Liam Keane and Lazuli Kleinhans
Fall 2024
Natural Language Processing Final

Question Answering:
Benchmarking a BERT vs. FLAN-T5 model on SQuAD v2.0

For us, one of the most intriguing topics in NLP is the ability to algorithmically answer questions about written information. Despite seeing the results repeatedly during this project and demystifying transformers in class, it is still quite surprising to us that these tools exist. In addition to our interest, solving this problem has obvious benefits to NLP as well as practical applications outside the field itself. Having to sift through large amounts of textual information is not uncommon for any person, regardless of their profession or background, so the ability to streamline this process benefits a large breadth of people.

A concrete example would be a historian doing research where answering specific questions about their research topic would save them large amounts of time. This would be time otherwise spent combing through source after source, looking for the answer to their query (I recognize I am considering this from the perspective of a lazy computer scientist, and I imagine historians tend to enjoy copious amounts of reading).

The Data

We used the **S**tanford **Qu**estion **A**nswering **D**ataset v2.0 (SQuAD) as our benchmarking data. It is a set of reading comprehension questions and answers (or lack thereof) cultivated by crowdworkers on a set of thirty-five Wikipedia articles. Each question corresponds to a specific chunk of text (or span) from their parent article, and a given model is expected to provide the string of text within the span containing the answer. SQuAD v2.0 builds off the original 100,000 questions from v1.1 with an additional 50,000 unanswerable questions written to look as though there should be an answer in the given text. These unanswerable questions were also written by crowdworkers. This adds an additional layer of robustness required for models to perform well on this dataset, as they will need to recognize an answer is not contained in the span.

Given our machines' computational limitations and the time it takes to query our models thousands of times, we were only able to use the first nine topics (articles) for our benchmarks. This left us with a total of roughly 3,000 questions, with an almost exact split between answerable and unanswerable questions.

The Metrics

Beginning with our rough estimates before moving on to our more practical measurements, we used a human heuristic to measure preference between answers given by the two models. We created a simple Python program that presents the user with each model's answer to a given question, with the author of each response obscured. We also provided the user with the context the model had access to. The user is then asked to choose which of the two options they think is better, completely based upon their own preferences. (the answers are presented in a random order each time to prevent any unintentional bias towards or against either model).

Our second measurement is slightly more robust, but still not accurate enough to be used in practice. Using the Gensim vector space modeling toolkit and the "glove-wiki-gigaword-50" dataset, we calculated the distance in vector space between each answer predicted by the model and the ground truth specified in the dataset, then added that distance to the sum corresponding to that model. Note that the lower the distance (the closer to zero) between words, the more semantically similar they are given the data they were trained on. From there we can visualize which model had a better score (a smaller sum).

Finally, our third measurement comes directly from an evaluation script written by the authors of SQuAD. Given the developer set (a JSON containing the questions, spans, and answers) and a set of predictions from a model, the script then calculates the accuracy and F1 measurement of answerable, unanswerable, and total combined questions.

The BERT model

BERT stands for "Bidirectional Encoder Representations from Transformers," and it is an open-source machine learning framework developed by Google in 2019. The transformer-based neural network makes up the core functionality of BERT models, allowing them to interpret human-readable language.[1] The "encoder" piece of the acronym describes the "encoder-only" architecture employed by the Google researchers who opted to remove the decoder module in later iterations, likely to emphasize analyzing human input rather than generating output.[2] The bidirectional piece was what made BERT such a dramatic improvement over current models; instead of considering a sentence sequentially (like

[1] GG's explanation of BERT
[2] Wikipedia's explanation of BERT

our n-gram model), BERT can consider context both to the left and right of a given word. The model is pre-trained on a large amount of unlabeled text to learn contextual embeddings then fine-tuned for a specific downstream task (like question answering) with additional data.

We opted to use Deepset's implementation of BERT trained on SQuAD v2.0 for our benchmarking because it is easily accessible through Hugging Face and was built using SQuAD v2.0's training set. The BERT's predictions for the questions of the first nine topics can be found here.

Human Metric:

Out of 100 BERT and FLAN-T5 response comparison prompts shown to human graders, a total of 87 response preferences were collected. From these 87 preference points, the BERT model was awarded **56/87** or ~**65%**.

Word Vector Metric:

The sum of the distance between all questions' predicted answers and their ground truths totaled to ~**722**.

Accuracy and F1 Metric:

Running the evaluation script on our results yielded the following statistics for questions both with and without answers as well as the overall performance:

	Accuracy	F1	# of Questions
Answerable	59.43%	66.42	1484
Unanswerable	69.75%	69.75	1461
OVERALL	**64.55%**	**68.07**	**2945**

Figure 1

The FLAN-T5 model

Another innovative model developed by Google that was coincidentally released the same year as BERT was the T5 model, which stands for "Text-To-Text Transfer Transformer."[3] The foundational idea behind this model is that it converts both input and output into text strings—this standardization simplifies the approach needed by the framework. Similar to BERT, a T5 model utilizes a transformer

[3] Medium's explanation of T5

architecture; however, unlike BERT, it makes use of both an encoder for processing input and a decoder for generating output. The model itself is pre-trained on a massive corpus of data before being fine-tuned for more specific tasks (such as question answering). That is where the FLAN piece of the model enters into the process. FLAN is a fine-tuning technique that trains the model on a varied set of instructions, which is meant to make the model perform better on all NLP tasks in general.

We chose to use Deepset's implementation of a FLAN-T5 model tuned using SQuAD v2.0 training data because it was trained using the same data as our BERT model and was created by the same startup, which affords some level of standardization for our benchmarking. The FLAN-T5's predictions for the questions of the first nine topics can be found here.

Human Metric:

Out of 100 BERT and FLAN-T5 response comparison prompts shown to human graders, a total of 87 response preferences were collected. From these 87 preference points, the FLAN-T5 model was awarded **31/87** or ~**35%**.

Word Vector Metric:

The sum of the distance between all questions' predicted answers and their ground truths totaled to ~**1463**.

Accuracy and F1 Metric:

Running the evaluation script on our results yielded the following statistics for questions both with and without answers as well as the overall performance:

	Accuracy	**F1**	**# of Questions**
Answerable	87.20%	91.86	1484
Unanswerable	1.16%	1.16	1461
OVERALL	**44.52%**	**46.86**	**2945**

Figure 2

Comparing the Results

Beginning with our human heuristic metric, the users preferred the BERT model's predictions 65% of the time and the FLAN-T5's only for 35% of the questions. The word vector distance sums revealed a similar disparity with BERT's sum of 722 being roughly *half* the FLAN-T5's total of 1463.

As a means of comparison between the two models, without regard to their quality, this offers clear evidence in favor of the BERT model's ability to answer and/or refrain from answering questions. However, the comparison is more complex when we consider their ability to handle answerable vs. unanswerable questions.

Looking at Figures 1 and 2, the disparity between the ability of the BERT and the FLAN-T5 model to accurately predict *answerable* questions is completely different from what we might expect given just the previous two metrics. The BERT model performs significantly worse than the FLAN-T5 model with an accuracy of around 60% and an F1 of 66.42 while the FLAN-T5 had an accuracy of around 87% with an F1 of 91.86. This result was quite surprising, not just because of our other metrics, but because of the design philosophy behind FLAN-trained models. This leads us to believe that they are "jack of all trades" models while BERT models tend to be tuned for specific downstream tasks. A possible explanation for this discrepancy could be the quality of the pre-training data. While both models were created by Google researchers, only the FLAN-T5 was cited as using Google's implementation as their base model. Deepset's BERT model is likely less robust than one developed by Google using their nearly unlimited resources.

Figures 1 and 2 show an even greater disparity between BERT and FLAN-T5 when looking at their responses to *unanswerable* questions. When comparing its performance with unanswerable questions to answerable questions, the BERT model performed similarly, with an accuracy of 69.75% and an F1 score of 69.75. The FLAN-T5 model's performance on the other hand completely plummeted, with an accuracy of 1.16% and an F1 score of 1.16. This is very likely due to FLAN-T5's near inability to provide a blank response to an impossible question, and will often just generate the first word of the question (i.e. generating "What" in response to "What is the boiling point of steam?"). A possible explanation for FLAN-T5's difficulty in not providing any answer to an unanswerable question could be that the fine-tuning performed was not effective enough. With FLAN-T5 trying to be more flexible and available for more use cases, its ability to be then trained to not provide *any* output when accurate output cannot be generated may be too extreme of a change.

Given more time we would like to explore these two questions further through benchmarking additional models. With the original T5 as a baseline, we could test additional variations of the T5 (including our FLAN-T5) to see whether they performed similarly or if they could recognize unanswerable questions. Similarly, we could test newer variations of BERT such as roBERTa and alBERT to see whether they improve over the original or close the gap with the FLAN-T5. Additionally, we would like to benchmark the speed it takes to query a model and take that into consideration when we are comparing models. During our testing, the BERT model could query in roughly less than a second while the FLAN-T5 model would consistently take around 8 seconds per query (it took a long time)! For reference, generating T5-FLAN responses for the first 15 topics took over 10 hours!

While the T5 model performed significantly better than the BERT model in both accuracy and F1 score for answerable questions, it really struggled with questions that were not answerable with the given information. The BERT model on the other hand performed decently well for both answerable and unanswerable questions. This matches the human heuristic results, as in most response comparisons the human graders preferred the BERT outputs. Overall, we believe that the BERT model performed better. However, given the choice to use one of these two models to answer a question we know can be answered using a provided document, we would choose the FLAN-T5 model. Even in the case where we were not entirely confident, fact-checking these models is simple in comparison to LLMs like Claude or ChatGPT since the responses they provide should mirror an excerpt from the text.

Based on what we have seen, bias enters a model through its training data and its developers. Both of these two models were trained using SQuAD v2.0, which was developed by the Stanford Natural Language Processing Group with crowdworkers of unspecified backgrounds. The topics themselves were chosen by Stanford researchers who clearly attempted to diversify the subjects, but they still tended to focus on Western ideals. The ability of the model to answer questions about a breadth of different topics requires training with a breadth of topics, so this might prevent the models from being as generalizable to other people and cultures as we might hope. As for our two chosen implementations, we know very little about Deepset's values or the design considerations they made when creating these models (as we discovered when trying to uncover how they trained their BERT model). While developer choices may introduce bias in more subtle ways than training data, it can still lead to similar problems of generalization. Consider our earlier example of a historian using a model to answer questions about text; a developer (or group of developers) may design a tool with a specific group of people in mind, like academics, instead of considering the larger context.

References

Khanna, C. (2021, May 15). *Question Answering with a fine-tuned BERT*. Medium; Towards Data Science. https://towardsdatascience.com/question-answering-with-a-fine-tuned-bert-bc4dafd45626

pawangfg. (2024, January 10). *Explanation of BERT Model - NLP*. GeeksforGeeks. https://www.geeksforgeeks.org/explanation-of-bert-model-nlp/

Pena, D. (2023, November 14). *BERT vs LLM: A Comparison*. SitePoint; SitePoint Pty. Ltd. https://www.sitepoint.com/bert-vs-llm-a-comparison/

Rajpurkar, P. (2018). *The Stanford Question Answering Dataset*. SQuAD; Stanford NLP Group. https://rajpurkar.github.io/SQuAD-explorer/

Wikipedia authors. (2024, November 18). *BERT (language model)*. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/BERT_(language_model)