Answer Pointer Alternatives for Unanswerable Questions in SQuAD2.0

Evin Yang
evin@stanford.edu

Problem/Task

SQuAD is a reading comprehension task where questions are answered by a segment of text in a passage. The most common approach is to train a model to point to the start and end indices of the answer span within the passage. This pointer method was first introduced with Pointer Net in 2015 [5] and applied to machine comprehension using match-LSTM architectures in 2016 [4]; answer pointer remains the prevailing approach for tackling span prediction. With the introduction of unanswerable questions in SQuAD2.0, the answer pointer method may be adapted to point to a zero-length span at the start of the input to indicate no answer, and we wondered whether this properly addresses the answerability problem. We investigate various alternative techniques in an attempt to beat the traditional answer pointer method.

Data

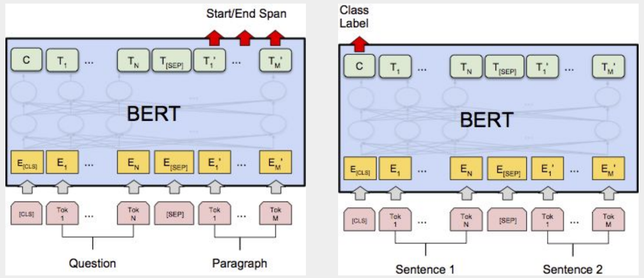
Train: Official SQuAD2.0 train set (130k examples)
Dev: Random half of official SQuAD2.0 dev set (6k)
Test: Other half of official SQuAD2.0 dev set (6k)

Approach

BiDAF and BERT answer pointer: Standard baselines to benchmark performance against

BERT pre-classifier and filtered model (MultiBERT): A binary classifier first determines whether or not the question is answerable based on the given passage. If it is, a separate BERT model trained only on answerable questions finds the answer span for the question within the passage.

Heat map span prediction: Three-class token level classification. For each token, determine: is this to the left, within, or to the right of the answer span and compute probabilities. Check all valid N^2 start/end index pairs and return maximum likelihood span.



Results

Table 1: Dev set results

Method or experiment	Batch size	Epochs	Eval	EM	F1
BiDAF baseline	64	30	All	56.91	60.43
BERT answer pointer	12	2	All	71.74	74.60
BERT pre-classifier	12	2	NoAns	55.59	55.59
Filtered BERT answer pointer	12	2	HasAns	80.69	88.42
MultiBERT	12	2	All	71.77	74.71
BERT heat map	24	1	All	47.84	60.11
BERT answer pointer	4	2	All	0.10	10.24
Freeze BERT, fine-tune QA layer	32	3	All	52.16	52.17
No answer everything	-	-	All	52.19	52.19

Analysis

With the classifier, we rely on BERT to instead encode the answerability classification within just a single vector of size H = 768 corresponding to the [CLS] token, as opposed to a vector for every token, up to 384 input tokens. It's possible that by "reducing" the problem to classification, we may have inadvertently introduced the very curse of dimensionality that attention mechanisms and non-recurrent Transformer architectures were designed to solve in the first place.
The filtered model performance is effectively equal to the original BERT. This tells us that including unanswerable questions and empty spans in SQuAD does not distract much from answerable questions and non-null spans—answerability classification and span prediction are similar, likely overlapping tasks.
In hindsight, there's a fairly apparent reason why heat map didn't work: by classifying every token in the input, we introduced less important or even irrelevant targets to minimize loss on, and treated them as equally important to the answer token classifications. Tokens far away from the ground truth span don't have much impact on the answer span itself, but tokens close to the answer span do, and this emphasis is not reflected in the loss function, resulting in very poor training.

Conclusion

Our inability to significantly outperform answer pointer highlights its architectural strength. Error analysis surfaced a couple insights for extending these methods or how to explore new approaches. Due to the nature of the task and the complexity of question answerability, binary classification is likely a dead end. However, filtered training and data segmentation strategies could prove useful for harder NLP tasks where noisy examples do actually "confuse" the model. Data/label augmentation, where additional meaningful labels are attached to examples and included in loss during training, could also help with performance. For heat map prediction, one possible solution is to localize and weight loss targets such that the closer a token is to the answer span, the higher priority given to minimizing its classification error. Reducing the three classes to two classes (inside/outside) could also support multiple-span prediction for difficult or lengthy passages where the answer appears multiple times (which is rare in the current version of SQuAD).

References

[1]Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of DeepBidirectional Transformers for Language Understanding.arXiv e-prints, art. arXiv:1810.04805,Oct 2018.
[2]Pranav Rajpurkar, Robin Jia, and Percy Liang. Know What You Don't Know: UnanswerableQuestions for SQuAD.arXiv e-prints, art. arXiv:1806.03822, Jun 2018.
[3]Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. BidirectionalAttention Flow for Machine Comprehension.arXiv e-prints, art. arXiv:1611.01603, Nov 2016.
[4]Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez,Lukasz Kaiser, and Illia Polosukhin.Attention Is All You Need.arXiv e-prints, art.arXiv:1706.03762, Jun 2017.
[5]Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer Networks.arXiv e-prints, art.arXiv:1506.03134, Jun 2015.
[6]Shuohang Wang and Jing Jiang. Machine Comprehension Using Match-LSTM and AnswerPointer.arXiv e-prints, art. arXiv:1608.07905, Aug 2016