Siamese LSTM with Convolutional Similarity for Similar Question Retrieval

Avinash Kamineni
International Institute of Information Technology
Hyderabad, India
avinash.kamineni@research.iiit.ac.in

Harish Yenala
International Institute of Information Technology
Hyderabad, India
harish.yenala@research.iiit.ac.in

Manish Shrivastava
International Institute of Information Technology
Hyderabad, India
m.shrivastava@iiit.ac.in

Manoj Chinnakotla
Microsoft
Hyderabad, India
manojc@microsoft.com

***Abstract*—In this paper, we model the similar question retrieval task as a binary classification problem. We propose a novel approach of "*1D-Siamese LSTM for cQA (1D-SLcQA)*" to find the semantic similarity between a new question and existing question(s). In 1D-SLcQA, we use a combination of twin LSTM networks and a contrastive loss function to effectively memorize the long term dependencies i.e., capture semantic similarity even when the length of the answers/questions is very large (200 words). The similarity of the questions is modeled using a single network with (1D) (feature) convolution between feature vectors learned from twin LSTM layers. Experiments on large scale real world Yahoo Answers dataset show that 1D-SLcQA outperform the state of the art approach of Siamese cQA approach(SCQA).**

***Index Terms*—Community Question Answering, Siamese Network, LSTM, CNN**

I. Introduction

The traction to cQA forums has gone up because users get accurate and concise answers. These forums organize the data with rich meta information like categories, sub categories, answer votes, approved answer, user expert level etc. However, one of the major problems with these cQA forums is "*question starvation*" [1] where a submitted question doesn't get answered immediately. It may sometimes take days to get an answer or it may also go unanswered. The cQA has answers well curated by expert users which can help reduce the problem of Answer Extraction in traditional QA systems. For the expert users, who answers the questions always find these similar questions again instead of new ones. To overcome these limitation, one solution could be to suggest answers from a semantically similar question which has already been answered. The problem of finding questions which are similar in intent is rendered difficult due to natural language phenomenon such as synonymy, polysemy and other phrase and syntactic ambiguities which allow expressing the same intent in various different ways. For example, the question "*how to get rid of my baby's cold?*" can also be expressed as "*how can I help my baby with a stuffy nose?*", "*how to treat my newborn with cold?*", "*what to do when a baby has running nose?*". There can be few answers by experts which include additional topics like "*sinus infections*", "*respiratory syncytial virus (RSV)*" etc., which can help identify semantically similar questions better. Due to the lack of large semantically equivalent question archives, techniques proposed so far for "*Similar Question Retrieval (SQR)*" mainly rely on previous QA archives for learning the ideal matching function for retrieving semantically similar questions. The existing approaches can be broadly divided into the following categories:

Term Weighing based Models: TF/IDF, BM25 [2], Language modeling based similarity [3] measures for scoring the extent of match between two questions. However, the matching mainly relies on the exact words between the two questions, and therefore are limited in the ability to retrieve semantically similar questions.

Translation Models: Translation models estimate the probability of translating the given question to a new candidate question with relatively little word overlap. The question with the highest translation probability is suggested to the user. Translations are learned at word level [4], [5], phrase level [6] from question-answer pairs in parallel corpora of same language. A combination of query likelihood language model and word based translation model [5] improved the performance of finding similar questions. However, recent works [7] state that questions and answers cannot be considered in parallel because they are heterogeneous at lexical level and user behaviors.

Latent Topic Models: Latent models [7]–[9] assume that the questions and answers share common topics and match questions not only on a term level but also on a topic level. A Fisher kernel [10] was used to model the fixed size representation of the variable length questions. The model enhances the embedding of the questions with the meta-data "category" involved with them. Learning representations [11] of words and question categories simultaneously are incorporated into traditional language models.

Deep learning based Models: Learning representations automatically at sentence level using variants [12]–[14] of neural network architectures are proposed to directly model the question-question pair similarity. Deep neural network

(DNN) [12] was used to map the question answer pairs to a common semantic space to compute the relevance of each answer given the query using cosine similarity between their vectors. The obtained semantic vectors are fed into a learning to rank (LTR) framework to learn the relative importance of each feature. Convolutional neural tensor network (CNTN) [13] combines sentence modeling and semantic matching. Deep structured topic modeling [14] combines topic model and paired convolutional networks to retrieve related questions. A twin convolutional neural networks [15] with shared parameters and a contrastive loss function joining them learns the similarity metric for question-question pairs. 1D convolution and 1D max pooling [16] operation for sequence modeling tasks captures more meaningful information by capturing long term dependencies. MaLSTM [17] uses Siamese-LSTM network with Manhattan metric to learn sentence representations in a highly structured space which can infer complex semantic relationships.

To summarize, the advantage of Deep Learning (DL) based techniques is that they usually do not require hand crafted feature engineering, could be trained end-end as a single model and offer competitive performance when compared to other off-the-shelf machine learning techniques. Previous DL based approaches for SQR [12]–[14] have mainly investigated the use of convolutional neural networks for learning the matching function between the questions. However, LSTMs are reasonably good to model the compositional nature of natural language. In this paper, we investigate the use of LSTM based architectures for SQR. More specifically, we propose a novel deep architecture for SQR, which is based on LSTMs, and uses a task-specific similarity function that automatically learns using CNNs. The resultant model is a combination of LSTMs and CNNs where the LSTM layer is employed to learn representations for the given question as well as the candidate question and the CNN layer is used for matching these representations and finally predict if the two are semantically similar.

The following are our main contributions in this paper:

- We propose a novel model 1DcQA based on one dimensional Convolutional Neural Network which learns latent features by doing feature (DSSM vectors) convolution between question-question pairs in a cQA dataset.
- We propose a novel approach of Siamese LSTM Network, which learns long term dependencies and capture sequential patterns present in the question and its related question, which was missing in the T-SCQA [15].

The rest of the paper is organized as follows: Section II describes the architecture 1DcQA. Section III describes the setup of of 1D-SLcQA with Siamese LSTM network with 1D Convolution similarity and explains the training and testing phases of SLcQA and 1DcQA, respectively. Section IV describes experimental set-up, details of the evaluation dataset and evaluation metrics. Section V shows quantitative and qualitative results and finally, Section VI concludes the paper.

II. 1D-Convolutional Neural Network for cQA (1DcQA)

Figure 1 shows the overall model consists of four parts: Input query embedding using Deep Structured Semantic Model (DSSM) Layer, one-dimensional Convolution Layer, Max pooling Layer, Rectified linear unit (ReLU) layer, and Fully Connected Layer. The details of different layers are described in the following sections.

A. Input query embedding using Deep Structured Semantic Model (DSSM)

DSSM [18] is a deep neural network (DNN) modeling technique for representing text strings in a continuous semantic space. It uses multiple hidden layers to project the word-hashed features into the semantic space. We take query text (Q) and related query text (R_q) and pass them through a trained DSSM model, which returns two vectors (Q_{vec}) and (Rq_{vec}) of dimension D, respectively. These vectors will be used in the subsequent layers.

B. One Dimensional Convolution

The vectors Q_{vec} and Rq_{vec} obtained from the DSSM layer are combined as matrix H of dimension $2 \times D$. We extract local feature by performing a 1D convolution i.e., sliding through only one axis along feature vector over H. In our case, the convolution operation involves a 1D filter $m \in \mathrm{I\!R}^{\mathrm{k} \times \mathrm{L}}$ where, $D = 300, k = 2, L = 20$. In the subsequent layers, we use the standard max-pooling with ReLU Unit followed by a fully connected layer with softmax node to compute the probability of semantic similarity between the two query texts.

III. Siamese LSTM Network with 1D-CNN for cQA (1D-SLcQA)

In the proposed architecture, we use LSTM layer in place of DSSM layer of the 1DcQA model shown in Figure 1. Our complete system of 1D-SLcQA can be seen in Figure 3. In the figure, the LSTM cells read the input word by word in sequential manner to produce learnt semantic representation vectors. The twin LSTM layers output a 200 dimensional vector for query and related query. These learned representations are sent to a one dimensional convolution network (1D-CNN) mentioned in 1DcQA architecture. The max pooling layer is performed on each of the feature maps which is then sent to a ReLU layer. ReLU simplifies back propagation and makes learning faster while avoiding saturation. The output of above step is fed to Fully Connected layer and in the final layer, softmax node outputs the probability of relatedness between Q and Rq. The following are the novel features of the proposed architecture 1D-SLcQA:

- Firstly, it has advantages of Siamese network of parameter sharing between query and related query.
- Secondly, using LSTM network learns long term dependencies for sequential patterns, and thus better semantic representation for individual queries.
- Thirdly, 1D-CNN network helps in improving the similarity metric between query and related query by using a

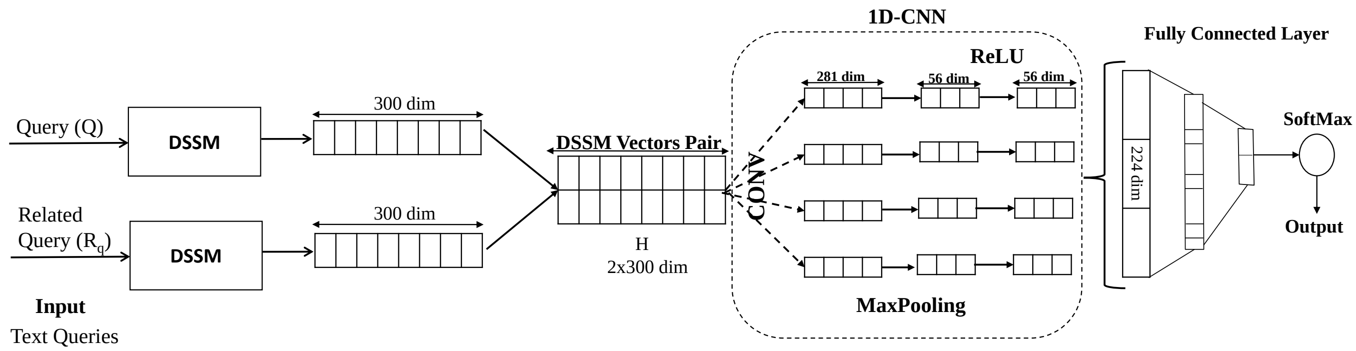

Fig. 1. Architecture of 1D-Convolutional NN(1DcQA) takes input as text queries: Query and a Related Query. The queries are passed through DSSM, Convolution, Fully Connected, Softmax layers to output a similarity score between the two queries.

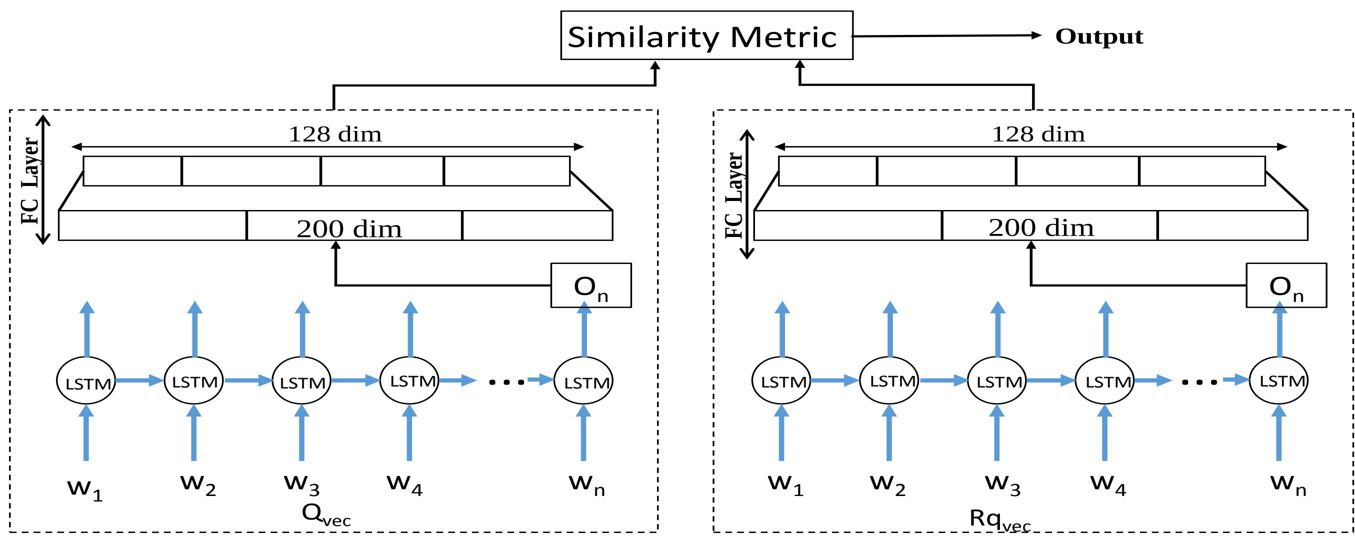

Fig. 2. Structure of Siamese LSTM Layer in SLCQA and 1D-SLcQA, used in place of DSSM layer to identify question representations using glove word embeddings.

Binary Cross Entropy (BCE) loss function i.e., we can reduce the Similar Question Retrieval (SQR) problem to a simple binary classification problem with prediction labels as similar or dissimilar.

In the subsequent sections, we detail the individual components of the 1D-SLcQA architecture.

A. Input Layer

For each word (W_i) in the input query (Q) and related query (Rq), we obtain its glove [19] word embedding (w_i) in D dimensions (in our case, $D = 300$) and use it to form the input query vectors (Q_{vec}, Rq_{vec}). We chose glove for representing the words since - a) it was trained over crawled data which covers diverse set of vocabulary and b) it moedls a vector space with meaningful linear substructure that helps in identifying similar words accurately. As the LSTM implementation requires fixed-length input, we pad each query with special symbol (UNK), indicating unknown words, at the end to ensure the length is equal to max sentence length n (in our case 200). These paddings have been masked in the subsequent layers.

B. LSTM Layer

Long-Short Term Memory (LSTMs) [20] are variants of Recurrent Neural Networks (RNN) [21], [22] architectures which - a) overcome the vanishing gradient problem of conventional RNNs and b) have the ability to capture long-term dependencies present in a sequential pattern due to their gating mechanisms which control information flow.

Given a sequence $Q_{vec} = w_1, w_2, ..., w_n$, where n is the length of input text, LSTM processes it word by word. At time-step t, the memory c_t and the hidden state h_t are updated with the following equations:

$$\begin{bmatrix} i_t \\ f_t \\ o_t \\ \hat{c}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} W \cdot [h_{t-1}, x_t] \tag{1}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \hat{c}_t \tag{2}$$

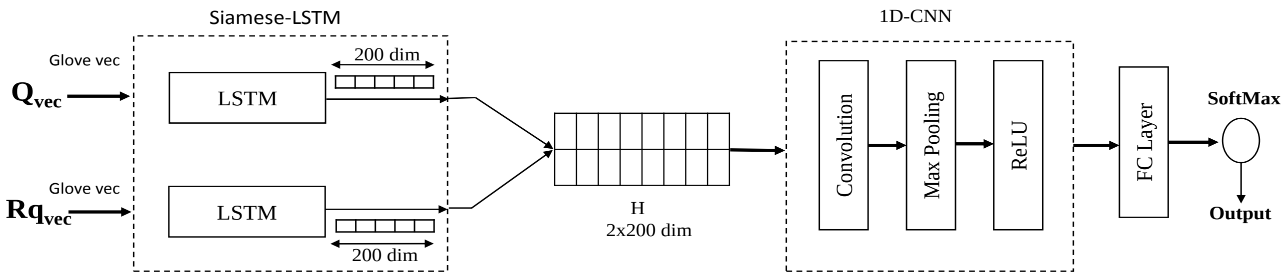

Fig. 3. System diagram of 1D-SLcQA, showcasing different layers to generate the similarity score for given pair of query and related query.

$$h_t = o_t \odot \tanh(c_t) \quad (3)$$

where x_t is the input at the current time-step, i, f and o is the input gate activation, forget gate activation and output gate activation respectively, $\hat{c}$ is new candidate vector, σ denotes the logistic sigmoid function and $\odot$ denotes element-wise multiplication [16].

In our architecture, the LSTM cells read (Q_{vec} and Rq_{vec}) word by word in sequential manner to produce semantic vector representations O_n^Q and O_n^{Rq}, respectively. Figure 2 demonstrates the LSTM layer used in our architecture.

C. Training with SLcQA

SLcQA is trained to learn the similarity between question and its relevant answers. SLcQA is different from the other deep learning counterparts due to its property of parameter sharing. Training the network with a shared set of parameters not only reduces number of parameters (thus, save lot of computations) but also ensures consistency of the representation of questions and answers in semantic space. Using a LSTM network compared to T-SCQA [15] helps in learning long term dependencies and detailed understanding of the sentence. Therefore, it helps in minimizing the semantic distance between the question and the relevant answers, and maximizing the semantic distance between the question and the irrelevant answers.

Given an input q_i, a_i where q_i and a_i are the i^{th} question answer pair, and a label y_i with $y_i \in 1, -1$, the loss function is defined as:

$$loss(q_i, a_i) = \begin{cases} 1 - \cos(q_i, a_i), & \text{if } y = 1; \\ \max(0, \cos(q_i, a_i) - m), & \text{if } y = -1; \end{cases}$$

where m is the margin which decides by how much distance dissimilar pairs should be moved away from each other. It generally varies between 0 to 1. The loss function is minimized such that question answer pairs with label 1 (question-relevant answer pair) are projected nearer to each other and that with label -1 (question-irrelevant answer pair) are projected far away from each other in the semantic space. The model is trained by minimizing the overall loss function in a batch. The objective is to minimize :

$$L(\Lambda) = \sum_{(q_i, a_i) \in C \cup C'} loss(q_i, a_i) \quad (4)$$

TABLE I
HYPER-PARAMETERS USED IN OUR EXPERIMENTS

Parameter	1DcQA	SLcQA	1D-SLCQA
Batch Size	5000	5000	5000
Depth of CNN	1	1	NA
Learning rate	0.01	0.001	0.001
Kernel width of Convolution	1x20	NA	1x20
Kernel width of MaxPooling	1x5	NA	1x5
#LSTM Cells	NA	200	200
Length of semantic vector	NA	128	200

where C contains batch of question-relevant answer pairs and C' contain batch of question-irrelevant answer pairs.The parameters shared by the convolutional sub-networks are updated using Stochastic Gradient Descent (SGD).

D. Testing on 1D-SLcQA

In testing phase, given a query Q we need to retrieve its related queries ($Rq_1, Rq_2, ...$) based on their similarity scores. In order to get the similarity score we will feed textual query Q and its related query Rq to 1D-SLcQA system. The system takes queries Q and Rq then it finds glove vectors for each word in Q and Rq. These vectors then fed into twin Siamese LSTM network, which outputs two semantic vectors which will form a matrix H as shown in the Figure 3. The matrix H is fed to the 1D-CNN to output convolved semantic feature vectors which will then be fed to Fully Connected layer. The final soft-max node in the Fully Connected layer outputs the probability of relatedness between the Q and Rq, which is our similarity score.

IV. EXPERIMENTS

For the experimental setup, we have taken Yahoo! Answers dataset from Yahoo! Labs Webscope We have used title, description, best answer information among all available details about each question.

For training dataset, we randomly selected 2 million data and extracted question-relevant answer pairs and question-irrelevant answer pairs from them to train 1DcQA, SLcQA and 1D-SLcQA networks. Similarly, our validation dataset

TABLE II

RESULTS ON YAHOO! ANSWERS DATASET. THE BEST RESULTS ARE OBTAINED BY 1D-SLCQA (BOLD FACED). SLCQA, 1DCQA, 1D-SLCQA RESULTS ARE STATISTICALLY SIGNIFICANT WITH $p \leq 0.001$.

Method	MAP	MRR	P@1
LMIR [3]	0.762	0.844	0.717
translation(word) [4]	0.786	0.870	0.807
translation+LM [5]	0.787	0.869	0.804
translation(phrase) [6]	0.789	0.875	0.817
Q-A topic model [8]	0.787	0.879	0.810
Q-A topic model(s) [7]	0.800	0.888	0.820
DSQA [14]	0.755	0.921	0.751
T-DSQA [14]	0.801	0.932	0.822
SCQA [15]	0.811	0.895	0.830
T-SCQA [15]	0.852	0.934	0.849
1DcQA **(ours)**	**0.856**	**0.938**	**0.882**
SLcQA **(ours)**	**0.861**	**0.933**	**0.898**
1D-SLcQA **(ours)**	**0.893**	**0.941**	**0.902**

contains 400,000 question answer pairs. Using this validation set the hyper-parameters mentioned in Table I have been tuned.

The experiments in this paper were evaluated on publicly available 1423 questions dataset, released by [7] as test set for all the models. On this gold data, we computed the performance of the models with three evaluation criteria: Mean Average Precision (MAP), Mean Reciprocal Rank (MRR) and Precision at 1 (P@1).

V. RESULTS

In Table II, we show a comparasion of the previous methods with proposed methods i.e., 1DcQA, SLcQA and 1D-SLcQA. In our experiments, the methods *SCQA* and *T-SCQA* use convolution in its twin sub-network. The *1DcQA*, one of the proposed model, shows improvement over *SCQA* and *T-SCQA* because of its 1D feature convolution technique. The results of our approaches *SLcQA* and *1D-SLcQA* show significant improvement over *SCQA* and *T-SCQA*. The results demonstrate that using an LSTM sub-network has better capabilities to learn context information and intrinsically understand the question.

A. Quantitative Analysis

$SLcQA$ and *1D-SLcQA* learns the semantic relationship, long-distance sequences between the question and their best and most voted answers. The weight is tuned in the validation dataset. We trained our model for several epochs and observed how the results varied with the epochs. We found that the evaluation metrics changed with increasing the number of epochs but stabilized after epoch 150.

The comparison of *SLcQA* and *1D-SLcQA* with the previously proposed models is shown in Table II. For baseline we considered the traditional language model *LMIR*. The results show translation based models outperform the baseline methods and topic based approaches outperform the translational methods. Also, it is observed that deep learning based solution with parameter sharing is more helpful for this task than without parameter sharing. Note, that the results of previous models stated in Table II differ from the original papers since we tried to re-implement those models with our training data (to the best of our capability). We used the same dataset for training and testing so the results shown have been taken from [15].

The increase of MAP due to inclusion of LSTM layer shows that the representations computed from LSTM network are feature rich compared to directly using DSSM based vectors. Many samples in dataset have pair with varying degree of sentence length which affected $P@1$ of previous approaches. In 1D-SLcQA, the $P@1$ at 0.902 is superior to other models as the model can effectively handle the long sentences by LSTM layer and moreover, the convolutional similarity on the latent features tries to classify with high degree of precision using CNN.

B. Qualitative Analysis

Qualitative analysis of our approaches compared with previous approaches is shown in Table III. In Q1, we compared LMIR approach with our *1D-SLcQA* results. LMIR looks for matching keywords "need", "help", "user", "name". Hence the output of the question "*I need help choosing a user name...?*" matches with "*I need help making a user name?*", whereas *1D-SLcQA* outputs the question "*I need a new user name, have any suggestions?*" because *1D-SLcQA* treats "have any suggestions?" as semantically similar to "help". In Q2, comparison is between *T-SCQA* and *1D-SLcQA*. For the question "*What's the most a nurse can earn a month an what type of nurses are there?*", T-SCQA captures contextual mapping "earn a month" to "salary". 1D-SLcQA captures the language (writing style) even if it is different for similar questions. In Q3, comparison is between two of our proposed approaches 1DcQA and SLcQA has been made. For the question "*Q3: What is a good iphone 4 case?*", 1DcQA finds semantically similar question and but 1D-SLcQA also captures features of the phone case. In Q4, comparison performed between SLcQA and 1D-SLcQA. For the question "*Can't start a new Yahoo Group?*", SLcQA returns "*How to start a new yahoo group ?*", where as 1D-SLcQA returns "*While starting a new group, when I enter a group email address, error msg.?*", along with sequential information, 1D-SLcQA also finds the context mapping between "Can't start" and "error msg".

VI. CONCLUSION

In this paper, we model the similarity question retrieval (SQR) problem as a binary classification problem using cross-entropy loss function. We proposed a novel architecture *SLcQA* for similar question retrieval which tries to learn long

TABLE III

QUALITATIVE COMPARATIVE ANALYSIS OF DIFFERENT MODELS FOR SAMPLE QUERIES CHOSEN FROM PUBLICLY RELEASED QUESTIONS. ALL THE MODELS ARE TRAINED FROM THE YAHOO ANSWERS DATASET

<table>
<tr><th colspan="2">Query</th><th>Comment</th></tr>
<tr><td colspan="2">Q1: I need help choosing a user name...?</td><td rowspan="3">LMIR looks for matching keywords "need", "help", "user", "name".</td></tr>
<tr><td>LMIR ([3])</td><td>I need help making a user name?</td></tr>
<tr><td>1D-SLcQA (ours)</td><td>I need a user name have any suggestions ?</td></tr>
<tr><td colspan="2">Q2: What's the most a nurse can earn a month an what type of nurses are there?</td><td rowspan="3">T-SCQA captures contextual mapping "earn a month" to "salary". 1D-SLcQA captures the language (writing style) even if it is different for similar questions</td></tr>
<tr><td>T-SCQA [15]</td><td>What type of nurse earns the highest salary?</td></tr>
<tr><td>1D-SLcQA (ours)</td><td>Which type of nurse get paid more ?</td></tr>
<tr><td colspan="2">Q3: What is a good iphone 4 case?</td><td rowspan="3">1DcQA finds semantically similar question and But 1D-SLcQA also captures features of the phone case.</td></tr>
<tr><td>1DcQA (ours)</td><td>Buying an iPhone 4 case.?</td></tr>
<tr><td>1D-SLcQA (ours)</td><td>Best iphone 4 case to get for protection /sturdiness?</td></tr>
<tr><td colspan="2">Q4: Can't start a new Yahoo Group?</td><td rowspan="3">Along with sequential information, 1D-SLcQA also finds the context mapping between "Can't start" and "error msg."</td></tr>
<tr><td>SLcQA (ours)</td><td>How to start a new yahoo group ?</td></tr>
<tr><td>1D-SLcQA (ours)</td><td>While starting a new group, when I enter a group email address, error msg.?</td></tr>
</table>

term dependencies and captures sequential patterns in query sample. *SLcQA* employs twin LSTM neural networks with shared parameters to learn the semantic similarity between the question and answer pairs. We can extend this work using Bi-Directional LSTMs in place of LSTM, which can help remember more terms. But, the data we experimented on, majority of answers are relatively smaller.

We also proposed *1DcQA*, a one dimensional feature convolution based similarity metric with improved results. *1D-SLcQA*, a hybrid approach combining *SLcQA* and *1DcQA* results in improved matching performance for both textual and semantic matching. Experiments on large scale real-life "Yahoo! Answers" dataset revealed that *1D-SLcQA* outperforms current state-of-the-art approaches based on translation models, latent topic models and the existing deep neural network based models which do not share parameters. As part of future work, we would like to enhance SLcQA with twin Bi-directional LSTM network, which covers both left and right context information of the question and answer pairs. Also we would like to take help of other meta-information like user expertise information, tags, answer quality information to improve the model. Data we experimented with, has question patterns very generic not a specific question pattern, to consider on first few words of question. We can also extend with emphasis boosting for Question phrases.

REFERENCES

[1] Li, B., King, I.: Routing questions to appropriate answerers in community question answering services. In: Proceedings of the 19th ACM international conference on Information and knowledge management, ACM (2010) 1585–1588

[2] Robertson, S.E., Walker, S., Jones, S., et al.: Okapi at trec-3. (1995)

[3] Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems (TOIS) **22**(2) (2004) 179–214

[4] Jeon, J., Croft, W.B., Lee, J.H.: Finding similar questions in large question and answer archives. In: Proceedings of the 14th ACM international conference on Information and knowledge management, ACM (2005) 84–90

[5] Xue, X., Jeon, J., Croft, W.B.: Retrieval models for question and answer archives. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, ACM (2008) 475–482

[6] Zhou, G., Cai, L., Zhao, J., Liu, K.: Phrase-based translation model for question retrieval in community question answer archives. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, Association for Computational Linguistics (2011) 653–662

[7] Zhang, K., Wu, W., Wu, H., Li, Z., Zhou, M.: Question retrieval with high quality answers in community question answering. In: Proceedings of the 23rd ACM international conference on conference on information and knowledge management, ACM (2014) 371–380

[8] Ji, Z., Xu, F., Wang, B., He, B.: Question-answer topic model for question retrieval in community question answering. In: Proceedings of the 21st ACM international conference on Information and knowledge management, ACM (2012) 2471–2474

[9] Cai, L., Zhou, G., Liu, K., Zhao, J.: Learning the latent topics for question retrieval in community qa. (2011)

[10] Zhou, G., He, T., Zhao, J., Hu, P.: Learning continuous word embedding with metadata for question retrieval in community question answering. In: ACL (1). (2015) 250–259

[11] Zhang, K., Wu, W., Wang, F., Zhou, M., Li, Z.: Learning distributed representations of data in community question answering for question retrieval. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, ACM (2016) 533–542

[12] Zhou, G., Zhou, Y., He, T., Wu, W.: Learning semantic representation with neural networks for community question answering retrieval. Knowledge-Based Systems **93** (2016) 75–83

[13] Qiu, X., Huang, X.: Convolutional neural tensor network architecture for community-based question answering. In: IJCAI. (2015) 1305–1311

[14] Das, A., Shrivastava, M., Chinnakotla, M.: Mirror on the wall: Finding similar questions with deep structured topic modeling. In: Pacific-

Asia Conference on Knowledge Discovery and Data Mining, Springer International Publishing (2016) 454–465
[15] Das, A., Yenala, H., Chinnakotla, M., Shrivastava, M.: Together we stand: Siamese networks for similar question retrieval. In: ACL 2016, Association for Computational Linguistics (2016) 378–387
[16] Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., Xu, B.: Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling. CoRR **abs/1611.06639** (2016)
[17] Mueller, J., Thyagarajan, A.: Siamese recurrent architectures for learning sentence similarity. (2016)
[18] Huang, P.S., He, X., Gao, J., Deng, L., Acero, A., Heck, L.: Learning deep structured semantic models for web search using clickthrough data. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management, ACM (2013) 2333–2338
[19] Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). (2014) 1532–1543
[20] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8) (1997) 1735–1780
[21] Werbos, P.J.: Backpropagation Through Time: What It Does and How to Do It. Proceedings of the IEEE **78**(10) (Oct 1990) 1550–1560
[22] Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Representations by Back-Propagating Errors. Cognitive modeling **5**(3) (1988) 1