firm’s future event sequences, based on its historical event sequences. Therefore, our prediction task is to solve a sequence-to-sequence problem. In particular, we use corporate event sequences in memory M to predict event sequences in forecasting horizon H . Historical event sequences are collected from corporate 8-K reports, and all events are identified by event types listed in Table 1. Memory M and forecasting horizon H are formed by smaller time intervals j and q , respectively. The time structure of our model is illustrated in Figure 1.

4.2 A Real-world Example

We can also view the sequence-to-sequence prediction as a story completion task. Particularly, once we know what events happened in the past, we can predict what events are likely to occur in the future, to complete the story. We present an example of corporate events sequence in Figure 2 to demonstrate the real-world practice and the importance of the problem.

Figure 2 shows an example of the company AT&T event sequence during time $t-M$ to $t+H$, and it illustrates how the historical corporate events during time $t-M$ to t can forecast and impact future corporate events during time t to $t+H$. For example, at time $t-M$, AT&T wins several wireless spectrum auctions from the Federal Communica-

tions Commission (FCC). Because of the auction win, AT&T needs more capital to support the new business. Therefore, we see AT&T has arranged loans and reported them in its 8-K afterward. In another stream, similar to its competitors, AT&T also desires to tap into the content industry. It announces the acquisition of Time Warner to support this corporate blueprint. As we have learned previously, because of the acquisition, AT&T needs more money to finance this deal. As a result, AT&T has filed a loan financing activity in its following 8-K report. Meanwhile, possibly because of the corporate strategy disagreement, AT&T’s business solution CEO, who was a supporter of expanding business in hardware instead of the content industry, announces retirement. Given what we have learned so far, we can foresee that several corporate events can have higher chances to become real in the future. For instance, if the previous financing amounts were not enough for the Time Warner acquisition, AT&T have to require more loans. We can observe the loan financing activity admittedly happened in the forecasting horizon. Moreover, because of the acquisition, AT&T would need to make arrangements for Time Warner’s executive members, which indeed happened in the forecasting window. AT&T announced Time Warner’s CEO duty after the acquisition was completed. Additionally, after the acquisition was completed, AT&T has realized additional financial needs for the combined business. Therefore, we notice AT&T reported another loan activity after the completion of the acquisition. This example tells us that historical corporate events can affect not only *what* type of event will happen in the future, but also *when* the event will occur in the future.

4.3 Formal Definition

Let’s formally define the problem as,

$$E_{(i)j} = g_{k=0}^{|K|}(S_{(i)jk}) \quad (1)$$

$$y_{(i)q} = f_{j=0}^M(E_{(i)j}) \quad (2)$$

$$y_{(i)q} \in Y_{(i)}^H \quad (3)$$

where,

- y denotes the event types in Table 1, Y^H denotes the event sequence during H , and sequence $Y^H = [y_1, \dots, y_H]$.

ID	Event Type	Code	Report Item	Examples
1	Business combination and restructuring	BC	1.01, 1.02, 2.01, 7.01, 8.01	merger, acquisition, join venture, separation, spin-off
2	Financial activities	FN	1.01, 1.02, 2.03, 2.04, 2.05, 2.06, 3.02, 6.01, 6.02, 6.03, 6.04, 6.05, 7.01, 8.01	lend, borrow, loan, Notes, payment, debt, stock, repurchase, dividend, asset-backed securiteis (ABS)
3	Operation activities	OA	1.01, 1.02, 7.01, 8.01	operation, contract, consulting, service, product, supply
4	Senior personnel change	PC	1.01, 1.02, 5.02, 7.01, 8.01	executive officer/director, retire, leave, appointment
5	Information disclosure	ID	2.02, 4.01, 4.02, 5.07, 5.08, 7.01, 8.01	conference, presentation, statement, exhibit
6	Document update	DU	3.03, 5.01, 5.03, 5.05, 5.06, 7.01, 8.01	by-laws, code of ethics
7	Intellectual property activities	IP	1.01, 1.02, 7.01, 8.01	intellectual property, patent approval
8	Litigation and lawsuit	LL	1.01, 1.02, 7.01, 8.01	settlement, litigation, lawsuit
9	Delisting, trading suspension	DL	3.01, 5.04, 7.01, 8.01	delisting, trading suspension
10	Bankruptcy	BK	1.03, 7.01, 8.01	bankruptcy

Table 1: Event Types (*target variables*)

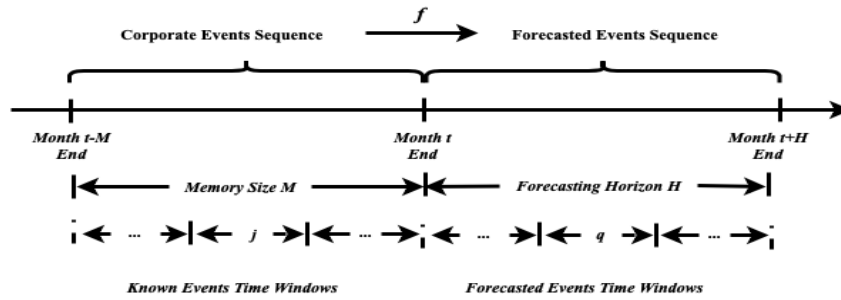


Figure 1: Time structure of the model

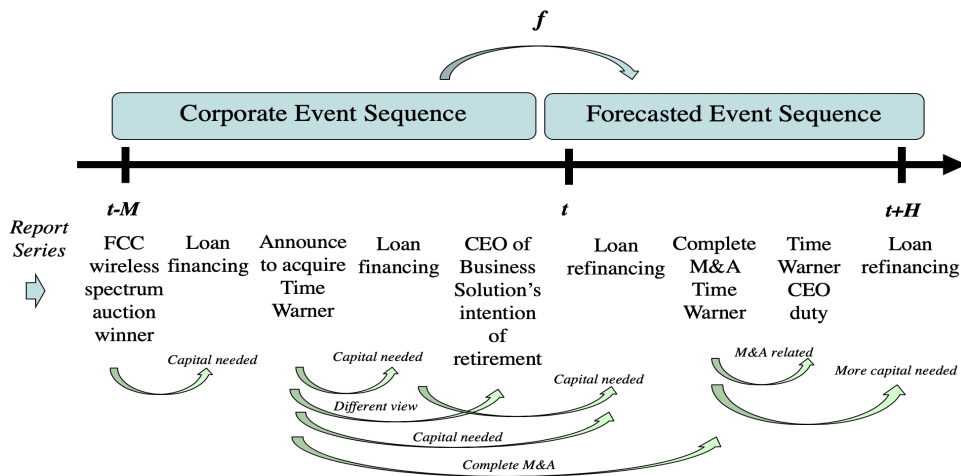


Figure 2: Corporate event sequence forecasting example: AT&T

- M denotes the size of memory, and H denotes the size of forecasting horizon, both are measured in terms of the number of time windows.
- i indexes companies.
- j indexes time window in M , and $j = \{1, \dots, M\}$.
- q indexes time window in H , and $q = \{1, \dots, H\}$.
- ev denotes event index, and $|Ev|$ denotes the total number of event types.
- $|K|$ denotes the total number of events per time window.
- $S_{(i)jk}$ is the embedding of the k th event of company C_i in time window j , and $E_{(i)j}$ is the aggregate event embedding of company C_i in time window j .
- g is a function that aggregates multiple event embeddings into one embedding.
- f is a learned model (function) that maps all event embeddings in memory to forecasting horizon.

In principle, g and f can be parameterized as any function approximator.

5 Model

5.1 GRU model

Since our task works on time sequences and is formed as a sequence-to-sequence (seq2seq) problem, we use *GRU* as the backbone of our model. Additionally, we use *encoder-decoder* framework as the architecture in our model.

In the encoder of the model, we train our event type embeddings in the *Event Embedding Layer*. Multiple reports can be filed at the same time window. Therefore, at each time window, we select the top $|K|$ event embeddings for each company, based on each event embedding’s L2 norm value. We institute various treatments of function g in Equation 1, such as attention mechanism. In the *Event Attention Layer*, we implement attention to the top $|K|$ event embeddings and obtain the weighted embedding at each time window.

We define our *Event Attention Layer* as,

$$\tilde{h}_{(i)j} = \text{sigmoid}(W_{(i)jk}S_{(i)jk} + b_{(i)j}) \quad (4)$$

$$\alpha_{(i)jk} = \text{softmax}(W_{(i)j}\tilde{h}_{(i)j}) \quad (5)$$

$$E_{(i)j} = \sum_{k=1}^{|K|} \alpha_{(i)jk}S_{(i)jk} \quad (6)$$

The weighted sum context vector $E_{(i)j}$ is used as the aggregated semantic representation of the company events at each time window. Next, $E_{(i)j}$ is fed into the following *Event Embedding Layer*.

In the vanilla GRU model, the last hidden state of the encoder is directly connected to the decoder. At every time t in the decoder, the hidden state h_{st} is used to predict the current timestamp event type in the *Prediction Layer*, and we use *softmax* function to compute y_t as,

$$y_t = \text{softmax}(W_{st}h_{st}) \quad (7)$$

5.2 GRU_attention model

We implement the *Alignment Attention Layer* in the GRU_attention model.

In the *encoder-decoder* framework, information from the encoder is carried over to the decoder. To be able to capture what events happened in history play more roles in the prediction horizon, we employ attention mechanism (Bahdanau et al., 2015; Luong et al., 2015; Kadlec et al., 2016; Cui et al., 2017) to capture the dynamics. The GRU_attention model’s decoder looks at every hidden state in the encoder. The *Alignment Attention Layer* aligns different attention values to hidden states in the encoder, and aggregates them. We follow the ”general” approach in Luong et al. (2015) and obtain attention scores between the target sequence and the input sequence as,

$$\text{score}(h_{tr}, h_{sr}) = h_{tr}^T W_a h_{sr} \quad (8)$$

, where h_{tr} is the hidden state of target sequence and h_{sr} is the hidden state of the source sequence.

The context vector c_t is the weighted sum of the product of attention scores and the hidden states in the encoder.

$$c_t = \sum_{j=0}^M \text{score}(h_{tr}, h_{sr})_j h_{sr_j} \quad (9)$$

$$h_{st} = \text{tanh}(W_c[c_t; h_{tr}]) \quad (10)$$

, where $[\cdot; \cdot]$ denotes concatenation along the sequence dimension.

We illustrate the GRU_attention model in Figure 3.

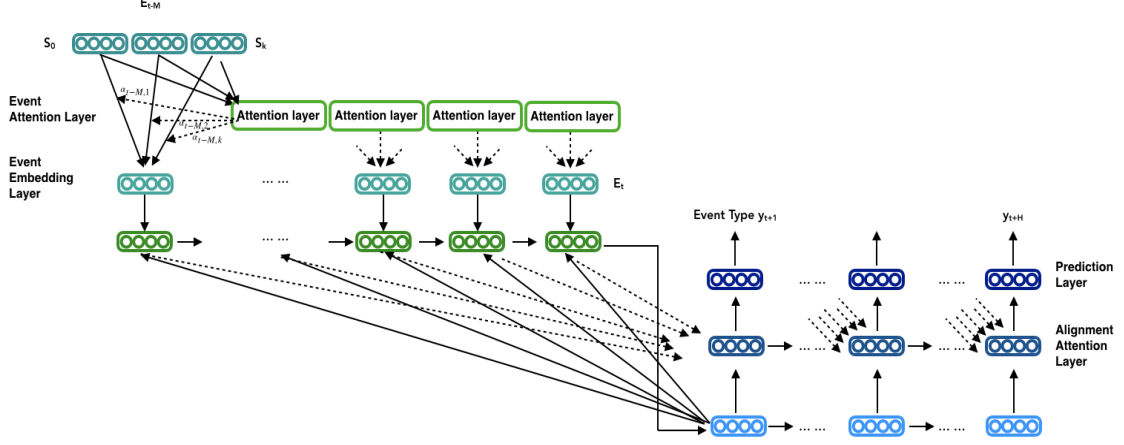


Figure 3: Corporate Event Sequence Forecasting: GRU_attention Model

6 Experiments

6.1 Evaluation Models

We evaluate the following models in our experiments:

- **MCMC_baseline:** Markov Chain Monte Carlo (MCMC) simulation. (Details are discussed in 6.2.)
- **GRU:** event sequence as input and without attention.
- **GRU_attention:** event sequence as input and with attention.

6.2 MCMC baseline

For every company, we gathered its event sequences during the entire experimental period, and constructed an event transition matrix.

Given the obtained transition matrix, we can implement the Markov Chain Monte Carlo (MCMC) simulations. In particular, we view each row of the transition matrix as each event type’s probability distribution. We recognize every event type at the last training timestamp as the current event type E_t , then we can draw the next event type E_{t+1} given E_t ’s probability distribution. By doing this step repetitively, we can sample E_{t+2} based on E_{t+1} ’s probability distribution, and so on. Finally, we reach E_{t+H} and complete the sampling process. In the experiment, for each event type E_t , we sample its sequences 100 times, and we use the averaged sequence performance as our model baseline.

6.3 Per Event Type Evaluation

We experiment a threshold to convert the *softmax* result of each event type at time t into binary format as,

$$Eval_{ev,t} = \begin{cases} 1 & \text{if } y_{ev,t} > \text{threshold} \\ 0 & \text{otherwise} \end{cases}$$

In experiments, we set the threshold value as 0.1. We evaluate classification performance per event type. In particular, we compute classification criteria, i.e., precision, recall, and F1 score, for each event type and evaluate our model results.

In addition, since predicting event type correctly within a reasonable temporal approximate period is also important in real business setting, we use two approaches, i.e. *precise evaluation* and *fuzzy evaluation* to evaluate our models.

For each event type, we compute its true positive (TP), false positive (FP), false positive (FP), and true negative (TN), and evaluate the model’s precision (Pr), recall (Re), and F-measure (F1) accordingly.

Precise evaluation: we compute the confusion matrix at the precise time t as,

$$Precision_{ev,t}(Pr_{ev,t}) = \frac{TP_{ev,t}}{TP_{ev,t} + FP_{ev,t}} \quad (11)$$

$$Recall_{ev,t}(Re_{ev,t}) = \frac{TP_{ev,t}}{TP_{ev,t} + FN_{ev,t}} \quad (12)$$

$$F1_{ev,t} = \frac{2 * Pr_{ev,t} * Re_{ev,t}}{Pr_{ev,t} + Re_{ev,t}} \quad (13)$$

Fuzzy evaluation: because correctly predicting event type that close to the exact time t also has

practical implications, i.e., forecasting accurately of the event type close to time t is also useful in reality, we compute the confusion matrix within $[t - z, t + z]$ time window, and $z \in [1, 2, \dots]$. We update the true positive for precision and recall for $t \in [t - z, t + z]$, and re-compute the precision, recall, and F1 measures for $t \in [t - z, t + z]$ as,

$$Pr_{ev,[t-z,t+z]} = \frac{TP_{ev,[t-z,t+z]}}{TP_{ev,[t-z,t+z]} + FP_{ev,t}} \quad (14)$$

$$Re_{ev,[t-z,t+z]} = \frac{TP_{ev,[t-z,t+z]}}{TP_{ev,[t-z,t+z]} + FN_{ev,t}} \quad (15)$$

$$F1_{ev,[t-z,t+z]} = \frac{2 * Pr_{ev,[t-z,t+z]} * Re_{ev,[t-z,t+z]}}{Pr_{ev,[t-z,t+z]} + Re_{ev,[t-z,t+z]}} \quad (16)$$

Precise evaluation is a special case of fuzzy evaluation when $z=0$. We report both precision evaluation and fuzzy evaluation $z=1$ results in the result section.

6.4 Data

We use 8-K Current Reports filed to SEC’s EDGAR system (*the Electronic Data Gathering, Analysis, and Retrieval system*) between August of the year 2004 and December of the year 2018 as our data. Our study focuses on the Fortune 1,000 companies, and we use them as our focal companies.

Given the time constraint, we only use 200 companies data in the training set, 45 companies data in the validation set, and 45 companies data in the testing set. We split the dataset by company. In the end, we had 8,400 sequences in training, 2,304 sequences in validation, and 2,090 sequences in testing.

6.5 Preprocessing

We extracted the company name, report content, and published date from each 8-K report in EDGAR. We use Python for reports downloading and content extraction. We use Spacy ⁴ fuzzy matching to map report content to event types in Table 1.

6.6 Model Training

We train event embeddings in our model. In the experiments, we define time window i to be a month and memory size $M = 36$. We define time window q to be a month and the forecasting horizon $H = 12$. Given the event reporting nature, we

⁴<https://spacy.io/>.

use $|K| = 2$ in our experiments. We use *softmax* (Goodfellow et al., 2016) as the activation function, Adam (Kingma and Ba, 2014) as the optimizer, and Cross Entropy (CE) as the loss function.

7 Discussion

We show our precise model performance in Table 2, fuzzy model performance in Table 3, and model perplexity in Table 4.

First of all, the model performance tells us the potential to predict firm event sequences using a sequence-to-sequence neural network. The performance tables show the direction of designing the corporate event sequence prediction problem as a story completion task is promising, although there are rooms to keep improving the model.

At the same time, one thing we want to point out is the dataset is unbalanced by following business nature. In other words, some event types happen less frequently than others. For instance, intellectual property activities, litigation and lawsuit, delisting, and bankruptcy happen much less frequently than financial activities, senior personnel change, and information disclosure. Therefore, some event types didn’t generate prediction results as high as other types.

From the model results, we can see (1) the proposed sequence-to-sequence neural network models perform better than the baseline simulation model, and (2) the attention mechanism is useful on certain event type predictions as well. They both demonstrate the promising direction of the proposed problem formulation.

In both Table 2 and Table 3, we can see the sequence-to-sequence models perform better than the baseline simulation model. Moreover, when attention mechanism is added, the with attention model gains prediction performance for event types including senior personnel change, information disclosure, document updates, intellectual property activities, and delisting, in precise evaluation. The with attention model shows better performance results for business combination and restructuring, document updates, intellectual property activities, litigation and lawsuit, delisting and bankruptcy event types, in fuzzy evaluation. They show the value of the model formulation. Meanwhile, when we compare models between with attention and without attention, we can identify the usefulness of the attention mechanism on certain

Event Type	MCMC Baseline	GRU	GRU_attention
business combination and restructuring	9.68%	19.80%	16.42%
financial activities	20.16%	39.01%	36.67%
operation activities	1.78%	4.20%	3.40%
senior personnel change	20.52%	30.79%	31.71%
information disclosure	33.34%	44.41%	46.26%
document updates	4.22%	6.32%	7.90%
intellectual property activities	0.54%	0.97%	1.01%
litigation and lawsuit	1.13%	1.94%	1.78%
delisting	0.44%	0.33%	0.62%
bankruptcy	0.03%	0.15%	0.14%

Table 2: Model performance (precise evaluation, F1%)

Event Type	MCMC Baseline	GRU	GRU_attention
business combination and restructuring	22.12%	35.59%	37.34%
financial activities	42.27%	62.03%	51.33%
operation activities	3.88%	9.28%	9.23%
senior personnel change	42.71%	56.96%	51.42%
information disclosure	58.49%	70.37%	59.65%
document updates	9.56%	19.66%	19.93%
intellectual property activities	1.43%	3.03%	3.38%
litigation and lawsuit	2.74%	4.69%	4.96%
delisting	1.58%	1.66%	2.20%
bankruptcy	0.05%	0.30%	0.34%

Table 3: Model performance (fuzzy evaluation, F1%)

Event Type	MCMC Baseline	GRU	GRU_attention
Perplexity	175.28	33.15	33.32

Table 4: Model Perplexity

event types. Perplexity results in Table 4 also verifies the promising model design direction as well.

In a real business setting, there are other data streams can also participate in the decision making process, such as company fundamental values. We are working on the integration of multiple data streams as well.

8 Conclusion and Future Work

In this paper, we proposed sequence-to-sequence (Seq2Seq) models to forecast firm material event sequences, based on firm historical event sequences. The proposed deep learning model

demonstrates promising performance and design rationale for the task of predicting firm future event sequences.

However, there are still rooms to improve our models in the future. We plan to incorporate other data streams and other techniques, such as variational autoencoder (VAE) (Kingma and Welling, 2013), Transformer (Vaswani et al., 2017) and/or BERT (Devlin et al., 2018), in the model architecture. We also plan to further investigate the economic implications of our formulation and solution.

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.
- Richard A Cazier and Ray J Pfeiffer. 2015. Why are 10-k filings so long? *Accounting Horizons*, 30(1):1–21.
- Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. [Attention-over-attention neural networks for reading comprehension](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 593–602. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2014. Using structured events to predict stock price movement: An empirical investigation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1415–1425.
- Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2015. Deep learning for event-driven stock prediction. In *the 24th International Joint Conference on Artificial Intelligence*, pages 2327–2333.
- Alex Edmans. 2011. Does the stock market fully value intangibles? employee satisfaction and equity prices. *Journal of Financial economics*, 101(3):621–640.
- Lily Fang and Joel Peress. 2009. Media coverage and the cross-section of stock returns. *The Journal of Finance*, 64(5):2023–2052.
- Stefan Feuerriegel and Nicolas Prölkochs. 2018. Investor reaction to financial disclosures across topics: An application of latent dirichlet allocation. *Decision Sciences*.
- Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.
- Jing He and A. Marlene Plumlee. 2019. Measuring disclosure using 8k filings. *SSRN*.
- Rudolf Kadlec, Martin Schmid, Ondřej Bajgar, and Jan Kleindienst. 2016. [Text understanding with the attention sum reader network](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 908–918. Association for Computational Linguistics.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.
- Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.
- Paul C Tetlock. 2010. Does public financial news resolve asymmetric information? *The Review of Financial Studies*, 23(9):3520–3557.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.
- Yequan Wang, Minlie Huang, Li Zhao, et al. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.
- Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.
- Xiaofei Zhao. 2016. Does information intensity matter for stock returns? evidence from form 8-k filings. *Management Science*, 63(5):1382–1404.