Paraphrase Question Identification

Yang Hsuan Ru, hyangap@connect.ust.hk

Abstract

In this paper, we explore methods of determining semantic equivalence between pairs of questions, with a dataset released by Quora. A deep learning approach that uses a Siamese GRU neural network connected to a distance-measure layer to determine the semantic equivalence is used in this paper. Variations of the network are explored to improve network performance. Applying an adapted attention mechanism on the GRU network has been found to improve model performance significantly. The effectiveness of several regularization techniques on the GRU cells are evaluated, improving the model performance close to the state-of-art level.

1 Introduction

Determining semantic similarity and equivalence has been a popular task in the field of natural language processing, including applications to semantic analysis and machine translation. With the recent development of deep learning methods, huge progress has been made in the field of measuring semantic similarity. For online question-answering platforms, such as Quora or Stackoverflow, detecting questions with semantic equivalence would enable them to provide a better quality of service. The answer of a duplicate question can be displayed instantly instead of requiring human identification and a manually added link to the duplicate question. In this project, different configurations of a RNN-based language model were experimented to further improve the performance on this task.

2 Related Works

2.1 Siamese Network

With the renaissance of neural network models, several deep learning approaches has been proposed to tackle this problem. A type of approach is based on the Siamese architecture[1]. In this framework, two identical neural network encoders (e.g., a CNN or a RNN) that shares the same parameters and configurations are applied to each of the input sentences individually, so that both of the two sentences are encoded into sentence vectors in the same embedding space. A matching decision is then made solely based on the two sentence vectors.

2.2 Attention Mechanism

The attention mechanism proposed by Bahdanau et al.[2] in machine translation allows the network to know where to look as it is performing its task. In a encoder-decoder model, instead of encoding the input sequence into a single fixed context vector, we let the model learn how to generate a context vector for each output time step.

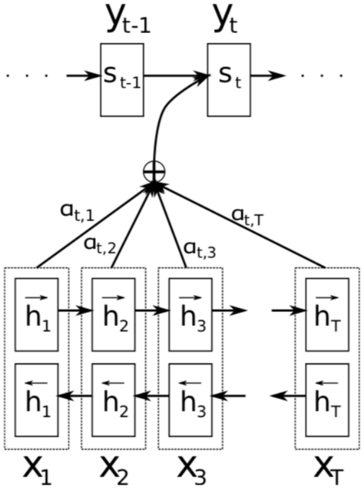

Figure 1: Illustration of the attention mechanism

The context vector c_i depends on a sequence of annotations $(h_1, , h_{T_x})$ to which an encoder maps the input sentence. Each annotation h_i contains information about the whole input sequence with a strong focus on the parts surrounding the i-th word of the input sequence. The context vector c_i is, then, computed as a weighted sum of these annotations h_i:

$$c_i = \sum_{j=1}^{T_x} a_{ij} h_i \tag{1}$$

The weight a_{ij} of each annotation h_j is computed by

$$a_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{Tx} exp(e_{ik})} \tag{2}$$

where

$$e_{ij} = a(s_{i1}, h_j) \tag{3}$$

is an alignment model which scores how well the inputs around position j and the output at position i match. The important part is that each decoder output word y_t now depends on a weighted combination of all the input states, not just the last state.

3 Approach

3.1 Data Preprocessing

The Quora dataset consists of 404,351 pairs of questions that are labeled as duplicate or not duplicate. The tokenizer from the keras text-preprocessing library is used to pre-process the data, which removes all punctuations and converts all characters to lower case. Each word is then converted into a unique integer with a corresponding ID mapping to the GloVe vectors. Each sentence is truncated to a maximum length of 30 words. The training/validation/test set split used in this project is same to those from Wang et al.[3]

is duplidate ?	question1	question2
1	What should I do to avoid sleeping in class?	How do I not sleep in a boring class ?
0	Do women support each other more than men do ?	Do women need more compliments than men
1	How can one root android devices ?	How do I root an Android device ?
0	How did Hitler come to power ?	Who followed Hitler to power ?

Table 1: Sample question pairs from the dataset

3.2 Network Architecture

The proposed network is consisted of three parts:

- An embedding layer for mapping each word in the sentence to its word embedding
- A Siamese GRU layer for encoding each sentence into a vector
- A fully connected layer that measures the distance between the encoded sentence pairs and predicts whether the sentence pair is duplicate.

The weights of the embedding layer is initialized by the 300-dimensional GloVe vectors pre-trained by Pennington et al.[4] , and the GRU cells and the fully connected layer are initialized using Xavier initialization while the biases are zero-initialized. A dropout value of 0.2 is applied after each layer of the fully connected layer.

3.3 Attention Mechanism

The concept of the attention mechanism were adapted in the model, encoding each sentence into a vector with the sum of different weights (pays different attention to) of each hidden state output.

3.4 Loss Function

The loss in the model is the binary-crossentropy loss with L2 regularization, defined by:

$$Loss = \frac{-1}{n} \sum (ylog(p)) + ((1-y)log(1-p)) + \lambda|\theta|^2 \tag{4}$$

Where θ is a 1-dimensional vector containing all of the weights of the GRU layer and the fully connected layer, excluding the bias. The Adam optimizer with an initial learning rate of 0.001 and a learning rate decay of 0.2 every 10 epochs is used for optimization. The back-propagation algorithm is provided by Googles TensorFlow library.

3.5 Embedding Dropout

As proposed by Gal & Ghahramani[5], embedding dropout is performing dropout on the embedding matrix at a word level, which is equivalent to dropping random words in the sentence. The remaining non-dropped-out word embeddings are scaled by $\frac{1}{1-p}$, where p is the probability of embedding dropout.

4 Experiments

4.1 Model Types

The performance of the Siamese network using GRU cells with three of its variations: Siamese network with Bi-directional GRU cells, Siamese network with attention mechanism, and Siamese network with Bi-directional GRU cells and attention mechanism are evaluated. The accuracy and the F1 scores (on the validation set) of the different model types tested are showed on Table 2.

Model	Accuracy (Val.)	F1 (Val.)	AUC (Val.)
BiGRU attention	87.80	81.97	91.37
GRU attention	87.37	81.62	92.07
BiGRU	87.22	81.38	91.87
GRU	86.96	80.73	91.67

Table 2: Results on effects of bi-directional GRU and attention

The best model was the bi-directional GRU Siamese network with the attention mechanism. The attention mechanism shows improvement in network performance in both uni-directional and bi-directional GRUs.

4.2 Increasing Loss

During the training phase of the model, an unusual behavior of the model was observed. While the training loss and error decreases normally, the validation error decreases but loss increases, as shown in figure 2.

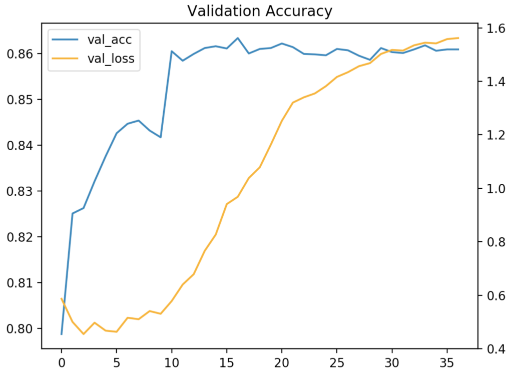

Figure 2: Increasing Validation Accuracy and Loss during training

The cause of this is that the model is making more correct predictions but predicting worse on false predictions, as shown in figure 3:

Since the loss function is concaved upwards, a similar change in the average predictions of the true predictions and the false predictions will actually lead to an increase in total loss. This might be a result of overfitting. To find out the part of the network that is contributing to overfitting, the weights of different parts of the network, the

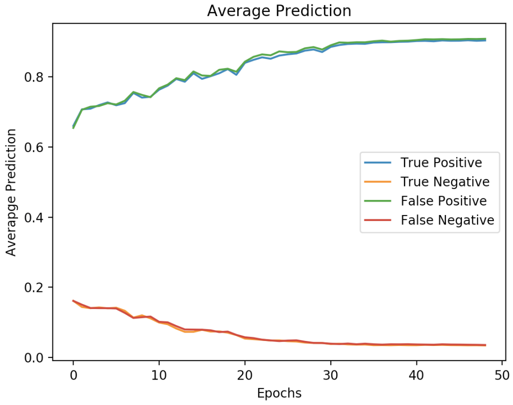

Figure 3: Average Prediction of the Model

GRU layer and the fully-connected layer, were frozen after several epochs in order to observe the change in loss. As

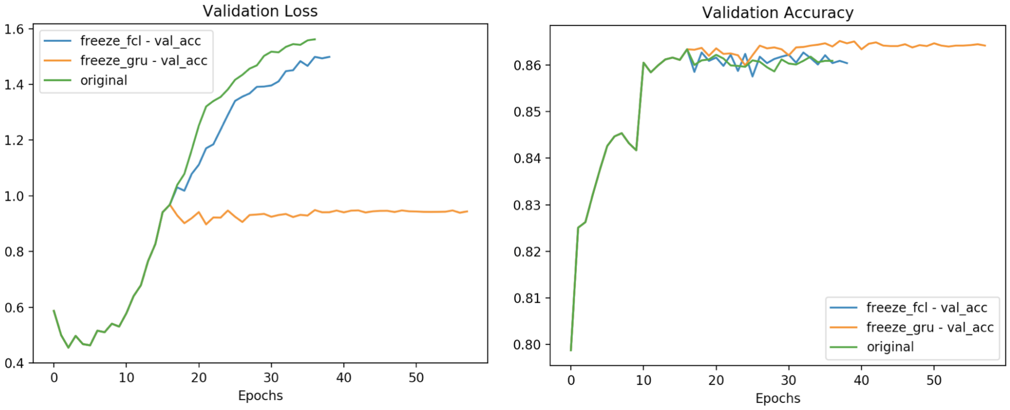

Figure 4: Change in Validation Accuracy and Loss after freezing weights

shown in figure 4, the loss stopped increasing while the accuracy slightly improved after freezing the GRU layer. It can thus be inferred that the GRU layer has been overfitting. Hence, several regularization techniques on the GRU layer were explored.

4.3 L2 Regularization

The first method is adding L2 regularization on the weights of the GRU cells.

The best parameter is $\lambda = 0.0005$ with an 0.7% increase in accuracy.

Regularization Coefficient	Accuracy (Val.)	F1 (Val.)	AUC(Val.)
0.0005	86.98	80.76	91.35
0.0001	86.78	80.37	89.73
0.001	86.41	80.08	90.98
0	86.34	79.96	90.69
0.005	85.61	78.98	91.49
0.01	85.07	78.34	91.65
0.05	84.77	77.74	91.68
0.1	82.66	74.29	89.59
0.5	81.57	72.62	88.76

Table 3: Results of L2 Regularization on the GRU cells

4.4 Embedding Dropout

The second method is embedding dropout. As mentioned in Section 3, random words in the sentence are dropped and the remaining non-dropped words scale according to the dropout value.

Model	Accuracy (Val.)	F1 (Val.)	AUC (Val.)
Embedding Dropout	87.06	81.72	92.45
GRU	86.96	80.73	91.67

Table 4: Results of embedding dropout

The embedding dropout has shown a positive impact on the model's performance.

4.5 Additional Dense Layer

To decrease the number of parameters of the GRU cells, an additional fully-connected layer is added between the embedding layer and the GRU layer, decreasing the word dimension from 300 to 100.

The additional dense layer has been shown to improve the model performance generally. It can be inferred that the most significant features of the original embedding vector is captured by the additional dense layer while decreasing the total parameters of the GRU cells.

Model	Accuracy(Val.)	F1(Val.)	AUC(Val.)
base+fcl	86.74	80.48	90.05
base	86.34	79.96	90.69
attention+fcl	86.89	80.61	90.27
attention	86.51	80.05	89.02
BiGRU+fcl	86.87	80.45	89.87
BiGRU	86.65	80.38	91.40
BiGRU+attention+fcl	87.79	82.05	91.43
BiGRU+attention	87.80	81.97	91.37

Table 5: Results of the additional dense layer

4.6 Putting it together

Combining the techniques mentioned above, a model combining bi-directional GRU with attention and the regularization techniques on the GRU cells, a test accuracy of 87.10% is achieved, which is close to the state-of-the-art performance (89%)[6].

4.7 Pre-Compute Sentence Vectors

In practical, the goal is to find duplicate questions in an existing database of questions for every new sentence. However, feeding all questions in the database into the model would be too computationally expensive. To solve this problem, we propose a candidate selection method. First, the model is separated into two pars: the sentence encoding layer (GRU layer), and the distance-measure layer (fully-connected layer).

All questions in the database can be pre-encoded into a vector through the GRU layer and be stored. For the new question, after encoding it through the GRU layer, the Euclidean distance among it and all the vectors in the database is then calculated, which is much cheaper in terms of computational resource compared to running the whole model. A candidate set is then selected based on the euclidean distance, and are fed into the fully connected layer along with the new question. Out of 1000 candidates selected, about 52 of them are classified as duplicate questions on average. Compared to that of randomly selected questions, an average of 4 out of 1000, the candidate selection method indeed serves as an efficient way to search for duplicate questions.

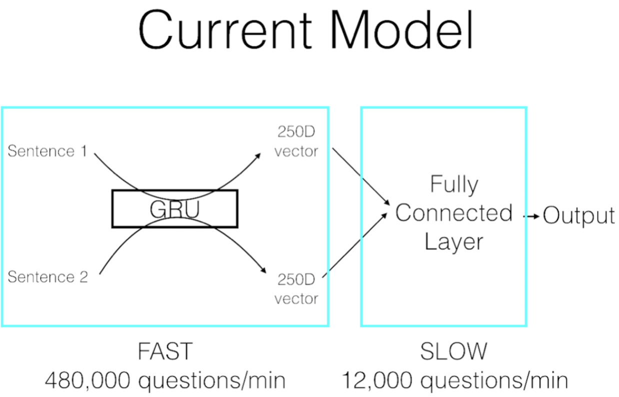

Figure 5: Illustration of current model architecture

5 Conclusion

Our research builds on the previous results of identifying duplicate questions using Siamese GRU Networks. The main findings in this report include: the effectiveness of bi-directional GRU cells and the adapted attention mechanism, which helps the network to capture the significant information in the sentences better; and the effectiveness of various regularization techniques on the GRU cells, including L2 regularization, embedding dropout, and an additional layer of fully connected layer before the GRU Layer. A test accuracy of 87.10% was achieved through combining all the aforementioned findings.

For future work, we believe data augmentation has a great potential to further improve model performance. Besides naive methods such as question pairs with identical questions or switching the questions in the pair, a possible way to generate data is using templates. For example, we can generate duplicate questions with the template "How can I be a good $< OCCUPATION >$?" and "What should I do to become a great $< OCCUPATION >$?". This type of template augmentation technique has been proved effective in VQA (Video Question Answering) tasks[7], and we believe it would as well lead to an improvement in the duplicate question identification task. For pre-computing sentence vectors, a possible improvement to the candidate selection process would be storing the pre-computed vectors in data structures such as k-d trees. In this way, computational time for selecting k candidates from the database can be improved from $O(n)$ to $O(klog(n))$ since computing the Euclidean distance among the new questions with all existing ones is not required any more.

References

[1] Jane Bromley, James W. Bentz, Leon Bottou, Isabelle Guyon, Yann LeCun, Cliff Moore, Ed- uard Sackinger, and Roopak Shah. Signature verification using a siamese time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence, 1993.*

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *Proceedings of the 2015 International Conference on Learning Representations.*

[3] Zhiguo Wang, Wael Hamza, and Radu Florian. Bilateral multi-perspective matching for natural language sentences. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*

[4] Richard Socher Jeffrey Pennington and Christopher D. Manning. Glove: Global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*

[5] Gal,Y.andGhahramani,Z.A theoretically grounded application of dropout in recurrent neural networks. *Proceedings of the 2016 Conference on Neural Information Processing Systems*

[6] Yichen Gong, Heng Luo, and Jian Zhang. Natural language inference over interaction space. *International Conference on Learning Representations, 2018.*

[7] Kushal Kafle and Christopher Kanan. 2017. Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding, 2017*