Augmenting BiDAF with Components from R-NET for Question and Answering on SQUAD 2.0

Brandon Vu, Nyle Wong, Richard Chen; *Mentor: Fenglu Hong*

Department of Computer Science

Introduction

- **SQuAD2.0** [1] is a question-answering dataset based on context-question-answer triplets with context from Wikipedia articles and question-passage pairs generated from Amazon Mechanical Turk
- Desired answer is certain span of words in the given context; goal is to find start and end token
- Notably, there are also "unanswerable" questions added in SQuAD2.0 where the answer cannot be found within the context, and models are expected to indicate this
- Our model aims to achieve a high performance on the SQuAD2.0 dataset without the use of any pretrained transformers

Background

Example Question

Context: The further decline of Byzantine state-of-affairs paved the road to a third attack in 1185…

Question: When did the Normans attack Dyrrachium?

Answer: 1185

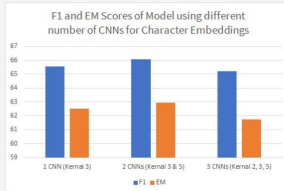
Previous Work

- The framework model and primary baseline for comparison we used was the Bidirectional Attention Flow network (BiDAF) [2].
- BiDAF incorporate key contributions: **temporally independent attention**, **bidirectional attention**, and **flowing attention for each time step**
- R-Net [3] included two novel innovations: a **gated attention recurrent network** and a **self matching attention layer**

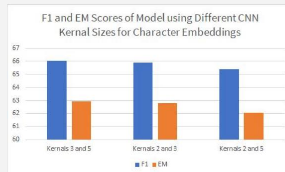
Methods

- We used the BiDAF model without any modifications as the baseline model
- Our first approach was to add **self-matching attention** from R-Net
- Next, we added **gated attention** from R-Net to the BiDAF attention layer
- We incorporated **character embeddings** using convolution neural networks (CNNs)
  - Tried combinations of 1 to 3 CNNs with kernel sizes of 2, 3, and 5
  - Kernel sizes generate embeddings of different subwords of the specified length
- We experimented with using a **Gated Recurrent Network (GRU)** in place of the original LSTM network
- We also experimented with varying **batch sizes** in training

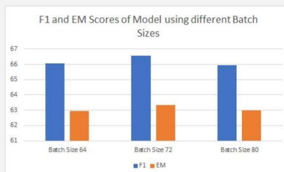
Experiments



- Experimented with multiple combinations of number of CNNs and kernel sizes used for character embeddings
- 2 CNNs with kernel sizes of 3 and 5 performed the best



- Further exploring kernel sizes for character embeddings show that different kernel sizes were not as optimal for 2 CNNs



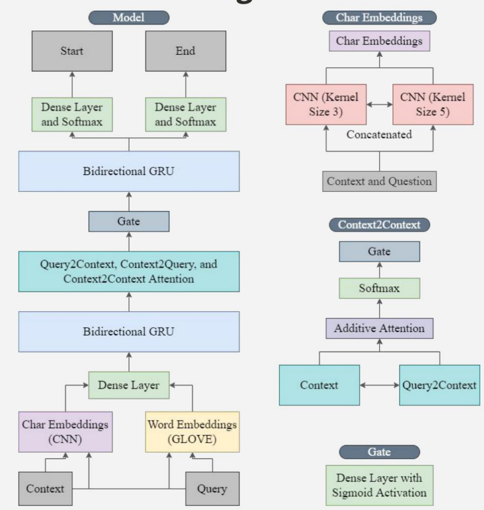
- Batch size differences showed similar results
- There was an increase in performance up to a batch size of 72, but decreased at larger sizes

Results

- Our best performing model built on the baseline BiDAF model with gated self attention, 2 CNN character embeddings with kernel sizes of 3 and 5, and trained at a batch size of 72

Model Description	Dev Set F1	Dev Set EM	Test Set F1	Test Set EM
Baseline	60.559	57.049	-	-
+ GRU + Gated Self Attn	62.90	59.40	-	-
+ 1 CNN Char Emb (Kernal 3)	65.55	62.51	-	-
+ 2 CNN Char Emb (Kernals 2 & 3)	65.9	62.78	-	-
+ 2 CNN Char Emb (Kernals 3 & 5)	66.05	62.95	-	-
+ Increase batch size 64 -> 72	**66.56**	**63.32**	**64.304**	**60.845**
+ Increase batch size 72 -> 80	65.93	62.98	-	-
+ 2 CNN Char Emb (Kernals 2 & 5)	65.38	62.07	-	-
+ 3 CNN Char Emb (Kernals 2 & 3 & 5)	65.20	61.72	-	-

Diagram



Conclusion

- Putting together the augmentations from R-Net, our finalized model improved upon the baseline on the dev set by **~10%**, bring the F1 and EM scores to **66.56** and **63.32** from **60.559** and **57.049** respectively
- There is room to further experiment with hyperparameter tuning to optimize the model
- It would be useful to explore ideas like pointer networks to replace the current output layer and condition the end location on the start location

References

[1] Konstantin Lopyrev Pranav Rajpurkar, Jian Zhang and Percy Liang. Squad: 100, 000+ questions for machine comprehension of text. 2016.
[2] Ali Farhadi Minjoon Seo, Aniruddha Kembhavi and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. 2016.
[3] Microsoft Research Natural Language Computing Group. R-net: Machine reading comprehension with self-matching networks. 2017.