AlterNet: Improving Span Conditioning for Q&A Systems

Simon Camacho, Eva Batelaan, and Ben Korngiebel

Department of Computer Science
Stanford University

Abstract

Recent improvements in Q&A have seen a progression from using RNNs to CNNs due to improved training and inference speeds. The QANet model, introduced in Yu et al. (2018) [1], combines CNNs with self-attention, first seen in Vaswani et al. (2017) [2]. We build upon the BiDAF model described in Seo et al. (2016) to create our own implementation of the QANet model, achieving a single-model dev F1 score of 65.47, 4.48 points higher than the baseline BiDAF model [3]. We complement the QANet model with our own extension on the conditional output layer described in Kim and Wolff [4]. We achieve an ensemble dev F1 score of 67.08. Our ensembled model achieves a test F1 score of 63.33.

Introduction

- Early Q&A models relied on sequential end-to-end structure; however, more recent models propose more parallelizable structures
- We create our own implementation of the QANet model. Our implementation achieves a similar performance score (61.03 F1) within an hour of training while it took the BiDAF baseline 2.5 hours to achieve 60.99 F1
- We extend our implementation of QANet by implementing the conditional output layer described in Kim and Wolff [4] and then create our own conditional output layer
- We further experiment with different novel changes on top of our baseline QANet model, including data augmentation, different model ensembling methods, and changing model sizes

Method

Baselines
- BiDAF [3], BiDAF + character embeddings, QANet [1]

Dataset
SQuAD 2.0

Evaluation Metrics
F1, EM, Training Time

Improving QANet

Data Augmentation
- Apply data augmentation by separately backtranslating context and answer from (context, question, answer) triple
- Include backtranslated question/answer pair if new answer appears in new context

Cross-Conditional Output Layers
- Based on Kim and Wolff [4], condition end probabilities on start probabilities and condition start probabilities on end probabilities (see Figure 1 for diagram of output layer)

Ensembling
- Implement segment and token max ensembling, where possible answers are maxed over their entire span vs. individual words

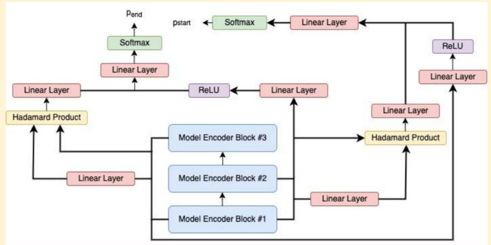

Figure 1: True forward-backward output layer diagram

Results

Model	F1	EM	AvNA	Hidden Size	Training Time
BiDAF	60.99	57.79	67.5	100	2.5 hrs
BiDAF w/ Character Embed	64.12	60.85	70.48	100	2.75 hrs
QANet	**65.47**	**61.55**	72.51	128	2.75 hrs
QANet (Double) w/ Data Augmentation v1	64.93	61.1	72.44	256	4.3 hrs
QANet (Double) w/ Data Augmentation v2 (early stop)	58.81	55.22	66.48	256	1.75 hrs
QANet+	63.37	59.72	70.91	128	1.85 hrs
QANet w/ Avg. Forward-Backward Cond.	62.55	58.78	70.43	128	2.8 hrs
QANet w/ True Forward-Backward Cond.	64.51	60.8	71.8	128	2.6 hrs
Segment Max Ensemble	**67.078**	**63.821**	N/A	128	N/A
Token Max Ensemble	66.761	63.099	N/A	128	N/A
Test Leaderboard (Seg. Max Ensemble)	**63.332**	**60.034**	N/A	128	N/A

Table 1: F1/EM scores from baseline, improved, and ensemble models (dev scores unless otherwise stated).

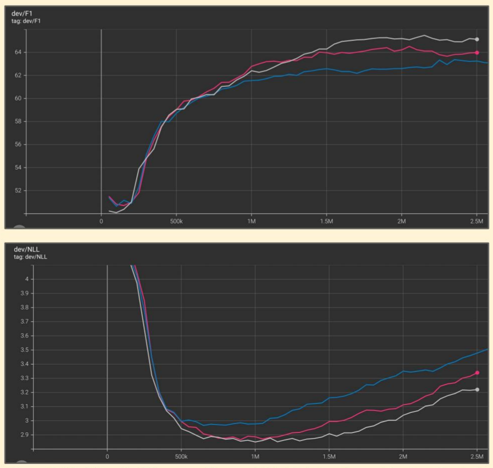

Figure 2: (Top) Dev Set F1 scores of top model (white), true forward-backward output model (pink), and Kim and Wolff [4] output model (blue); (bottom) Dev Set NLL

Discussion

Data Augmentation
- Improved performance for larger (i.e. x2 hidden size) models
- Decreased performance for regular models
- Poorly backtranslated answers introduce incorrect answer spans in the context which can result in poor performance

Forward-Backward Output Layer
- Improved performance over our implementation of Kim and Wolff [4] by using conditional probabilities for start and end
- Achieved lower overall performance than best model but improvement over [4] indicates their might be reason to continue exploring bi-directional conditionalities for the output layer

Ensembling
- Four models (QANet, QANet+, QANet Avg., QANet True)
- We saw overall improvement of 1.608 F1 from the baseline QANet model through segment max ensembling
- Both ensembling techniques leverage the individual strengths of each model, hence their improved performance
- Segment max demonstrates improved performance over token max as it maxes over entire existing answers whereas token max could lead it to potentially create an unseen result

Regularization and Layernorms
- Attempted various regularization techniques, such as dropout, layer dropout, non-linear activations, L2 weight decay
- Overfitting was still an issue, and occasionally became worse when some of these techniques were employed. Use of stochastic layer dropout meant that later layers (self-attention layer) would be dropped out more frequently than earlier layers (CNN layers), leading to loss of global interaction information and overfitting

References

[1] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi,and Quoc V. Le. QANet: Combining local convolution with global self-attention for reading comprehension. In International Conference on Learning Representations (ICLR), 2018.
[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In CoRR, abs/1706.03762, 2017.
[3] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hananneh Hajishirzi. Bidirectional attention flow for machine comprehension. In International Conference on Learning Representations (ICLR), 2016.
[4] Moo Kim and Christopher Wolff. Qanet+: Improving qanet for question answering. In CS224N Default Final Project, 2020.
[5]Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable Questions for SQuAD. In arXiv preprint arXiv:1806.03822, 2018.
[6] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. 2017.
[7] Jeffrey Pennington, Richard Socher, and Chris Manning. https://nlp.stanford.edu/pubs/
glove.pdf. 2014.
[8] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In CoRR, abs/1606.05250, 2016.