Question Answering on SQuAD 2.0

Liz Guo
Department of Electrical Engineering
Stanford University
`lizguo@stanford.edu`

1 Introduction

Machine reading comprehension style question answering and automated question answering have gained significant popularity over recent years because of its wide usage in applications and also because its theoretical values in natural language processing. In this project, we use SQuAD2.0 dataset [1] and build an end-to-end system to perform automated question answering base on the given context. System is supposed to provide correct answer to an answerable question about a given context by selecting a segment of text from corresponding paragraph, and abstain when presented with a query that cannot be answered based on given passage. Our model combines BiDAF, Self-attention and Encoder block inspired by [2] and achieves EM score 65.65, F1 score 68.79 after ensemble 9 single models.

2 Related Work

The baseline model we use here is highly based on Bidirectional attention flow mechanism discussed in [4]. It incorporates both context-question and question-context attention. The second paper [3] focuses on reading comprehension which presents a novel structure called *Gated Self-Matching Networks* which is the refinement of basic seq-to-seq with attention model. The key innovation is its passage self-matching layer which matches passage against itself and effectively encodes information from the whole passage [3]. The idea of encoder block comes from [2] achieves better performance than bi-directional LSTM to learn temporal dependencies between words in the question and context.

3 Approach

In this section, we first describe our baseline model and then introduce our primary model architecture which combines idea from BiDAF [4], QANet [2] and R-NET [3].

3.1 Baseline model

Our baseline model is to incorporate character-level embedding into BiDAF model. We code learnable character embedding rather than pretrained one since we finds it works better in our task. More specifically, for each character c, we look up a dense character embedding, apply 1-dimensional convolution, max pooling and highway network [5] to get character level word embedding. We set the dimension of character level word embedding to be the same as the pretrained word embedding in our implementation.

3.2 Primary model

Our primary model contains embedding layer, encoding layer, bi-directional attention layer, self-matching attention layer and output layer. Figure 1 gives the multi-stage model architecture.

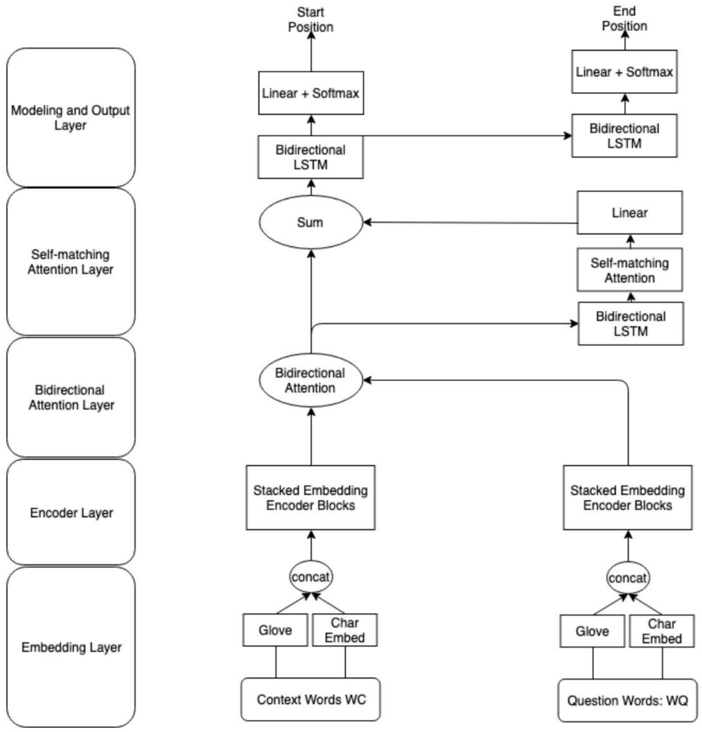


Figure 1: Overview of model architecture

3.2.1 Input Embedding layer

We use both word-level embedding and character-level embedding to represent context and question. In terms of word-level embedding, we use pretrained 300-dimensional Glove vectors. For the character-level embedding, we use the techniques described in section 3.1 but set the dimension to be 200. Then we concatenate them to obtain the representation of each word. Therefore after the embedding layer, context words can be represented as [$\mathbf{c}_1$,...,$\mathbf{c}_N$] where $c_i \in \mathrm{R}^{500}$ and question words can be represented as [$\mathbf{q}_1$,...,$\mathbf{q}_M$] where $q_i \in \mathrm{R}^{500}$.

3.2.2 Encoder layer

This layer allows our model to learn temporal dependencies between timesteps of the embedding layer's output. Instead of using a bi-directional LSTM to encode context and question, we followed [2] and built encoder layer as a stack of the following basic unit: [conv_layer * 4 + self_attention + feed_forward]. For the convolution layer, we use depthwise separable convolutions. For the self-attention layer, we adopt the idea from [7]. We also apply the layer normalization and residual connection as indicated in the paper.

3.2.3 Bidirectional attention layer

After the encoder layer, we have dense representation of context **C** and question **Q** where **C** $\in \mathrm{R}^{clen*H}$ and **Q** $\in \mathrm{R}^{qlen*H}$, H is the hidden size, clen is context length and qlen is question length. In this layer, we fuse context representation into question representation and fuse question representation into context representation and concatenate the result. The Context-to-query (C2Q) attention signifies which query words are most relevant to each context word and the Query-to-context (Q2C) attention signifies which context words have the closest similarity to one of the query words and are hence critical for answering the query [4]. Therefore a combination of the two attentions gives question-aware context information. The similarity of i-th context word and j-th question word S_{ij} can be computed as:

$$S_{ij} = W_{sim}^T [c_i; q_j; c_i \circ q_j] \tag{1}$$

3.2.4 Self-matching attention layer

As mentioned in R-NET [4], question-aware passage representation has limited knowledge of surrounding context in practice. So it proposes self-attention to directly match the question-aware passage representation against itself. As shown in Figure 1, we first pass the question-aware context information G a bidirectional LSTM layer to get G'. And then we adopt the similarity-based attention idea to construct another similarity matrix S' where

$$S'_{ij} = {W'_{sim}}^T [g'_i; g'_j; g'_i \circ g'_j] \tag{2}$$

$W'_{sim} \in \mathrm{R}^{6H}$ and it is a trainable weight. S'_{ij} represents the similarity of i-th context word and j-th context word. According to [8], we compute the attention score a' as

$$a'_t = softmax(S'_{t:}) \tag{3}$$

And we compute the attention vector **M** as

$$m_t = \sum_{i=1}^{N} a_{ti} * g'_i \tag{4}$$

The final context representation by self attention M' can be represented as:

$$m'_t = g_t + ReLU(W''_m[m_t; g'_t; m_t \circ g'_t]) \tag{5}$$

$W''_{sim} \in \mathrm{R}^{6H}$ and it is a trainable weight.

3.2.5 Modeling and Output layer

After self attention layer, the modeling layer integrates temporal information between context representations conditioned on the question. Here we use a two layer bi-directional LSTM. For the output layer, we produce a vector of probabilities corresponding to each position in the context being start or end of an answer span. We adopt negative sum of log probabilities of prediction as objective function.

3.3 Model ensemble

To further improve model performance we also use model ensemble. We give random initialization of 9 primary models and train each of them using different hyperparameters such as learning rate, batch size, whether using learning rate warm up or not and take the max vote of start and end index. If after max vote, the start index is larger than end index, then we treat the prediction as no answer.

4 Experiment

4.1 Dataset

SQuAD 2.0 is a new reading comprehension dataset that combines 100,000 answerable questions from previous version SQuAD 1.1 with 53,775 new, unanswerable questions about the same paragraphs. We use the custom train and dev set which contains 130319 and 6078 examples respectively for tuning and evaluating model performance.

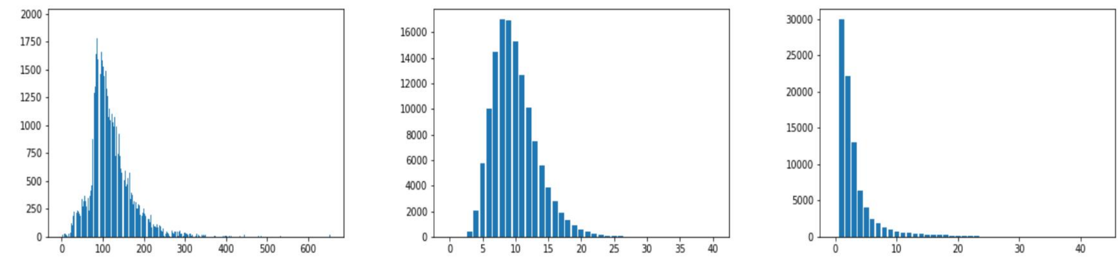
Before building models, we analyze data which can help us get a general understanding of the distribution of question and context. The following analysis is based on our training dataset. Figure 2 is a histogram plot of the number of tokens for context, question and answer. From the graph it can be seen that the mode length of context is 87 and 8 for question. In terms of question type, as Figure 2b shows, among the 66.63% answerable questions almost half are interested in "What" questions.

4.2 Evaluation method

The main evaluation metrics we used are EM and F1 score which is standard for SQuAD dataset. We also check AvNA (Answer vs. No Answer) which measures the classification accuracy in determining having answer and no-answer predictions.

4.3 Experiment details

In order to do model ensemble, we also ran our primary model on Amazon EC2 p2.xlarge for many times with different hyperparameters (it takes about an hour to run a epoch). For instance, we change the learning rate from default 0.5 to 0.3, batch size varies from 16 to 128 and whether using learning rate warm up or not. We also tried to use Adam optimizer but the performance is worse than Adadelta. For each model we run at least 30 epoches and might continue to run if the performance still improve.



(a) Context Length (b) Question Length (c) Answer Length

Figure 2: Histogram plot of the number of tokens for context, question, answer in the training dataset

5 Results and Analysis

5.1 Model performance

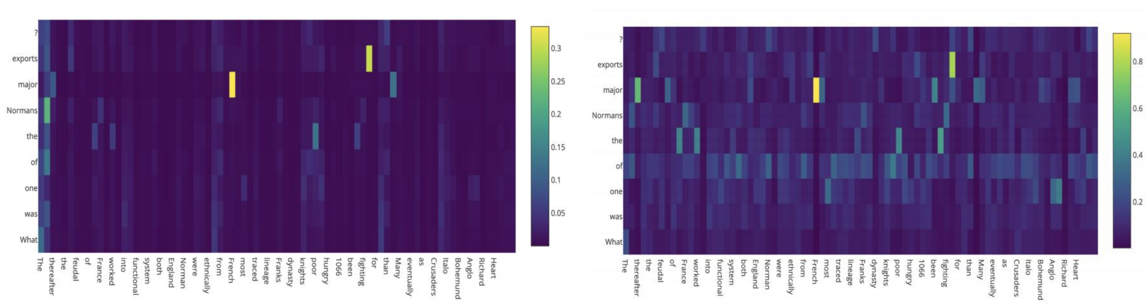
The F1 and EM results in the development set are listed in Table 1 below:

Table 1: F1 and EM results

	EM	F1	AvNA
pure baseline	57.82	61.03	67.65
extend baseline	60.091	63.285	69.92
Single primary model	62.91	66.38	73.04
Ensemble primary model	65.65	68.79	74.01

5.2 Attention visualization

In Figure 3 we show the Q2C and C2Q attention matrix of 37th example in dev_eval.json.



(a) Q2C example (37th) in the dev set (b) C2Q example (37th) in the dev set

Figure 3: Attention Visualization

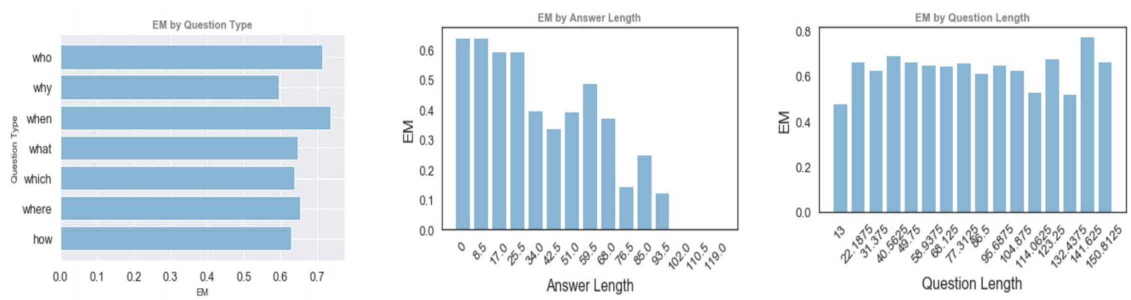
Our attention mechanism can successfully capture the word "horsemen" in the context which correspond to exports in the question. The answer to this question is "fighting horsemen". The attention also captures other words such as French because it is highly related to the word major in the question. This might lead to a wrong prediction of our model since our model thinks major and French is highly correlated, but French is not related to the answer in this question. The high correlation might come from word embedding.

C2Q attention signifies which question words are most relevant to each context word. As Figure 3b shows, the word "Normans", "major", "exports" are relevant to the context word which is correct in this example.

5.3 Error analysis

5.3.1 Quantitative error analysis

To gain deeper understanding of the model performance, we divide the questions into different types based on the start word in questions and visualize average EM scores on it, as Figure 4(a) shown. We notice our model preforms best on questions start with "when", but provide worst results on questions start with "why". One reason we think about it is questions which relate to time tend to have short and relatively obvious answer which are easy to locate in context, while questions interested in reason usually have relative long and complex answers.
To further verify our assumption, we analyze EM score on different answer length based on characters in Figure 4(b). As the answer length increases, the model gives worse performance. We also try to explore the relationship between EM and question length in Figure 4(c), which doesn't provide too much useful pattern except that it shows we have the highest EM in the second last bucket.



(a) EM by question types (b) EM by answer length (c) EM by question length

Figure 4: EM score by different question types, answer length and question length

5.3.2 Qualitative error analysis

We provide a representative example to illustrate the future improvement of our model. As shown in Figure 5, our model does not capture the word in the latter half which is "and also by Cherokee". This suggests that our model needs improvement in understanding the context information better. Although self attention can give certain knowledge to surrounding context, it is not enough in this case. A more complex attention mechanism is required.

- **Question:** What tribes supported British?
- **Context:** Further south the Southeast interior was dominated by Siouan-speaking Catawba, Muskogee-speaking Creek and Choctaw, and the Iroquoian-speaking Cherokee tribes. When war broke out, the French used their trading connections to recruit fighters from tribes in western portions of the Great Lakes region (an area not directly subject to the conflict between the French and British), including the Huron, Mississauga, Ojibwa, Winnebago, and Potawatomi. The British were supported in the war by the Iroquois Six Nations, and also by the Cherokee – until differences sparked the Anglo-Cherokee War in 1758. In 1758 the Pennsylvania government successfully negotiated the Treaty of Easton, in which a number of tribes in the Ohio Country promised neutrality in exchange for land concessions and other considerations. Most of the other northern tribes sided with the French, their primary trading partner and supplier of arms. The Creek and Cherokee were subject to diplomatic efforts by both the French and British to gain either their support or neutrality in the conflict. It was not uncommon for small bands to participate on the "other side" of the conflict from formally negotiated agreements, as most tribes were decentralized and bands made their own decisions about warfare.
- **Answer:** Iroquois Six Nations, and also by the Cherokee
- **Prediction:** Iroquois Six Nations

Figure 5: Wrong prediction due to shortsightness

6 Conclusion

In this project, we implemented several end-to-end models to perform automated question answering based on the given context. Our best model combines idea from BiDAF, QANet, R-NET and our ensemble model achieves EM 65.65, F1 68.79 and is rank top 10 in the development set as of March 18. From the error analysis, we find our model has bad performance when dealing with long contexts and long answers. This is a future improvement of our model.

Acknowledgments

We thank all the instructors of CS230 for bringing me such a meaningful experience during this quarter.

References

[1] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. arXiv preprint arXiv:1806.03822, 2018.

[2] Yu, Adams Wei, et al. "Qanet: Combining local convolution with global self-attention for reading comprehension." arXiv preprint arXiv:1804.09541 (2018).

[3] Wang, Wenhui, et al. "Gated self-matching networks for reading comprehension and question answering." Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vol. 1. 2017.

[4] Seo, Minjoon, et al. "Bidirectional attention flow for machine comprehension." arXiv preprint arXiv:1611.01603 (2016).

[5] Srivastava, Rupesh Kumar, Klaus Greff, and Jürgen Schmidhuber. "Highway networks." arXiv preprint arXiv:1505.00387 (2015).

[6] Luong, Minh-Thang, and Christopher D. Manning. "Achieving open vocabulary neural machine translation with hybrid word-character models." arXiv preprint arXiv:1604.00788 (2016).

[7] Vaswani, Ashish, et al. "Attention is all you need." Advances in Neural Information Processing Systems. 2017.

[8] Web.Stanford.Edu, 2019, https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/6878267.pdf.