Diverse Ensembling with Bert and its variations for Question Answering on SQuAD 2.0

Stanford University

CS 224N Final Project | Winter 2019

Weiquan Mao Zhen Qin Zhining Zhu

Problem

The goal is to answer questions correctly given paragraph context from SQuAD 2.0. The target answer would be the span of text or N/A if there is no answer in paragraph. We use BiDAF as baseline, BERT-based architecture as the core, L1 regularization and other architecture changes on BERT. Ensembling method is also applied for improvement, which combines multiple models into a more robust Question Answering system by several different ensemble mechanisms

Dataset/Task

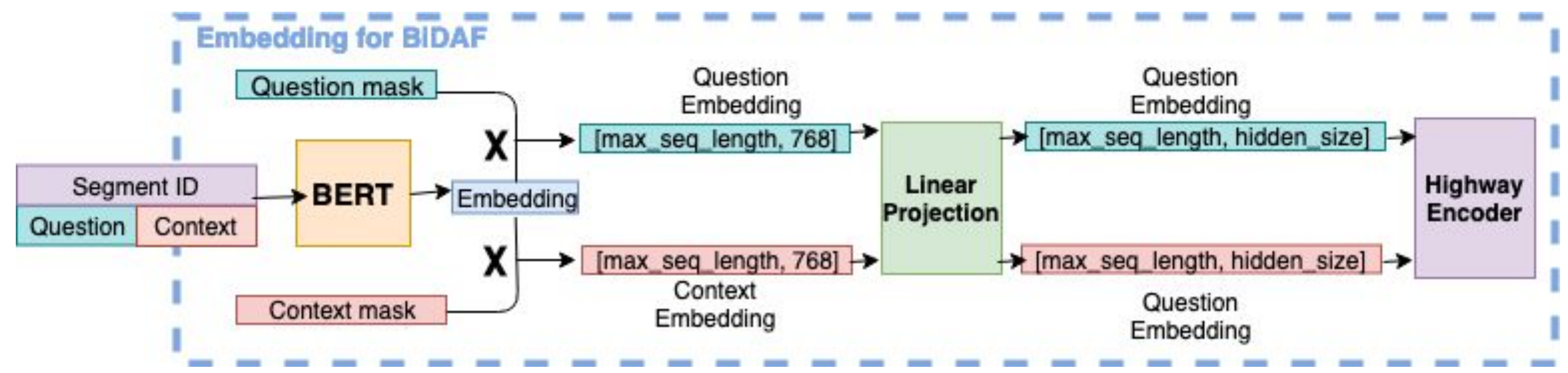
We use SQuAD 2.0 as the reading comprehension data set. Every answerable SQuAD question has three answers provided.
Dataset has been split into: 129941 training examples, 6078 dev examples, 5291 test examples

Conclusions

After training 26 BiDAF-based, BERT-based models, and ensemble them with two algorithms, we push test F1 score to 78.841 and Test EM to 76.010. In conclusion, all our architectural changes increase the versatility of models and ensembling further amplifies such versatility in reducing training variances and achieving better performances.

Approach

1. **L1 Regularization:** $J(w,b) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^i, y^i) + \frac{\lambda}{2m}\|w\|_1$
2. **Go "deeper":** Add one more fully-connected layer to the output of BERT
3. **Freeze shallow transformer layers:** Freeze the weights of the first few layers of BERT (embedding layer, and the first few transformer layers)
4. **Use BERT's contextual embedding on BiDAF**

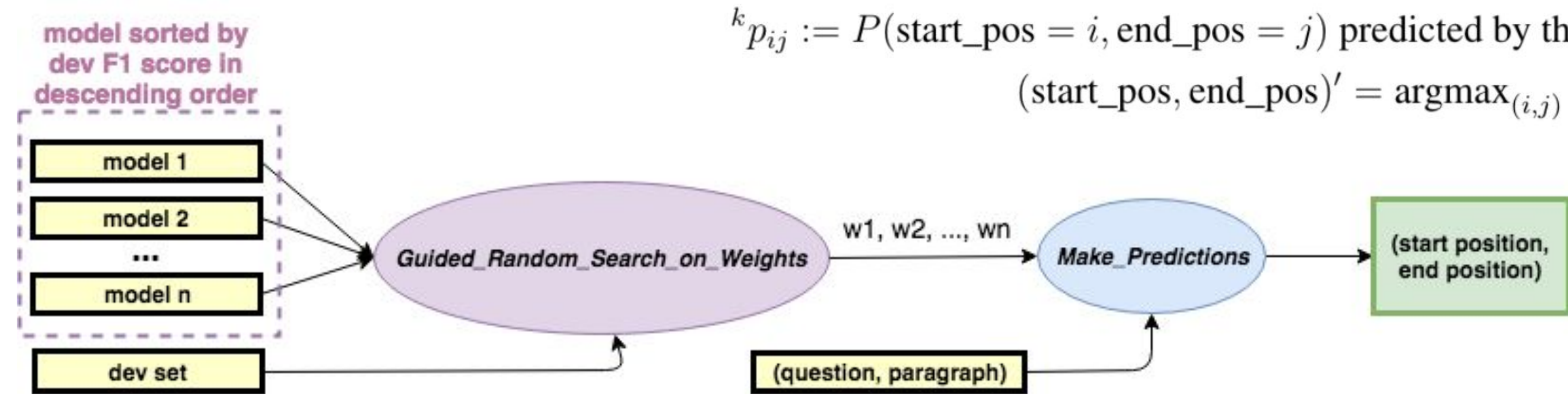


5. **Ensembling**

(a) Guided Random Search for Weighted Average Ensembling

$^{k}p_{ij} := P(\text{start\_pos} = i, \text{end\_pos} = j)$ predicted by the k-th model

$(\text{start\_pos}, \text{end\_pos})' = \text{argmax}_{(i,j)} \sum_k w_k \times p^k_{ij}$

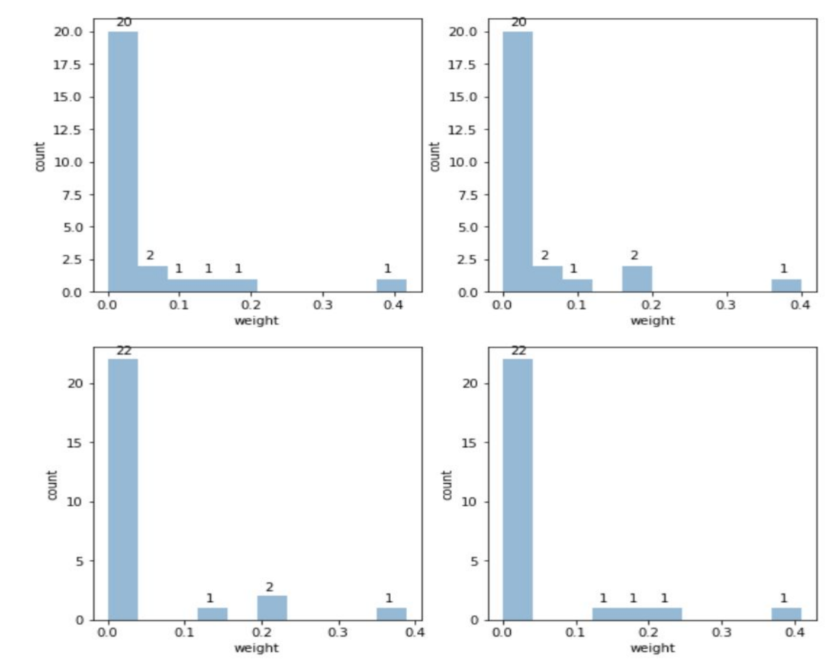


(b) Follow the Most Confident prediction

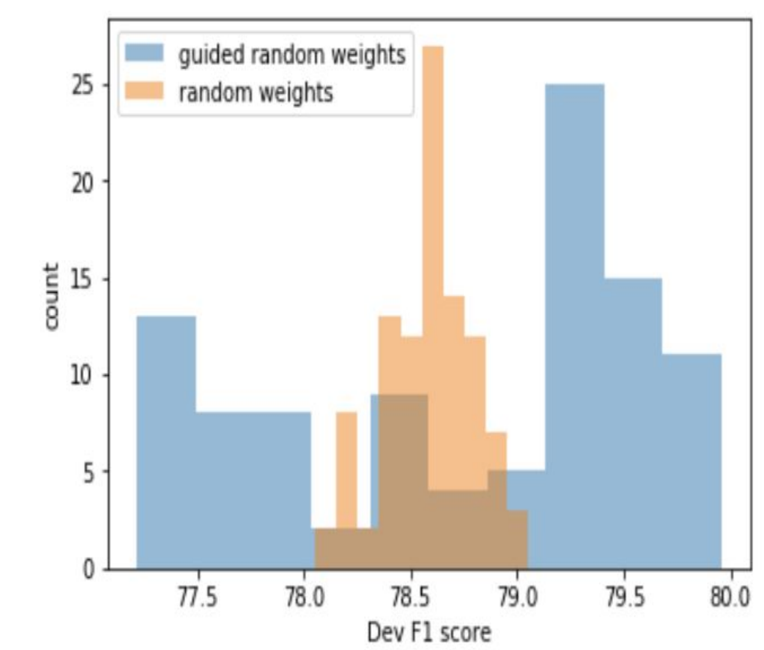
$(\text{start\_pos}, \text{end\_pos})^{k\prime}, p^k = \text{argmax}_{(i,j)} p^k_{ij}, \max_{(i,j)} p^k_{ij}$

$(\text{start\_pos}, \text{end\_pos})' = (\text{start\_pos}, \text{end\_pos})^{argmax_k p^k\prime}$

Analysis



This plot visualizes the weights for the top 4 Ensembling models in one run(100 iters) of Guided Random Search of weights by plotting the distribution in histograms. Most of the models only bears a weight in the order of 0.001.



F1 scores in one run(100 iters) of Guided Random Search of weights vs F1 scores in 100 iterations of random weights with no guide. The guided random search pushes F1 scores to higher.

References

Seo, Minjoon, et al. "Bidirectional attention flow for machine comprehension." arXiv preprint arXiv:1611.01603 (2016).
Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

Experiments & Results

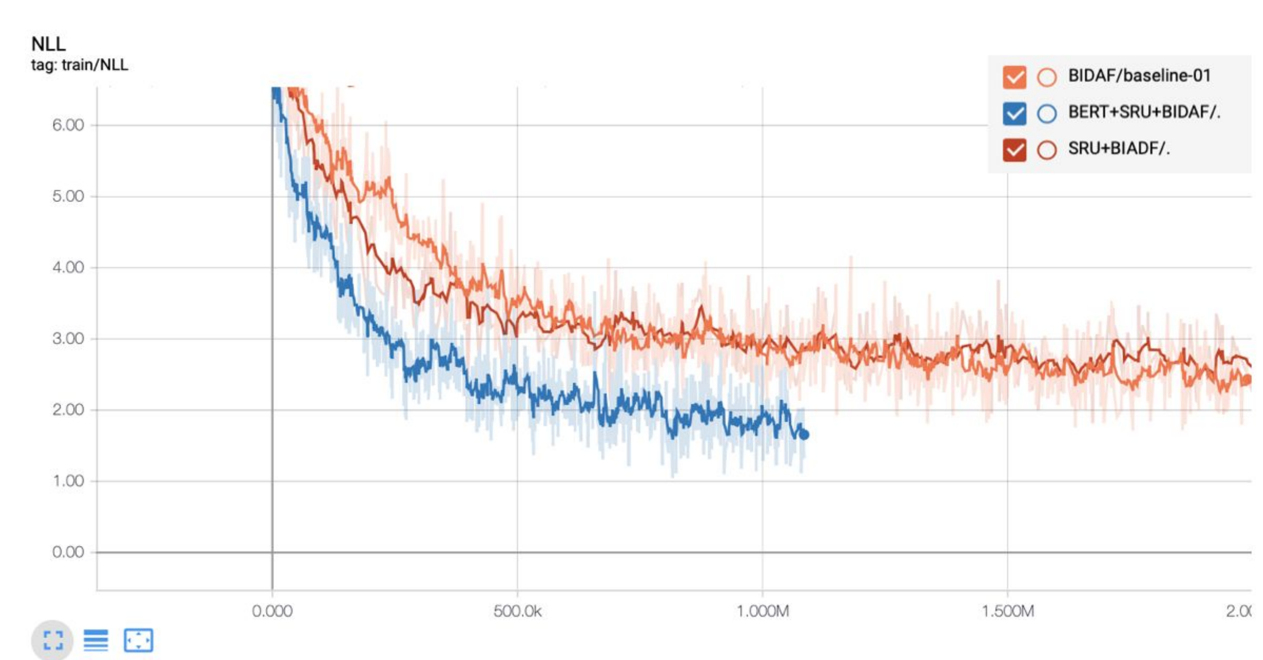
1. L1 Regularization: Train BERT with L1 regularization on weights of output classification and varies the coefficient. Increasing regularization strength helps improve the F1 and EM score.

Regularization Coefficient	Dev F1	Dev EM
0	74.679	71.915
1e-4	75.705	73.001
1e-3	76.666	73.824
1e-2	76.76	73.955

Freezed Layers	Dev F1	Dev EM
No Freeze	74.679	71.915
Embedding + 1 transformer layers	76.841	73.939
Embedding + 3 transformer layers	74.702	71.8
Embedding + 5 transformer layers	74.306	71.405
Embedding + 10 transformer layers	59.536	56.038

2. Freeze shallow transformer layers: Freeze BERT's embedding layer and its first 1, 3, 5, and 10 self-attention transformer layers while fine-tuning. Freeze first 1 layer improves the original model.

3. Use BERT's contextual embedding on BiDAF
The blue line represents training loss for BiDAF with BERT embedding. It converges faster than the original BiDAF model



Ensembling Model	Dev F1	Dev EM	Test F1	Test EM
Guided Random Search for Weighted Average	79.944	77.081	78.841	76.01
Follow the Most Confident Prediction	77.941	75.930	N/A	N/A

4. Final Ensembling Result

- Ensembling is effective in decreasing variances and reducing over-fitting.
- Quantitatively, our Ensembling model turns out to generalize well to the test set, with only a 1.3% decrease in both Test F1 and Test EM from Dev F1 and Dev EM, respectively