**Introduction**

Question Answering is a task in Natural Language Processing where an algorithm attempts to answer a question given context that would help answer it. For this project, ALBERT, a BERT-based neural network was evaluated on the SQuAD question answering dataset. Previously, some hyperparameters were tuned such as learning rate and batch size to gain a better understanding of what affects model performance. Through this tuning, we achieve a F1 Score of 90.3. With this model, we analyze 500 incorrect predictions from the validation set to understand the errors that the model produces.

**Data**

500 samples were randomly selected from the SQuAD validation set that produced incorrect predictions were analyzed in this work. Furthermore, a TSNE visualization of all embeddings of the SQuAD validation set were used to see if incorrect predictions were clustered together.

**Results**

**Reasons of Failure by Percent**

Out of the 500 incorrect predictions, the errors were categorized into 28 categories. For the specific categories and their counts, check the “Reasons of Failure” workbook in the included spreadsheet. Notably, the majority of the errors (36.6%) are due to span-mismatches between the predicted and the gold-standard output. All of these “errors” were technically correct answers, but did not exactly match the span the answer bank was looking for. An example of this is when the model predicts “green spaces” as the answer to “What type of space in Warsaw are the Botanic Garden and University Library garden?”, and the gold-standard specified “green” as the only acceptable answer. These, as well as other mismatches, such as punctuation that was included in the answer or prediction, accounts for a plurality of the errors. 22.4% of the time, the model simply failed to extract the correct answer.

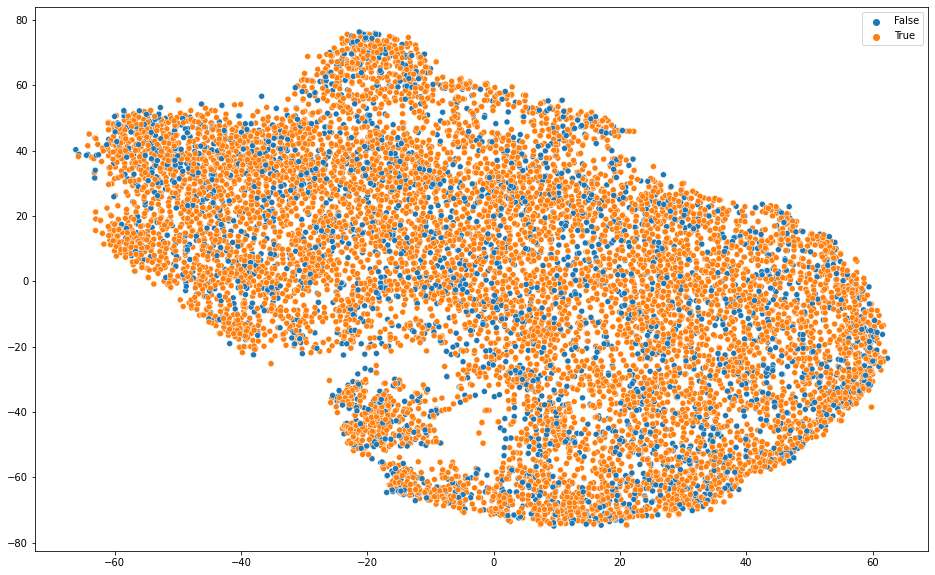
**Types of Answer**

**Types of Answer When Model Failure Occurs**

Also analyzed were the types of answers that generated the most errors. 41% of the time, the question was looking for a “thing” in the text. The next largest category that caused errors were “concepts”, or more abstract ideas described in the passages. Note, this should not be viewed as “the hardest type of answer”, as we do not know the dataset characteristics. If 41% of answers (both correct and incorrect) are of type “thing”, then the model does not struggle with a certain type of answer, rather it struggles equally against all classes of questions. If we break down the cases in which the model simply failed to predict the correct answer, the data paints a different picture. We can see that Concept and Reason answer types occur much more frequently in this subset than “simpler” answer types such as people.

**Relative Occurrence of Answer Type between Model Failure Case and entire Dataset**

Here, we see that the model struggles with Concept and Reason answering, while errors when the answer is a location, person, or number typically occur due to other reasons than straight-up failure cases. This is probably due to the Concept and Reason answer types are much more abstract compared to extracting a name of a person or number from the text. It would be interesting to do similar analysis between incorrect answers in the “model failure” compared to correct classifications.



**TSNE Projection of Answer Embeddings**

From the TSNE plot, we can see that the embeddings of correct and incorrect predictions are pretty well interspersed with one another. However, the incorrect predictions seem to be in small clusters throughout the plot. It would be interesting to see if the datapoints in these clusters have commonalities such as the type of answer they are looking for, but this was not accomplished.

**Conclusion**

In conclusion, after analyzing 500 incorrect predictions on the SQuAD dataset, we see that a lot of the predictions are “correct” in the sense that they make sense to humans, but unfortunately do not exactly match the gold-standards curated in the dataset. This can be improved with better data labelling techniques. Additionally, erroneous punctuation or tokens such as “the” should be considered before saying that a prediction is incorrect, and clean those from the predicted output. However, there are many instances where the predictions do not come anywhere close to correct, and this is where more training data and better model training could help to improve performance.

**References**

The code used is from this HuggingFace tutorial notebook: <https://colab.research.google.com/github/huggingface/notebooks/blob/master/examples/question_answering.ipynb>

Some light modifications to use Weights & Biases were found here:

<https://docs.wandb.ai/guides/integrations/huggingface>

Papers with Code SQuAD Leaderboard:

<https://paperswithcode.com/sota/question-answering-on-squad11?metric=F1>

ALBERT Paper:

@article{DBLP:journals/corr/abs-1909-11942,

author = {Zhenzhong Lan and

Mingda Chen and

Sebastian Goodman and

Kevin Gimpel and

Piyush Sharma and

Radu Soricut},

title = {{ALBERT:} {A} Lite {BERT} for Self-supervised Learning of Language

Representations},

journal = {CoRR},

volume = {abs/1909.11942},

year = {2019},

url = {http://arxiv.org/abs/1909.11942},

eprinttype = {arXiv},

eprint = {1909.11942},

timestamp = {Fri, 27 Sep 2019 13:04:21 +0200},

biburl = {https://dblp.org/rec/journals/corr/abs-1909-11942.bib},

bibsource = {dblp computer science bibliography, https://dblp.org}

}

SQuAD Paper:

@article{DBLP:journals/corr/RajpurkarZLL16,

author = {Pranav Rajpurkar and

Jian Zhang and

Konstantin Lopyrev and

Percy Liang},

title = {SQuAD: 100, 000+ Questions for Machine Comprehension of Text},

journal = {CoRR},

volume = {abs/1606.05250},

year = {2016},

url = {http://arxiv.org/abs/1606.05250},

eprinttype = {arXiv},

eprint = {1606.05250},

timestamp = {Mon, 24 Aug 2020 14:01:25 +0200},

biburl = {https://dblp.org/rec/journals/corr/RajpurkarZLL16.bib},

bibsource = {dblp computer science bibliography, https://dblp.org}

}

Code can be accessed on GitHub: https://github.com/parkererickson/csci8980-hw2