YNU-HPCC at SemEval-2019 Task 8: Using A LSTM-Attention Model for Fact-Checking in Community Forums

Peng Liu, Jin Wang and **Xuejie Zhang**
School of Information Science and Engineering
Yunnan University
Kunming, P.R. China
Contact:xjzhang@ynu.edu.cn

Abstract

The objective of the task, Fact-Checking in Community Forums , is to determine whether an answer to a factual question is true, false, or whether it even constitutes a proper answer. In this paper, we propose a system that uses a long short-term memory with attention mechanism (LSTM-Attention) model to complete the task. The LSTM-Attention model uses two LSTM(Long Short-Term Memory) to extract the features of the question and answer pair. Then, each of the features is sequentially composed using the Attention mechanism, concatenating the two vectors into one. Finally, the concatenated vector is used as input for the MLP (Multi-Layer Perceptron) and the MLP's output layer uses the softmax function to classify the provided answers into three categories. This model is capable of extracting the features of the question and answer pair well. The results show that the proposed system outperforms the baseline algorithm.

1 Introduction

Many questions pertaining to various fields are posted to QA forums by users every day, where they collect answers. However, the answers do not always address the question asked. Indeed, in some cases, the answer has nothing to do with the question. There are several reasons why this is the case. For example, the responder could have misunderstood the question and so provided a wrong answer. Most QA forums have little control over the quality of the answers posted. Moreover, in our dynamic world, the true answer was true in the past, but it may be false now . Figure 1 presents an example from the Qatar Living forum. In this case, all three answers could be considered to be good since they formally answer the question. Nevertheless, a1 contains false information, whereas a2 and a3 are correct, as can be established from the official government website.

q: I have heard its not possible to extend visit visa more than 6 months? Can U please answer me.. Thankzzz...

a_1: Maximum period is 9 Months....

a_2: 6 months maximum

a_3: This has been anwered in QL so many times. Please do search for information regarding this. BTW answer is 6 months.

Figure 1: An example from the Qatar Living forum

In this study, we aim to solve the problem of detecting true factual information in online forums. Given a question requesting factual information, the goal is to classify the provided answers into the following categories.

(i) Factual - True: The answer is true and can be proved by cross referencing with an external resource.

(ii) Factual - False: The answer gives a factual response, but it is either false, partially false, or the responder is uncertain about their response.

(iii) Non-Factual: The answer does not provide factual information relevant to the question; it is either an opinion or an advice that cannot be verified.

To the best of our knowledge, various approaches have been proposed for the purposes of fact-checking in community forums (Mihaylova et al., 2018), such as long short-term memory (Gers et al., 2000) .

In this paper, we provide an LSTM-Attention model for fact-checking in community question answering forums. In our approach, we use pretrained word vectors for word embedding. The LSTM layer is used to extract features from the question and answer sentences. Finally, these features are used by the Attention Mechanism (Vaswani et al., 2017) with a focus on extracting useful information from the features that are significantly relevant to the current output.

The remainder of this paper is organized as fol-

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 1180–1184
Minneapolis, Minnesota, USA, June 6–7, 2019.

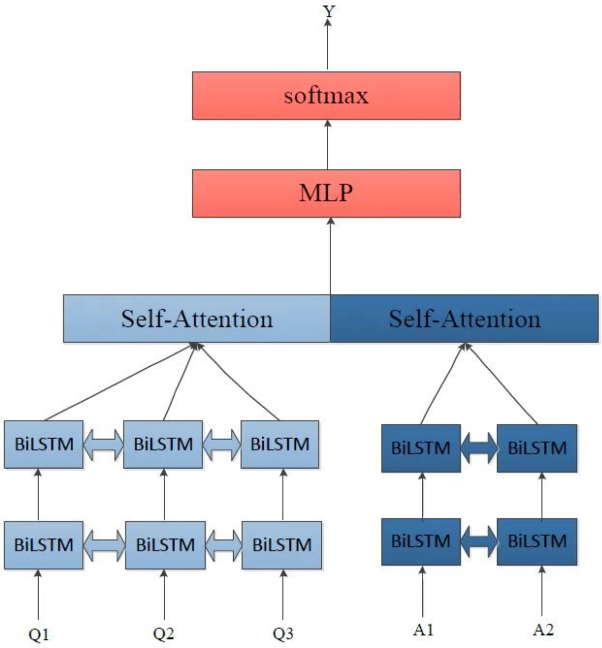

Figure 2: LSTM-Attention Model

lows. In section 2, we described the LSTM, Attention model, and their combination. Section 3 summarizes the comparative results of the proposed model against the baseline algorithm. Section 4 concludes the paper.

2 LSTM-Attention Model for Fact-Checking

Figure 2 shows the architecture of our model. First, a sentence is transformed into a feature matrix. The feature matrix is then passed into the LSTM to extract salient features.

A simple tokenizer is used to transform each sentence into an array of tokens, which constitute the input to the model. This is then mapped into a feature matrix or sentence matrix by an embedding layer. The n-gram features are extracted when the feature matrix passes through the LSTM, and the output of the LSTM is passed into the Self-Attention layer. This layer composes the useful features to output the final regression results by means of a linear decoder.

2.1 Embedding Layer

Vectors encoded using the one-hot method have large dimensions and are sparse. Suppose we encounter a 2,000-word dictionary in natural language processing (NLP). When the one-hot method is used for coding, each word will be represented by a vector containing 2,000 integers. If the dictionary is larger, this method will be very inefficient.

The one-hot-vector method has many defects when used for word encoding. One is that it has too much redundancy; the other is that the dimension of the vector is too high. The vector will have as many dimensions as there are words, which will increase the computational complexity. Word-embedding Mikolov et al. (2013) transforms an original high-dimensional redundant vector into a low-latitude vector with strong information content. No matter how many words there are, the converted vector generally has only 256 dimensions to 1024 dimensions.

The embedding layer is the first layer of the model. Each sentence is regarded as a sequence of word tokens $t_1, t_2...t_n$, where n is the length of the token vector.

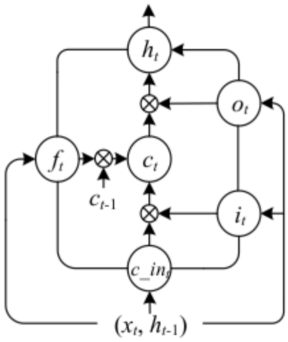

Figure 3: LSTM

2.2 Long Short-Term Memory

In theory, RNN Tsoi and Back (1994) should be able to handle such long-term dependency. We can pick and choose the parameters carefully to solve the most elementary form of this type of problem (Le et al., 2015). However, in practice, RNN is not able to learn this knowledge successfully. Therefore, the LSTM was designed to solve the problem of long-term dependency. In practice, the LSTM excels at dealing with long-term dependency information rather than the ability to acquire it at great cost. RNN has a chain of repeating neural network modules. In standard RNN, the repeating module has a very simple structure. LSTM has the same structure, but the structure of repeating modules is more complex. This is different from that of the single neural network layer. Figure 3 shows the detailed structure of an LSTM. The LSTM cal-

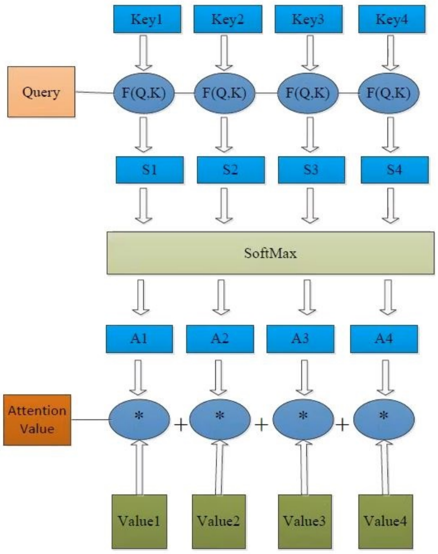

Figure 4: Attention

culates hidden states H_t and outputs C_t using the following equations.

• Gates:

$$\begin{aligned} i_t &= \sigma\left(W_{xi}x_t + W_{xi}h_{t-1} + b_i\right) \\ f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \\ o_t &= \sigma\left(W_{xo}x_t + W_{xo}h_{t-1} + b_o\right) \end{aligned} \tag{1}$$

• Input transformation:

$$c\_in_t = tanh\left(W_{xi}x_t + W_{xi}h_{t-1} + b_{in}\right) \tag{2}$$

• State update:

$$\begin{aligned} c_t &= f_t \otimes c_{t-1} + i_t \otimes c\_in_t \\ h_t &= o_t \otimes tanh\left(c_t\right) \end{aligned} \tag{3}$$

Here, x_t is the input vector; c_t is the cell state vector; W and b are layer parameters; f_t , i_t , and o_t are gate vectors; and σ is the sigmoid function. Note that $\otimes$ denotes the Hadamard product. Bidirectional LSTM comprises a forward LSTM and a reverse LSTM. It captures context feature information very well as compared to LSTM. Therefore, bidirectional LSTM usually performs better than LSTM and we use it to process the sequences. Among the many hidden layers of deep neural networks, the earlier layers learn simple low-level features, and later layers combine simple features to predict more complex things. Therefore, we use several hidden layers to make predictions more accurate.

2.3 Attention Mechanism

The concept of the Attention mechanism came from the human visual attention mechanism (Butterworth and Cochran, 1980). When people perceive things visually, they usually do not observe the scene end-to-end. Instead, they tend to observe specific parts according to their needs. When people find that a scene has something they want to observe in a certain part, they will learn to pay attention to that part in the future when similar scenes appear. With RNN or LSTM, the information accumulation of several time steps is needed to connect long-distance interdependent features. However, the longer the distance is, the less likely it is to be captured effectively. In the Attention calculation process, the connection between any two words in a sentence is directly established through one calculation step. Thus, the distance between long-distance dependent features is greatly shortened, which is conducive to the effective use of these features. Obviously, it is easier to capture the long-distance interdependent features in sentences after the introduction of Attention. In figure 4, self attention can be described as mapping a query and a set of key-value pairs to an output. The calculation of Attention is mainly divided into three steps. The first step calculates the similarity between query and each key to get the weight. The second step uses a softmax function (Jean et al., 2015) to normalize these weights. Finally, the weight and the corresponding key value are weighted and summed to get the final Attention. Currently, in NLP research, the key and value are always the same, that is, key=value. In this part, we use self-attention, which is denoted as key=value=query (Firat et al., 2016).

$$\begin{aligned} &Attention(Q, K) = \\ &\sum_{n=1}^{n} Similarity(Q, K_i) * V_i \end{aligned} \tag{4}$$

2.4 MLP Layer

This layer is a fully connected layer that multiplies the results of the previous layer with a weight matrix and adds a bias vector. The ReLU (Jarrett et al., 2009) activation function is also applied in this layer. The final result vectors are finally input to the output layer.

2.5 Output Layer

This layer outputs the final classification result. It is a fully connected layer that uses softmax as an activation function.

3 Experiments and Evaluation

3.1 Data Preprocessing

The organizers of the competition provided the training data that included one question and a number of answers. Each of answer was to be classified into the categories: (Factual - TRUE, Factual - FALSE, Non-Factual). We extracted the questions and corresponding answers, and then concatenated them into the form of a question-answer pair. As all of the data was provided by the ”Qatar Living” forum, the content primarily contained English text, and all non-english characters were ignored. We converted all letters into lower case to accommodate the known tokens in word2vec pretrained word vectors. We counted the sentence length of questions and answers. Most of them were no more than 80 words. Therefore, we set the length of the sentence to 80 words. The word2vec pretrained data was used to initialize the weight of the embedding layer. word2vec is a popular unsupervised machine learning algorithm to acquire word embedding vectors. We used 100-dimension word vectors to initialize the weight of the embedding layer.

3.2 Implementation

We used Keras with TensorFlow backend. The hyper-parameters were tuned in train and dev sets using the scikit-learn grid search function that can iterate through all possible parameter combinations to identify the one that provides the best performance. The optimal parameters found are as follows. The LSTM layer count is 2, and the dimension of the LSTM hidden layer (d) is 200. The dropout rate is 0.3. The training has a batch size of 128 and runs for 30 epochs. The results also revealed that the model using pre-trained word2vec vectors and an Adam optimizer achieved the best performance.

3.3 Evaluation Metrics

The system was scored based on Accuracy, macro-F1, and AvgRec where the ”Factual - True” instances were considered to be positive, and the remaining instances to be negative.

3.4 Results and Discussion

To prove the advantages of our system architecture, we ran a 6-fold cross validation on different sets of layers. On training data, the trial data experiment results shown in Table 1:

Model	F_1-score
CNN	0.483
LSTM	0.498
BiLSTM	0.514
BiLSTM-Attention	**0.548**

Table 1: The trial data experiment results.

Our system achieved 0.548 accuracy on Subtask B. The evaluation results revealed that our proposed system showed considerable improvement over the average baseline, which we attribute to our LSTM with Attention architecture. Our system can effectively extract features from question and answer. Using this, prediction can be made on whether the answers are actually factual and whether the fact is true or not.

4 Conclusion

In this paper, we described our submission to the SemEval 2019 Workshop Task 8, which involved Fact-Checking in Community Forums. The proposed LSTM-Attention model combines LSTM and Attention. LSTM extracts local information within both the answer and question. The Attention Mechanism resolves the issue of poor learning effect on the long input sequence. The official results reveal that our system output performed all baseline algorithms and ranked 9th on Subtask B. In future work, we will query a search engine to fetch relevant documents from the Internet to achieve an improved classification system.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants No.61702443 and No.61762091, and in part by Educational Commission of Yunnan Province of China under Grant No.2017ZZX030. The authors would like to thank the anonymous reviewers and the area chairs for their constructive comments.

References

George Butterworth and Edward Cochran. 1980. Towards a mechanism of joint visual attention in human infancy. *International Journal of Behavioral Development*, 3(3):253–272.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism.

F. A. Gers, J Schmidhuber, and F Cummins. 2000. Learning to forget: continual prediction with lstm. *Neural Computation*, 12(10):2451–2471.

Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann Lecun. 2009. What is the best multi-stage architecture for object recognition? volume 12.

Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for neural machine translation. *Computer Science*.

Quoc V. Le, Navdeep Jaitly, and Geoffrey E. Hinton. 2015. A simple way to initialize recurrent networks of rectified linear units. *Computer Science*.

Tsvetomila Mihaylova, Preslav Nakov, Lluis Marquez, Alberto Barron-Cedeno, and James Glass. 2018. Fact checking in community forums.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Computer Science*.

A C Tsoi and A D Back. 1994. Locally recurrent globally feedforward networks: a critical review of architectures. *IEEE Transactions on Neural Networks*, 5(2):229–39.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.