Question Answering on the SQuAD Dataset

TG Sido
osido@stanford.edu | Computer Science, Stanford University

Background

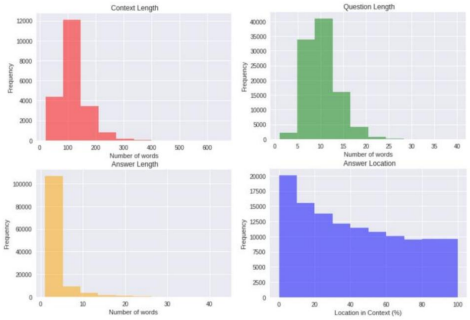
Stanford **Qu**estion **A**nswering **D**ataset (SQuAD) is a reading comprehension dataset comprised of questions from Wikipedia articles and answers that are continuous spans of text within those articles. **Q**uestion **A**nswering (QA) on SQuAD is a difficult task; nevertheless, it has important applications for chatbots and virtual assistants.

Goal

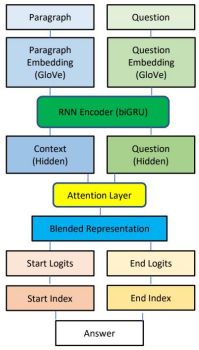
Leverage complex attention mechanisms and ensemble approaches to find the start and end indices of the answer within the text.

Dataset

107,785 examples from 23,215 paragraphs in the following format:
P: The South African Schools Act of 1996 recognizes two categories of schools: "public" (state-controlled) and "independent"…
Q: What South African law recognized two types of schools?
A: South African Schools Act
Q: In what year was the South African Schools Act passed?
A: 1996
Q: Along with public schools, what type of school was recognized under the South African Schools Act?
A: independent
Legend: **P** (paragraph); **Q** (question); **A** (answer)



Solution Architecture



Attention Mechanisms

Given N context hidden states (C) and M question hidden states (Q), our baseline uses basic dot-product attention from context to question (C2Q). However, we can achieve better results when we allow attention to flow from question to context (Q2C) as well.

Bi-Directional Attention Flow (BiDAF)

- Create similarity matrix S (NxM)
- C2Q: get a (weighted sum of question states)
- Q2C: get c′ (weighted sum of context states)
- *Attention layer output*: stacked combination of a and c′

Dynamic Coattention Network (DCN)

- Create affinity matrix L ((N+1)x(M + 1))
- C2Q: get a (weighted sum of question states)
- Q2C: get b (weighted sum of context states)
- 2nd level attention: get s (weighted sum of b states)
- *Attention layer output*: biLSTM encoding of stacked combination of s and a

Smart Span Selection

Since the same blended distribution is used to determine the start and end indices and span lengths rarely exceed 15 characters (see answer length histogram), we use a bit mask to enforce: startIndex ≤ endIndex ≤ startIndex + 15

Results

Model	Train F1	Train EM	Dev F1	Dev EM
Baseline	61.1	49.2	39.4	29.1
BiDAF (dropout = .15, context_len = 600)	54.4	42.6	44.1	32.2
BiDAF (dropout = .20, context_len = 300)	66.3	54.6	45.2	33.1
DCN (dropout = .15, context_len = 600)	72.9	58.6	57.9	43.5
DCN (dropout = .20, context_len = 300)	72.5	58.8	58.1	43.3
BiDAF + DCN Ensemble (dropout = .20, context_len = 300)	73.3	60.1	58.3	43.2
PAML + BERT (Ensemble) (state-of-the-art)	**n/a**	**n/a**	**85.9**	**83.4**

Conclusions

- Regularization – due to increasing dropout probability or early stopping – can sometimes provide a small boost in performance
- Since most paragraphs (contexts) are less than 300 words in length, using 300 instead of 600 for the *context_len* hyperparameter provides tremendous speed improvements in training and testing
- DCN performs better than BiDAF due to the following:
  - trainable sentinel states that are added to both the context and question, utilizing the growing knowledge base
  - biLSTM encodes more query-aware information into the attention layer

Future Work

- Implement smart span selection with an LSTM to condition probability of end index on the start index
- Use character-level CNN to augment the GloVe word embeddings, mitigating against out-of-vocabulary words
- Experiment with averaging, adding, or max-pooling the forward and hidden states from the RNN Encoder

Acknowledgements: Pedro Garzon, Steven Chen