CS 224N Improved QANet on SQuAD 2.0

Chenyang Dai (qddaichy@stanford.edu)

Objectives

The goal of this project it to build reading comprehension systems for the Stanford Question Answering Dataset (SQuAD) 2.0 wuthout pre-trained models. We have three contributsions:

1. Improve the baseline BiDAF[1] model.
2. Implement QANet[2] from scratch and search for the best model size.
3. Improve QANet performance with using an input embedding refine layer and a condition output layer

BiDAF

We improved the embedding layer of BiDAF model by introducing learnable character level embedding. We also introduced a fusion function after the co-attention layer to better fuse different attention components. Fused attention is computed as:

$$A_{fuse} = ReLU(W[c, a, c \odot a, c \odot b] + b)$$

QANet

QANet is a feed-forward model that consists of only convolutions and self-attention. The core building block of QANet is **encoder block**. It consists of a sinusoidal positional encoding layer, followed by X convolution layers, a self multi-head attention layer and a feed-forward layer (Figure 1).

QANet adopts the core **bi-directional attention** idea from BiDA. It is computed as:

Compute similarity matrix S and normalize over each row and column to get $\overline{S}$ and $\overline{\overline{S}}$ respectively

the context-to-query attention is $A = \overline{S} \cdot Q^T$, and the

the query to context attention is $B = \overline{S} \cdot \overline{\overline{S}}^T \cdot C^T$

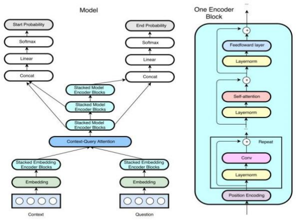

Figure 1: QANet architecture

Improved QANet Layers

Input embedding layer with convolution and linear projections

We added two additional 1D convolutions while adopting the linear projection. The first convolution refines character embedding into 128-D hidden size and the second convolution further refines the 256-D concatenated embedding into the final 128-D representation. (Figure 2)

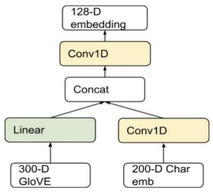

Figure 2: Improved Input Embedding Layer

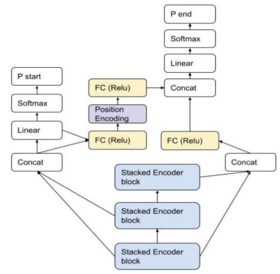

Figure 3: Improved Output Layer

Output layer with conditioning end prediction on start prediction

It's helpful to know where the answer starts when predicting the end of the answer. We designed the new output layer with this conditioning (Figure 3). The P_{start} is computed the same as before, the calculations for P_{end} are as follows:

$$A = [M_0; M_1] \quad B = [M_0; M_2]$$

$$A_{weight} = W_0 A \quad B_2 = ReLU(W_1 B)$$

$$A_{weighted} = A \odot A_{weight}$$

$$A_2 = ReLU(W_2 A_{weighted})$$

$$A_2^{'} = PositionEncoding(A_2)$$

$$A_3 = ReLU(W_3 A_2^{'})$$

$$P_{end} = softmax(W_4[A_3; B_2])$$

In $A_{weighted}$ words with higher probability of being the answer start will be more activated. The $A_{weighted}$ is then sent through a position encoding function for position information hardening. Finally, the output is used as additional information when predicting the end position.

Experiment Results

Model Name	Dev F1	Dev EM	Dev AvNA	Test F1	Test EM
BiDAF					
BiDAF (Baseline)	60.90	57.65	68.14	-	-
BiDAF + Char Embedding	63.25	59.97	70.87	-	-
BiDAF + Char Embedding + Fusion	**65.39**	**62.28**	**71.52**	-	-
Scaled QANet					
QANet small (3 blocks, single head)	64.80	61.35	71.11	-	-
QANet medium (5 blocks, 4 heads)	65.39	61.5	72.09	-	-
QANet (7 blocks, 8 heads)	**66.63**	**63.25**	**72.89**	65.11	61.52
QANet large (9 blocks, 8 heads)	66.15	62.87	71.18	-	-
QANet large (2) (160 hidden size)	65.16	61.86	-	-	-
Improved QANet					
QANet + Condition Output Layer	68.10	64.73	73.70	-	-
QANet + Input Emb Refine Layer + Condition Output Layer	**68.61**	**65.54**	**73.90**	**67.81**	**64.82**
Ensemble Model					
Bidaf + QANet + QANet Improved	69.26	66.21	74.26	-	-

Table 1: Performance of experimented models

Analysis

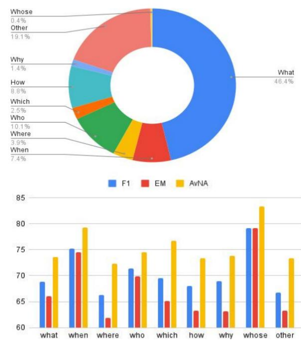

Figure 4: Question categories breakdown

We break down questions by common question words. We observed relatively consistent performance across all categories (61.9-79.1 EM, 66.2-79.17 F1). The top 3 model performing categories are "Whose", "When" and "Who".

The 3 least performing categories are "Where", "How" and "Why". "How" and "Why" are naturally more difficult to answer because these categories require reasoning.

Reference

[1] Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. arXiv: 1611.01603, 2016.

[2] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. Qanet: Combining local convolution with global self-attention for reading comprehension. arXiv: 1804.09541, 2018.