Exploring Embedding and Attention Improvements to BiDAF SQuAD Model

Colin Dolese
Stanford University
cdolese@stanford.edu

Abstract

This project aims to improve upon a baseline Bi-Directional Attention Flow model focused on solving the SQuAD question answering task using Non Pre-Trained Contextual Embedding methods. These improvements are targeted primarily at the embedding and attention layers of the current model. In particular three additional models are explored in this research. The first is a BiDAF model with character-level embeddings, extending the baseline BiDAF model's word embedding layer. The second is a BiDAF model with an added self-attention layer based on the R-Net model. The third model is a BiDAF model using a GRU RNN rather than the original LSTM RNN. Beyond individual experimentation on each of these three models, cross-experimentation was done to determine the best performing combination. In the end a character-level embeddings BiDAF model using a GRU RNN achieved the best overall performance.

1 Introduction

Question answering as a reading comprehension task has been the focus of many recent advancements in natural language processing research. One of the most popular datasets for measuring success at this task is the Stanford Question Answering Dataset (SQuAD) [4]. SQuAD consists of context paragraph, question, and answer triplets. A model that solves this task takes in the context paragraph and question as input, and predicts the answer to this question as a span of text from the context paragraph. The metrics used for measuring a model's success are "Exact-Match" (EM), the percent of predicted answers that match the ground truth perfectly, and F1, the harmonic mean of precision and recall. SQuAD 2.0 also introduces unanswerable questions and motivates an additional metric, "Answer vs No Answer" (AvNA), which measures the models accuracy at classifying questions as answerable or unanswerable. The results of various models on these metrics are posted on an official SQuAD leaderboard.

This paper will describe efforts to improve a baseline Bi-Directional Attention Flow (BiDAF) model that uses purely word-level embeddings [5]. All of the most successful models on the current SQuAD leaderboard use pre-trained contextual embeddings (PCE). Specifically, PCE models use pre-trained word embeddings that depend on the context of the paragraph they appear in. The models described in this paper will not use PCE, although it is reasonable to assume that the models could incorporate PCE and achieve overall better results.

To improve the baseline BiDAF model this paper will focus on improvements to the embedding layer and the attention layer. For the embeddings layer we will add character-level embeddings to the current word-level embeddings. This improvement follows directly from the original BiDAF model, which also used character-level embeddings. For the attention layer, we will add a new self-attention layer as described by Microsoft Research in their R-Net paper. Beyond adding and testing these improvements, we will perform further testing by combining these two approaches and exploring

varitions to the type of RNN used by the BiDAF model.

Our final analysis and comparison of results will be done using both quantitative measurement of EM and F1 scores and qualitative observation of answer predictions and their corresponding question and context paragraph. Where possible, discrepancies in qualitative results will be analyzed and explained.

2 Related Work

Question-answering is one example of a set of NLP tasks known as reading comprehension or machine comprehension (MC). Early MC data-sets were small and did not lend themselves well to training end-to-end neural models. The massive cloze-style CNN/DailyMail and Children's Book Test data-sets, released in 2015 and 2016 respectively, allowed neural models to successfully tackle MC tasks. Cloze-style MC data-sets contain query, document, answer triplets, and the goal of a model is to predict the answer to the query using the context. SQuAD is also a cloze-style data-set.

Recent neural models that have performed well on cloze-style data-sets distinguish themselves mainly by how they perform attention. Specifically, there are three primary ways that these models handled attention prior to BiDAF and R-Net. In the first model type attention weights are updated dynamically, a process known as "dynamic attention." One of the earliest state-of-the-art models for the CNN/DailyMail data-set was the "Attentive Reader" model proposed by Hermann et al. This model performs attention at the sentence level and updates word embeddings based on attention scores. As a result, future attention scores are influenced by previous attention scores, and for this reason dynamic attention models are referred to as "memory networks."

In the second model type attention weights are computed only once. An example of this is Cui et al.'s "attention-over-attention" method [2]. Cui et al. obtain both a "document-to-query" and "query-to-document" attention and then combine these two attentions to get an "attended-attention" (i.e. attention-over-attention). This method was able to outperform state-of-the-art methods at the time of publication on both the CNN/DailyMail and Child Book data-sets. BiDAF also follows this general attention paradigm.

In the third model type attention is repeatedly computed through multiple layers alternating between document attention and query attention, a technique known as "multi-hop." Sordoni et al.'s method is one example of a multi-hop architecture [6]. Sordoni et al. propose an "iterative alternating attention layer," whereby query-attention and document-attention are each computed separately given the previous hidden state in a repeating process. The goal of this alternating iteration is to produce an inference chain from the document and query leading to the answer. Sordoni et al.'s method also outperformed state-of-the-art methods at the time of publication on both the CNN/DailyMail and Child Book data-sets.

With the release of SQuAD 1.1 in 2016 all new CM models have reported relative performance on this data-set. BiDAF and R-Net were two of the earliest models to focus on SQuAD as part of their research and achieve impressive results, and as such they helped established the initial state-of-the-art. The next section will discuss these models in more detail and explain how they fit into the model in this paper.

3 Approach

As previously discussed, the work in this paper focuses on Non-PCE methods for question answering and builds off of a provided BiDAF model that uses word-level embeddings. The BiDAF model was proposed by Seo et al. and improved upon the work by Cui et al. and other single-computation attention methods. Specifically, BiDAF does not summarize the context-to-question and question-to-context attentions and instead allows both attentions to flow into the modeling layer.

3.1 Characer-Level Embeddings Model

The first model this paper will explore is an improved embedding layer for the BiDAF model. In the original BiDAF paper the authors use both character-level and word-level embedddings for their embedding layer. Character embeddings are computed first by embedding each word into a character-level vector and feeding this vector into a convolutional neural network (CNN), then max pooling the result to obtain an embedding vector of uniform size for each word. Then, word-level embeddings are computed using pre-trained GloVe word vectors. Finally, character-level and word-level embeddings are concatenated and fed through a highway network to encode these concatenated embeddings. Relative to the baseline code, the character embedding step and the concatenation of character and word embeddings are the only additions to achieve this.

3.2 Self-Attention Model

The second model this paper will explore is the addition of a self-attention layer to the BiDAF model. The implementation of this self-attention layer will be based on R-Net's "Self-Matching Attention" layer [3]. The original R-Net model first uses a "Gated Attention-Based Recurrent Networks" layer that incorporates question information into passage representations (i.e. a context-to-question attention layer). The results of this layer are then fed into the self-attention layer. The baseline BiDAF model also computes a "context-to-question" attention in addition to a "question-to-context" attention. The final attention output of the BiDAF attention layer g_i is a concatenation of the context hidden state, the context-to-question attention, and the question-to-context attention:

$$g_i = [c_i; a_i; c_i \circ a_i; c_i \circ b_i]$$

Eq. 1: Final BiDAF attention. c_i is the context hidden state,
a_i is the context-to-question attention, and b_i is the question-to-context attention.

An interesting area for experimentation with the self-attention model will be formulating various possible outputs from the BiDAF attention layer and feeding them into the self-attention layer. This will be covered in more detail in the experiments section.

In the R-Net paper, the self-attention layer first computes an attention pooling vector c_t for each of n question-aware context passage word representations v_t^P. Then v_t^P and c_t are concatenated and fed through a Bi-directional Recurrent Neural Network (BiRNN) to produce a final passage representation h_t^P. The calculation of c_t is shown below.

$$s_j^t = \nu^T \tanh(W_1 v_j^P + W_2 v_t^P)$$
$$a_i^t = \frac{\exp(s_i^t)}{\sum_{j=1}^{n} \exp(s_j^t)}$$
$$c_t = \sum_{i=1}^{n} a_i^t v_i^P$$

Eq. 2: Self-Attention attention pooling vector.
W_1, W_2,and ν are weight parameters.

The result of this is a new vector representation of the context passage where each word vector representation has information from the entire context passage it is contained within. The baseline BiDAF model also feeds the attention output through a BiRNN (referred to as the "modeling layer"). To incorporate self-attention into the existing BiDAF model, this same BiRNN will be used.

The self-attention model for this project will first compute some initial attention based on the BiDAF attention. Then this initial attention output will be fed into the self-attention layer to obtain a self-attended attention. Finally the initial attention and self-attended attention will be concatenated to obtain the final attention, which will then be fed through the original modeling layer.

3.3 Type of RNN

An additional area for experimentation will be the type of RNN used in the BiDAF model (both in the encoding layer and modeling layer). Although the original BiDAF paper used a Long-Short-Term (LSTM) RNN, the R-Net paper used a Gated Recurrent Unit (GRU) RNN. This paper will experiment with both types of RNN to determine which is better suited in this hybrid BiDAF- self-attention context. This will be covered more in the experimentation section.

4 Experiments

4.1 Data

The data used is the provided Stanford Question Answering Dataset (SQuAD) with altered development and test sets (each half of the official development set). For word embeddings we use the provided pre-trained GloVE word vectors.

4.2 Evaluation Method

Quantitative evaluation of the training results is done using Tensorboard and its provided graphs displaying the EM, F1, and AvNA metrics for each model. The best training results from these models were also compared.

Qualitative evaluation involved examining example text results (i.e. question, context, answer, and predicted answer) from training. Comparison was made between the results of different models and particular attention was given to observed variation.

4.3 Experimental Details

The first experiment tested a BiDAF with character-level embeddings over 1.8 million iterations. This result was tested directly against the baseline.

Multiple experiments were run with the self-attention model by varying the output of the first attention layer. Using the same notation from eq. 1, testing was done with each of $g_a = [a_i]$, $g_{ca} = [c_i, a_i, c_i \circ a_i]$ and $g_{cab} = [c_i, a_i, c_i \circ a_i, c_i \circ b_i]$ for initial attention. Each of these experiments were run for at least 1.3 million iterations, although some were run for much longer. Although it may have taken longer in some cases to maximize output for each model, the purpose of these experiments was to determine the best overall self-attention model as quickly as possible. 1.3 iterations was deemed enough to determine whether a model would perform better or not.

Self-attention introduced significant memory overhead to the model resulting in space limitations on the virtual machines used for testing. As a result batch size could not be kept constant across experiments, with batch size decreasing as the size of the initial attention increased.

The baseline model was also tested with a GRU RNN to compare against the initial LSTM RNN. This experiment was used as evidence to for using one type of RNN over the other in final tests. This was done for the sake of time, although more sophisticated experiments with varying model types would likely need to be done to fully inform the best type of RNN to use in the context of these models.

With the best self-attention model identified, character-level embeddings and self-attention were also combined for a final experimental run over 2.5 million iterations.

4.4 Results

Table 1: Overall Test Results

Model	Dev NLL	F1	EM	AvNA
Baseline BiDAF	2.95	58.62	55.49	64.68
Self-Attention using g_{ca}	2.95	58.81	55.67	65.94
Character-Embeddings + Self-Attention using g_{ca}	2.96	60.05	56.80	66.95
Baseline BiDAF with GRU	2.93	61.25	57.72	67.97
Character-Embeddings	2.8	61.37	58.18	67.05
Character-Embeddings with GRU	**2.79**	**64.12**	**60.85**	**70.22**
Test Leaderboard Subm. (CharEmbedGRU-CD)	N/A	63.7	60.37	N/A

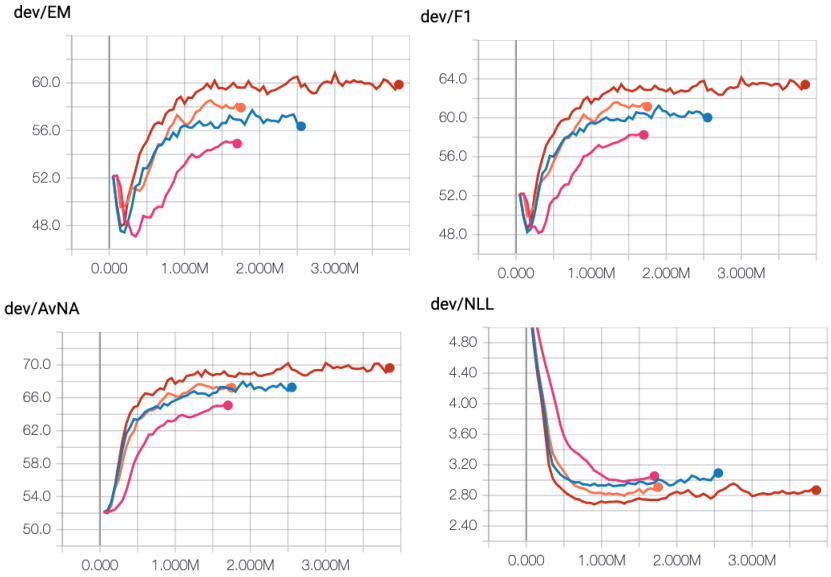

Figure 1: Training results for BiDAF with Character Embeddings (blue), GRU (orange), Character Embeddings with GRU (red) vs Baseline BiDAF (pink)

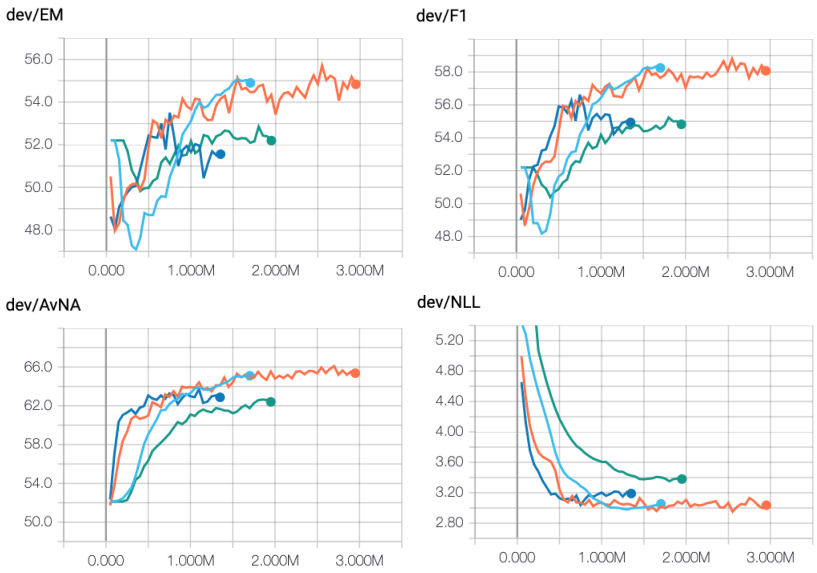

Figure 2: Training results for BiDAF with various Self-Attention models: g_a (green), g_{ca} (orange), g_{cab} (green) vs Baseline BiDAF (light blue)

5 Analysis

5.1 Quantitative Analysis

Adding character embeddings to the embedding layer of the BiDAF model immediately improved performance on all metrics (F1, EM, AvNA). This was expected given that this followed the original BiDAF embedding architecture.

Performance when adding the self-attention layer did not improve upon each metric as initially expected. Experiments demonstrated that an initial attention output $g_{ca} = [c, a, c \circ a]$ produced the best quantitative results, although this was only a slight improvement on the BiDAF baseline. Although unexpected, this result is somewhat consistent with results from the original BiDAF research. BiDAF was shown to perform comparably with R-Net on SQuAD 1.1 (only answerable questions).

It is somewhat surprising that an initial attention output of $g_{cab} = [c, a, c \circ a, c \circ b]$ did not outperform g_{ca}. One possible explanation for this is that the addition of question-to-context attention information in the initial attention adds noise to the self-attention process and prevents as much meaningful information from being extracted. In the original R-Net research each question-aware context word representation gains information about every other question-aware context word. However, when more complex concatenated attentions are put through self-attention each word representation receives information from every other word representation across the context, question-aware context, and context-aware question. It's possible this abundance of information adds noise and prevents meaningful context-based information from being extracted. Reinforcing this explanation, self-attention training loss variance increased significantly as the size of the initial attention output increased. Although other explanations are possible, the issue seems to be with how the self-attention layer is incorporated with the existing BiDAF model, and not with the self-attention layer itself. The self-attention layer itself was thoroughly tested per the R-Net specifications.

Replacing the LSTM with a GRU RNN improved performance over the baseline on all metrics. Additionally models trained with the GRU RNN converged to their best result much faster. The character-embedding model was also tested with a GRU and achieved the best overall results of any model. It has been shown that relative performance of LSTMs and GRUs depends on the specific context and dataset [1]. The results of this paper suggest that in the context of SQuAD a GRU RNN

- **Question:** What types of alternatives did CYCLADES give?
- **Context:** The CYCLADES packet switching network was a French research network designed and directed by Louis Pouzin. First demonstrated in 1973, it was developed to explore alternatives to the early ARPANET design and to support network research generally. It was the first network to make the hosts responsible for reliable delivery of data, rather than the network itself, using unreliable datagrams and associated end-to-end protocol mechanisms. Concepts of this network influenced later ARPANET architecture.
- **Answer:** N/A
- **Prediction:** support network research

- **Question:** What types of alternatives did CYCLADES give?
- **Context:** The CYCLADES packet switching network was a French research network designed and directed by Louis Pouzin. First demonstrated in 1973, it was developed to explore alternatives to the early ARPANET design and to support network research generally. It was the first network to make the hosts responsible for reliable delivery of data, rather than the network itself, using unreliable datagrams and associated end-to-end protocol mechanisms. Concepts of this network influenced later ARPANET architecture.
- **Answer:** N/A
- **Prediction:** N/A

Figure 3: Example baseline output (top) and example with character embeddings (bottom)

will perform better.

Final tests combining character-level embeddings and self-attention with a GRU RNN did not outperform the character-level embeddings GRU model. Following the discussion of self-attention results above, this result may be due to noise introduced into the final attention preventing practical context information.

5.2 Qualitative Analysis

A significant reason for the greater performance of adding character embeddings over the baseline BiDAF is an improved ability to handle no-answer questions. Observing example outputs reveals that the baseline predicts both answers on no-answer questions and "N/A" on answerable questions more often than the character-embeddings model. An example of this is shown in fig. 3. The baseline is fooled by the use of "explore alternative" in the context and its similarity to "types of alternatives" in the question. It would appear that with character embeddings the model is better able to appreciate the nuance between how "alternatives" is used in the context and question to avoid answering.

There were also differences observed between example outputs of different self-attention models. When the initial attention layer only output context-to-question attention, the model was worse at correctly predicting answers but would tend to be more similar to the ground truth when it did correctly predict an answer. Examples from g_{ca} and g_{cab} as initial attention showed some difference in ability to recognize answerable questions and which information they focused on, but in general they were fairly similar.

Comparing results between the LSTM character embeddings model and GRU character embeddings model shows that the GRU model is able to successfully answer certain questions character embeddings had previously failed at. In many cases the GRU model successfully answers questions which the LSTM had predicted "N/A." Even more interestingly, on some questions where both models fail, it is clear that the GRU model was drawing on more relevant information than the LSTM model. We see an example of this in fig. 4 with a very difficult question requiring synthesis across the entire context paragraph. Although neither model was correct, we can see that if "Marches" was the name of a country the GRU model's prediction is logically drawn from the sentence. In contrast, the LSTM model predicts a place that has no real relevance to the question. One possible explanation for this is that the LSTM is removing important information through its "forget" gate and then predicting answer that lack as much relevance.

6 Conclusion

Overall the most successful EM, F1, and AvNA results came from a BiDAF model with character-level embeddings using a GRU RNN. Although a combined self-attention and character-level embeddings

- **Question:** What country was under the control of Norman barons?
- **Context:** Subsequent to the Conquest, however, the Marches came completely under the dominance of William's most trusted Norman barons, including Bernard de Neufmarché, Roger of Montgomery in Shropshire and Hugh Lupus in Cheshire. These Normans began a long period of slow conquest during which almost all of Wales was at some point subject to Norman interference. Norman words, such as baron (barwn), first entered Welsh at that time.
- **Answer:** Wales
- **Prediction:** Cheshire

- **Question:** What country was under the control of Norman barons?
- **Context:** Subsequent to the Conquest, however, the Marches came completely under the dominance of William's most trusted Norman barons, including Bernard de Neufmarché, Roger of Montgomery in Shropshire and Hugh Lupus in Cheshire. These Normans began a long period of slow conquest during which almost all of Wales was at some point subject to Norman interference. Norman words, such as baron (barwn), first entered Welsh at that time.
- **Answer:** Wales
- **Prediction:** Marches

Figure 4: Example character-embeddings with LSTM output (top) and example character embeddings with GRU output (bottom)

model was tested, it did not improve upon the results obtained without a self-attention layer. This paper suggests the reason for this lack of improvement is likely due to how the self-attention layer has been incorporated into the BiDAF model. Alteration to the current BiDAF self-attention architecture and sufficient experimentation would likely yield overall improved results, as self-attention has been shown to improve performance on SQuAD. This paper also demonstrates that the BiDAF model performs better in this SQuAD context using a GRU RNN rather than an LSTM RNN. Although there was insufficient time to thoroughly experiment with hyper-parameter tuning, this would be a worthwhile next step for additional experimentation.

References

[1] Junyoung Chung et al. “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling”. In: *CoRR* abs/1412.3555 (2014). arXiv: 1412.3555. URL: http://arxiv.org/abs/1412.3555.

[2] Yiming Cui et al. “Attention-over-Attention Neural Networks for Reading Comprehension”. In: *CoRR* abs/1607.04423 (2016). arXiv: 1607.04423. URL: http://arxiv.org/abs/1607.04423.

[3] Natural Language Computing Group. “R-NET: Machine Reading Comprehension with Self-matching Networks”. In: (May 2017). URL: https://www.microsoft.com/en-us/research/publication/mcr/.

[4] Pranav Rajpurkar et al. “SQuAD: 100, 000+ Questions for Machine Comprehension of Text”. In: *CoRR* abs/1606.05250 (2016). arXiv: 1606.05250. URL: http://arxiv.org/abs/1606.05250.

[5] Min Joon Seo et al. “Bidirectional Attention Flow for Machine Comprehension”. In: *CoRR* abs/1611.01603 (2016). arXiv: 1611.01603. URL: http://arxiv.org/abs/1611.01603.

[6] Alessandro Sordoni, Phillip Bachman, and Yoshua Bengio. “Iterative Alternating Neural Attention for Machine Reading”. In: *CoRR* abs/1606.02245 (2016). arXiv: 1606.02245. URL: http://arxiv.org/abs/1606.02245.