Sarcouncil Journal of Engineering and Computer Sciences

**ISSN(Online): 2945-3585**

Volume- 03| Issue- 5| 2024

**Research Article** **Received:** 27-03-2024| **Accepted:** 15-04-2024 | **Published:** 28-05-2024

# Sentence Representation Using Lstm for Finding Question

***Dinh Khanh Linh***

*Thai Nguyen University of Information and Communication Technology, Thai Nguyen, Vietnam*

**Abstract:** Learning sentence representation with semantic fulls of document is a challenge in natural language processing problems because if the semantic representation finding vector of the sentence is good, it will increase the performance of similar question problems. In this paper, we propose to implement a series of LSTM models with different ways of extracting sentence representations and apply them to question retrieval for the purpose of exploiting hidden semantics of sentences. T hese methods give sentence representation from hidden layers of the LSTM model. The results show that the technique using a combination of both Maxpooling and Meanpooling gives the highest results on the 2017 semeval dataset for the problem of finding similarity questions.

**Keywords:** LSTM, NLP, Deep Learning, QA, CQA.

## INTRODUCTION

Finding similar questions in community Q&A (cQA) systems is one of the intractable problems in Natural Language Processing and has attracted much attention from researchers and industry recently. There are many web forums such as Stack Overflow (https://stackoverflow.com/) and Qatar Living (https://www.qatarliving.com/forum), which are becoming popular and versatile to provide information to users (Zhou, G. *et al.*, 2013). Users can post questions and potentially receive many answers from others. In order for users to automatically receive answers from existing answers in existing questions, the problem of finding similar questions is posed. This is why it is necessary to build a tool that automatically finds questions related to the query.

The problem of finding related questions is defined as follows: Given a new question q and a set of existing questions in the data warehouse $\{q_1, q_2, \ldots, q_n\}$, the output requires a response. about a list of questions similar to q such that the most relevant questions come before the less relevant questions.

Previous studies (Zhou, G. *et al.*, 2015) have shown that the biggest challenge of this problem is the vocabulary gap. That means the usage of the words and phrases of the first question is different from the words and phrases of the second question even though the two sentences have the same meaning. Below is an example of two questions that are considered similar despite the different wording taken from the semeval 2017 dataset(Cai, L. *et al.*, 2011; Wu, W. *et al.*, 2018).

*Question 1: where can I buy good oil for massage? Question 2: Hi there, I can see a lot of massage centers here, but I dont which one is better. Can someone help me which massage center is good…and how much will it cost me? Tks.*

These two questions have the same meaning but have different interpretations. In question number 2, there is also a lot of content that explains the question and has a spoken tone, containing many abbreviations. A key challenge of this task lies in the complex and flexible semantic relationships observed between the question and the passage question. In the example above, question 1 has only 11 words, while question 2 uses 39 words to explain. On the other hand, question number 2 contains a group of words that includes information that is not directly related to the question. Additionally, while a good answer should be related to the question, they often do not share common lexical units. This problem can confuse simple word association systems. Therefore, these challenges make handcrafted features much less desirable than deep learning approaches. Furthermore, they also require our system to learn to distinguish useful parts from irrelevant parts and, furthermore, to focus more on the former.

This problem is often approached as a pairwise ranking problem; the best strategy for capturing question-to-question associations remains a matter of research. Established approaches often suffer from the following weaknesses: First, previous work, such as (Gheibi, O. *et al.*, 2021; Moravvej, S. V. *et al.*, 2021) uses convolutional neural networks (CNN) or recurrent neural networks ( RNN) respectively. However, CNN emphasizes local interactions in n-grams, while RNN is designed to capture long-range information and forget unimportant local information via the last layer hidden vector.





***Corresponding Author:** Dinh Khanh Linh

DOI- https://doi.org/ 10.5281/zenodo.11372432

In this paper, we propose a series of learning models to address the above weaknesses. We start with a basic LSTM model that uses hidden vectors at the last layer to provide a sentence representation. Then we synthesize sentence representations using Max pooling and Mean pooling strategies to synthesize sentence representations across hidden layers in the LSTM network, and finally we evaluate the model when combining both Max features. and Mean pooling.

In the next part of the article we present: (2) Related works; (3) Proposed models; (4) Results and discussion; (5. Conclusion.

## RESEARCH METHODS

In recent years, many related studies have been proposed to solve the problem of finding similar questions and achieved many positive results. As follows:

Previous work on question finding problems often used technical features, linguistic tools, and external knowledge. For example, semantic features are built based on Wordnet (Dhandapani, A., & Vadivel, V. 2021). This model pairs semantically related words based on the semantic relationships of words.

In the Semeval 2017 conference, the top model in the competition on the Semeval data set uses very complex technical features (Robertson, S. E., *et al.*, 1995) such as exploring kernel functions or extracting tree kernel features from analyzing trees. syntax. Another study exploited different similarity features such as cosine measure, Euclidean measure of lexical distance, syntax and semantics (Gheibi, O. *et al.*, 2021) to represent sentences learned from SVM model.

Studies on the problem of finding answers (Jiang, Z. *et al.*, 2021; Chauhan, U., & Shah, A. 2021; Nakov, P. *et al.*, 2019; Filice, S. *et al.*, 2017) in the CQA system have yielded good results using neural networks without having to use manually extracted features. These models learn sentence representations, then perform question-to-question and question-to-answer similarity measurements (Chauhan, U., & Shah, A. 2021).

In this paper, we propose a series of learning models to address the above weaknesses. We start with a basic LSTM model that uses hidden vectors at the last layer to provide a sentence representation. Then we synthesize sentence representations using Max pooling and Mean pooling strategies to synthesize sentence representations across hidden layers in the LSTM network, and finally we evaluate the model when combining both Max features. and Mean pooling.

## Proposed Models

### LSTM original model

We first briefly present the LSTM model [13]. LSTM is a special type of neural network RNN based on sequence data. LSTM uses several gate vectors at each position to control the transfer of information along the sequence and thus improve modeling of long-range dependencies. While there are different variations of LSTM. We use $X = (x_1, x_2, ..., X_N)$ to represent an input sequence, where $x_k \in R^L$ $(1 \le k \le N)$. These vectors are used together to create a d-dimensional hidden state $_{hk}$ as follows[11]:

$$\begin{aligned} i_k &= \sigma\left(\mathrm{W}^i x_k + V^i h_{k-1} + b^i\right), \\ f_k &= \sigma\left(\mathrm{W}^f x_k + V^f h_{k-1} + b^f\right), \\ o_k &= \sigma\left(\mathrm{W}^o x_k + V^o h_{k-1} + b^f\right), \\ c_k &= f_k \square\ c_{k-1} + i_k \square\ \tanh\left(\mathrm{W}^c x_k + V^c h_{k-1} + b^c\right) \\ h_k &= o_k \square\ \tanh\left(c_k\right) \end{aligned} \quad (1)$$

In which: **i, f, o** are input gates, forget gates and output gates respectively, matrices **W** , **V** and **b** are matrices learned from the model.

Sentence representation methods

Figure 1 describes how to get a sentence representation using the last hidden layer in the problem of finding similar questions.

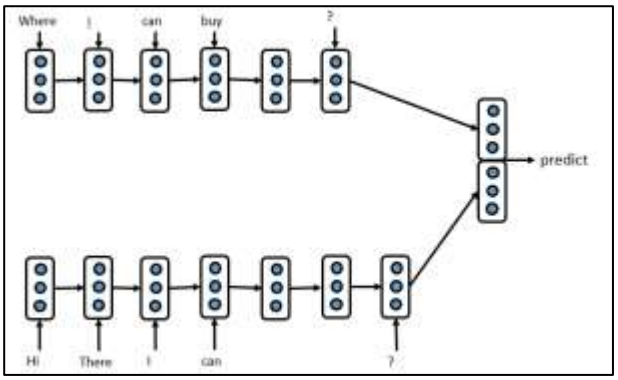


**Figure 1.** LSTM model uses hidden vectors at the last layer to represent sentences.

Figure 2 describes the method to get sentence representation using max pooling operation of hidden layers. Max pooling means taking the maximum value of each component in the hidden layers.





Publisher: SARC Publisher

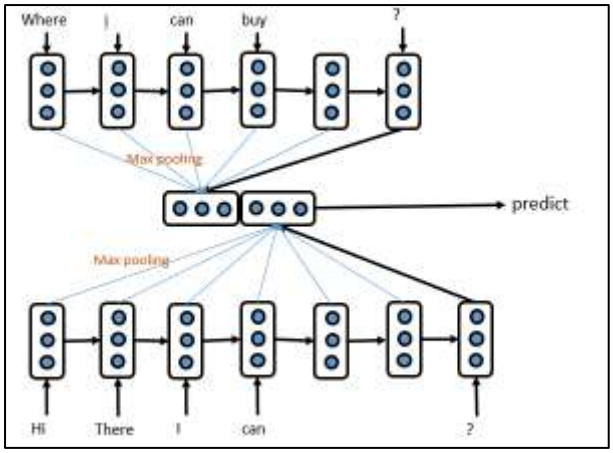


**Figure 2** . The LSTM model uses the max pooling operation to get a sentence representation.

Figure 3 below describes the method of getting sentence representation using Mean pooling operation of hidden layers. Mean pooling is calculating the average value of each component in hidden layers.

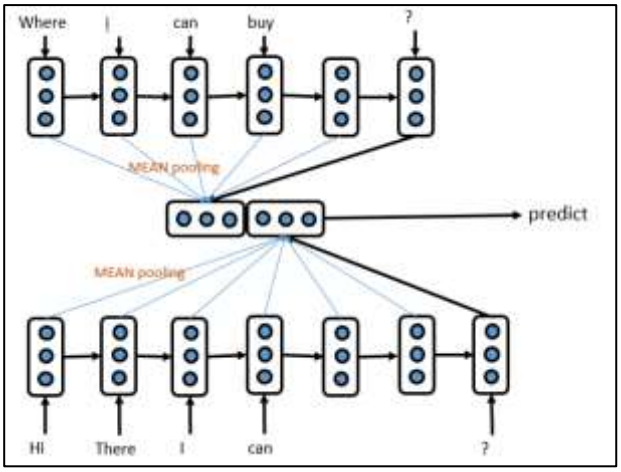


**Figure 3:** The LSTM model uses the MEAN pooling operation to obtain sentence representation.

Finally, we use both MEAN and Max retrieval techniques combined to make sentence predictions.

The loss function is the cross entropy function:

$$L_{model} = -\frac{1}{S}\sum(y \log \hat{y} + (1-y)\log(1-\hat{y})) + \frac{\gamma}{2S}\|\boldsymbol{W}\|_2^2 (2)$$

In which, S is the number of question pairs in the training set, γ is the model's tuning parameter, **W** is the model's set of weight matrices.

## RESULTS AND DISCUSSION

### Data set

We use the Semeval 2017 dataset to evaluate the proposed models. This dataset (Nakov, P. *et al.*, 2019) is taken from the Qatar living forum. This is a forum to discuss all issues for foreigners living in Qatar. The data set is labeled and divided into 3 sets: training set, development set and test set. Table 1 lists the number of question pairs in the data set.

**Table 1.** Statistical table of question pairs in the Semeval 2017 data set(Nakov, P. et al., 2019)

| | **Semeval 2017** |
|---|---|
| Training session | 3170 |
| Practice development | 700 |
| Test set | 880 |

We use MAP and MRR[9] metrics to evaluate the effectiveness of the proposed model.

$$\text{MAP} = \frac{1}{|N|}\sum_{j=1}^{|N|}\frac{1}{m_j}\sum_{k=1}^{|m_j|} Precision\left(R_{jk}\right) (3)$$

### Model Parameters

We use the representation from the 300-dimensional Glove fed into the model in the input layer. OOV words that are not in the dictionary are randomly initialized. The number of hidden layer dimensions in the LSTM model is set to 400 dimensions. The Adam optimization algorithm is used with the learning rate set to 0.0001, parameter γ selected to 0.0001, batch-size to 64, drop-out to 30%. The model is implemented on tensorflow and run on google colab . We evaluate the performance of the model on the dev set and select the best selected parameters on the dev set and then set the test parameters on the test set.

### Result

Table 2 shows the test results on the models:





Publisher: SARC Publisher

**Table 2.** Results of the proposed model

| **Model** | **MAP** |
|---|---|
| LSTM uses the last hidden layer | 40.03 |
| LSTM-max pooling | 40,50 |
| LSTM-Mean pooling | 40.51 |
| **LSTM-Mean+Max pooling** | **41.07** |

Looking at the results in table 2, we see that when using the Max and Mean pooling technique, the Map measure increases from 40% to 40.5% . This proves that, when the sentence representation vector is synthesized from hidden layers, it is capable of exploiting more semantic information of the sentence than using the last hidden layer. Furthermore, when synthesizing sentence representations combining both Mean and max pooling, the MAP result increases to 41.07%. Thus, when connecting the two vectors Mean and Max, pooling makes it better to contain sentence synthesis information. Therefore, the model's prediction results are better.

## CONCLUSION

In this article, we have proposed to use the LSTM model with different sentence representation synthesis techniques for the problem of finding similar questions. Through experiments, we see that using both Mean and Max pooling strategies also affects the results of predicting pairs of similar questions. In the future, we will conduct experiments on biLSTM and CNN models and combine the models as well as use attention mechanisms on this problem.

## REFERENCES


1. Zhou, G., Chen, Y., Zeng, D., & Zhao, J. "Towards faster and better retrieval models for question search." *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2013.
2. Zhou, G., He, T., Zhao, J., & Hu, P. "Learning continuous word embedding with metadata for question retrieval in community question answering." *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2015.
3. Cai, L., Zhou, G., Liu, K., & Zhao, J. "Learning the latent topics for question retrieval in community qa." *Proceedings of 5th international joint conference on natural language processing*. 2011.
4. Wu, W., Sun, X., & Wang, H. "Question condensing networks for answer selection in community question answering." *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2018.
5. Gheibi, O., Weyns, D., & Quin, F. "Applying machine learning in self-adaptive systems: A systematic literature review." *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 15.3 (2021): 1-37.
6. Moravvej, S. V., Kahaki, M. J. M., Sartakhti, M. S., & Mirzaei, A. "A method based on attention mechanism using bidirectional long-short term memory (BLSTM) for question answering." *2021 29th Iranian Conference on Electrical Engineering (ICEE)*. IEEE, 2021.
7. Dhandapani, A., & Vadivel, V. "Question answering system over semantic web." *IEEE Access* 9 (2021): 46900-46910.
8. Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., & Gatford, M. "Okapi at TREC-3." *Nist Special Publication Sp* 109 (1995): 109.
9. Jiang, Z., Araki, J., Ding, H., & Neubig, G. "How can we know when language models know? on the calibration of language models for question answering." *Transactions of the Association for Computational Linguistics* 9 (2021): 962-977.
10. Chauhan, U., & Shah, A. "Topic modeling using latent Dirichlet allocation: A survey." *ACM Computing Surveys (CSUR)* 54.7 (2021): 1-35.
11. Nakov, P., Hoogeveen, D., Màrquez, L., Moschitti, A., Mubarak, H., Baldwin, T., & Verspoor, K. "SemEval-2017 task 3: Community question answering." *arXiv preprint arXiv:1912.00730* (2019).
12. Filice, S., Da San Martino, G., & Moschitti, A. "Kelp at semeval-2017 task 3: Learning pairwise patterns in community question answering." *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. 2017
13. Tan, M., Santos, C. D., Xiang, B., & Zhou, B. "Lstm-based deep learning models for non-factoid answer selection." *arXiv preprint arXiv:1511.04108* (2015)






**Publisher: SARC Publisher**

**Source of support:** Nil; **Conflict of interest:** Nil.

**Cite this article as:**
Hang, N.T.T. and Phuong, D.N." Design of A Monitoring System for Noise, Dust and $Co_2$ Concentration."
*Sarcouncil Journal of Engineering and Computer Sciences* 3.5 (2024): pp 1-5.





**Publisher: SARC Publisher**