CS230

# Who is Ernie and if so how many? A multitasking Bert for question answering with discrete reasoning

**Barthold Albrecht (bholdia)**
**Yanzhuo Wang (yzw)**
**Xiaofang Zhu (zhuxf)**

## Abstract

In this paper we show that a SQuAD-style BERT question answering model can be successfully extended beyond span extraction in a multitask setting. We demonstrate this on the newly released DROP dataset which requires discrete reasoning like counting, adding/subtracting, sorting or comparing for finding the correct answer. By adding further downstream tasks besides the span predictor we are able to improve the performance particularly for questions that must be answered with an infered number. We also have our model learn an answer type selector as meta task. Therefore, our design is open for the addition of further tasks to capture the broad range of challenges posed by DROP and other question answering datasets.

## 1 Introduction

Question Answering (QA) is a prime task for assessing the reading comprehension capabilities of an algorithm. A standard benchmark for this assessment is the Stanford Question Answering Dataset SQuAD (Rajpurkar et al., 2016)[1]: Given a paragraph and a question the task is to output the correct answer, which is a span of text directly from the paragraph (with SQuAD 2.0 additional complexity has been introduced as some questions may be impossible to answer). However, with the introduction of pre-trained language models like ELMo (Peters et al., 2017)[2], Open AI GPT (Radford et al., 2018)[3] and, in particular, BERT (Devlin et al.; 2018)[4] human level performance has been surpassed now for both versions of SQuAD.

Therefore, new benchmarks are introduced to stimulate advances towards structurally richer natural language understanding. One such benchmark is "DROP: Discrete Reasoning Over Paragraphs" (Dua et al.; 2019)[5] which was published just some weeks ago. Here, producing the answers might require a mapping of multiple references from the question to the paragraph in order to execute discrete operations like counting, sorting or comparing. In this setting the F1 score of BERT, for example, drops by more than 50 points as compared to SQuAD (Dua et al., 2019)[5].

However, in this project we show that BERT is adaptive and can serve as a powerful building block also in such a new setting. We do so in extending the SQuAD-style span prediction by number prediction where this number could be arrived at through counting or through adding/subtracting. In such a multitask setting we are able to increase the performance as compared to the "traditional" use of BERT which also served as a baseline in (Dua et al., 2019)[5].

,

## 2 Related work

Multitask models have gained considerable attention for NLP tasks as outlined by Ruder (2017)[7] and demonstrated, e.g., by McCann et al. (2018)[8]. This line of research indicates that the new challenges introduced by DROP fit such an approach as different sub-tasks need to be executed to construct an answer. A multitask setting is also adopted in the NAQANet - a numerically-aware QANet submitted by Dua et al. (2019)[5] which serves as the first benchmark and is currently state of the art. The authors of this paper even envisaged combining their model with BERT as future work. In that sense our paper can be seen as a contribution to this new and challenging research in the the field of reading comprehension.

## 3 Dataset and Features

### 3.1 Overview

We use the dataset "DROP" which can be downloaded from the respective website for the competition (https://leaderboard.allenai.org/drop/submissions/get-started). This dataset consists of 96k question-answer pairs which have been adversarially created: the examples that made into the dataset are the ones that could not be answered by BiDAF (Seo et al., 2017)[6], a previously best-performing SQuAD-style reading comprehension model. The paragraphs are drawn from Wikipedia with an emphasis on sports and history. The most distinguishable characteristic of the DROP dataset is that all questions require the system to perform discrete reasoning like counting, adding/subtracting, comparing or co-reference resolution to find correct answers. Accordingly, the answers can be of various type: a) a single or multiple spans in the paragraph b) a single or multiple spans in the question c) a date or d) a number. The analysis from the paper reveals that over 60% of the answers are numbers and that on average more than two spans must be considered to produce a correct answer. Sample questions from Dua et al., 2019[5] are given in the appendix.

### 3.2 Data conversion

The DROP dataset provided by AllenNlp has a different format than what is accepted by BERT. Specifically, the answers in the DROP dataset is in plain text. The BERT model, on the other hand, expects the plain answer text plus the passage index of the first character of the answer. There exists a script from AllneNlp that does most of the conversion. Another big challenge is to convert the date and number answers from DROP to the BERT style as BERT previously assumes that the all the answers are expressed as spans in the passage. To accomplish this, first the data and number answers are converted to new fields in the SQuAD json file and a new end-to-end pipeline is created in the BERT model that takes the json file as inputs and plugs them into the BERT framework.

## 4 Methods

Before introducing the NAQANet Dua et al. (2019)[5] ran several established QA models on the dataset for baselining. Among those, a version of BERT which is finetuned for SQuAD-style question answering and implemented by huggingface (https://github.com/huggingface/pytorch-pretrained-BERT) performes best with EM 30.10 and F1 33.36 on the dev set.

As a first milestone for our project we reproduced this baseline with the same implementation from huggingface, yielding a performance of EM 27.63 and F1 31.36 on the dev set. This result was achieved with maximum sequence length of 278, batch size of 20 and training over 3 epochs. The slightly weaker performance of our implementation is presumably due to the fact that Dua et al. (2019)[5] employ the BERT large version with 340 million parameters whereas we use BERT base with 110 m parameters due to memory constraints of our virtual machine.

To establish this baseline we use the allennlp framework (https://github.com/allenai/allennlp) for data preprocessing, in particular for reading the DROP dataset into memory and converting it into SQuAD-style question/answer-pairs. Only for this baseline during training all examples that can not be answered by a span from the passage are skipped which reduces the size of the training set by 45% to 42842. For evaluation, all examples from the dev set regardless their nature are passed to the model and evaluated.

In a second step we successively added other downstream tasks to our model. The largest impact on the performance is achieved with a "Count" module. Here, we add a two layer Feed Forward network which performs a classification task over ten digits (0 to 9). A second two layer Feed Forward network predicts signs (0, 1 for "+" and 2 for "-") for each of the extracted numbers in the passage in order to add or subtract them. Finally, a third Feed Forward network predicts the answer type ("Count" or "Add/Subtract") and picks the most probable answer accordingly. However, this type selector only get's used when the "traditional" "Span" module does not return an answer and, hence, predicts that the answer is impossible to find in a span from the passage.

# 5 Experiments/Results/Discussion

## 5.1 Experiments

We use the hyperparameters suggested by the huggingface BERT repo, which is fine-tuned for the SQuAD dataset. Since these hyperparameters are proven to achieve top results in the original BERT paper(Devlin et al.; 2018)[4], we keep them mostly untouched. They are summarized in the following table:

## 5.2 Results

We adopt the evaluation metrics from DROP paper(Dua et al., 2019)[5]: EM(Exact Match) and F1 score. The formula for EM is exactly the same as the one in the original SQuAD paper(Rajpurkar et al., 2016)[1], while the F1 score calulation takes into account of the numerical values, such that it gives a score of 0 when there is a number mismatch between the true answer and the predicted one.

The following table summaries the detailed breakdown of results for each model.

## 5.3 Discussion

Overall, there is a moderate increase in both EM and F1 scores as we integrate more tasks to the BERT baseline model. This is expected. However, by breaking down the result by the answer types and the model types, there are some interesting findings.

First the "Count" task substantially increases the model's performance on the "Number" type answers, by 3 to 4 points in both EM and F1 score. This behaviour is aligned with the result presented in the DROP paper(Dua et al., 2019)[5] because our implementation of the "Count" task on BERT is mostly similar to the one in NAQANet from the DROP paper(Dua et al., 2019)[5]. However, our "Count" task has some limitations compared to the NAQANet. In NAQANet, there can exist multiple valid "Count" answers in one example, whereas the our "Count" task only allows a single answer. This is due to the fact that the pytorch CrossEntropy function we used to compute loss does not take multiple targets as inputs.

Another observation is that adding the ability to predict answer as a question span has little effects on overall score (third column in Table 2). A further analysis reveals that for single-span answers, about 52% of them can be answered by only the passage spans, 42% can be answered by spans from both the passage and question, and only 4% of the them can be answered by only question spans. This means even if we implemented a perfect predictor for question spans, the maximum improvement is very limited. Moreover, the existing passage span task receives very little benefit from the question span task because these two tasks are virtually learning the same thing.

The overall improvement from the Addition Subtraction task is also marginal. Several reasons contribute this effect. First, the input sequence accepted by the BERT model is tokenized by the WordPiece tokenizer. As an example, suppose an answer "1367" is arrived by adding two numbers "1365" and "2" from the passage. The WordPiece tokenizer splits "1365" into "136" and "5", then there is no more valid expression that evaluate to "1367" given "136", "5" and "2" tokens. As a result, the add/sub task learns that there exists no answer in this training example. The second reason is that BERT model breaks a long passage into different chunks by a sliding window, each chunk is turned into an individual example. Thus there is a chance that two numbers in a valid expression will not end up in the same chunk, and again, the add/sub task misses a learning point.

| **Reasoning** | **Passage** (some parts shortened) | **Question** | **Answer** | **BiDAF** |
|---|---|---|---|---|
| Subtraction (28.8%) | That year, his **Untitled (1981)**, a painting of a haloed, black-headed man with a bright red skeletal body, depicted amid the artists signature scrawls, was **sold by Robert Lehrman for $16.3 million, well above its $12 million high estimate**. | How many more dollars was the Untitled (1981) painting sold for than the 12 million dollar estimation? | 4300000 | $16.3 million |
| Comparison (18.2%) | In **1517, the seventeen-year-old King sailed to Castile**. There, his Flemish court .... **In May 1518, Charles traveled to Barcelona in Aragon**. | Where did Charles travel to first, Castile or Barcelona? | Castile | Aragon |
| Selection (19.4%) | In 1970, to commemorate the 100th anniversary of the founding of Baldwin City, **Baker University professor and playwright Don Mueller and Phyllis E. Braun, Business Manager, produced a musical play entitled The Ballad Of Black Jack** to tell the story of the events that led up to the battle. | Who was the University professor that helped produce The Ballad Of Black Jack, Ivan Boyd or Don Mueller? | Don Mueller | Baker |
| Addition (11.7%) | Before the UNPROFOR fully deployed, the HV clashed with an armed force of the RSK in the village of Nos Kalik, located in a pink zone near Šibenik, and captured the village at 4:45 p.m. on **2 March 1992**. The JNA formed a battlegroup to counterattack the **next day**. | What date did the JNA form a battlegroup to counterattack after the village of Nos Kalik was captured? | 3 March 1992 | 2 March 1992 |
| Count (16.5%) and Sort (11.7%) | Denver would retake the lead with kicker **Matt Prater nailing a 43-yard field goal**, yet Carolina answered as kicker **John Kasay ties the game with a 39-yard field goal**. ... Carolina closed out the half with **Kasay nailing a 44-yard field goal**. ... In the fourth quarter, Carolina sealed the win with **Kasay's 42-yard field goal**. | Which kicker kicked the most field goals? | John Kasay | Matt Prater |
| Coreference Resolution (3.7%) | **James Douglas** was the second son of Sir George Douglas of Pittendreich, and Elizabeth Douglas, daughter David Douglas of Pittendreich. Before **1543 he married Elizabeth**, daughter of James Douglas, 3rd Earl of Morton. **In 1553 James Douglas succeeded to the title and estates of his father-in-law**. | How many years after he married Elizabeth did James Douglas succeed to the title and estates of his father-in-law? | 10 | 1553 |
| Other Arithmetic (3.2%) | Although the movement initially gathered some **60,000 adherents**, the subsequent establishment of the Bulgarian Exarchate **reduced their number by some 75%**. | How many adherents were left after the establishment of the Bulgarian Exarchate? | 15000 | 60,000 |
| Set of spans (6.0%) | According to some sources 363 civilians were killed in **Kavadarci**, 230 in **Negotino** and 40 in **Vatasha**. | What were the 3 villages that people were killed in? | Kavadarci, Negotino, Vatasha | Negotino and 40 in Vatasha |
| Other (6.8%) | This **Annual Financial Report** is our principal financial statement of accountability. The **AFR gives a comprehensive view** of the Department's financial activities ... | What does AFR stand for? | Annual Financial Report | one of the Big Four audit firms |

Figure 1: Question and answer types from the DROP dataset and the required reasoning as presented by Dua et al., 2019[5].

| Model | Learning rate | Batch size | Max seq length of input | Stride length | Training epochs |
|---|---|---|---|---|---|
| bert-base-uncased | 3e-5 | 20 | 278 | 128 | 2∼3 |

Table 1: hyperparameters

| | BERT | | BERT+Count | | BERT+Count+Q Span | | BERT+Count+Add/Sub | | BERT+Count+Add/Sub 3 Epochs | |
|---|---|---|---|---|---|---|---|---|---|---|
| | EM | F1 | EM | F1 | EM | F1 | EM | F1 | EM | F1 |
| Overall | 24.2 | 28.1 | 26.3 | 29.7 | 26.3 | 29.5 | 27.6 | 30.7 | 28.9 | 32.2 |
| Date (1.5%) | 31.3 | 36.2 | 30.8 | 34.3 | 29.1 | 32.1 | 29.9 | 31.2 | 28.2 | 30.1 |
| Number (61.9%) | 13.8 | 14.3 | 17.7 | 18.0 | 16.1 | 16.4 | 17.8 | 18.0 | 18.8 | 19.0 |
| Span (31.7%) | 47.9 | 55.8 | 46.7 | 53.8 | 49.9 | 56.2 | 50.7 | 56.8 | 53.1 | 59.4 |
| Spans (4.9%) | 0 | 20.8 | 0 | 20.9 | 0 | 20.6 | 0 | 21.0 | 0 | 23.0 |

Table 2: Experiment results

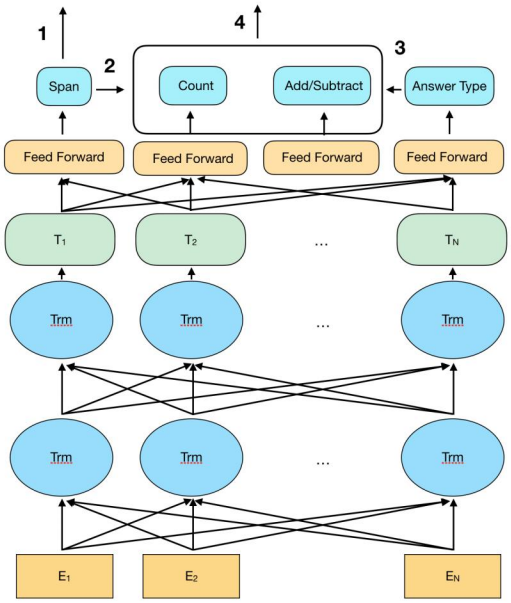


Figure 2: The embedded question and passage are encoded through the transformer blocks of BERT. The returned output sequence is then fed to four separate fully connected networks. If (1) "Span" returns a prediction it is chosen as answer. If (2)"Span" returns that the answer can not be found in the passage, an alternative answer will be chosen: the "Answer Type"-selector will decide whether the question can better be answered by "Count" or by "Add/Subtract" and (4) the answer is given accordingly.

## 6 Conclusion/Future Work

We believe that our model serves as a good platform for further improvements on the drop dataset. Therefore, we plan to further enhance the arithmetic capabilities of our model and to tackle the issue of multiple spans needed for the correct answer.

## 7 Contributions

All team members contributed equally to the project. Yanzhuo Wang put particular emphasis on the data conversion and the question span part, Barthold Albrecht delved mainly into the other downstream tasks and the model design, and Xiaofang Zhu concentrated on the performance evaluation and submission format.

## References

[1] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang, (2016). *Squad: 100,000+ questions for machine comprehension of text*. arXiv preprint arXiv:1606.05250

[2] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Ken-ton Lee, and Luke Zettlemoyer, (2018). *Deep contextualized word representations*. arXiv preprint arXiv:1802.05365.

[3] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, (2018). *Improving language understanding with unsupervised learning*. Technical report, OpenAI.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv preprint arXiv:1810.04805

[5] Dua, D., Wang, Y., Dasigi, P., Stanovsky, G., Singh, S., Gardner, M, (2019). *DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs*. arXiv preprint arXiv:1903.00161.

[6] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi, (2016). *Bidirectional attention flow for machine comprehension*. arXiv preprint arXiv:1611.01603.

[7] Sebastian Ruder, (2017). *An Overview of Multi-Task Learning in Deep Neural Networks*. arXiv preprint arXiv:1903.00161.

[8] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, Richard Socher, (2018). *The Natural Language Decathlon: Multitask Learning as Question Answering*. arXiv preprint arXiv:1806.08730v1.