# 주가 예측을 위한 어텐션 메커니즘의 비교분석

유연국*, 천용상*, 조민희*, 김윤중*
*한밭대학교 컴퓨터공학과
e-mail:ryk012@naver.com

# Comparison and Analysis of the Attention Mechanism for Stock Prediction

Yeonguk Yu*, Yongsang Cheon*, Min-Hee Cho*, Yoon-Joong Kim*
*Dept of Computer Engineering, Hanbat National University

요 약

주가 예측은 상업적인 매력 때문에 많은 이목이 끌리는 분야이지만, 주가의 불확실성과 변동성 때문에 주가 예측은 어려운 작업이다. 최근에는 주가 예측 모델에 어텐션 메커니즘을 사용하여 주가 예측에 많은 인자들이 사용되어 생기는 성능 하락 문제를 해결하여 좋은 성능을 보여주는 연구가 존재한다. 본 연구에서는 그 모델 중 하나인 Dual-Stage Attention-Based Recurrent Neural Network (DARNN)의 어텐션 메커니즘을 변경해가며 어떤 어텐션 메커니즘이 주가 예측에 적합한지를 알아본다. KOSPI100 지수의 예측실험을 통해 location 스코어함수를 사용한 어텐션 메커니즘이 가장 뛰어난 성능을 보여주는 것을 확인하였고, 이는 기존의 스코어함수를 사용한 DARNN에 비해 약 10% 향상된 성능으로 스코어 함수가 모델의 중요한 영향을 끼치는 것을 확인하였다.

## 1. 서론

주식시장에 대한 예측 기술은 상업적인 매력 때문에 여러 분야의 연구자들이 관심을 갖고 있는 분야이다. 그렇지만 주식의 불확실성과 비선형성 그리고 여러 노이즈가 섞여 있는 주식시장의 환경 때문에 주가를 예측하는 것은 어려운 작업으로 여겨진다. 여러가지 기계학습 알고리즘이 주식시장의 예측을 위해 사용되고 있지만, 그 중에서도 인공신경망이 많이 사용된다. 인공신경망의 입력과 출력의 복잡한 비선형 매핑을 수행할 수 있는 능력과 노이즈에 대한 내성은 이미 잘 확립되어 있고[1], 여러 연구[2, 3]에서 인공신경망을 사용하여 주식시장의 예측 시스템을 제안하였다.

Recurrent neural networks (RNNs)은 순차적, 시간적 변화 패턴을 학습할 수 있도록 설계되어 있는 인공신경망의 한 종류이다[4]. 그 중에서도 Long short-term memory (LSTM) [5]은 RNN의 여러 구조들 중 대표적으로 사용하고 있는 구조이다. Jia의 연구[6]에서는 LSTM이 주가 예측 작업에서 가지는 장점에 대해 조사하였다. 그리고 [1]에서는 LSTM을 사용한 모델이 moving average, exponential moving average, support vector machine을 사용한 모델에 대해 우수성을 보이는 것을 확인하였다. 하지만 최근의 연구[7, 8]에서는 LSTM만을 사용한 모델의 경우, 주가와의 낮은 상관관계를 가지고 있는 인자들로 생기는 성능하락 문제가 있는 것을 보여준다.

[8]의 연구에서는 기존의 기계번역 연구[9]에서 사용되었던 어텐션 메커니즘을 기반으로 하여 낮은 상관 관계의 인자들로부터 발생하는 성능하락을 막을 수 있는 강화된 LSTM을 제안하였다. 이 연구에서는 어텐션 메커니즘으로 강화된 LSTM을 사용한 dual-stage attention-based recurrent neural network (DARNN)을 제안하였으며, 주가 예측 및 실내 온도 예측에서 좋은 성능을 보여주는 것을 확인하였다.

어텐션 메커니즘은 모델이 입력과 출력의 연관관계를 계산할 수 있도록 하는 일종의 방법이다. 어텐션 메커니즘을 사용한 모델은 출력을 만들기 위해 중요한 입력이 무엇인지 판단하기 위해 점수를 계산하는 스코어 함수를 사용한다. 어텐션 메커니즘은 [9]의 연구에서 기계번역을 위해 소개된 이후, 다양한 분야에서 다양한 변형으로 응용되어 사용되고 있다. 예를 들어, [10]의 연구에서는 주어진 그림에 대한 설명을 만드는 이미지 캡션 작업에서 어텐션 메커니즘을 사용했으며, [11]의 연구에서는 기계번역을 위해 사용되는 다양한 어텐션 메커니즘 및 스코어 함수를 비교하였다. 위처럼 다양한 어텐션 메커니즘 및 스코어 함수는 다양함에도 불구하고, 주가 예측을 위해서 어떤 스코어 함수를 사용해야 하는지에 대한 연구는 없는 것 보인다.

본 연구에서는 어떠한 스코어 함수를 사용한 어텐션 메커니즘이 주가 예측을 위해 최적의 성능을 낼 수 있는지 알아보기 위한 실험을 진행하였고 결과를 통해 스코어 함수를 비교하였다. 실험에서는 위에서 언급된 DARNN모델에 사용된 스코어 함수를 여러 스코어 함수로 교체해보며 실험을 진행하였다.

논문의 나머지 내용은 다음과 같이 구성되어 있다. 2절에서는 예측 문제 정의, 본 연구의 모델로 사용된 DARNN에 대해 설명한다. 3절에서는 실험에 사용된 스코어 함수들에 대해 설명한다. 4절에서는 실험환경 및 데이터에 대해 설명한다. 5절에서는 실험결과 및 그에 대한 분석을 한다. 6절에서는 본 논문의 결론을 이끌어낸다.

## 2. 문제 정의 및 DARNN 모델

예측 모델의 입력은 주식 종목들의 가격 데이터인 $\boldsymbol{X}$와 주식 종목들로부터 계산된 지수의 데이터인 $\boldsymbol{y}$이다. 종목들의 가격 데이터인 $\boldsymbol{X} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_T})$는 $T$개의 종목들의 데이터 모음이며, $T$는 예측 모델의 윈도우 사이즈이다. $\boldsymbol{X}$는 원소 $\boldsymbol{x_t} = (x_t^1, x_t^2, \dots, x_t^K)$로 구성되어 있으며, $K$는 예측에 사용되는 종목들의 총 개수이고 $x_t^k$는 k 번째 종목의 날짜 $t$에서의 종가 데이터이다. 지수 데이터인 $\boldsymbol{y} = (y_1, y_2, \dots, y_T)$로 종목들로 계산되는 지수의 T일간의 종가 데이터이다. 예측 모델의 목표는 입력 $\boldsymbol{X}$, $\boldsymbol{y}$를 사용하여 다음날의 지수 종가 $y_{t+1}$를 예측하는 것이므로 식 1과 같이 표기될 수 있다.

$$\overline{y_{t+1}} = F(\boldsymbol{y}, \boldsymbol{X}) \tag{1}$$

그러므로 F()가 예측 모델로, 훈련하고자 하는 맵핑함수이다.

[8]의 연구에서는 LSTM을 기초로 하여 입력되는 벡터를 어텐션 메커니즘으로 강화하는 DARNN 모델을 제안하였다. DARNN은 인코더와 디코더라고 불리는 어텐션 메커니즘을 응용한 LSTM이 사용되는 모델이다. 본 연구의 실험에서도 같은 모델을 사용하였으며, 스코어 함수만을 변경해보며 실험을 진행하였다. DARNN에 사용되는 어텐션 메커니즘으로 강화된 LSTM의 계산과정은 다음과 같다:

$$e_t^k = v_e^\top \tanh(W_e[h_{t-1}; s_{t-1}] + W_x x^k) \tag{2}$$

$$\alpha_t^k = \frac{\exp(e_t^k)}{\sum_{i=1}^{K} \exp(e_t^i)} \tag{3}$$

$$\overline{x_t} = (\alpha_t^1 x_t^1, \alpha_t^2 x_t^2, \dots, \alpha_t^K x_t^K) \tag{4}$$

2-4의 식은 LSTM의 입력 벡터를 강화하는 과정으로 어텐션 메커니즘을 나타낸다.

$$f_t = \sigma(W_f[h_{t-1}; \overline{x_t}] + b_f) \tag{5}$$

$$i_t = \sigma(W_i[h_{t-1}; \overline{x_t}] + b_i) \tag{6}$$

$$o_t = \sigma(W_o[h_{t-1}; \overline{x_t}] + b_o) \tag{7}$$

$$s_t = f_t \circ s_{t-1} + i_t \circ \tanh(W_s[h_{t-1}; \overline{x_t}] + b_s) \tag{8}$$

$$h_t = o_t \circ \tanh(s_t) \tag{9}$$

5-9의 식은 4번의 식에서 얻어진 강화 입력 벡터를 사용하여 기존의 LSTM과 같은 계산을 통해 히든스테이트 $s_t$와 LSTM의 출력 $h_t$를 만들어내는 과정이다. 위의 식에서 $\sigma$, $\circ$, []는 각각 sigmoid 함수, 요소간 곱셈, concatenation을 나타내며 $W$로 표기된 행렬은 훈련에서 업데이트되는 웨이트 파라메터이고, $b$로 표기된 벡터또한 훈련을 통해 업데이트되는 웨이트 파라메터이다. DARNN에서는 2단계의 강화 LSTM을 거친 후, fully connected layer를 사용하여 최종적인 모델의 출력 $\overline{y_{t+1}}$을 얻어낸다.

## 3. 스코어 함수

DARNN에서 사용된 어텐션 메커니즘은 식 2-4이다. 하지만 다양한 분야에서 다양한 어텐션 메커니즘이 사용되고 있고, 이를 테스트해보기 위해 사용한 스코어 함수는 5가지로 표1에 설명되어 있다.

<표 1 > 스코어 함수의 종류 (Basic : DARNN모델의 기본 스코어 함수, Additive : [9]에서 사용된 스코어 함수, Content : [12]에서 사용된 스코어 함수로 cosine_distance은 코사인 디스턴스의 계산 함수를 나타냄, Dot, Location : [11]에서 소개된 스코어 함수)

| function name | function |
|---|---|
| Basic | $e_t^k = v_e^\top \tanh(W_e[h_{t-1}; s_{t-1}] + W_x x^k)$ |
| Additive | $e_t^k = v_e^\top \tanh(W_e[h_{t-1}; x^k])$ |
| Content | $e_t^k = \text{cosine_distance}(W_h h_{t-1}, x^k)$ |
| Dot | $e_t^k = (W_h h_{t-1})^\top x^k$ |
| Location | $e_t^k = (W_a h_{t-1})^k$ |

표1에 나와있는 각각의 스코어 함수는 모델의 계산과정에서 식 2를 대신하여 사용될 수 있다. Basic을 사용한 경우, DARNN 모델과 동일하게 계산되었으며, Location 스코어 함수의 경우는 LSTM의 이전 타임스텝의 출력만이 스코어 함수에 사용되었다. $W$로 표기된 행렬은 훈련에서 업데이트 되는 웨이트 파라메터이다. bias는 간결함을 위해 생략되었다.

## 4. 실험 데이터 및 실험 환경

모델의 주식 예측 성능을 테스트하기 위해 코스피 주식시장의 데이터를 사용하였다. 전체 데이터의 길이는 2011-3-28일부터 2019-5-10일 까지의 코스피시장이 열렸을 때 가져온 2000일이다. 예측에 사용되는 지수 데이터로는 코스피100 지수를 사용하였으며, 종목들은 코스피 시장의 2019-5-10일에 상위 200종목중 2000일간의 데이터가 존재하는 171개 종목의 데이터를 사용하였다. 만들어진 2000일간의 데이터를 8:2의 비율로 나누어 각각을 훈련용, 평가용으로 사용하였다. 각각의 데이터세트에 대한 정보는 표 2에 나타나 있다.

<표 2> 훈련용 데이터세트 및 평가용 데이터세트

| Dataset | Using Stock | Target | Size | Start date | End date |
|---|---|---|---|---|---|
| Training Set | 171 | Kospi100 index | 1600 | 2011-03-28 | 2017-09-27 |
| Evaluation Set | 171 | Kospi100 index | 390 | 2017-09-28 | 2019-05-10 |

표2에서 볼 수 있는 것과 같이 훈련용 데이터세트는 1600개의 입력데이터 및 타겟, 평가용 데이터세트는 390개의 입력데이터 및 타겟으로 구성되어 있다. 입력데이터는 10일을 윈도우 사이즈로 설정하였으며, 1일씩 나누어 입력데이터 및 타겟을 만들어 구성하였다.

추가적으로 어텐션 메커니즘의 성능을 알아보기위해 어텐션 메커니즘이 사용되지 않은 LSTM 1층과 fully connected layer 1층으로 구성된 모델도 실험에 사용하였다. LSTM모델의 1층 LSTM의 cell unit은 DARNN모델에 사용된 LSTM의 cell unit과 같이 64개로 설정하였으며, fully connected layer는 LSTM의 마지막 타임스텝의 출력을 입력으로 받아 1개의 출력을 내놓도록 설정되어 있다.

실험은 스코어 함수를 변경한 각 모델에 대하여 4번씩 반복되었으며, 각각의 실험은 훈련용 데이터세트를 사용하여 모델을 훈련한 후, 평가용 데이터 세트를 사용하여 모델의 성능을 계산하였다. 모델의 성능은 root mean square error (RMSE)로 나타내었으며, 자세한 식은 식(10)에 나타나 있다.

$$\mathrm{RMSE}(Y_{t+1}, \overline{Y}_{t+1}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{t+1,i} - \overline{y}_{t+1,i})^2} \quad (10)$$

실험에 사용된 Adam optimizer의 learning rate은 0.01로 설정하였다. 최종적으로 모델의 성능은 500epoch동안 평가용 데이터세트에 대해 얻어진 가장 낮은 RMSE를 선택하였다. 실험은 python 환경에서 tensorflow 프레임워크를 사용하여 프로그래밍 되었으며, 엔비디아 GTX1070 그래픽카드를 장착한 데스크탑에서 실험이 진행되었다.

## 5. 실험 결과

4절에서 설명한 바와 같이 진행된 실험의 결과는 표3에 나타나 있다.

<표 3> 스코어함수별 RMSE값 (평균값은 볼드체로 표기됨)

| Model | Basic | Additive | Content | Dot | location | LSTM |
|---|---|---|---|---|---|---|
| 1 | 25.227 | 22.815 | 23.363 | 32.563 | 20.942 | 26.631 |
| 2 | 21.723 | 23.591 | 22.580 | 21.866 | 22.231 | 29.984 |
| 3 | 26.183 | 36.579 | 22.267 | 26.084 | 21.585 | 34.432 |
| 4 | 23.204 | 26.991 | 21.039 | 25.900 | 22.119 | 37.709 |
| average | **24.085** | **27.494** | **22.312** | **26.603** | **21.719** | **32.189** |

표 3에서 볼 수 있는 스코어함수별 모델의 RMSE값을 확인해보면 location 스코어 함수를 사용한 모델이 가장 좋은 성능을 보여주는 것을 확인할 수 있다. 이는 어텐션 메커니즘이 사용되지 않은 LSTM모델에 대해서는 32.53% 향상된 성능이며, 기존의 DARNN모델에 대해서는 9.82% 향상된 성능이다. 어텐션 메커니즘을 사용한 모든 모델은 LSTM에 비해 크게 향상된 성능을 보여주었는데, 이를 통해 어텐션 메커니즘을 통해 모델이 중요한 종목을 선택할 수 있도록 하여 낮은 상관관계를 가진 인자로부터 성능향상을 방지하여 예측 성능을 향상시키는 것을 확인 할 수 있다. 또한 스코어 함수에 따라 예측 성능이 달라지는 것을 확인하였는데, 가장 좋은 성능을 보여주었던 location 스코어 함수를 사용한 모델은 다른 스코어 함수를 사용한 모델이 4번의 실험에서 약간씩 차이가 있는 성능을 보여준 것과 달리 거의 일관되게 좋은 성능을 보여주었다. 이는 입력 값이 스코어 함수에 사용되지 않고, 이전 타임스텝의 히든스테이트만 사용하는 단순한 구조로 되어있는 location 스코어 함수를 사용함에 따라 훈련과정에서 웨이트 파라메터의 최적화가 다른 스코어 함수에 비해 쉽게 되고 있는 것을 뜻하는 것으로 보인다.

## 6. 결론

본 연구에서는 DARNN 모델의 어텐션 메커니즘에 사용되는 스코어 함수를 다양하게 변경한 후, 주식 지수 예측 실험을 통해 스코어 함수가 모델의 성능에 영향을 얼마나 미치는 지를 확인하였다. 실험에서는 KOSPI 100지수의 예측을 위해 코스피 상위 200종목의 주식 데이터를 사용하였으며, 비교 대상으로 기초적인 LSTM모델과 dot, additive, location, content 등 다른 연구에서 사용되었던 스코어 함수를 사용한 모델을 선택하였다. 실험결과를 통해 location 스코어 함수를 사용한 모델이 가장 좋은 성능을 보여주었으며, LSTM모델과 기존 DARNN 모델에 비해 각각 약 30%, 약 10%의 성능향상을 보여주는 것을 확인할 수 있었다. 이를 통해 스코어함수 및 어텐션 메커니즘이 모델의 예측 성능에 중요한 역할을 하는 것을 확인하였다. 또한 다른 스코어 함수와는 달리 입력 값 벡터를 사용하지 않고 히든스테이트만 사용하는 비교적 단순한 구조로 된 스코어 함수를 사용함으로써, 웨이트 파라메터의 최적화가 더 잘 이루어질 수 있었던 것을 알 수 있었다.

향후에는 location 스코어 함수가 단순히 주가 예측뿐만 아니라 미세먼지 예측, 비즈니스에서의 매출액 예측등 다른 시계열 예측 분야에서도 좋은 성능을 보여주는지를 확인해볼 수 있을 것이다. 또한 location 스코어 함수의 성능을 상회할 수 있는 새로운 스코어 함수에 대해서 연구해 볼 수 있을 것이다.

## 참고문헌

[1] Gao, T., Chai, Y., Liu, Y. “Applying long short term memory neural networks for predicting stock closing price” In Proceedings of the 2017 8th IEEE International Conference on Software Engineering and

Service Science (ICSESS), 2017.
[2] Yaqub M U, Al-Ahmadi M S. “Application of Combined Arma-Neural Network Models to Predict Stock Prices”. The Multidisciplinary International Social Networks Conference on Socialinformatics 2016.
[3] Jabin S. “Stock Market Prediction using Feed-forward Artificial Neural Network” International Journal of Computer Applications, 2014.
[4] Jain L C, Medsker L R. “Recurrent Neural Networks: Design and Applications” Joint Conference on Neural Networks. 1999.
[5] Hochreiter S., Schmidhuber J. “Long short-term memory” Neural Computation, 1997.
[6] Jia H. “Investigation into The Effectiveness of Long Short-Term Memory Networks for Stock Price Prediction” . arXiv:1603.07893, 2016.
[7] Hao Li, Yanyan Shen, and Yanmin Zhu. “Stock Price Prediction Using Attention-based Multi-Input LSTM” Proceedings of Machine Learning Research, 2018
[8] Y. Qin, D. J. Song, H. Cheng, W. Cheng, G. Jiang, and G. W. Cottrell. “A dual-stage attention-based recurrent neural network for time series prediction,” Proceedings of the 26th International Joint Conference on Artificial Intelligence, 2017.
[9] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. “Neural machine translation by jointly learning to align and translate” CoRR, abs/1409.0473, 2014.
[10] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. “Show, attend and tell: Neural image caption generation with visual attention“ In International Conference on Machine Learning (ICML), 2015.
[11] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. “Effective approaches to attentionbased neural machine translation”. arXiv preprint arXiv:1508.04025, 2015
[12] Alex Graves, Greg Wayne, and Ivo Danihelka. “Neural Turing machines” arXiv preprint arXiv:1410.5401, 2014.