IID SQuAD track: Comparing QANet with BiDAF

CS224N Default Final Project: Bingqi Sun, Ruochen(Chloe) Liu, Shanduojiao Jiang

Stanford
Computer Science

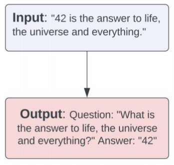
Introduction

In the task of reading comprehension or question answering, a model will be given a paragraph and a question about that paragraph, as input. The goal is to answer the question correctly. This is an interesting task as it could be used to evaluate how well a model can "understand" text. Current end-to-end machine reading and question answering models such as BiDAF[1] can achieve relatively good results. However, given how powerful transformer models are for tasks such as translation and text summarization, we want to further improve the QA system performance by borrowing ideas from transformers.

Methods

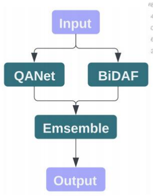
In an attempt to improve the QA system performance, we proposed the following workflow:

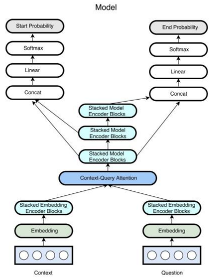
1. Improve on the given BiDAF-based model by adding character level embedding, as it allows us to *condition on the internal structure of words* (morphology), and *better handle out-of-vocabulary words*
2. Use data augmentation to further improve the BiDAF-based model. We used a T-5 based model that could generate questions by simply passing the text.
3. Implement QANet[2] and compare its performance with the BiDAF model
4. Explore other methods such as ensembling to improve the model performance

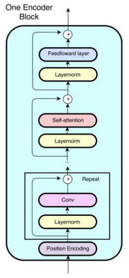


Input: "42 is the answer to life, the universe and everything."

Output: Question: "What is the answer to life, the universe and everything?" Answer: "42"




Model
Start Probability
Softmax
Linear
Concat
End Probability
Softmax
Linear
Concat
Stacked Model Encoder Blocks
Stacked Model Encoder Blocks
Stacked Model Encoder Blocks
Context-Query Attention
Stacked Embedding Encoder Blocks
Embedding
Context
Stacked Embedding Encoder Blocks
Embedding
Question



The five layers in QANet:

1. **Input Embedding Layer**
   The output of a given word x is $[x_w; x_c] \in \mathbf{R}^{p_1+p_2}$ where x_w and x_c are the word embedding and the convolution output of character embedding of x respectively
2. **Embedding Encoder Layer**
   This layer is a stack of the following building blocks: [convolution-layer × # + self-attention-layer + feed-forward-layer]
3. **Context-Query Attention Layer**
   A bidirectional attention flow layer
4. **Model Encoder Layer**
   Refines the sequence of vectors
5. **Output Layer**
   Produces a vector of probabilities corresponding to each position in the context

Experiments & Results

Dataset

We run all of our experiments on the Stanford Question Answering Dataset (SQuAD) 2.0 [3] dataset. There are around 150k questions in total, and roughly half of the questions cannot be answered using the provided paragraph. If the question is answerable, the answer is a chunk of text taken directly from the paragraph.

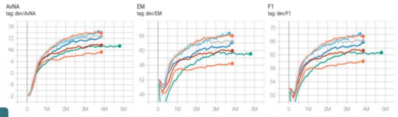
Question: Why was Tesla returned to Gospic?
Context paragraph: …Tesla was returned to Gospic under police guard for not having a residence permit. On 17 April 1879 Milutin Tesla died at the age…
Answer: not having a residence permit

Evaluation Metrics

Evaluation is based on Exact Match (EM) score and F1 score, comparing the prediction with three provided answers by the crowd worker, where the maximum score will be considered. We also report AvNA.

- **Exact Match** is a binary measure of whether the output matches the ground truth answer exactly
- **F1** score is less strict, it is calculated with 2 × prediction × recall / (precision + recall)
- **AvNA** measures the accuracy when only considering the answer vs. no-answer predictions.

Result



Bottom Orange: BiDAF, Green: BiDAF + Char.Emb + Data Augmentation, Red: BiDAF + Char.Emb, Dark blue: QANet(4), Grey: QANet(2), Top Orange: QANet(3), Light blue: QANet(1)

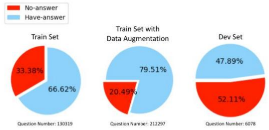
	Hidden Size	# Heads	RNN
QANet(1)	128	1	LSTM
QANet(2)	128	4	LSTM
QANet(3)	256	8	LSTM
QANet(4)	128	1	GRU

	F1	EM	AvNA
BiDAF	60.42	56.74	67.7
BiDAF + Char.Emb	63.79	60.34	70.07
BiDAF + Char.Emb + Data Augmentation	62.75	59.15	69.55
QANet(1) + Char.Emb	68.46	64.86	74.47
QANet(2) + Char.Emb	64.76	62.58	68.78
QANet(3) + Char.Emb	61.5	59.22	67.17
QANet(4) + Char.Emb	66.13	62.49	73.25
Ensemble (Average)	**70.09**	**67.50**	**74.36**
Ensemble (Manual)	69.96	67.36	N/A

Analysis

1. Dataset analysis

- Bias between datasets
- Potential reason why model with data augmentation doesn't work well on the dev set: it decreased the proportion of unanswerable questions



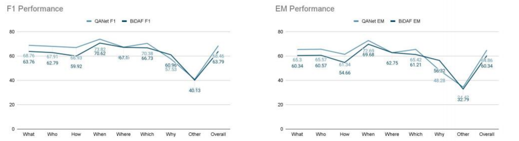
2. Comparing QANet and BiDAF

- **Answer vs. No-answer predictions**

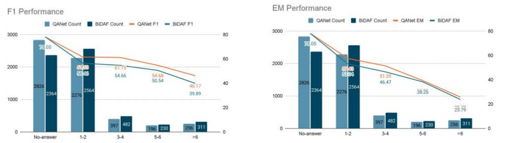
*QANet in this section refers to QANet(1)

	QANet + Char.Emb Prediction		BiDAF + Char.Emb Prediction	
	Answerable	Unanswerable	Answerable	Unanswerable
Answerable	37.42% (TP)	10.44% (FN)	39.1% (TP)	8.75% (FN)
Unanswerable	15.09% (FP)	37.05% (TN)	21.17% (FP)	30.97% (TN)

- **Performance by Question Type**
  Though QANet achieves higher performance on most of question types, BiDAF model perform better on 'Why' questions



- **Performance by Answer Length**



1) Both models achieved higher performance for shorter answers. 2) QANet achieved higher performance for all answer lengths. 3) QANet is more likely to predict "unanswerable" questions while BiDAF is more likely to predict long answers. 4) EM is close to F1 when answer is short, but the gap becomes larger when the answer length gets larger.

3. Manual Classification Ensemble Model

- Use BiDAF when the question is "why" type
- Prefer using shorter answers
- Use "Unanswerable"when either model outputs "Unanswerable"

References

[1] https://arxiv.org/pdf/1611.01603.pdf
[2] https://arxiv.org/pdf/1804.09541.pdf
[3] https://rajpurkar.github.io/SQuAD-explorer/