Supervised attention for answer selection in community question answering

Thi-Thanh Ha[1], Atsuhiro Takasu[2], Thanh-Chinh Nguyen[3], Kiem-Hieu Nguyen[4], Van-Nha Nguyen[5], Kim-Anh Nguyen[6], Giang-Son Tran[7]
[1]Hanoi University of Science and Technology, VietNam
[2]National Institute of Informatics, Japan
[1,3,4,5,6]ThaiNguyen University of Information and Communication Technology, VietNam
[7]University of Science and Technology of Hanoi, VietNam

Article Info

ABSTRACT

Answer selection is an important task in community Question Answering (cQA). In recent years, attention-based neural networks have been extensively studied in various natural language processing problems, including question answering. This paper explores *match-LSTM* for answer selection in cQA. A lexical gap in cQA is more challenging as questions and answers typical contain multiple sentences, irrelevant information, and noisy expressions. In our investigation, word-by-word attention in the original model does not work well on social question-answer pairs. We propose integrating supervised attention into *match-LSTM*. Specifically, we leverage lexical semantic from external to guide the learning of attention weights for question-answer pairs. The proposed model learns more meaningful attention that allows performing better than the basic model. Our performance is among the top on SemEval datasets.

Corresponding Author:

Thi-Thanh Ha,
Hanoi University of Science and Technology,
1-Dai Co Viet Street, Hai Ba Trung District, Hanoi, Vietnam.
Tel: (+84)982266009
Email: htthanh@ictu.edu.vn

1. INTRODUCTION

Answer selection of Community Question Answering (cQA) is one of the most problems in Natural Language Processing and attracts much interest by 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 is obtaining popularity and flexibility to provide to a user [1]. A user can post a question and likely to receive many answers from others. It is difficult for a user to become aware of correct answers in a few restrictions. Moreover, it is time-consuming for a user to check over them all. For these reasons, it is necessary to build a tool automatically identifying the right answers.

The answer selection problem is defined as follows: Given a question and set of candidate answers, we need to identify which candidates are correct. It is an essential problem in question answering and has drawn much attention from the community [2, 3] Lexical gap, i.e. the mismatch between vocabularies used in questions and answers, is one of the main challenges in answer selection. The problem becomes more complicated in cQA as questions and answers typically contain multiple sentences and extraneous information

irrelevant to the main question, with substantial noise such as greetings, emoji, and sentiment. These characteristics can be seen in Figure 1. There are redundant, noise, and the average of questions and answers are long as shown in Table 1.

> ***Example 1:***
> **Subject:** Nationalities banned in Qatar
> **Question:** Hello! Can you help me, is there anyone knows the list of nationalities who are banned and cannot apply employment visa in Qatar?
> **Answer (good):** Pakistanis are facing severe problems. There is no ban on Visa but it is very hard, near impossible to get.
> **Answer (bad):** Hi are you suspecting your nationality .:)
> ***Example 2:***
> **Subject**: Seafood:
> **Question:** All this talking about crab and fishing trip made me hungry! :o) What kind of seafood place would you recommend in Doha? What do you enjoy eating there? (fish, shrimps, crabs,...?) How much does it cost to treat yourself in there? And (of course I'd ask!) which places are best to avoid and why?
> **Answer (good):** i went to these places in Doha, like Best Fish or Golden Fish- and they were really horrible. oh well, i guess we only have Sheraton and Intercont :o) If you're looking for a problem, you're probably gonna find one.
> **Answer (bad):** Dont know about any particular restaurant that is really good here. But let me know if you find any GOd, I am hungry again....

Figure 1. Two examples of a question and its answers in SemEval dataset

A huge of research methods in recent years have focused on end-to-end approaches based on deep neural networks and attention mechanism without depending on feature engineering or external knowledge bases for the purpose to handlethese problems [4]. Attention mechanism has shown great success in various NLP tasks [5] such as machine translation, natural language inference, reading comprehension, and question answering [6]. Furthermore, attention calculation makes the redundant and noisy segments provided with less importance, followed by emphasizing the representation of significant segments. Thus, the attention-based deep learning model is suitable for processing text in CQA.

In this paper, we study word-by-word attention in matching questions and answers on social forums. We explore *match-LSTM* [7] and propose integrating supervised attention into this model. *Match-LSTM* works well in natural language inference by matching important words between premise and hypothesis sentences. However, our initial investigation shows that the model fails to learn meaningful attention in cQA context as shown in Figure 2, where both questions and answers are long and noisy. The experiments show that supervised attention helps to learn meaningful matching that allows to better select correct answers. Our proposed model achieves performance on a par with top results on SemEval datasets.
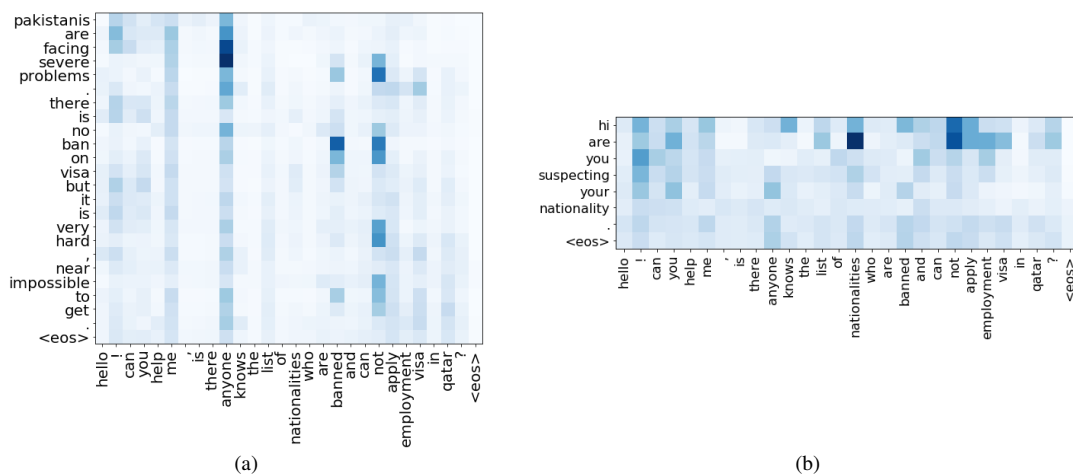

Figure 2. An example of word alignment learned by *match-LSTM*. Content words are weakly aligned, while much of attention is paid to stopwords, (a) A pair of question and its good answer, (b) A pair of question and its bad answer

2.   RELATED WORK

Previous work in answer selection bases on handcrafted features such as semantic role annotations [8], parse trees [9], tree kernels [10]. Then, researchers started using deep neural networks for answer selection, for example, Yu et al [11] propose a convolutional bigram model to classify a candidate answer as correct or incorrect. Tan et al [4] used an attentive BiLSTM component that performs importance weighting before pooling based on the relatedness of segments in the candidate answer to the question.

In neural machine translation, word-by-word attention tries to learn soft alignment, which mimics the task of word alignment between source and target sentences [6]. Rocktaschel et al [12] proposed using two LSTMs to read premise and hypothesis sentences and learn word-by-word alignment to help predict their textual entailment. Following this direction, *match-LSTM* was proposed to add a so-called mLSTM to better capture word alignment and directly use the last hidden state of this LSTM for prediction [7]. Furthermore, their model was extended to tackle machine comprehension by combining with pointer networks.

Supervision has been shown to improve attention quality in some natural language tasks such as machine translation, sentiment analysis, and event detection [13, 14]. Mi at el [13] argued that unsupervised soft alignment in seq2seq model [6] suffers from the lack of context after current word in the target sentence. They proposed using supervised word alignment to guide the learning of attention weights that, in turn, helps to generate more accurate translation. Zou at el [15] used a sentiment lexicon to guide their model to attend to sentiment words. Similarly, neural models were asked to pay attention to argument information when detecting event triggers. Top systems in the SemEval cQA campaign utilize classifiers with rich features, from dependency tree to text similarity, and other task-specific features. Recently, [2] proposed an CNN with question subject-body attention.

3.   OUR MODEL

Figure 3 shows our model based on *match-LSTM* with supervised attention tailored for question answering in social forums.
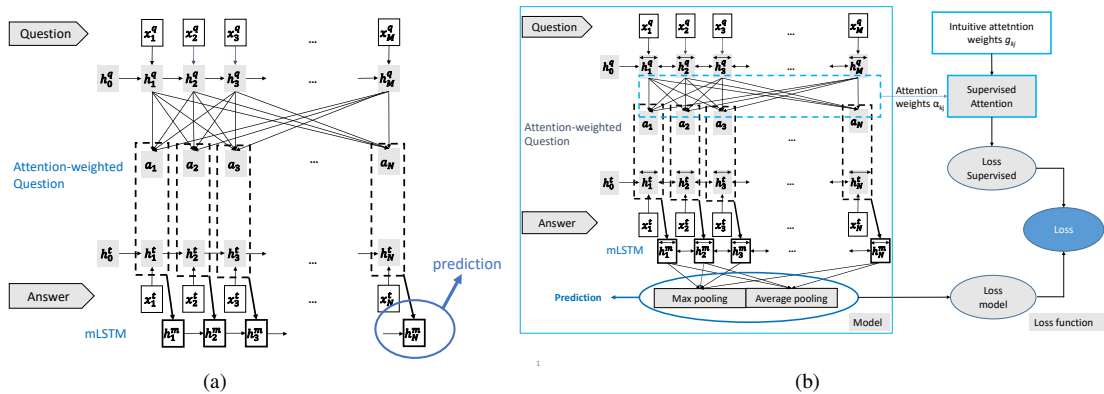

Figure 3. Our model were extended from Match-LSTM, (a) Match-LSTM (provided by the author [7]), (b) Our model

3.1.   LSTM model

LSTM [16] is a particular model of recurrent neural network (RNN). It process sequence data capturing semantic information to neural gates that adaptively read or discard information to/from internal memory states. Specifically, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N)$ is used to denote input sequence, where $\mathbf{x}_k \in \mathbf{R}^l$. At position k, hidden state $\mathbf{h}_k$ is generated as follows:

$$\mathbf{i}_k = \sigma(\mathbf{W}^i \mathbf{x}_k + \mathbf{V}^i \mathbf{h}_{k-1} + \mathbf{b}^i),$$
$$\mathbf{f}_k = \sigma(\mathbf{W}^f \mathbf{x}_k + \mathbf{V}^f \mathbf{h}_{k-1} + \mathbf{b}^f),$$
$$\mathbf{o}_k = \sigma(\mathbf{W}^o \mathbf{x}_k + \mathbf{V}^o \mathbf{h}_{k-1} + \mathbf{b}^o), \quad (1)$$
$$\mathbf{c}_k = \mathbf{f}_k \odot \mathbf{c}_{k-1} + \mathbf{i}_k \odot \tanh(\mathbf{W}^c \mathbf{x}_k + \mathbf{V}^c \mathbf{h}_{k-1} + \mathbf{b}^c),$$
$$\mathbf{h}_k = \mathbf{o}_k \odot \tanh(\mathbf{c}_k),$$

where $\mathbf{i}$, $\mathbf{f}$, $\mathbf{o}$ are input, forget and output gates, respectively, σ is the sigmoid function, $\odot$ is the elementwise multiplication of two vectors and all $\mathbf{W} \in \mathbf{R}^{d \times l}$, $\mathbf{V} \in \mathbf{R}^{d \times d}$, $\mathbf{b} \in \mathbf{R}^d$ are weight matrices and vectors to be learned from the model.

3.2. match-LSTM

match-LSTM(Fig 3a) [7] was originally proposed for sentence-level natural language inference. Its application to cQA answer selection is straightforward. We denote a question and an answer as $\mathbf{X}^q = (\mathbf{x}_1^q, \mathbf{x}_2^q, ..., \mathbf{x}_N^q)$ and $\mathbf{X}^t = (\mathbf{x}_1^t, \mathbf{x}_2^t, ..., \mathbf{x}_M^t)$, respectively; where each $\mathbf{x}_i$ is an embedding vector of corresponding word. Our goal is to predict a binary label y (In SemEval datasets, positive corresponds to *good* while negative corresponds to *potentially useful* and *bad*).

The attention vector $\mathbf{a}_k$ is generated as follows:

$$e_{kj} = \mathbf{w}^e . \tanh(\mathbf{W}^q \mathbf{h}_j^q + \mathbf{W}^t \mathbf{h}_k^t + \mathbf{W}^m \mathbf{h}_{k-1}^m) \tag{2}$$

$$\alpha_{kj} = \frac{exp(e_{kj})}{\sum_{j'} exp(e_{kj'})} \tag{3}$$

$$\mathbf{a}_k = \sum_{j=1}^{M} \alpha_{kj} \mathbf{h}_j^q, \tag{4}$$

where α_{kj} is attention weight of kth word in the answer and j^{th} word in the question; $\mathbf{h}_j^q$ and $\mathbf{h}_k^t$ are hidden states of two LSTMs representing question and answer, respectively; $\mathbf{h}_{k-1}^m$ is the hidden state of mLSTM of $(k-1)^{th}$ word. The central idea lies in mLSTM, which takes the concatenation of $\mathbf{a}_k$, attention-weighted version of the question, and $\mathbf{h}_t^k$, hidden state of kth word itself as input. mLSTM could learn to *forget* unimportant matching and *remember* important ones. The last hidden state of mLSTM is used for prediction.

3.3. Our extension

The first, we used biLSTM to learn character-level word vectors and concatenated it with pre-trained word embeddings in the input layer. Character embeddings were proved to be useful for both formal and informal texts without preprocessing data. Because there are quantities of informal language usage in CQA systems such as abbreviations, typos, emoticons, and grammatical mistakes, using character embeddings helps to attenuate the OOV problem. It is especially useful for the small dataset which has a large number of OOV words as SemEval dataset. To represent questions and answers, we also use two LSTMs to capture both forward and backward sequential contexts.

The second, instead of using only the last hidden state for prediction, we used the concatenation of max pooling and average pooling of all hidden states of mLSTM to capture local information better.
The loss function is regularized binary cross-entropy:

$$L_{model} = -\frac{1}{S} \sum (y \log \hat{y} + (1-y) log(1 - \hat{y})) + \frac{\gamma}{2S} ||\mathbf{W}||_2^2, \tag{5}$$

where S is the number of question-answer pairs and γ is a regularized parameter. The last, supervised attention was integrated into the extended model to learn meaningful matching between answer and question (detailed in below section 3.4.)

3.4. Supervised attention

We denote g_{kj} as intuitive attention weight between k^{th} word of the answer and j^{th} word of the question, where $\sum_j g_{kj} = 1$. The difference between intuitive attention weights and learned attention weights (3) is computed as squared element difference:

$$L_{supervised} = \frac{1}{S} \sum (\sum_{k,j} (g_{kj} - \alpha_{kj})^2) \tag{6}$$

Our goal is to minimize the loss in (5) and (6) simultaneously:

$$L = L_{model} + \lambda L_{supervised}, \tag{7}$$

where λ is a regularized coefficient to control the effect of attention difference.

Intuitively, we want *i)* words semantically close to each other would be matched by our model, and *ii)* answer words are aligned to important question words. We realize the first intuition by cosine similarity between word vectors learned by *fasttext* on texts of an unannotated dataset from all English cQA tasks of SemEval 2016 and 2017. Secondly, we utilize *tfidf* weighting for question words to emphasize important contents.

$$g_{kj} = tfidf(w_j^q)cosine(\mathbf{w}_k^t, \mathbf{w}_j^q), \tag{8}$$

where $\mathbf{w}_k^t$ and $\mathbf{w}_j^q$ are word vectors learned by *fasttext*; to calculate *tfidf* weighting, each document is a question or answer on hold of the unannotated dataset. Similarly to *matchLSTM*, we insert a special token $<eos>$, which allows unimportant words in answer to align with it. Finally, g_{kj} is normalized by a softmax function.

4. EXPERIMENTS AND RESULTS

4.1. Dataset and evaluation metrics

We used SemEval dataset to evaluate our method. It is based on data from Qatar Living forum [1] and was divided three datasets: Training, Development, Testing. Table 1 demonstrates statistics of the dataset on pairs of question and answer. Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR) were used as evaluation metrics with evaluation scripts provided by SemEval organizers.

Table 1. Static of number pairs of question and answer in CQA datasets

	Semeval 2016	Semeval 2017
Train	36,198	39,468
Dev	2,440	3,270
Test	3,270	2,930
Average length of body	49.4	45.8
Average length of answer	38.8	38.0
Size of Vocabulary	6,1271	6,3758

Table 2. Performance of our models on SemEval 2016 and 2017 datasets

	Models	SemEval 2017		SemEval 2016	
		MAP	MRR	MAP	MRR
(A)	QA-LSTM	86.68	91.01	74.36	83.4
(B)	QA-LSTM-CNN	87.17	92.59	74.97	83.56
(C)	QA-LSTM-attention	87.39	91.50	75.87	82.88
(D)	Enhanced LSTM	87.23	93.04	76.46	83.51
(E)	match-LSTM	86.51	92.12	77.70	83.76
(F)	Enhanced match-LSTM	87.76	92.28	78.10	84.21
(G)	**Enhanced match-LSTM + sup. att.**	88.38	**93.13**	78.62	84.56
(H)	QCN	**88.51**	-	-	-
(I)	KELP	88.43	92.82	**79.19**	**86.42**
(J)	ECNU	86.72	91.45	77.28	84.09

4.2. Hyperparameters

We used Glove pretrained word embeddings with 300 dimensions in input layer. Out-of-vocabulary words were initialized randomly. The dimension of two LSTMs for character representation was set to 50. The dimension of other word-level LSTMs was 400. Word vectors for calculating similarity were learned by *fasttext* with a dimension of 100. We used Adam optimizer with initial learning rate 0.0001 and learning rate decays $\beta_1 = 0.9$, $\beta_2 = 0.999$; L2 and supervised attention regularized coefficients λ and γ are both set to 0.0001. The batch size was set to 64. To avoid overfitting, we applied a drop-out of 30% units in all hidden layers and early stopping on dev set. The models were implemented with Tensorflow and all experiments were conducted on GPU Nvidia Tesla p100 16Gb. We used the accuracy on the validation set to decide on the best hyper-parameter settings for testing.

4.3. Results and discussions

In this section, we show detailed experimental results on SemEval datasets. Table 2 is divided into three parts as flows: From Row (A) to (D) is a group of LSTM models used in question answering, Row (E) to (F) indicates the developing from match-LSTM to our proposed model and Rows(H-J) lists the results of state of the art models on SemEval datasets. We evaluated our model with some approached as followed:

(a) QCN [2] models attention between question subject and body and utilizes CNN for question and answer representation.

(b) KELP [17] uses syntactic kernel with text similarity and other task-specific features to learn a feature-rich classifier.

(c) ECNU [18] is an ensemble of feature-based classifiers and CNN.

(d) QA-LSTM, QA-LSTM-CNN, QA-LSTM attention [4]: These models were projected matching answers to questions accommodating their complicated semantic relations. In which, QA-LSTM-CNN is the hybrid model between Convolutional and LSTM. After that, attention mechanism was put forward into QA-LSTM to construct better answer representations according to the input question. Each output vector of LSTM on the answer side at time step t was updated by the question representation and attention parameters.

(e) Enhance-LSTM [19]: This model is proposed for natural language inference by considering recursive architectures in both local inference modeling and inference composition.

The models from row A to G in Table 2 were implemented in Tensorflow, and the results of SOTA models in rows (H,I,J) were reported from original papers. From Table 2, It can be seen that the performance of Enhanced match-LSTM is also better than typical LSTM models. Moreover, when supervised attention is put into this model, the performance increases steadily as well on both SemevalCQA2016 and SemevalCQA2017. This suggested that supervised attention can learn semantic of question and answer better than previous LSTM models. Specifically, supervised attention not only learns more meaningful word alignment (as discussed later in Section 4.4.), but also supports the main task of answer selection. For example, the MRR score of our model surpass the winer KELP team in Semeval 2017 with 93.13% and the MAP performance is on par with top results on SemEval 2016 and 2017.

4.4. Attention visualization

Figure 4 and Figure 5 visualize word-by-word attention between answer (Y-axis) and question (X-axis) to explain our model. These splots present the alignment weights α_{kj} between answer and question, where a darker color correlates with a larger value of α_{kj}. Overall, Our model interpret word relationship better than basic *match-LSTM* as depicted in Figure 2.

In Figure 4, content words in the answer (e.g. 'Pakistanis', 'ban', and 'get') and question (e.g. 'nationalities', 'banned', and 'apply') are correctly aligned. While 'ban' and 'banned' basically have the same root form, we anticipate that text similarity is especially helpful for other alignments like 'Pakistanis' and 'nationalities', or 'get' and 'apply'. Last but not least, as we look more deeply into Figure 4a, stopwords and punct are still aligned. Whereas in Figure 4b, thanks to $tfidf$ weighting, stopwords and punct in the answer are leaned towards the final $<eos>$ token of question, as indicating by multiple blue cells in the last column. We could also observe that stopwords and punct in the question are no longer highlighted. Therefore, the greetings, questions that do not mean to be asked are not attended.

The same goes for a pair of question and bad answer in Figure 5. Some words in answer ('your nationality. :)') are aligned the most highlightly with 'nationalities' in the question as shown in Figure 5a. It is evident that our model can learn essential parts of the question and answer better than the original model.
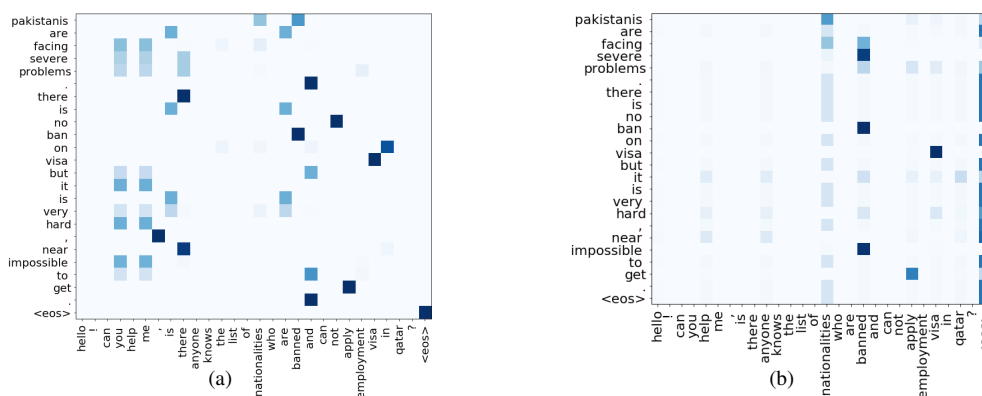

Figure 4. A pair between question and good answer example of attention learned by our model with supervision, (a) Supervised attention with similarity, (b) Supervised attention with similarity and *tfidf*
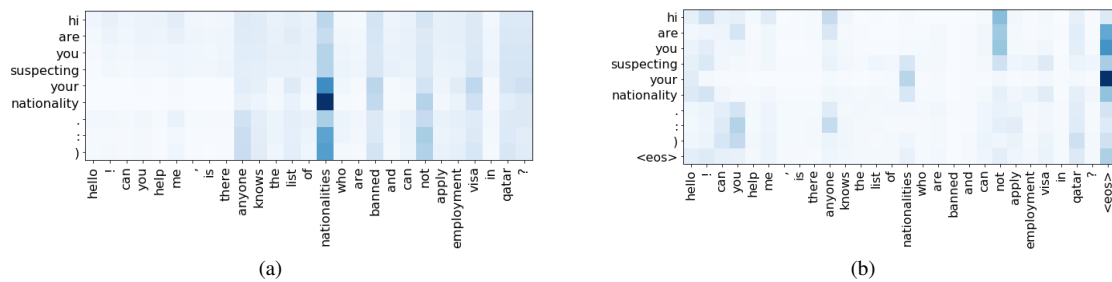
(a)   (b)

Figure 5. A pair between question and bad answer example of attention learned by our model with supervision, (a) Supervised attention with similarity, (b) Supervised attention with similarity and *tfidf*

5.   CONCLUSIONS AND FUTURE WORKS

In this paper, we propose to extend *match-LSTM* with supervised attention. We empirically demonstrate that our solution is useful in cQA answer selection. In the future, we are going to investigate cQA in popular forums such as Yahoo Answers and Stack Overflow and then use Transformer model [20] instead of LSTM model. Such forums also provide useful meta-data and related tasks such as expert findings. In another direction, we are going to study siamese architecture and CNN with phrase-level representation.

ACKNOWLEDGEMENT

REFERENCES

[1]   P. Nakov, D. Hoogeveen, L. Màrquez, A. Moschitti, H. Mubarak, T. Baldwin, and K. Verspoor, "SemEval-2017 task 3: Community question answering," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*.   Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, pp. 27–48. [Online]. Available: https://www.aclweb.org/anthology/S17-2003

[2]   W. Wu, X. SUN, and H. WANG, "Question condensing networks for answer selection in community question answering," in *Proceedings of the 56 th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.   Association for Computational Linguistics, 2018, pp. 1746–1755. [Online]. Available: http://aclweb.org/anthology/P18-1162

[3]   N. K. Tran and C. Niedereée, "Multihop attention networks for question answer matching," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, ser. SIGIR '18.   New York, NY, USA: ACM, 2018, pp. 325–334. [Online]. Available: http://doi.acm.org/10.1145/3209978.3210009

[4]   M. Tan, C. dos Santos, B. Xiang, and B. Zhou, "Improved representation learning for question answer matching," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.   Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 464–473.

[5]   J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "Generating token-level explanations for natural language inference," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.   Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 963–969. [Online]. Available: https://www.aclweb.org/anthology/N19-1101

[6]   D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. [Online]. Available: http://arxiv.org/abs/1409.0473

[7]   S. Wang and J. Jiang, "Learning natural language inference with LSTM," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human*

Language Technologies. San Diego, California: Association for Computational Linguistics, Jun. 2016, pp. 1442–1451. [Online]. Available: https://www.aclweb.org/anthology/N16-1170

[8] P. Jansen, M. Surdeanu, and P. Clark, "Discourse complements lexical semantics for non-factoid answer reranking," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 977–986. [Online]. Available: https://www.aclweb.org/anthology/P14-1092

[9] M. Wang and C. Manning, "Probabilistic tree-edit models with structured latent variables for textual entailment and question answering," in *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. Beijing, China: Coling 2010 Organizing Committee, Aug. 2010, pp. 1164–1172. [Online]. Available: https://www.aclweb.org/anthology/C10-1131

[10] A. Severyn and A. Moschitti, "Structural relationships for large-scale learning of answer re-ranking," in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '12. New York, NY, USA: ACM, 2012, pp. 741–750. [Online]. Available: http://doi.acm.org/10.1145/2348283.2348383

[11] L. Yu, K. M. Hermann, P. Blunsom, and S. Pulman, "Deep learning for answer sentence selection," *CoRR*, vol. abs/1412.1632, 2014. [Online]. Available: http://arxiv.org/abs/1412.1632

[12] T. Rocktäschel, E. Grefenstette, K. M. Hermann, T. Kocisky, and P. Blunsom, "Reasoning about entailment with neural attention," in *International Conference on Learning Representations (ICLR)*, 2016.

[13] H. Mi, Z. Wang, and A. Ittycheriah, "Supervised attentions for neural machine translation," *CoRR*, vol. abs/1608.00112, 2016. [Online]. Available: http://arxiv.org/abs/1608.00112

[14] M. Nguyen and T. H. Nguyen, "Who is killed by police: Introducing supervised attention for hierarchical lstms," *CoRR*, vol. abs/1807.03409, 2018. [Online]. Available: http://arxiv.org/abs/1807.03409

[15] Y. Zou, T. Gui, Q. Zhang, and X. Huang, "A lexicon-based supervised attention model for neural sentiment analysis," in *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 868–877. [Online]. Available: https://www.aclweb.org/anthology/C18-1074

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735

[17] S. Filice, G. Da San Martino, and A. Moschitti, "Kelp at semeval-2017 task 3: Learning pairwise patterns in community question answering," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017, pp. 326–333. [Online]. Available: http://aclweb.org/anthology/S17-2053

[18] G. Wu, Y. Sheng, M. Lan, and Y. Wu, "ECNU at SemEval-2017 task 3: Using traditional and deep learning methods to address community question answering task," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, pp. 365–369. [Online]. Available: https://www.aclweb.org/anthology/S17-2060

[19] Q. Chen, X. Zhu, Z.-H. Ling, S. Wei, H. Jiang, and D. Inkpen, "Enhanced LSTM for natural language inference," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1657–1668.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

BIOGRAPHIES OF AUTHORS

Thi-Thanh Ha is a PhD student at Data Science Laboratory, HaNoi University of Science and Technology, VietNam. She obtained Bachelor Degree of Science in Applied Mathematics and Informatics from University of Natural Science, Vietnam National University in 2004. She now is a lecturer at Information Technology and Communication University, Thainguyen University. Her researches are fields of deep learning in Natural Language Processing.

Atsuhiro Takasu received Doctoral Degrees in Engineering from Tokyo University in 1989. Now, he is a Digital Content and Media Sciences Research Division Professor/Director. He is interested in statistical data processing, information extraction and integration and applying the developed technologies to information sharing systems and digital libraries. Main research topics are Data Engineering, Machine Learning, Data Mining, and Text Mining, Data and Information Integration, Information Retrieval, and Digital Library.

Kiem-Hieu Nguyen received his BCS degree from Hanoi University of Science and Technology, Vietnam in 2007, and his Ph.D. degree from School of Computer Engineering and Information Technology, University of Ulsan, Korea in 2013. He was a post-doctoral researcher at LIMSI-CNRS and CEA LIST, France during two years. He is currently a lecturer/researcher at Hanoi University of Science and Technology, Vietnam. His research interest is natural language processing, event extraction and community Question answering.

Giang-Son Tran Dr Tran Giang Son is a lecturer and researcher at University of Science and Technology of Hanoi (USTH). He received his Ph.D. in Computer Science in 2014 at the IRIT laboratory, Institute National Polytechnique de Toulouse, France. His Ph.D. was funded by a scholarship from Vietnamese government. Earlier, he obtained his engineer degree of science and technology diploma at Hanoi University of Science and Technology, Vietnam in 2008. His research interests include computer vision, machine learning, high performance computing, cloud management systems and mobile platforms.

Kim-Anh Nguyen received her Ph.D. degree in Computer Science from HaNoi University of Science and Technology in 1994. She is currently an associate professor at Hanoi University of Science and Technology, Vietnam. Her research interest is machine learning, topic model and community Question answering.
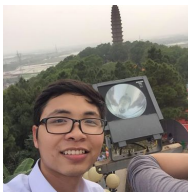
Van-Nha Nguyen received Bachelor of Engineering degree from HaNoi University of Science and Technology in 2018. He is currently a MSc student at Hanoi University of Science and Technology, Vietnam. His research interest is topic model, deep learning, natural language processing and community Question answering.

Thanh-Chinh Nguyen has just received Bachelor of Engineering degree from Hanoi University of Science and Technology in July 2019. He is currently an engineer at Brains Technology, Inc, Japan. His research interest is natural language processing, deep learning and community Question answering.