Siamese Networks for Similar Question Retrieval

Anuj Bansal*, Rishabh Chandra*, Snehasis Pal*, Yash Mandilwar*

(* - *contributed equally)*

1. Introduction

The Community Question Answering (cQA) forums have emerged as popular and effective means of information exchange on the Web. cQA services like Yahoo! Answers, Baidu Zhidao ,Quora , StackOverflow etc. provide a platform for interaction with experts and help users to obtain precise and accurate answers to their questions. But there are cases when some questions go unanswered.

One solution for this would be to find an already answered question and use it's answer as a solution for the unanswered question. To achieve this task it is important that we find the similarity between the questions asked not just syntactically but also semantically. Our project is to implement an algorithm designed by Arpita Das et. al. that follows a new approach of using a Convolutional Neural Network in a Siamese Network using Question-Answer pair dataset. This new method is named as Siamese Convolutional Neural Network for cQA (SCQA).

We train the network using Question-Answer pairs to find a better mapping between easy words (mostly used by a novice) and it's synonymous complicated words (mostly used by professionals). This is better that training with just Question-Question pairs because the latter will only develop mapping with mostly simple words since questions may mostly be asked by a novice and would not find similarity in question if a complicated word but it's semantic meaning is the same as that of a novice's question.

Following sections discuss as follows: Section 2 describes the basic understanding of Siamese Network. Section 3 discusses the archtecture used in the project.

2. Siamese Network

Siamese network is an artificial neural network that use the same weights while working in tandem on two different input vectors to compute comparable output vectors. Often one of the output vectors are precomputed, thus forming a baseline the other output vector are compared against.

In this project we use siamese network with convolution layers to find the similarity between random questions from dataset containing question answer pairs. To see how this works, refer Figure 1. The working is as follows: Let, F (X) be the family of functions with set of parameters W . F (X) is assumed to be differentiable with respect to W . Siamese network seeks a value of the parameter W such that the symmetric similarity metric is small if X1 and X2 belong to the same category, and large if they belong to different categories. Here we consider our1 and X2 to be Q and A which stands for Question and Answer respectively. We use the Question and it's best Answer to train the network to identify similar question and unrelated Question-Answer pairs to train the network to identify dissimilarity in Questions. Thus F(Q) represents a question input in one of the nework and F(A) represents either a best or an unrelated answer. And so we will call S(Q,A) as the output from the Siamese network that measures the semantic relatedness between question answer pair (Q,A) can be de- fined as:

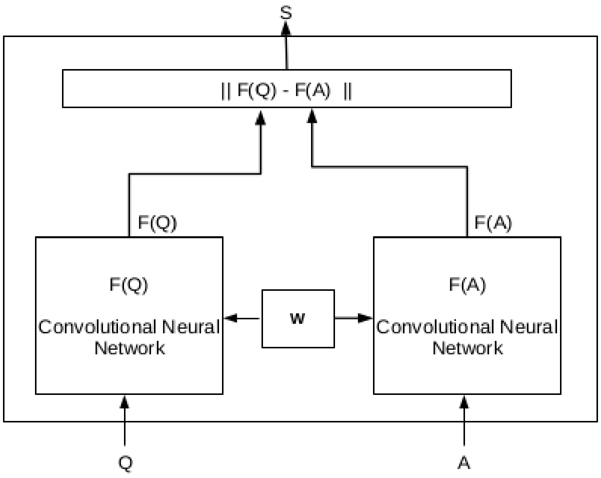


Figure 1: Architecture of Siamese network.

$$S(Q, A) = \| F(Q) - F(A) \|$$

3. Architecture of SCQA

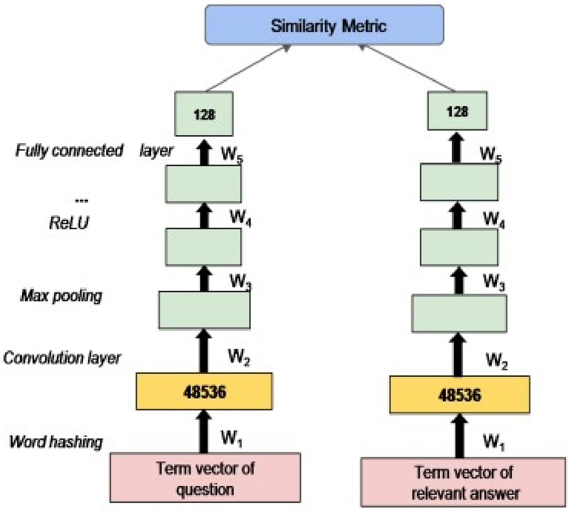


Figure 2: Architecture of SCQA. The network consists of repeating convolution, max pooling and ReLU layers and a fully connected layer. Also the weights W_1 to W_5 are shared between the sub-networks.

SCQA consists of a pair of deep convolutional neural networks (CNN) with convolution, max pooling and rectified linear (ReLU) layers and a fully connected layer at the top. CNN gives a non linear projection of the question and answer term vectors in the semantic space. The semantic vectors yielded are connected to a layer that measures distance or similarity between them. The contrastive loss function combines the distance measure and the label. The gradient of the loss function with respect to the weights and biases shared by the sub-networks, is computed using backpropagation. Stochastic Gradient Descent method is used to update the parameters of the sub-networks.

3.1 Inputs to SCQA

The input to the twin networks of SCQA are word hashed term vectors of the question and answer pair and a label. We one-hot encode the hashed vector. The label indicates whether the sample should be placed nearer or farther in the semantic space. For positive samples (which are expected to be nearer in the semantic space), twin networks are fed with word hashed vectors of question and relevant answers which are marked as “best-answer” or “most voted answers” in the cQA dataset of Yahoo! Answers (question-relevant answer pair). For negative samples (which are expected to be far away from each other in the semantic space), twin networks are fed with word hashed vectors of question and answer of any other random question from the dataset (question-irrelevant answer pair).

3.2 Convolution

After feeding the network with question and answer pair Q_i and A_i respectively, we apply a convolution to the one-hot coded vector of size *w* where *w* is the length of the hashed dictionary that was created by the dataset. Since the input vector is one dimensional, our convolution filter of weights will also be of just one dimension. The filter is slided across the length of vector such that the resulting connectivity looks like a series of overlapping receptive fields which output of width *w*.

3.3 Max Pooling

Max pooling performs a kind of non-linear downsampling. It splits the filter outputs into small nonoverlapping grids (larger grids result to greater the signal reduction), and take the maximum value in each grid as the value in the output of reduced size. Max pooling layer is applied on top of the output given by convolutional network to extract the crucial local features to form a fixed-length feature vector.

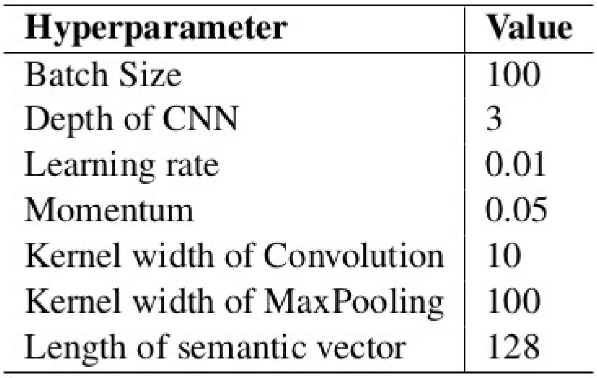
3.4 ReLU

Non-linear function Rectified linear unit (ReLU) is applied element-wise to the output of max pooling layer. ReLU is defined as $f(x) = max(0, x)$. ReLU is preferred because it simplifies backpropagation, makes learning faster and also avoids saturation.

3.5 Fully Connected layer

The terminal layer of the convolutional neural subnetworks is a fully connected layer. It converts the output of the last ReLU layer into a fixed-length semantic vector s. s ∈ R n s of the input to the subnetwork. We have set the output layer to be 128 as mentioned in the paper.

3.6 Hyperparameters



Hyperparameter	Value
Batch Size	100
Depth of CNN	3
Learning rate	0.01
Momentum	0.05
Kernel width of Convolution	10
Kernel width of MaxPooling	100
Length of semantic vector	128

Table 1: Hyperparameters of $SCQA$.

We have followed the same hyperparameters mentioned in the paper.

4. Training

To begin our training, we would have our question and answer pair to feed into our Siamese Architecture. The question and answer pair could be a question and it’s relevant answer pair and a question with it’s non relevant answer. This is then trained with the Siamese Network described above with the hyperparameters mentioned in Table 1. Given the the two pairs we recieve a loss between the two and is backpropagated which changes the shared weights *w*. The loss function is defined as follows:

If the label is 1 then loss(Q_i,A_i) is $1- cos(F(Q_i,A_i))$ and if the label is 0 then the the loss is $max(0,cos(Q_i,A_i) – m)$ where *m* is the margin which decides by how much distance dissimilar pairs should be moved away from each other. It varies between 0 and 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 parameters shared by the convolutional sub-networks are updated using Stochastic Gradient descent (SGD).

5. Testing

There are two phase to training. One is calculating the semantic similarity and other is the textual similarity. In the semantic similarity we take two question-question pairs and measure it’s metric using the parameters learnt in the training stage. This metric tells how similar are the two questions semantically regardless if matching words were found in both the questions or not. Whereas the textual similarity matches the similar words between the question. This model will give better performance in datasets with good mix of questions that are lexically and semantically similar. To use both the metrics we have followed the weighted maetrics mentioned by the authors as mentioned below:

$$\text{score} = \alpha * \text{SSS} + (1 - \alpha) * \text{TSS}$$

where SSS is Semantic Similar Score anf TSS and Textual Similar score

where α is a parameter where α control the weights given to semantic metric and textual similarity metric models. It range from 0 to 1.

6. Result

Example 1:

```
Question - how to attract a girl

Best Three Answers
How should I act on a date?
Is a transponder required to fly in class C airspace?
How do I end a date with a girl I'm not interested in?
```

Example 2:

```
Question - best place to eat

Best Three Answers
best place to meet guys in the bay area?
Whats a good place to eat in LA?
Where's the best place to get my FICO score?
```

Example 3:

```
Question - best internet site

Best Three Answers
what is the best news site on the net?
How many websites are on the internet?
What is the singel most important thing you are missing on the internet ?
```

Example 4:

```
Question – how do i find peace?

Best Three Answers
Who won the first nobel peace prize?
Where can I find help about a war?
need to find reserch about work ethics?
```

Example 5:

```
Question - which fruit is good for health

Best Three Answers
Why is red wine good for your heart?
Why are blueberries so good for your health?
How do you get a toddler to eat what's good for him?
```

Example 6:

```
Question - Who is most famous celebrity

Best Three Answers
Who is the most famous woman athlete of all time?
What is emo?
What is "I"?
```

7. Conclusions

In this project we have implemented a T-SCQA which tries to find the the similarity between two questions both semantically and textually using Siamese Network with CNN. With respect to above results we get about 66% good questions retrievals. The model was trained with 5000 examples out of 2 million examples available to us. The above result could have been improved with more training time and more training examples.

8. Contributions

1. Creating Three Hash Vector: Anuj Bansal, Rishabh Chandra

2. Cleaning Dataset: Snehasis Pal

3. Siamese Architecture: Yash Mandilwar

4. Loss Function and Textual Similarity: Rishabh Chandra

5. Training: Snehashis Pal, Yash Mandilwar

6. Testing: Anuj Bansal, Snehasis Pal

7. Report: Yash Mandilwar, Anuj Bansal

8. Presentation: Rishabh Chandra

Acknowledgements

Yahoo! Answer Dataset: https://webscope.sandbox.yahoo.com/catalog.php?datatype=l (L-6)

Paper: https://pdfs.semanticscholar.org/62a9/7ac04e742ad1513cf164760e4d6a25d93203.pdf

TA: Avinash Kumar

Faculty: Dr. Ravi Kiran Sarvadevabhatla

Project Link: https://github.com/devloper13/SiameseNetworkProject

Drive Link:

1. **Data**:

`https://drive.google.com/open?id=17VMN5CJA05vTPEs15gw-W2ocxmEITQEH`

2. **Three hash**:

`https://drive.google.com/openid=1Jhj9OazxPnvLcuuZsZvNFfpnnsFg88I7`