Using Character Embedding and QANet to Improve Perfomance on Question-Answering Task (SQuAD)

Stanford CS224N Default Project

Gurdeep Sullan, Takara Truong
Department of Computer Science
Stanford University
gksullan@stanford.edu, takaraet@stanford.edu

Abstract

Given a baseline model BiDAF without character embeddings, we aim to build an NLP model that performs better than baseline. The two main approaches we take are to (1) create a character level embedding and (2) build a QANet Model. We perform an ablation study to understand how these model architecture features influence performance. Results show that the model with QANet and character embedding performs the best, with a score (evaluated on the test set) EM = 60.592, F1 = 64.353. Further analysis compares context to query attention between the baseline and the top performing models.

1 Key Information to include

- Mentor: Yian Zhang
- External Collaborators: N/A
- Sharing project: N/A

2 Introduction

While models such as BiDAF [1] show decent results on question-answer datasets like SQuAD, the reliance on recurrent structures makes both training and inference slow. The slow and computationally expensive training time also makes a larger impact on the carbon footprint of these models. Compounded, these issues prevent such models from being deployed in industry.

In this project, we aim to reduce both the computation time and carbon footprint while achieving better performance than the BiDAF baseline on SQuAD [2]. Towards this end, we implement two improvements: adding character level embeddings, and adapting the architecture to follow the QA-Net model.

3 Related Work

RNN models have been shown to have good performance on the SQUAD dataset in the question-answering task. As mentioned above, the BiDAF based RNN model is one such architecture. The BiDAF model consists of an Embedding layer, an Encoder Layer, an Attention Layer, a Modeling Layer, and an Output layer. The embedding layer takes input word indices and converts them into word embeddings. The encoder and model layer use a recurrent structure, specifically a bi-directional LSTM. The attention layer makes use of the bidirectional attention flow and calculates both context-to-query attention and query-to-context attention. The final output layer takes the return of the model layer and outputs start and stop location probabilities of the potential answer within the context.

There are papers which present extensions to the traditional RNN structure. For example, Zaman et al have combined an RNN with a convolutional neural network to better preserve local information [3].

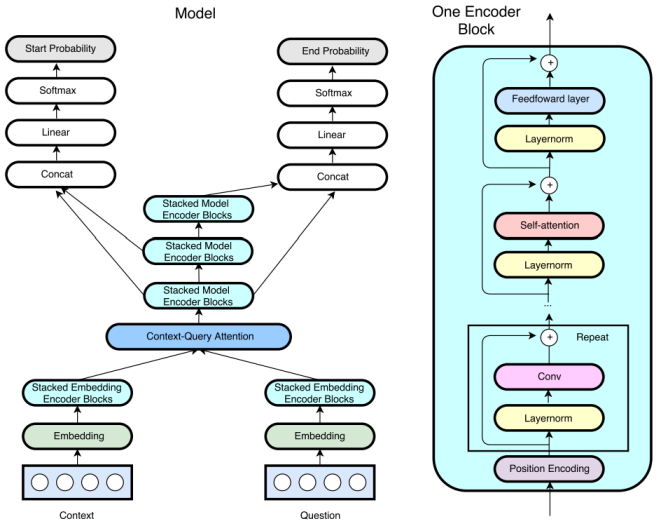
Another extension of the RNN model is the Query Reduction Network + RNN to specifically address the issue of multi-hop question answering [4]. This is when question-answering specifically requires the understanding of multiple facts within the context to answer the question. The structure of the model is a single recurrent unit that is a variant of an RNN with an update gate which provides a local sigmoid attention and a reducing function the reduces the query. The authors achieved a performance of their model with an error rate of 9.9% on the bAbI QA dataset.

Transformer based models are capable of outperforming many RNN based networks by using self-attention and positional encoding. Self-attention gives the model the ability to attend to all words within a sequence, offsetting the drawbacks that RNN based encoders have, that is, the forgetting of information over time. Additionally, RNN's must process the sequence in order which retains positional information but is not parallelizable . In contrast, attention based models encodes position into the hidden states of the sequence so that the entire sequence can be processed at once, hence, being parallelizable . The breakthrough of the transformer model, has led to recent successes such as BERT [5] and GPT3 [6].

4 Approach

The first variant we implemented was the addition of character level embedding to our baseline BiDAF model. The reason behind adding character embeddings was because as shown in previous work [7] [1], including character embeddings improves reading comprehension and the BiDAF model. To implement this, we loaded the randomly generated character vector embeddings, applied a convolution, and concatenated the resultant character embeddings with the word embeddings. We then passed this concatenated result through a linear layer to reduce the new hidden dimension size from 2 * hidden_size to hidden_size, so that it was the proper shape for the next layer, the Highway Encoder, and future layers.

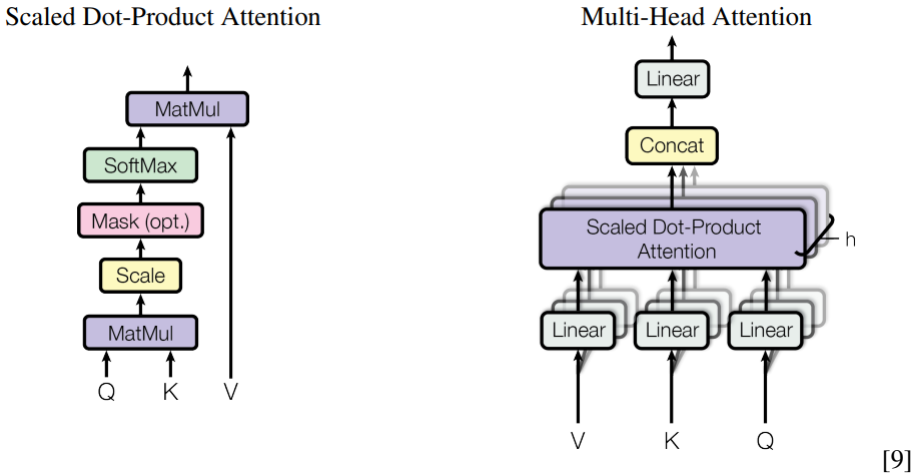
The second variant we have implemented is the QANet model. This model takes advantage of the fact that there is no RNN, making it more time-efficient due to parallelization. The drawback is that it can be less memory-efficient. The general structure of the QANet model is an embedding layer, an embedding encoder block, a context-query attention layer, a series of model encoder blocks, and a final output layer. The key repeated structure here is the encoder block, which is comprised of a positional encoder, a series of convolutional layers, a self attention layer, and a feedforward layer. Additionally, there is a residual that is being kept track of between each of these pieces in the block, as well as an applied layernorm. The general structure of the QANet is below [8]:



The encoder block structure first passes the input data through a position encoder, which is used to provide information on where words are in the sequence. The formula for the positional encoding [9] was calculated like so:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

After positional encoding of the word+char embedding, a series of convolutions is applied which is used to model local interactions, and then multiheaded attention is applied, which is used to model global interactions [9]. As opposed to the single headed approach, the multiheaded approach is able to filter and focus on different "edges"/"features" for each head, much like a filter in a CNN.



[9]

The final step of the encoder block is a simple feed forward layer, the result of which is passed to the next layer, which is the context-query attention. Our model made use of the BiDAF attention that was provided in the started code for the baseline for this layer.

After calculating context query attention, the model is passed through a stacked model encoder block. We implemented this by initializing the same encoder block as above, stacked n=7 times. The inputs were passed through this same stacked model encoder block three times. The final output layer concatenated the three outputs of these three pass-thrus of the stacked encoder block to produce output probabilities for start and end positions of the answer.

5 Experiments

To understand how model architecture features influence performance, we perform an ablation study. To do so, we start with the baseline, BiDAf model, and incrementally add model changes. The experiments that we run include the following: BiDAf, BiDAf + character embedding, QANet, and character embedding + QANet.

5.1 Data

This project uses the SQuAD dataset [2], comprising of context-question-answering tasks. An example is shown below:

- **Question:** What was the Song dynasty's capital?
- **Context:** After strengthening his government in northern China, Kublai pursued an expansionist policy in line with the tradition of Mongol and Chinese imperialism. He renewed a massive drive against the Song dynasty to the south. Kublai besieged Xiangyang between 1268 and 1273, the last obstacle in his way to capture the rich Yangzi River basin. An unsuccessful naval expedition was undertaken against Japan in 1274. Kublai captured the Song capital of Hangzhou in 1276, the wealthiest city of China. Song loyalists escaped from the capital and enthroned a young child as Emperor Bing of Song. The Mongols defeated the loyalists at the battle of Yamen in 1279. The last Song emperor drowned, bringing an end to the Song dynasty. The conquest of the Song reunited northern and southern China for the first time in three hundred years.
- **Answer:** Hangzhou
- **Prediction:** Hangzhou

Table 1: Model Performance

Model	EM	F1	Carboon Footprint [kgC02eq]
BiDAF	61.32	58.13	0.32
QANet	59.45	63.20	2.08
BiDAF + char-embed	64.82	61.65	0.35
QANet + char-embed	**67.75**	**64.06**	2.09

5.2 Evaluation method

We use three evaluation metrics: EM, F1, and the carbon footprint of training the model. Both EM and F1 are standard metrics used to compare models on SQuAD. Estimations of carbon footprint were conducted using the MachineLearning Impact calculator presented in [10].

5.3 Experimental details

The model and training configurations for QAnet will be presented in this section. We follow the QAnet paper and use the ADAM optimizer with $\beta_1 = 0.8$, $\beta_1 = 0.999$ and $\epsilon = 10^{-7}$. For regularization, we use dropout with a value of 0.1 in all layers except for the character embedding which uses half this value. For the encoder block, the hidden size and number of convolution filters are 100. The number of convolution layers in the embedding encoder and modeling encoder are the same as the QAnet paper with 4 layers and 2 layers with kernel size 7 and 5, respectively. The block numbers for the embedding encoder is 1, while the modeling encoder has 7.

5.4 Results

Results of the experiments show that the QANet has some improvement over the BiDAF model as noted by a large increase in F1 score and a slight decrease in EM score, Table 1. Additionally, character embedding improves both the BiDAF and QANet model significantly. From this experiment, we find that the QANet + character embedding model has the best performance on the devset when compared to other variants and use this model on the test set. The result of this model on the **test set: EM = 60.592, F1 = 64.353**.

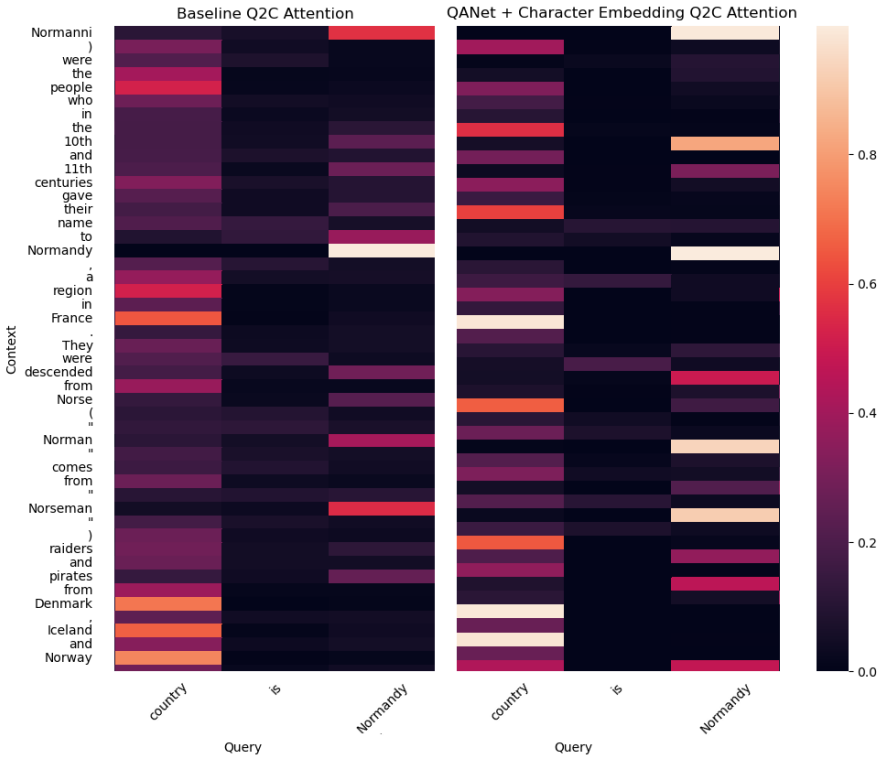
Looking at the carbon footprint of each architectural feature, we see that character embedding is a more eco-friendly option than changing the architecture, Table 1. In perspective, the carbon footprint of training QANet + character embedding is equivalent to driving an average car for 8.37 km or burning about .97 kg of coal, compared to BiDAF with character embedding at 1.29 km or burning 0.19 kg of coal [10].

6 Analysis

We qualitatively examine how the model variants use attention for the context and query. Below are heatmaps of Query to Context attention on an example datapoint for the baseline (BiDAF) and top performing model (QANet + char embedding).

In Figure 1 is an example that contains part of a query: "In what country is Normandy located?", and part of a context (zoomed in for ease of readability to a window of words containing the answer). For both the baseline and the QANet + char embedding models, we can see the query word "country" attends highly to "France", "Denmark", "Iceland" and "Norway" in the context. Additionally, we can see the query word "Normandy" attends to the words "Normanni" and "Normandy". In this case, "France" is the answer, and overall the BiDAF attention layer seems to capture meaningful attention from the query to the context after word embedding and the encoding layers. However, we can see from the heatmap that the QANet attention is better at focusing on the relevant terms, whereas in the baseline model, the signal is slightly muted. We speculate this improvement in the QANet may be due to extra processing and filtering done in the QANet encoder block to create more meaningful word representations.

Figure 1: Query 2 Context Attention



Below, we present some text examples of predictions made on the DEV dataset by both the worst and best performing models from our ablation study. The first example is one of the baseline BiDAF model correctly identifying the answer from the context. The second example is one of the baseline BiDAF model incorrectly not finding the answer when there is one in the context.

- **Question:** When was the military-political complex reflected upon within the scope of understanding imperialism?
- **Context:** The correlation between capitalism, aristocracy, and imperialism has long been debated among historians and political theorists. Much of the debate was pioneered by such theorists as J. A. Hobson (1858–1940), Joseph Schumpeter (1883–1950), Thorstein Veblen (1857–1929), and Norman Angell (1872–1967). While these non-Marxist writers were at their most prolific before World War I, they remained active in the interwar years. Their combined work informed the study of imperialism and it's impact on Europe, as well as contributed to reflections on the rise of the military-political complex in the United States from the 1950s. Hobson argued that domestic social reforms could cure the international disease of imperialism by removing its economic foundation. Hobson theorized that state intervention through taxation could boost broader consumption, create wealth, and encourage a peaceful, tolerant, multipolar world order.
- **Answer:** the 1950s
- **Prediction:** 1950s

- **Question:** How are the total numbers of seats allocated to parties?
- **Context:** The total number of seats in the Parliament are allocated to parties proportionally to the number of votes received in the second vote of the ballot using the d'Hondt method. For example, to determine who is awarded the first list seat, the number of list votes cast for each party is divided by one plus the number of seats the party won in the region (at this point just constituency seats). The party with the highest quotient is awarded the seat, which is then added to its constituency seats in allocating the second seat. This is repeated iteratively until all available list seats are allocated.
- **Answer:** proportionally to the number of votes received
- **Prediction:** N/A

In the positive example, we can see that the model correctly identifies the "1950's" as the correct answer (although, missing the "the"), which is presented in the later part of the context. However, in

the negative example, it was not able to find the answer, which is presented in the first line of the context. One explanation for this incorrect prediction may be the relatively long sequence length between the answer and the end of the sequence, which results in RNN memory loss over processing of the information.

The next two examples are from the QANet + character embedding model; the first is an example of correct prediction and the second is an example of incorrect prediction.

- **Question:** What co-receptor recruits molecules inside the T cell that are responsible for cell activation?
- **Context:** Helper T cells express T cell receptors (TCR) that recognize antigen bound to Class II MHC molecules. The MHC:antigen complex is also recognized by the helper cell's CD4 co-receptor, which recruits molecules inside the T cell (e.g., Lck) that are responsible for the T cell's activation. Helper T cells have a weaker association with the MHC:antigen complex than observed for killer T cells, meaning many receptors (around 200–300) on the helper T cell must be bound by an MHC:antigen in order to activate the helper cell, while killer T cells can be activated by engagement of a single MHC:antigen molecule. Helper T cell activation also requires longer duration of engagement with an antigen-presenting cell. The activation of a resting helper T cell causes it to release cytokines that influence the activity of many cell types. Cytokine signals produced by helper T cells enhance the microbicidal function of macrophages and the activity of killer T cells. In addition, helper T cell activation causes an upregulation of molecules expressed on the T cell's surface, such as CD40 ligand (also called CD154), which provide extra stimulatory signals typically required to activate antibody-producing B cells.
- **Answer:** CD4 co-receptor
- **Prediction:** CD4

- **Question:** Where was the Gate of King Hugo?
- **Context:** In this last connection, the name could suggest the derogatory inference of superstitious worship; popular fancy held that Huguon, the gate of King Hugo, was haunted by the ghost of le roi Huguet (regarded by Roman Catholics as an infamous scoundrel) and other spirits, who instead of being in Purgatory came back to harm the living at night. It was in this place in Tours that the prétendus réformés ("these supposedly 'reformed'") habitually gathered at night, both for political purposes, and for prayer and singing psalms. Such explanations have been traced to the contemporary, Reguier de la Plancha (d. 1560), who in De l'Estat de France offered the following account as to the origin of the name, as cited by The Cape Monthly:
- **Answer:** Tours
- **Prediction:** Huguon

We can see that the QANet model in the "CD4" example is able to find the answer though most of the text was highly technical. It may be possible that the character-level embedding provides additional information at the sub-word level that allows it to learn how to answer this question that has a non-traditional English word answer. Additionally, it is able to find the answer at the beginning of a relatively long stretch of context, which the BiDAF baseline model was struggling to do. In the negative example, we see that the prediction incorrectly returns "Huguon". One reason for this might be the self-attention mechanism in the encoder block as well as the context-query attention, which caused the model to attend to words similar to "Hugo". Additionally, this example had quite a bit of non-English words in the context and answer, and it may be possible that the model just did not train on enough examples with French words to learn a meaningful representation of the language.

7 Conclusion

Overall, we present character embedding and QANet as solutions to the question-answer task that give decent improvements to the performance over the baseline model. Through the process of building several elements of this model architecture from scratch, we gained a deeper understanding of NLP. Specifically, we implemented the entire encoder block from scratch, including the multiheaded self-attention. However, we did notice that performance improved with the use of the Pytorch multiheaded attention module; thus this is a limitation of our current approach. Future work would be to optimize our multi-headed attention to get performance on par with the PyTorch module.

We noticed that our performance in EM and F1 scores is somewhat below the paper implementation. Future work to extend this model would be to apply the stochastic layer dropout as mentioned in the paper to improve generalizability. Additionally, we could implement the context-query attention that offered slightly better performance compared to the BiDAF attention used in our project.

References

[1] Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603, 2016.

[2] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics (ACL)*, 2018.

[3] M. M. Arefin Zaman and Sadia Zaman Mishu. Convolutional recurrent neural network for question answering. In *2017 3rd International Conference on Electrical Information and Communication Technology (EICT)*, pages 1–6, 2017.

[4] Min Joon Seo, Hannaneh Hajishirzi, and Ali Farhadi. Query-regression networks for machine comprehension. *CoRR*, abs/1606.04582, 2016.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[7] Zhilin Yang, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W. Cohen, and Ruslan Salakhutdinov. Words or characters? fine-grained gating for reading comprehension. *CoRR*, abs/1611.01724, 2016.

[8] 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. *CoRR*, abs/1804.09541, 2018.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[10] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

A Appendix (optional)

If you wish, you can include an appendix, which should be part of the main PDF, and does not count towards the 6-8 page limit. Appendices can be useful to supply extra details, examples, figures, results, visualizations, etc., that you couldn't fit into the main paper. However, your grader *does not* have to read your appendix, and you should assume that you will be graded based on the content of the main part of your paper only.