

주식데이터를 활용한 Attention과 LSTM의 성능 비교

Performance Comparison of Attention and LSTM Using Stock Data

Yunsang Yoo¹ · Dong-ho Shin^{2*}

¹Student, My Paul School, Education administration,
Goesan, 28054 Republic of Korea

²Teacher, My Paul School, Education administration,
Goesan, 28054 Republic of Korea

ABSTRACT

In recent years, there has been significant progress in the fields of Attention-based Transformers[1, 2]. In this paper, we compare the performance of an Attention encoder and an LSTM[3] for the stock prediction task, evaluating the performance based on the model's size and input length. We utilize stock history data for Apple, Samsung, and Amazon, obtained from Kaggle. The Attention Encoder model outperforms the LSTM baseline model on all datasets. This demonstrates the superior performance of the Attention Encoder model. Additionally, increasing the input sequence dimension of the Attention Encoder network leads to improved performance. However, there is a concern regarding overfitting of the Attention Encoder model due to the rapid changes in the data.

Keywords : time series prediction, stock price prediction, dataset analysis, performance comparison

I. 서 론

현재 NLP(Natural-Language-Processing) 분야에선 Transformer가 가장 두각을 나타내고 있지만, Transformer[4]의 출현 이전에는 Long-Short Term

Memory(LSTM)이 NLP 분야에서 가장 많이 사용되었다. 하지만 LSTM이 가지는 기울기 소실 및 폭발과 같은 고질적인 문제에 의해 현재는 GPT[1], BERT[2]와 같은 Transformer-기반의 모델들이 많이 쓰이는 추세이다.

주식 예측은 time-series prediction 중 하나이며, 많은 현대인의 관심사로 여러 연구가 진행되었으며, 주로 순환 신경망으로 문제를 해결한 연구[5, 6]가 진행되었다. 학습의 주체는 주어진 몇 개의 주식 가격을 기반으로 그 후에 이어지는 주식값을 예측해야 하며, 모델은 입력으로 주어지는 주식 가격 간의 연관성을 파악하는 것이 중요하다.

본 연구에서는 Transformer Encoder과 LSTM 모델에서 주식 예측을 진행했을 때 어떤 네트워크의 성능이 뛰어난지를 비교하기 위해 모델의 크기, 입력의 크기를 기준으로 실험을 진행했다. 두 가지 평가 기준은 모델 크기가 커짐에 따라 성능이 증가하는지, 그리고 Transformer 기반의 모델이 입력 정보를 얼마나 활용할 수 있는지를 측정하기 위해 선정되었다.

실험 과정에서 사용된 데이터셋은 Kaggle에 공개된 Samsung, Apple, 그리고 Amazon의 주식 데이터가 사용되었으며, 파이썬 라이브러리 pandas로 numpy 형식으로 변환하는 과정과 주식 가격을 -1과 1 사이의 값으로 변환하는 전처리 과정을 포함한다.

기존의 주식 예측 문제에서 LSTM 기반의 모델을 구현한 [5]는 평가 과정에서 모델의 예측이 기존의 주식 그래프에서 크게 벗어나는 것이 관찰됐다. 본 논문에선 Transformer 기반의 모델이 주식 예측 성능을 향상시킬 수 있으며, NLP뿐만 아니라 주식 예측 문제에서도 유망한 성능을 보인다는 것을 보여준다.

II. 연구 방법

본 연구에서는 3가지 주식데이터를 사용해서 LSTM 과 Transformer Encoder 모델의 성능을 비교하기 위해

Received 29 May 2023, Revised 8 June 2023, Accepted 26 June 2023

* Corresponding Author Dong Ho Shin (E-mail: eavatar@hanmail.net, Tel: +82-2029-1309)

My Paul School, Teacher, Education administration, Goesan, 28054 Republic of Korea

Open Access <http://doi.org/10.6109/jkiice.2023.27.7.887>

print ISSN: 2234-4772 online ISSN: 2288-4165

모델 크기에 대한 평가와 입력 크기가 주는 영향을 실험하기 위한 Transformer Encoder 단독적인 평가가 진행되었다.

2.1. 예측 모델

본 연구에는 Transformer 기반의 모델과 비교하기 위한 LSTM 모델이 실험에 사용되었다, LSTM과 Gated-Recurrent-Unit(GRU)는 때에 따라 성능이 달라지기[7] 때문에 본 연구에서는 LSTM을 사용한다.

LSTM은 RNN으로부터 파생된 네트워크 구조이다. 기존의 RNN은 단순히 연속된 입력을 순차적으로 처리하는 방식을 사용하며, 입력의 크기가 커질수록 정보가 손실된다는 문제가 있었다. 이것을 장기 의존성 문제라고 한다.

LSTM은 RNN에 있었던 장기 의존성을 해결하기 위한 모델 구조이며, LSTM은 4개의 선형 게이트를 업데이트하여 이를 통해 cell에 필요한 정보를 저장하고, 불필요한 정보는 버릴 수 있다.

하지만 LSTM도 순환신경망이기 때문에 이러한 구조를 가진 구조가 가진 문제인 기울기 소실/폭발이 있다. 이를 해결하기 위한 연구가 진행되어 있지만, 여전히 LSTM은 한계가 있다.

Transformer는 기존에 순환신경망의 성능을 높이기 위해 사용된 attention network[4]로 구성된 네트워크 구조이며, 기존의 LSTM과 GRU[7]와 같은 순환적인 구조는 사용되지 않는다.

Transformer를 구성하는 Attention network는 query, key, value 세 가지 벡터값을 사용하여 scaled dot-production 연산을 수행하여 attention을 계산하며 [4], 이는 나뉜 각 헤드마다 수행된다(1).

$$\begin{aligned} & MultiHeadAttn(Q, K, V) \\ &= Concat(head_1, \dots, head_h) W^O \end{aligned} \quad (1)$$

2.2. 학습 데이터

모델을 학습하고 평가하기 위해 Kaggle 사이트에 공개된 Samsung, Apple, Amazon의 3가지 주식 데이터가 사용되었다. 실험 과정에서 모델이 주식 가격의 급격한 변화를 예측할 수 있는지 판단하기 위해 변동 폭이 큰 데이터가 사용되었다.

Samsung, Apple, Amazon의 길이는 각각 5621, 5568,

6098로 모두 5500이 넘는 데이터가 수집되었다. 주가가 급등하는 특징에 따라 모델의 예측 난이도는 높아지며, 모델의 주식 가격 예측이 성공적으로 이뤄진다면 모델의 학습이 성공적으로 진행되었다고 볼 수 있다.

2.3. 데이터 전처리

원본 데이터는 .csv 형식이므로, 파이썬 라이브러리 pandas를 활용하여 계산하기 용이한 numpy 형식으로 변환된다. 주식 가격은 -1과 1 사이의 값으로 정규화 과정을 거친다.

데이터셋은 학습/검증/평가로 각각 8:1:1로 나뉘었다. 이와 같이 데이터를 나누게 되면 평가 데이터셋에 급상승하는 부분이 배정됨으로써 과적합 현상을 관찰할 수 있다.

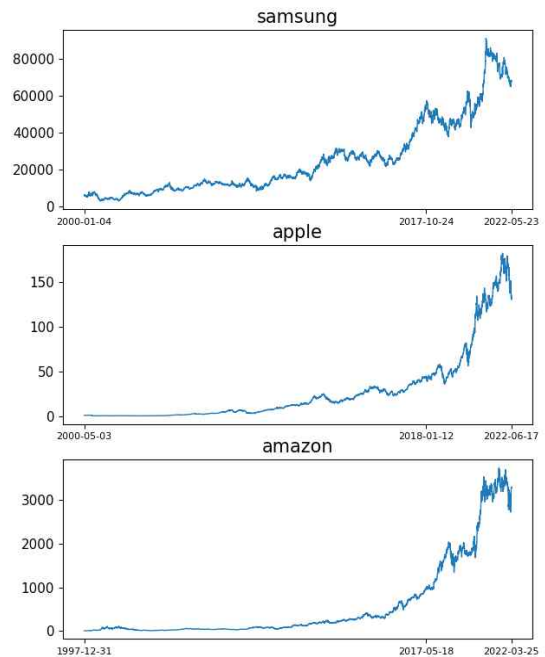


Fig. 1 Stock Data visualization(Samsung, Apple, Amazon)

III. 실험 결과

3.1. 실험 세팅

실험에 사용된 모델은 LSTM 모델과 Transformer Encoder 모델이다. 모델 오차는 MSE 오차를 사용했다.

파라미터 표시는 BERT[2]와 동일하게 하며(L은 레이어 수, H는 히든 사이즈, A는 어텐션 헤드 수) LSTM 모델 파라미터는 작은 모델(H=512, L=3), 큰 모델(H=1024, L=6)으로 구성되며 Transformer Encoder 모델은 작은 모델(H=512, A=8, L=3), 큰 모델(H=1024, A=16, L=6)으로 설정되었다.

3.2. 모델 크기에 따른 성능 평가

해당 실험은 모델 사이즈 변화로 인한 성능 변화 평가한다. 이때 입력과 출력 크기는 (120, 30)으로 고정한다. Transformer Encoder는 임의로 Encoder로 나타낸다. Loss는 1000을 곱해 나타낸다.

Table. 1 Evaluation of MSE Loss with respect to model size changes

Model	Samsung	Apple	Amazon	Average
LSTM small	51 / 73	13 / 14	12 / 4	25 / 30
LSTM large	32 / 49	42 / 24	21 / 86	11 / 10
Encode small	0.6 / 0.9	0.4 / 1.4	0.3 / 0.7	0.4 / 1
Encoder large	0.9 / 0.9	0.19 / 2.4	0.8 / 0.16	1.2 / 1.6

모든 모델은 동일하게 50 에폭 동안 학습되었으며, 실험 과정에서 LSTM은 긴 출력에 대한 예측을 하지 못하는 것을 알 수 있었다. Transformer Encoder 모델은 모두 낮은 MSE 오차값을 출력했다.

Transformer Encoder의 작은 모델이 큰 모델보다 더 낮은 오차값에 도달한 것이 확인되었는데, 학습 스텝을 80까지 증가시킨 결과 작은 모델보다 더 낮은 loss에 도달했다.

3.3. 입력 크기에 따른 성능 평가

해당 실험 파트에서는 모델의 입력 길이에 따라 Transformer Encoder 모델의 성능 변화를 평가했으며, 짧은 입력 길이에서는 모델이 제한된 입력 정보를 최대한 활용하여 주식 가격을 예측할 수 있는지 실험하고 긴 입력에서는 정보를 충분히 활용할 수 있는지, 얼마나 높은 성능에 도달하는지를 관찰한다.

실험에서 출력 길이는 30으로 고정되며, 입력 길이를

변화시킨다. 가장 짧은 입력 길이는 120, 중간 길이 170, 그리고 가장 긴 길이는 200으로 설정되었다. 해당 오차 또한 1000을 곱해 나타낸다.

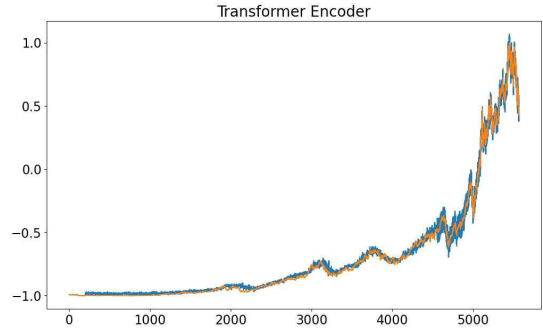


Fig. 2 Transformer Encoder output example

Table. 2 Evaluation of MSE Loss with respect to model size changes

Input length	Samsung	Apple	Amazon	Average
120-len	0.9	1.4	0.7	1
170-len	1.4	1.8	2.1	1.7
200-len	0.9	0.3	0.6	0.6

모든 입력 길이 세팅에서 200-len인 모델이 가장 낮은 오차값에 도달했으며, 모델이 긴 입력의 정보를 온전히 활용하여 성능이 증가했음을 알 수 있다.

IV. 결 론

본 논문에서는 주식 예측에서 LSTM과 Transformer Encoder 모델의 성능을 분석했다. 모델은 급격히 변화하는 주식값을 예측해야 하며, 모든 실험 과정에서 Transformer Encoder 모델의 성능이 가장 우수한 결과를 보였다. 실험 중에 Encoder-small의 오차값이 Encoder-large의 오차값이 더 높은 것이 관찰되었는데, 이는 늘어난 파라미터를 충분히 학습하기에 시간이 짧았던 것으로 추측된다. 모델의 입력 크기의 변화에 따른 성능 실험에서는 입력 길이가 120일 때보다, 200일 때 오차값이 더 낮았으며, Encoder 모델은 늘어난 입력

크기에 따른 정보를 온전히 활용할 수 있는 것으로 판단된다.

다만 BERT[2]와 같은 Transformer encoder 기반의 모델은 많은 컴퓨팅 예산이 필요로 한다. 따라서 상용화 가능한 모델을 위해 향후에 아키텍처 최적화 연구를 진행할 예정이다.

ACKNOWLEDGEMENTS

This paper was supported by my parents and advisor Dr. dong-ho Shin, who has been my invaluable mentor. I would like to express my gratitude to everyone who assisted me in writing this paper.

REFERENCES

- [1] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. (2018). Improving language understanding by generative pre-training [Internet]. Available: https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.
- [2] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, Oct. 2018. DOI: 10.48550/arXiv.1810.04805.
- [3] S. Hochreiter and J. Schmidhuber "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735-1780, Nov. 1997. DOI: 10.1162/neco.1997.9.8.1735.
- [4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiserand, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems 30 (NIPS 2017)*, [Internet]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- [5] A. Moghar and M. Hamiche, "Stock Market Prediction Using LSTM Recurrent Neural Network," *Procedia Computer Science*, vol. 170, no. 1, pp. 1168-1173, Apr. 2020. DOI: 10.1016/j.procs.2020.03.049.
- [6] Y. Wen, P. Lin, and X. Nie, "Research of stock price prediction based on PCA-LSTM model," in *IOP Conference Series: Materials Science and Engineering*, vol. 790, no. 1, p. 012109, Mar. 2020. DOI: 10.1088/1757-899x/790/1/012109.
- [7] J. Chung, C. Gulcehre, K. H. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling" *arXiv preprint arXiv:1412.3555*, Dec. 2014. DOI: 10.48550/arXiv.1412.3555.