Question Answering on Natural Questions

Yiling Chen, Rena Sha
{yilingc, renasha}@stanford.edu
Stanford University

Problem Overview

Motivation We tackle Natural Questions (NQ), a question answering (QA) challenge released by Google. QA is an important natural language processing task where a system, given a question and a context document, returns the correct answer to the question.

Task Given a question and a relevant Wikipedia page, our task is to:
1. Identify the long answer (if any) from the document
2. Identify the short answer span (if any) from the document

Overview We focus on predicting long and short answers using a BERT Bi-GRU model that is computationally more efficient.

Related Work
- **DecAtt+DocReader Baseline** (Google)
  - Long Answer Selection: Decomposable Attention
  - Short Answer Select: Document Reader (DrQA)
- **BERT Based Baseline** (Google)
  - Sliding context window generating multiple instances with overlapping text
  - Application of pre-trained BERT model

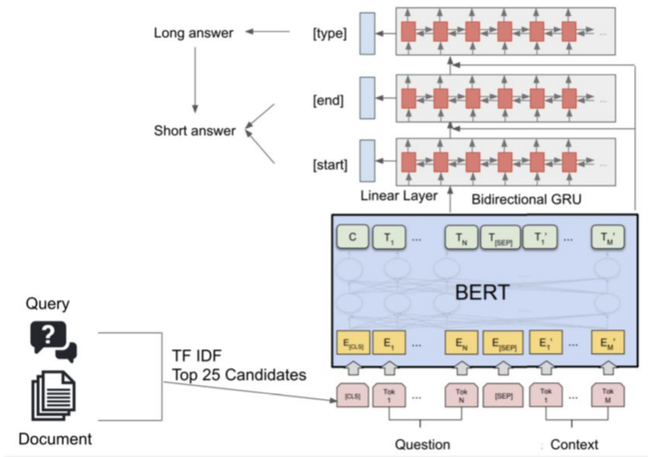
Dataset & Output

Dataset The NQ dataset contains real queries issued to the Google search engine and a corresponding Wikipedia article. An example takes the form {question, wikipedia page, long answer candidates, annotation}.

The NQ dataset totals 42GB larger than existing popular QA datasets (SQuAD 2.0 is 44MB). We aggressively downsampled:
- Training Data
  - 115K training instances (2 per training example)
- Test Data
  - Used 200/7830 development examples as dev set
  - Used full 7830 development examples as test set

Output We output the start, end location and answer type.

Model Architecture



Layers

1. **TF-IDF Candidates Retrieval**
   Selects top candidates by cosine similarity of tf-idf values of query and candidates

$$tf(t,d) = 0.5 + 0.5 \cdot \frac{f_{t,d}}{\max\{f_{t',d} : t' \in d\}},\ idf(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

$$tfidf = tf(t,d) \cdot idf(t,D)$$

2. **BERT Layer**
   We use the BERT-base-uncased model. BERT uses Transformer Architecture which has a "Multi-Head Attention" block. The Multi-Head attention block computes multiple attention weighted sums, attention is calculated by:

$$\texttt{Attention}_i(\mathbf{h}_j) = \sum_k \texttt{softmax}(\frac{W_i^q \mathbf{h}_j \cdot W_i^k \mathbf{h}_k}{\sqrt{d/n}})$$

3. **Bi-GRU Output Layer**
   We sequentially predict the start location, end location and answer type in 3 different output layers. Each output layer consists of a bi-GRU layer followed by a feed forward layer.

Loss Function

The loss of our model is defined as follows:

$$L = H(l_s, s) + H(l_e, e) + H(l_t, t)$$

where $H(x, class) = -x[class] + log(\sum_j \exp(x[j]))$ is the cross entropy loss between x and $class$, $l_s, l_e \in R^{512}$ is the output of the linear layer corresponding to the start/end position, and s, e are the ground truth of start/end position from the training example annotation. $l_t \in R^5$ is the output of the linear layer corresponding to the type which we defined in section 3.1, and t is the ground truth of the type of the instance.

Results & Discussion

	Long Answer Dev F1	Long Answer Dev P	Long Answer Dev R	Long Answer Test F1	Long Answer Test P	Long Answer Test R	Short Answer Dev F1	Short Answer Dev P	Short Answer Dev R	Short Answer Test F1	Short Answer Test P	Short Answer Test R
Baseline[1]	33.7	44.4	27.2	-	-	-	0	0	0	-	-	-
Baseline + Type[1]	33.7	28.8	40.8	-	-	-	5.6	6.0	5.3	-	-	-
Bi-GRU + TD-IDF[1]	49.0	49.5	48.5	**56.4**	51.6	62.1	26.3	29.0	24.0	20.4	18.6	22.6
DecAtt+DocReader [2]	54.8	52.7	57.0	55.0	54.3	55.7	31.4	34.3	28.9	31.5	31.9	31.1
BERT$_{joint}$ [3]	64.7	61.3	68.4	66.2	64.1	68.3	52.7	59.5	47.3	52.1	63.8	44.0
Human [2]	73.4	80.4	67.6	-	-	-	57.5	63.4	52.6	-	-	-

- ✓ Outperform the Google DecAtt+DocReader Model in the long answer prediction task
- × Not performing as well on the short answer prediction task
- ✓ Evaluates much faster! <2hr on M60 GPU vs. 5hr on P100 GPU

Ablation Analysis

	Long Answer F1	Long Answer P	Long Answer R	Short Answer F1	Short Answer P	Short Answer R
Bi-GRU + TD-IDF (all data)	49.0 (**+10.2**)	49.5	48.5	26.3 (**+14.4**)	29.0	24.0
Bi-GRU + TD-IDF (2% data)	38.8 (0.0)	32.3	48.5	11.9 (0.0)	55.6	6.7
Bi-GRU	34.0 (-4.8)	28.7	41.7	4.0 (-7.9)	2.9	6.7
Bi-GRU 50 instance per example	35.1 (+1.1)	52.9	26.2	8.7 (+4.7)	9.7	8.0
Feedforward	27.1 (-6.9)	23.6	32.0	8.2 (+4.2)	17.4	5.3

- Using more training examples helped!
- Narrowing down with TD-IDF improves performance despite being ~90% accurate
- More instances per example gives marginal improvement, but runtime is much longer
- More complex architecture learns better

Contribution
- A BERT based Bi-GRU model that is computationally more efficient than current work and outperforms the DecAtt+DocReader baseline in long answer prediction.
- Experimentation and analysis of the model

Lessons Learned
- Information retrieval techniques such as tf-idf can be used to reduce scope in question answering
- Bi-GRU's ability to retain features across sequential input allows for better performance

Future Work
- ★ Test out sliding context window data pre-processing.
- ★ Train network with more data and training time.

Reference

[1] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: a benchmark for question answering research. Transactions of the Association of Computational Linguistics, 2019.
[2] Chris Alberti, Kenton Lee, and Michael Collins. A bert baseline for the natural questions. CoRR,abs/1901.08634, 2019