SQuAD Model Exploration:

BiDAF and Input Features

Ben Barnett and Jeffrey Chen
Department of Computer Science
Stanford University
{ben.barnett, jchen623}@stanford.edu

Abstract

In this paper, we explore various deep learning techniques and approaches to implementing a reading comprehension model using the Stanford Question Answering Dataset (SQuAD), a reading comprehension dataset that includes a set of Wikipedia articles, and crowdsourced pairs of 100k+ questions and answers to those corresponding articles. Specifically, each corresponding answer is taken directly from each question, meaning each answer is an exact excerpt or "span" of the original context. Through a model inspired by "Bidirectional attention flow for machine comprehension" [1] and "Reading Wikipedia to answer open-domain questions" [2], we found that adding BiDAF to an RNN model with additional "word embedding" and "exact match" input features improved upon our original baseline model, with an F1 score of .43 and an EM score of .30.

1 Introduction

1.1 Objective

Our task, is to use various deep learning techniques to develop a model that performs well on the Stanford Question Answering Dataset (SQuAD). Machine reading comprehension and question answering is a complex, yet fundamental problem in Natural Language Processing with endless possible approaches. Our goal is to be able to take a context paragraph and question, and return an answer. For example, consider the context and question below:

Context: "during his time at his lab , tesla observed unusual signals from his receiver which he concluded may be communications from another planet . he mentioned them in a letter to reporter julian hawthorne at the philadelphia north american on 8 december 1899 and in a december 1900 letter about possible discoveries in the new century to the red cross society where he referred to messages " from another world " that read " 1 ... 2 ... 3 ... " . reporters treated it as a sensational story and jumped to the conclusion tesla was hearing signals from mars . he expanded on the signals he heard in a 9 february 1901 collier's weekly article " talking with planets " where he said it had not been immediately apparent to him that he was hearing " intelligently controlled signals " and that the signals could come from mars , venus , or other planets . it has been hypothesized that he may have intercepted marconi 's european experiments in july 1899—marconi may have transmitted the letter s ... in a naval demonstration , the same three impulses that tesla hinted at hearing in colorado—or signals from another experimenter in wireless transmission."

Question: "To what did tesla attribute the unknown signals his radio received ?"

Given this context and question, our model should return the answer "communications from another planet." Notice that this answer is a "span" or direct excerpt from the given passage.

1.2 Performance Metrics

The passage above is a direct data point from SQuAD, and the performance on the dataset is measured with 2 metrics: F1 and Exact Match (EM) scores. F1 is defined as 2×prediction×recall/(precision+recall). EM, or "exact match" on the other hand is a binary measurement of whether the predicted value matches the true answer exactly. For reference, human performance on this dataset is an F1 score of .86 [3]. Without sophisticated knowledge of linguistics and semantics, we attempt to use deep neural networks to perform as well as possible on SQuAD.

1.2 Related Works

There has been an extensive amount of research regarding SQuAD and an even greater concentration on Machine Reading Comprehension as a whole. However there are a few notable papers in which we gathered inspiration for our own improvements. "Bidirectional attention flow for machine comprehension" [1] describes the appropriate architecture for implementing BiDAF. This paper also describes many other improvements to be made to the baseline model like Logistic Regression, Co-attention, and Fine-Grained Gating, but we believed that BiDAF was the primary modification that resulted in their model's high performance. We also added additional input features with the inspiration of "Reading Wikipedia to answer open-domain questions" [2]. This paper describes significant improvements from adding a few other additional features to the word embeddings. They found that adding these features below specifically improved F1 scores most significantly.

1. Representing if each context token appeared inside the question
2. Adding Part-of-Speech tags and Named Entity types to each context
3. Using the word embedding for each context to attend to the word embeddings for the question

2 Approach

The general architecture of the final model is the combination of several modifications to the baseline model provided by the CS224N course staff. The following sections outline the architecture of the baseline model and the individual modifications to the baseline model explored in this project.

2.1 Baseline Model

The baseline model consists of three components: a RNN encoder layer that encodes contexts and questions into hidden states, an attention layer that combines context and question hidden states into a single representation, and an output layer, which outputs the starting and end location of the predicted answer span within the context.

2.1.1 RNN Encoder Layer

Every context and question word is mapped to its corresponding d-dimensional, pre-trained GLoVe embedding. These fixed embeddings are then fed into a 1-layer bidirectional GRU, which produces a sequence of h-dimensional forward hidden states and h-dimensional backward hidden states for both contexts and questions:

$$\{\overrightarrow{\boldsymbol{c}_1}, \overleftarrow{\boldsymbol{c}_1}, \ldots, \overrightarrow{\boldsymbol{c}_N}, \overleftarrow{\boldsymbol{c}_N}\} = \text{biGRU}(\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\})$$
$$\{\overrightarrow{\boldsymbol{q}_1}, \overleftarrow{\boldsymbol{q}_1}, \ldots, \overrightarrow{\boldsymbol{q}_M}, \overleftarrow{\boldsymbol{q}_M}\} = \text{biGRU}(\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M\})$$

Figure 1: Output of Bidirectional GRU within RNN Encoding Layer

Then, the forward and backward hidden states are concatenated, which comprises the output hidden states for each corresponding question and context word from the encoding layer:

$$\boldsymbol{c}_i = [\overrightarrow{\boldsymbol{c}_i}; \overleftarrow{\boldsymbol{c}_i}] \in \mathbb{R}^{2h} \quad \forall i \in \{1, \ldots, N\}$$
$$\boldsymbol{q}_j = [\overrightarrow{\boldsymbol{q}_j}; \overleftarrow{\boldsymbol{q}_j}] \in \mathbb{R}^{2h} \quad \forall j \in \{1, \ldots, M\}$$

Figure 2: Final Hidden Representations Output from RNN Encoder Layer

2.1.2 Attention Layer

The baseline attention layer consists of a basic dot-product attention in which context hidden states attend to the question hidden states. This layer outputs an N long sequence of blended representations wherein each representation corresponds to one context word and N is the context length.

2.1.3 Output Layer

Each blended representation, b_i, is fed through a fully connected layer followed by a ReLU activation function, which is then passed through a linear layer that computes the starting score of each context word:

$$\boldsymbol{b}'_i = \mathrm{ReLU}(\boldsymbol{W}_{FC}\boldsymbol{b}_i + \boldsymbol{v}_{FC}) \in \mathbb{R}^h \quad \forall i \in \{1, \ldots, N\}$$
$$\mathrm{logits}_i^{\mathrm{start}} = \boldsymbol{w}_{\mathrm{start}}^T \boldsymbol{b}'_i + u_{\mathrm{start}} \in \mathbb{R} \quad \forall i \in \{1, \ldots, N\}$$

Figure 3: Computation of Context Blended Representations

To obtain the probability distribution of start words, $\mathrm{logits}_i^{\mathrm{start}}$ is passed through a final softmax layer. Computing the predicted end word is a parallel process, with the only difference being the weight and bias terms w_{end} and u_{end} replacing w_{start} and u_{start}.

2.1.4 Loss and Predictions

The loss function for a single example is computed as the sum of the cross-entropy loss for start and end locations, using i_{start} and i_{end} for the true starting and ending word locations:

$$\mathrm{loss} = -\log p^{\mathrm{start}}(i_{\mathrm{start}}) - \log p^{\mathrm{end}}(i_{\mathrm{end}})$$

Figure 4: Loss Calculation for a Single Context

During training, loss is minimized across the average of each batch, using an Adam optimizer.

Note that during testing, rather than making start and end word predictions based on their respective probability distributions, the start and end words with the highest respective scores are output as predictions.

2.2 Modification 1: Bidirectional Attention Flow

Bidirectional Attention Flow ("BiDAF") is a modification to the baseline model that improves the attention layer by allowing attention to flow in both directions. In the baseline model, the context attends to the question. However, in BiDAF, the question attends to the context as well, resulting in a higher dimensional blended representation of each context word whose features are hopefully more informative for the output layer of the model. The C2Q component is computed similarly to the baseline model.

All equations and formulas for computation can be found on the final project handout, though the key part of BiDAF is the altered blended representation of each context word, which contains twice as many features at the end of the layer (for a total of $8h$-dimensionality). The blended representation for a single word is as follows:

$$\boldsymbol{b}_i = [\boldsymbol{c}_i; \boldsymbol{a}_i; \boldsymbol{c}_i \circ \boldsymbol{a}_i; \boldsymbol{c}_i \circ \boldsymbol{c}'] \in \mathbb{R}^{8h} \quad \forall i \in \{1, \ldots, N\}$$

Figure 5: Final Blended Representation from BiDAF Attention Layer

wherein c_i is the context word's embedding and a_i is the C2Q attention output—a metric of the context word to all question words. c' is the Q2C attention output that is new to the model from the previous baseline implementation which used only C2Q, and is computed as follows:

$$\boldsymbol{S}_{ij} = \boldsymbol{w}_{\text{sim}}^T [\boldsymbol{c}_i; \boldsymbol{q}_j; \boldsymbol{c}_i \circ \boldsymbol{q}_j] \in \mathbb{R}$$

$$\boldsymbol{m}_i = \max_j \boldsymbol{S}_{ij} \in \mathbb{R} \quad \forall i \in \{1, \ldots, N\}$$

$$\beta = \text{softmax}(\boldsymbol{m}) \in \mathbb{R}^N$$

$$\boldsymbol{c}' = \sum_{i=1}^{N} \beta_i \boldsymbol{c}_i \in \mathbb{R}^{2h}$$

Figure 6: Calculation of c', the Q2C Attention Output during BiDAF

In the equations above, S_{ij} is the similarity score between c_i and q_j. In order to compute similarities efficiently, a single similarity matrix $S \in R^{N \times M}$ is computed once for every context, question pair in the batch, whose entries represent the similarity between each combination of word pairs within the context and question.

2.3 Modification 2: Additional Input Features

In our final model, we concatenated additional input features to the output produced in the baseline encoding layer. Two features were appended to each context word embedding:

1. A positive integer metric, representing whether the context word appears in the question.
2. A scalar representing the minimum Euclidean distance from the context word's GLoVe embedding and all question word embeddings.

More formally, given a context word's GLoVe embedding, $c\text{-}emb_i \in R^H$ (the GLoVe embedding hidden size), the context word's final embedding output from the model's encoding layer *before passing through the bidirectional GRU* is as follows:

$[c\text{-}emb_i; a; b] \in R^{H+2}$

where a is the number of times that the context word appears in the question (most often just a 0 or 1) and

$b = min \; || \; c\text{-}emb_i - q\text{-}emb_j \; || \; j \in \{1, \ldots, M\}$.

The intuition behind these features is that often when trying to find the answer to a question in a passage, words in the passage that appear in the question are likely to be near or part of the answer. This is also part of why basic dot-product attention is so effective, as similarities between context and question words are often a strong indicator of where the answer lies.

Incorporating this technique/intuition into the encoding layer seemed like a good way to directly 'encode' the importance of similarity between a context word and its existence within the corresponding question.

3 Experiments

3.1 Dataset

SQuAD is a crowdsourced reading comprehension dataset containing context, question, answer triplets from Wikipedia entries [3]. The dataset used includes about 100,000 questions in total, though note that there are fewer distinct contexts, as multiple questions and answers may come from the same context. More information about SQuAD can be found at https://rajpurkar.github.io/SQuAD-explorer/ explore/1.1/dev/.

3.2 Evaluation Metrics

The primary performance metrics used were F1 and EM scores, whose definitions can be found in the introduction section.

3.3 Results

Table 1: Baseline

Model configuration	Default baseline model as described above
Learning rate	0.001
Batch size	32
Num iterations reached	40k
Training time	15 hrs.

Table 2: Baseline + BiDAF

Model configuration	Default baseline model as described above with the exception of the model's attention layer, which was substituted with a BiDAF implementation
Learning rate	0.001
Batch size	32
Num iterations reached	12k
Training time	28 hrs.

Table 3: Baseline + BiDAF + additional input features

Model configuration	Default baseline model with modified RNN encoder layer to include additional input features as well as substitution of the default attention layer for a BiDAF attention layer
Learning rate	0.001
Batch size	32
Num iterations reached	15k
Training time	40 hrs.

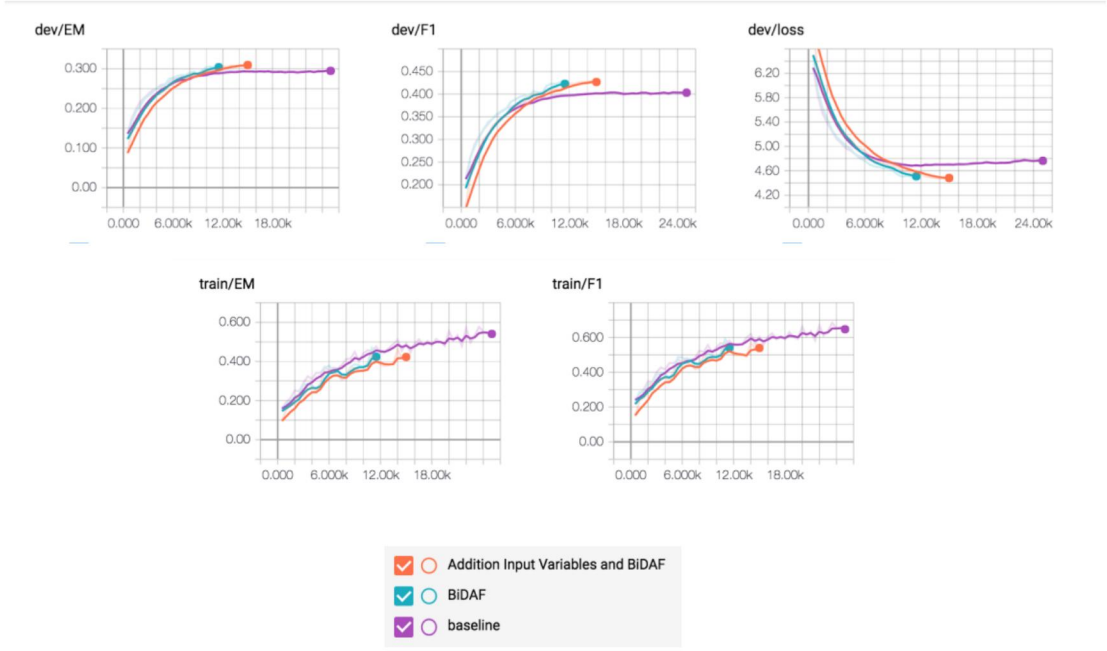

Figure 7: Various model results overlaid over iterations run

From Figure 7 above, we can see that BiDAF with our additional input features achieves the highest F1 score, plateauing at .43. This model also performs the best in terms of with a score of .31. By analyzing individual example answer spans that our model predicted, we were able to identify specific strengths and weaknesses of our implementation through the following examples.

EXAMPLE 1

CONTEXT: a piece of paper was later found on which Luther had written his last statement. The statement was in Latin , apart from " we are beggars , " which was in German.

QUESTION: what was later discovered written by luther ?

TRUE ANSWER: his last statement

PREDICTED ANSWER:

F1 SCORE ANSWER: 0.000

EM SCORE: False

EXAMPLE 2

CONTEXT: after leaving edison 's company tesla partnered with two businessmen in 1886 , robert lane and benjamin vail , who agreed to finance an electric lighting company in tesla 's name , tesla electric light & manufacturing . the company installed electrical arc light based illumination systems designed by tesla and also had designs for dynamo electric machine commutators , the first patents issued to tesla in the us .

QUESTION: what was produced at tesla 's company ?

TRUE ANSWER: dynamo electric machine commutators

PREDICTED ANSWER: electrical arc light based illumination systems

F1 SCORE ANSWER: 0.000

EM SCORE: False

EXAMPLE 3

CONTEXT: the ipcc receives funding through the ipcc trust fund , established in 1989 by the united nations environment programme (unep) and the world meteorological organization (wmo) , costs of the secretary and of housing the secretariat are provided by the wmo , while unep meets the cost of the depute secretary . annual cash contributions to the trust fund are made by the wmo , by unep , and by ipcc members ; the scale of payments is determined by the ipcc panel , which is also responsible for considering and adopting by consensus the annual budget . the organisation is required to comply with the financial regulations and rules of the wmo .

QUESTION: who funds the ipcc 's deputy secretary ?

TRUE ANSWER: united nations environment programme

PREDICTED ANSWER: united nations environment programme (unep) and the world meteorological organization (wmo) , costs of the secretary and of housing the secretariat are provided by the wmo

F1 SCORE ANSWER: 0.320

EM SCORE: False

EXAMPLE 4

CONTEXT: in june 1978 , arledge created the newsmagazine 20/20 ; after its first episode received harshly negative reviews , the program – which debuted as a summer series , before becoming a year-round program in 1979 – was immediately revamped to feature a mix of in-depth stories and interviews , with hugh downs appointed as its anchor (later paired alongside his former today colleague barbara walters) . in february 1979 , abc sold its recording division to mca inc. for $ 20 million ; the label was discontinued by march 5 of that year , and all of its 300 employees were laid off (the rights to the works of abc records and all of mca 's other labels have since been acquired by universal music group) .

QUESTION: when was the newsmagazine 20/20 first created ?

TRUE ANSWER: june 1978

PREDICTED ANSWER: june 1978

F1 SCORE ANSWER: 1.000

EM SCORE: True

From example 1, you can see that our model fails to make any prediction whatsoever. This is because the span start-index was predicted to be after the end-index. Furthermore, in example 3, we can see that the correct answer is only 4 words long, whereas our model predicted the answer to have the gold truth starting index, but to span 25 words. Although this answer is technically correct (in that it contains the correct answer), it received a low F1 score because it was much longer than necessary. From these examples, among others, we determined that regulating span would be an important modification to make in the future. Specifically, ensuring that span start indices were never predicted to be after end indices and that span lengths are always under a predetermined size would not only improve accuracy and F1 scores, but also the efficiency of our model.

Example 2 highlights another shortcoming of our model. The question asks “what was produced at tesla 's company?” The true answer is “dynamo electric machine commutators”, but our model instead predicts the answer to be “electrical arc light based illumination systems.” We attribute this error to the additional input feature which adds a similarity metric between GloVe contexts and questions to each word embedding. This input feature directly affects our model in this example because the question specifically asks “what was produced…?” and our model likely uses the fact that “designed” has a closer GloVe embedding to “produced” than “installed.”

However, this same feature seems to help our model perform better on questions inquiring about numbers and dates. In example 4, our model scores a perfect F1 of 1.00. This is likely because numbers exist in a very specific space within the entire GloVe embedding space, which results in greater certainty in our model when searching for numbers and context tokens similar to numbers. This ultimately improves the model's ability to find the correct answer span when prompted for an answer to contain dates or numbers.

5 Conclusion

Bidirectional attention flow and additional input features both resulted in improvements from the baseline model. However, in analyzing the results of example sentences (and the model's predictions), we noticed that there still seems to be a large opportunity for "easy" model improvement by training the model to more heavily consider the answer span in the output layer of the model before simply predicting start and end words independently. For example, the final model often outputs no answer at all because its predicted end word is prior to the predicted start word; preventing this alone by requiring the start word to be before the end word may alone cause a large model improvement without any significant increase in the computational expense of training. Further, it's possible that improvements would be visible if the model were to consider joint probabilities between the start and end word as opposed to calculating the most likely start and end words independently.

One other broad takeaway outside of the primary objective of investigating the best possible SQuAD model is the importance of the model's training efficiency; although our final model outperformed the baseline model in all evaluation metrics, it took roughly 7 times slower to train, making it expensive to tune hyperparameters. In the future, given a perhaps even larger dataset, it would be vital to find ways to improve the efficiency of the model, especially the implementation of BiDAF.

6 References

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

[2] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. arXiv preprint arXiv:1704.00051, 2017.

[3] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100, 000+ questions for machine comprehension of text. CoRR, abs/1606.05250, 2016.