Understanding Attention for Question Answering models

Akshatha Kamath[1]
[1]Stanford University

Problem

Question Answering involves a short paragraph (context) and a question (query) with a goal to output the location of the answer within the given text. Since PCE based methods require a large amount of computational time and costs, we explore and compare the performances of non Pre-trained Contextual Embeddings (PCE) based approaches - Bidirectional Attention Flow (BiDAF), Dynamic Coattention Network (DCN), and FusionNet and also consider ensemble based approaches of FusionNet to get an idea of attention focused models' effectiveness on the SQuAD-2[2] dataset.

Background

Dynamic Coattention Networks

Xiong and Zhong [3] proposed a "coattention model" to give attention to both the question and the context at the same time.

- Built a codependent representation of them simultaneously which they use to predict the starting and ending point of the answer within the context.
- Introduced a dynamic point decoder, which iteratively moves through the context to determine which start and endpoints in the context satisfy the question best

FusionNet[1]

- Non-PCE reading comprehension model.
- Simple model encoding with RNN features like pre-trained word vectors, term frequencies, part-of-speech tags, name entity relations, and whether a context word is in the question or not.
- PointNet is used to learn a global, vectorized representation of the query sentence, followed by a convolution over the context word embeddings to learn a representation of each word within its local context

Coattention module

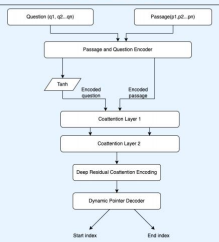

Figure 1. Dynamic Coattention network module

FusionNet module

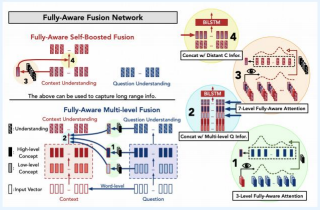

Figure 2. FusionNet architecture (taken from the original paper)

Experiments

Our experiments mainly just consisted of training our model on the Squad 2 dataset and seeing how it performed on unseen dev sets of data; we trained 3 different models and saw their performance on the F1, EM and AvNA metrics provided by Tensorboard.

Coattention

- Converted the character-indexed answer spans from SQuAD2.0 to token indexing.
- We used the Adam optimizer with learning rate 0.01. The weights and biases and hidden states were initialized to zero whereas the encoding sentinels were randomly initialized. The start and end indices were initialized to the beginning of the context. We used LSTM hidden layers of size 200, BiLSTMs thus produce hidden states of dimension 400. We used a maxpool size of 16.
- We stopped training after 100000 steps since the model's F1 score no longer increased on the dev set after that point.

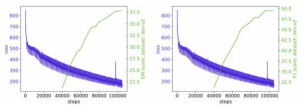

Figure 3. The training loss, F1 and EM scores of our coattention model.

FusionNet

- Since the original paper is evaluated on Squad 1, we prepend a OOV (Out of Vocabulary) token to the beginning of each context when the question is unanswerable.
- Learning rate 0.003, adamax optimizer, Exponential moving average decay rate 0.999, maximum gradient norm for gradient clipping 5.0, and a dropout probability 0.35.
- **FusionNet Ensemble**: We use 6 models from 6 runs with slightly different hyperparameters (36 models in total), giving us a dev F1 of 67.735 and EM score of 65.23.

Analysis

Model evaluated on DEV	Dev NLL	F1	EM	AvNA
BiDAF baseline	02.98	61.81	58.31	68.56
Coattention	-	49.67	38.39	51.34
FusionNet single	06.21	64.49	60.56	72.00
Fusion Net ensemble	05.81	64.38	61.79	74.13

Both, the FusionNet and the FusionNet ensemble methods perform better than our baseline. However, the coattention model performs far worse than the baseline.

Model robustness upon Adversial Attack

We change the dev set by substituting two words in a sentence and by replacing words by random words to explore how our model behaves. We see a nearly clear linear correlation between the number of values changed and the drop in performance. While there is a performance drop, the linear relations also show some level of robustness to changes in the input sentences.

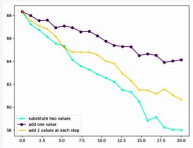

Figure 4. Robustness of the FusionNet single model to adversarial attacks of changing words in the sentences)

Conclusions

We tried to replicate the coattention network with a similar behaviour as shown in the paper, but resulted in significantly decreased performance. The FusionNet single and ensemble modules boost the performance on the question-answering tasks and show some level of robustness to adversarial attacks. We consider the following as possibilities for future work:

- **Lower training time**: We wish to speed the implementation of the DCN so it takes lesser time to execute.
- **Combination of different types of attention**: Developing a model structure combining the different types of attention layers either internally in the model or externally.
- **Initialisation techniques**: We could also explore the effect of alternative initialisation schemes, such as Xavier initialisation for our LSTMs [?].
- **Training alterations to improve robustness to adversarial attacks**: Perhaps the next version of Squad might include context with misspelled words or grammatically incorrect sentences to encourage models that promote robustness.

References

[1] Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. *arXiv preprint arXiv:1711.07341*, 2017.

[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] Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016.

kamathak@stanford.edu