Decoding Sherlock Holmes: A Comparative Analysis of BERT Models' Performances

Ria Patel, Devanshi Patel, Sai Dasari, Seoyoung Amy An, and Jihun Kim
Department of Electrical Engineering and Computer Science,
University of Tennessee, Knoxville, TN 37996, USA

***Abstract*—This project evaluates the one-shot learning capabilities of BERT-based pre-trained models, including SpanBERT and RoBERTa variants, in question-answering tasks. The models were tested using ten plot-related questions related to Sir Arthur Conan Doyle's Sherlock Holmes series, requiring them to extract relevant answers from the narrative texts. Text chunking was employed to split the narratives into manageable sections that fit the models' input size limitations to handle the long contexts of the stories. Parallel processing pipelines and GPU acceleration optimize computational efficiency and reduce execution time. Models fine-tuned for question answering, such as RoBERTa-SQuAD2.0, were observed to handle the complexity of narrative contexts more effectively compared to general-purpose models like BERT-base and BERT-large. The study highlights how different BERT models handle question-answering tasks by evaluating different models' performances.**

***Index Terms*—BERT models, Plot Analysis, Performance Comparison, Natural Language Processing,**

I. Introduction

Models such as BERT (Bidirectional Encoder Representations from Transformers), developed by Google, are trained to understand and handle context-sensitive language, making them well-suited for task answering and particularly effective in analyzing complex plots. BERT is a self-supervised language model trained on a large corpus of English text to perform masked language modeling (MLM) and next-sentence prediction (NSP) [1]. In MLM, the model randomly masks 15% of the words in the given sentence and then predicts the masked words, learning the bidirectional representations of the sentence. NSP involves concatenating two masked sentences and training the model to determine if they follow a sequential order. These two objectives help the model learn relevant language features that other models might miss. However, the original BERT model contains a huge number of parameters, making it computationally expensive to train and run.

In this project, we compared and analyzed the performance of various BERT-based models in the context of one-shot learning rather than fine-tuning. One-shot learning refers to a process where a model learns to recognize new data with minimal examples during training [2]. Traditional deep learning models typically require large training datasets, but one-shot learning explores the possibility of down-scaling the amount of data needed. Specifically, we evaluated the models using a set of 10 plot-related questions regarding Sir Arthur Conan Doyle's Sherlock Holmes series. We assessed each model's ability to provide correct and coherent answers by comparing their responses to the correct answers which we found from actually reading the book. Then, we analyzed how each model leverages its pre-training strategies to handle complex narrative contexts.

II. Related Work

A. Text Chunking for Contextual Analysis

Handling long contexts is another critical study area for transformer-based models, as their fixed input size frequently limits their applicability to long texts. Lin et al. investigated chunking and sliding window approaches for splitting long documents into digestible chunks while maintaining semantic integrity, which is especially useful for processing prolonged narratives [3]. Sparse attention methods, applied in models such as Longformer and BigBird, increase transformer input capacity by selectively paying to relevant areas of the text, allowing for efficient processing of large documents [4], [5]. These strategies are vital for narrative-driven QA assignments that require understanding story progression and character interactions across multiple situations.

Also, Studies show that chunking texts allows models to focus on meaningful sections, particularly when dealing with long documents that exceed the token limitations of models like BERT. For instance, J. Devlin et al. demonstrated that splitting data into smaller segments preserves semantic integrity [1]. They argued that BERT's bidirectional pre-training, which looks at the context from both the left and right of a word, significantly improves performance on NLP tasks. The paper also highlighted the importance of chunking long texts into smaller sections to fit within model token constraints, ensuring that semantic integrity is maintained [1]. This paper is relevant to our project as it addresses the challenge of processing long texts, such as those in Sir Arthur Conan Doyle's works. By chunking these texts, we ensure they can be fed into different BERT models without losing context, improving text analysis.

B. Advancements in Pre-trained Language Models

Natural language processing (NLP) has greatly benefited from pre-trained language models, which allow models to perform very well on a variety of tasks, such as query response and contextual language understanding. These models take advantage of large-scale training on varied datasets, allowing them to generalize successfully. However, continuous improvements in this area have been fueled by constraints in efficiency, computation demands, and context handling.

For the study, we used three different versions of pre-trained BERT models: basic BERT, RoBERTa (Robustly Optimized BERT Pretraining Approach), and SpanBERT (Span-based BERT). RoBERTa is an optimization of BERT that improves its performance by modifying the pretraining process. SpanBERT, another pretraining approach, focuses on learning better representations of text by predicting spans of text, rather than just individual tokens. According to Liu et al, BERT is shown to be significantly undertrained, and hence, RoBERTa and SpanBERT were proposed to address inefficiencies in model training and inference while ensuring that these models remain small enough to be accessible to those with limited computational resources [6], [7]. RoBERTa improves on the base BERT model by training on more data, using larger batch sizes, and longer sequences, while removing the NSP objective. It also dynamically changes the applied pattern to the masked sentence prediction (MSP) objective. SpanBERT differs from RoBERTa in that, after fine-tuning the base BERT model, it introduces a new training objective, the span-boundary objective (SBO), which trains the model to predict entire masked spans of text based on the context of the tokens at the boundary.

C. One-Shot Learning in NLP

One-shot learning has drawn a lot of interest as a possible remedy for challenges with little labeled data. Prompt-based learning was developed by Schick and Schütze, who showed that performance in one-shot and few-shot scenarios could be greatly improved by matching input forms with pre-training goals [8]. Meta-learning frameworks, as researched by Yin et al., have also shown promise in equipping models with the ability to adapt fast to new tasks with minimum examples, providing a valuable method for low-resource applications [9]. But it's also unclear if pre-trained models can generalize well without task-specific fine-tuning, especially in intricate fields like narrative QA.

III. Methods

A. Data Preparation and Data Engineering

We used three Sherlock Holmes novels from Project Gutenberg as our data source: The Hound of Baskervilles, The Valley of Fear, and A Study in Scarlet [10]–[12]. The text was fetched directly from each book's Project Gutenberg link. Using the headers and footer of a fetched book, we identified the beginning and the end and removed the unnecessary spaces and newline characters to make sure the text only contains the contents from the book.

Models like BERT have a maximum input restriction, meaning that if the input text is long, processing it is necessary so that BERT can handle it well. We wanted to improve work efficiency by removing unnecessary stopwords from the text. This work is done by simply referring to NLTK's stopwords. After removing stopwords, the text has been chunked based on 512 words.

We used the pipeline function provided by the Hugging Face for the efficiency of the work. The pipeline can load the specific BERT model and configure it for specific tasks using pre-processing model inputs, such as a tokenizer or processor. By using this function, we could simplify complex tasks such as tokenization and model configuration. In addition, using the Question-Answering pipeline, each BERT model found the best answer by outputting answers and trust scores from given questions and input text. Among all answers, the answer with the best trust score is selected and given as the model's final output. In addition, by setting up the device to the GPU of Google Colab, it was able to secure a big advantage in run time.

B. Questions

To evaluate our model, we came up with the following 10 questions:

1) Who supports Sherlock Holmes on the investigation?
2) Who is the victim?
3) Who killed the victim?
4) Where does the murder take place?
5) What is the murder weapon?
6) When does Sherlock Holmes begin to unravel the details that lead to solving the murder?
7) How does Sherlock find the murderer?
8) What is the motive for the murder?
9) What is the evidence that led Sherlock Holmes to the murderer?
10) What is the plot twist of the story?

Using these questions, we tested how well each model can find answers to different levels of questions from the text. The first question is a low-level question that can be answered easily to see if the model is working since it is very obvious that Dr. Watson supports Sherlock Holmes in the investigation as well as Lestrade in the Sherlock Holmes series. Questions 2 through 5 are also the questions that can be found fairly easily from the texts although some of the texts that we used included a major plot twist at the end of the book. Questions 6 through 10 are high-level questions to see if a model can answer a hard complex question such as which part Holmes was able to start getting hints about the murderer or what led him to the truth. Through different levels of questions, we wanted to analyze how each model answers the literal questions that use basic facts to deeper questions that require analyzing the characters, plot, and setting.

C. Pretrained Language Models

To compare the performance of the BERT models, we used total of eight different versions of BERT-based models: two versions from basic BERT, two versions of SpanBERT, and four versions of RoBERTa.

First, we use basic BERT models. The bert-base-uncased is designed to be used for mask language modeling or the next sentence prediction and aims to fine-tune the task of making a decision such as question and answer with 110M parameters [1]. The bert-large-cased model has a similar function to the bert-base-uncased but consists of more parameters of 336M [13]. By observing the performance of this basic model, we wanted to use it as an indicator to observe the performance difference with other improved models.

SpanBERT is designed to better represent and predict spans of text. SpanBERT builds on BERT by masking random contiguous spans of text instead of individual tokens. It trains the representations at the span boundaries to predict the full content of the masked span, rather than focusing on individual token representations. This approach enhances performance on span-based tasks, such as question answering [7]. Spanbert-base-cased has 110M parameters, while spanbert-large-cased has 340M parameters. Since SpanBERT is a model that is more optimized for text processing than the basic BERT model, we expected that it would show better performance.

RoBERTa improved the model's performance by adjusting BERT's hyperparameters and the impact of training data size [6]. The roberta-base-squad2 fine-tunes the roberta-base model, which is pre-trained on 11,038 unpublished books, English Wikipedia, and 63M English newspaper articles, using the Stanford Question Answering Dataset (SQuAD). The model has 124M parameters and is trained on question-answer pairs containing unanswerable questions for extractive question-answering tasks. The roberta-large-squad2 functions similarly to roberta-base-squad2, but has a parameter of 354M. We decided to use this model, expecting that it would produce the best results if we used a model optimized for question answering through various books and datasets. We also researched whether there are any significant differences in performance using tinyroberta-squad2 and roberta-base-squad2-distilled which are distilled versions of the previous two models, roberta-base-squad2 and roberta-large-squad2 respectively.

We input texts and questions into eight different BERT-based models to analyze their execution speed and performance based on the answers they produce. To handle the large number of questions efficiently, we used a ThreadPoolExecutor to process multiple questions in parallel. This approach significantly reduces the overall execution time while maintaining the ability to evaluate each model's accuracy and efficiency effectively.

D. Challenges

Because it is best to run all the models together to compare their performance, it was necessary to build an execution environment optimized for all models. Since we were running many models, we also had to consider the execution time. We were able to dramatically reduce the code execution time by using the pipeline function to run the models on the GPU on Google Colab.

To compare the performance of the BERT models, we provided the minimum data processing necessary for the BERT model to accept the text well and configured each model in a form optimized for Q&A through the pipeline. For this, we needed to understand the execution structure of each BERT model.

Furthermore, it was important to develop questions at various levels to check each model's Q&A ability and find the correct answers. We needed to know the plot of each book to some extent to understand the possible impact of the difficulty of each plot and compare the accuracy of the models, so we spent a great deal of time understanding the books' contents.

IV. Experiments

A. Result

Using pipeline, we were able to generate CSV files that contains answer from all questions and run times for each model. In Figure 7, 8, and 9, we can observe that the BERT models provide strings of nonsensical responses, some answers repeating for some of the questions, for all three books. We can also observe that the SpanBERT models do something similar with BERT, however, it generates a varying length sequence of words depending on the books. Finally, we can observe that RoBERTa models give the most succinct answers that are contextually relevant to the questions asked. However, RoBERTa also struggles with the more complex questions just like the other models.

In Fig. 10, the models with negligible run times generate the most sensical answers out of all the models. This is due to models following different training objectives in comparison to the original BERT models. Since NSP is removed from approaches in SpanBERT and RoBERTa, it is fair to assume that it is one of the factors that contribute to these long inference times of BERT, along with undertraining.

B. Analysis

To analyze the generated answers, we created an answer key for each question corresponding to each book. Then, we checked if the answer of the model matches the answer key. If it matches, we put 1 on the spreadsheet and 0 for the ones that did not match. After creating 3 CSV files for each book, we generated graphs of frequencies of 1s, representing how many questions each model answered correctly. Then, we turned it into a graph that shows the number of questions correct for each model as Figure 1, 3, and 5. We also graphed the frequency of correct answers for each question to see how the accuracy changes for different levels of questions.

C. Model Accuracy

Overall, RoBERTa performed very well compared to the other two models. The maximum number of answers that a model guessed right was 3 out of a total of 10 questions. Most of the correct answers were found on the lower-level questions from 1 through 5, as shown in Figure 2, 4, and 6.

In both A Study in Scarlet and The Hound of Baskervilles (Figure 1 and 3), only RoBERTa was able to give some correct answers, finding the victim and murderer from the given text. Particularly in A Study in Scarlet, RoBERTa correctly identifies Lestrade as Holme's one of the supports. A Study in Scarlet was the first book that introduced Lestrade in the

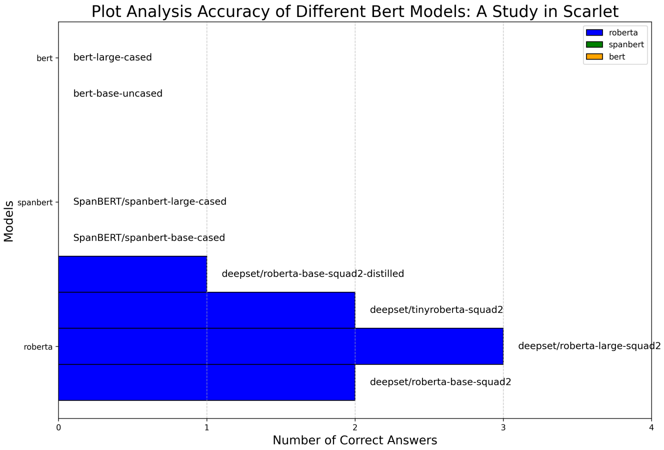


Fig. 1. Plot Analysis Accuracy of Bert Models: A Study in Scarlet

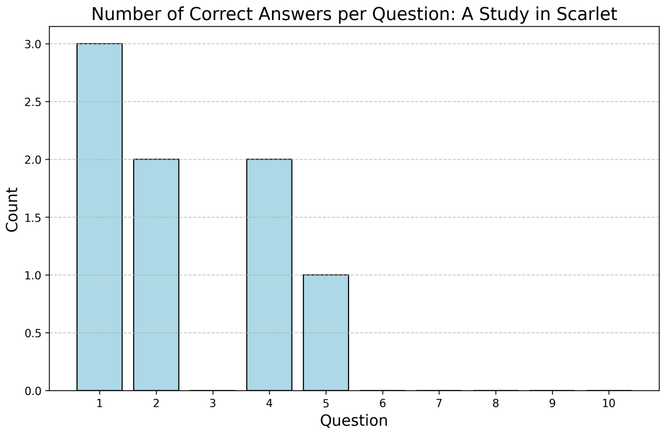


Fig. 2. Correct Answer Frequency of Models per Question: A Study in Scarlet

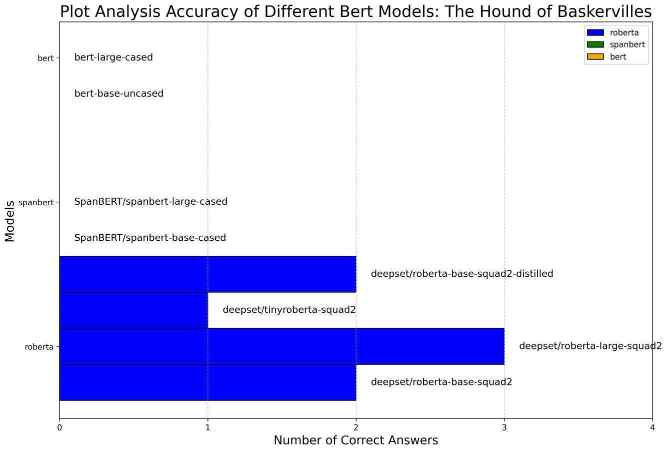


Fig. 3. Plot Analysis Accuracy of Bert Models: The Hound of Baskervilles

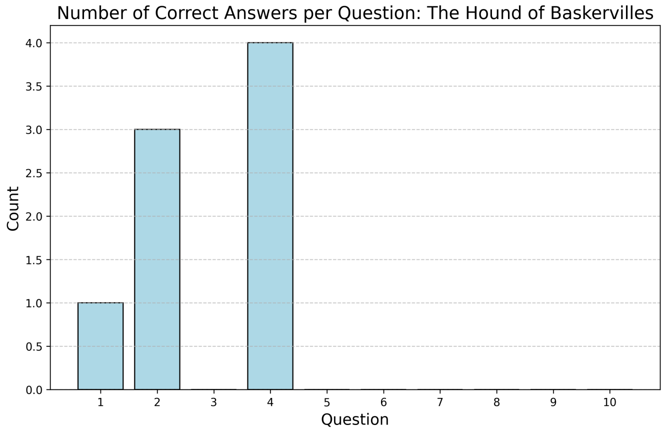


Fig. 4. Correct Answer Frequency of Models per Question: The Hound of Baskervilles

Sherlock Holmes series, and he relied on Holmes to solve the case, being cooperative. This indicates that the RoBERTa was able to identify that one of the key characters Lestrade was a person who supported Holmes during the investigation.

The Valley of Fear had the worst performance, yet the most interesting. The reason behind this is because, at the end of the book, it turns out that the victim and murderer are switched and the person who was known as the victim, John Douglas, was a fake identification. Therefore, there was a big plot twist. Furthermore, after Sherlock found out about the real identification of the body and who killed it, the murderer was killed by someone, who is believed to be James Moriarty. Switched identification of the victim and murderer and the death of the murderer is very confusing, making it harder to answer the question correctly. roberta-base-squad2 and roberta-large-squad2 answered that John Douglas or Birdy Edwards is the victim, but since he was the main murderer that Sherlock was chasing, we considered it as the wrong answer. Another interesting result is that roberta-large-squad2 gave a correct answer to question 10, which is one of the higher-level questions asking about the plot twist. The model gave the name of the real identity of the murderer as the answer to the question asking about the plot twist.

Overall, bert-base-uncased model tends to give the same word to all questions. Bert-large-cased model answers in a list of words. Both Bert models were not able to find correct answers, due to undertraining. RoBERTa models are good at answering lower-level questions. RoBERTa has the best performance since it is trained with a dataset of questions. Among 4 different versions of RoBERTa models, roberta-large-squad2 worked the best because it has the most parameters out of all RoBERTa models. SpanBERT models give answers in sentences with punctuations, which might be due to the fact that it does span masking, trying to predict the words that are in the mask.

V. Conclusion

To conclude, RoBERTa proved to be the most effective model for the question-answering task, surpassing both BERT and SpanBERT in terms of delivering accurate and contextually relevant responses. Its advanced pre-training methodology, which incorporates dynamic masking and utilizes larger-scale

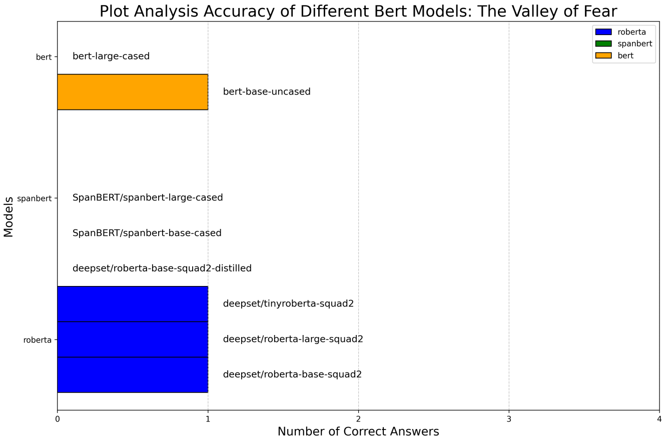


Fig. 5. Plot Analysis Accuracy of Bert Models: The Valley of Fear

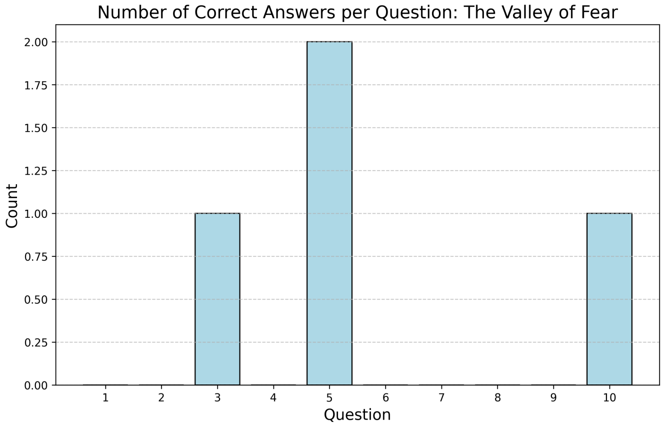


Fig. 6. Correct Answer Frequency of Models per Question: The Valley of Fear

datasets specifically targetting question-answering, played a crucial role in enhancing its performance. RoBERTa's strength in managing complex, context-dependent queries along with huge parameters made it especially well-suited for analyzing the intricate texts of Sir Arthur Conan Doyle's works, having the best performance in one-shot analysis among all models that we examined.

A. Future Work

For future work, we would like to try different ways of tokenization methods to see if they impact performance. We assume that the performance will not drastically change since all models we used already use different fine-tuning and tokenization methods of their own for specific tasks. We also want to examine different types of BERT models such as DistilBERT which is another BERT model that also has a version trained for question-answering.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[2] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[3] B. Y. Lin et al. Few-shot learning with long-tailed data distribution. In *Proceedings of AAAI*, 2021.

[4] I. Beltagy, M. E. Peters, and A. Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[5] M. Zaheer et al. Bigbird: Transformers for longer sequences. In *NeurIPS*, 2020.

[6] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

[7] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans, 2020.

[8] T. Schick and H. Schütze. It's not just size that matters: Prompt engineering for few-shot learning. In *Proceedings of ACL*, 2021.

[9] P. Yin et al. Meta-learning for low-resource nlp tasks. In *Proceedings of EMNLP*, 2020.

[10] Arthur Conan Doyle. The hound of the baskervilles, 1902. Accessed: 2024-12-02.

[11] Arthur Conan Doyle. The valley of fear, 1915. Accessed: 2024-12-02.

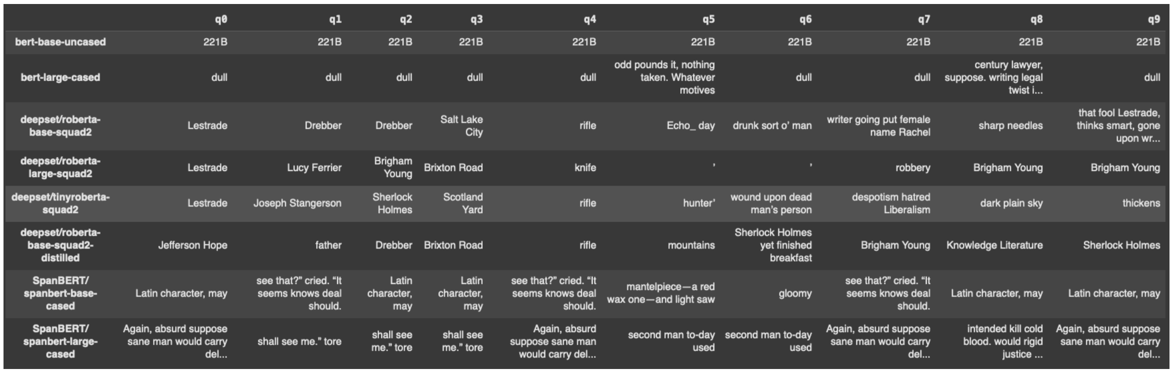
[12] Arthur Conan Doyle. A study in scarlet, 1887. Accessed: 2024-12-02.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

VI. Appendix

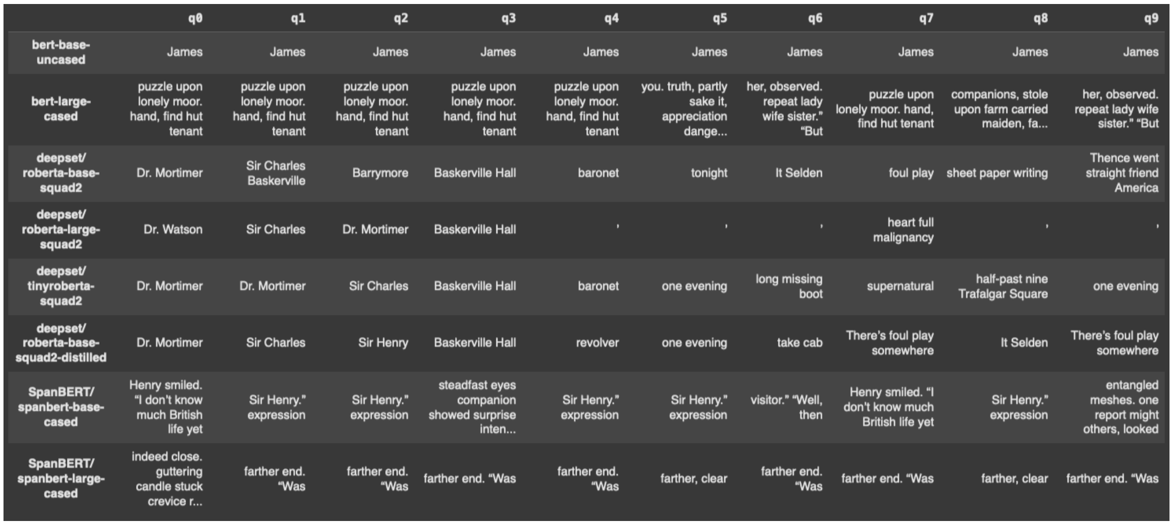
Please see the next two pages for model responses (Figs. 7, 8, 9) and model inference times figure (Fig. 10 for each book given.

Fig. 7. Model responses for A Study in Scarlet



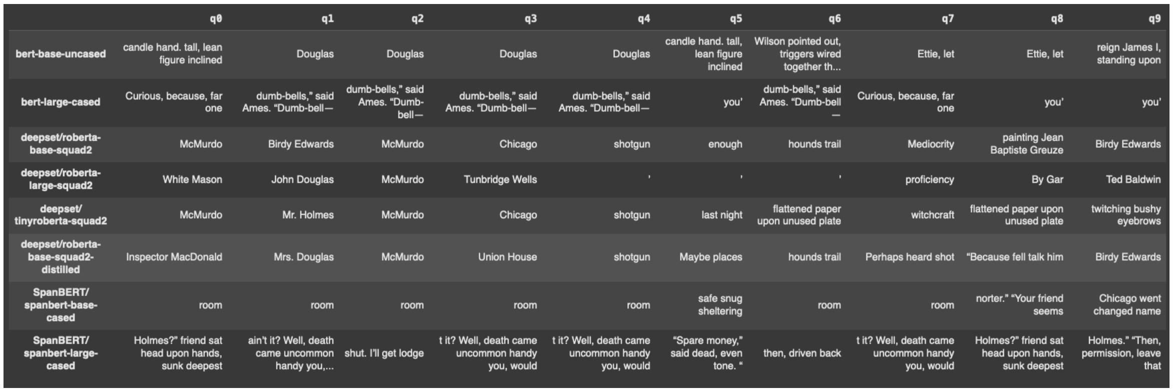
	q0	q1	q2	q3	q4	q5	q6	q7	q8	q9
bert-base-uncased	221B	221B	221B	221B	221B	221B	221B	221B	221B	221B
bert-large-cased	dull	dull	dull	dull	dull	odd pounds it, nothing taken. Whatever motives	dull	dull	century lawyer, suppose. writing legal twist i...	dull
deepset/roberta-base-squad2	Lestrade	Drebber	Drebber	Salt Lake City	rifle	Echo_ day	drunk sort o’ man	writer going put female name Rachel	sharp needles	that fool Lestrade, thinks smart, gone upon wr...
deepset/roberta-large-squad2	Lestrade	Lucy Ferrier	Brigham Young	Brixton Road	knife	’	’	robbery	Brigham Young	Brigham Young
deepset/tinyroberta-squad2	Lestrade	Joseph Stangerson	Sherlock Holmes	Scotland Yard	rifle	hunter’	wound upon dead man’s person	despotism hatred Liberalism	dark plain sky	thickens
deepset/roberta-base-squad2-distilled	Jefferson Hope	father	Drebber	Brixton Road	rifle	mountains	Sherlock Holmes yet finished breakfast	Brigham Young	Knowledge Literature	Sherlock Holmes
SpanBERT/spanbert-base-cased	Latin character, may	see that?” cried. “It seems knows deal should.	Latin character, may	Latin character, may	see that?” cried. “It seems knows deal should.	mantelpiece—a red wax one—and light saw	gloomy	see that?” cried. “It seems knows deal should.	Latin character, may	Latin character, may
SpanBERT/spanbert-large-cased	Again, absurd suppose sane man would carry del...	shall see me.” tore	shall see me.” tore	shall see me.” tore	Again, absurd suppose sane man would carry del...	second man to-day used	second man to-day used	Again, absurd suppose sane man would carry del...	intended kill cold blood. would rigid justice ...	Again, absurd suppose sane man would carry del...

Fig. 8. Model responses for The Hound of Baskervilles



	q0	q1	q2	q3	q4	q5	q6	q7	q8	q9
bert-base-uncased	James	James	James	James	James	James	James	James	James	James
bert-large-cased	puzzle upon lonely moor. hand, find hut tenant	puzzle upon lonely moor. hand, find hut tenant	puzzle upon lonely moor. hand, find hut tenant	puzzle upon lonely moor. hand, find hut tenant	puzzle upon lonely moor. hand, find hut tenant	you. truth, partly sake it, appreciation dange...	her, observed. repeat lady wife sister.” “But	puzzle upon lonely moor. hand, find hut tenant	companions, stole upon farm carried maiden, fa...	her, observed. repeat lady wife sister.” “But
deepset/roberta-base-squad2	Dr. Mortimer	Sir Charles Baskerville	Barrymore	Baskerville Hall	baronet	tonight	It Selden	foul play	sheet paper writing	Thence went straight friend America
deepset/roberta-large-squad2	Dr. Watson	Sir Charles	Dr. Mortimer	Baskerville Hall	’	’	’	heart full malignancy	’	’
deepset/tinyroberta-squad2	Dr. Mortimer	Dr. Mortimer	Sir Charles	Baskerville Hall	baronet	one evening	long missing boot	supernatural	half-past nine Trafalgar Square	one evening
deepset/roberta-base-squad2-distilled	Dr. Mortimer	Sir Charles	Sir Henry	Baskerville Hall	revolver	one evening	take cab	There’s foul play somewhere	It Selden	There’s foul play somewhere
SpanBERT/spanbert-base-cased	Henry smiled. “I don’t know much British life yet	Sir Henry.” expression	Sir Henry.” expression	steadfast eyes companion showed surprise inten...	Sir Henry.” expression	Sir Henry.” expression	visitor.” “Well, then	Henry smiled. “I don’t know much British life yet	Sir Henry.” expression	entangled meshes. one report might others, looked
SpanBERT/spanbert-large-cased	indeed close. guttering candle stuck crevice r...	farther end. “Was	farther end. “Was	farther end. “Was	farther end. “Was	farther, clear	farther end. “Was	farther end. “Was	farther, clear	farther end. “Was

Fig. 9. Model responses for The Valley of Fear



	q0	q1	q2	q3	q4	q5	q6	q7	q8	q9
bert-base-uncased	candle hand. tall, lean figure inclined	Douglas	Douglas	Douglas	Douglas	candle hand. tall, lean figure inclined	Wilson pointed out, triggers wired together th...	Ettie, let	Ettie, let	reign James I, standing upon
bert-large-cased	Curious, because, far one	dumb-bells,” said Ames. “Dumb-bell—	dumb-bells,” said Ames. “Dumb-bell—	dumb-bells,” said Ames. “Dumb-bell—	dumb-bells,” said Ames. “Dumb-bell—	you’	dumb-bells,” said Ames. “Dumb-bell —	Curious, because, far one	you’	you’
deepset/roberta-base-squad2	McMurdo	Birdy Edwards	McMurdo	Chicago	shotgun	enough	hounds trail	Mediocrity	painting Jean Baptiste Greuze	Birdy Edwards
deepset/roberta-large-squad2	White Mason	John Douglas	McMurdo	Tunbridge Wells	’	’	’	proficiency	By Gar	Ted Baldwin
deepset/tinyroberta-squad2	McMurdo	Mr. Holmes	McMurdo	Chicago	shotgun	last night	flattened paper upon unused plate	witchcraft	flattened paper upon unused plate	twitching bushy eyebrows
deepset/roberta-base-squad2-distilled	Inspector MacDonald	Mrs. Douglas	McMurdo	Union House	shotgun	Maybe places	hounds trail	Perhaps heard shot	“Because fell talk him	Birdy Edwards
SpanBERT/spanbert-base-cased	room	room	room	room	room	safe snug sheltering	room	room	norter.” “Your friend seems	Chicago went changed name
SpanBERT/spanbert-large-cased	Holmes?” friend sat head upon hands, sunk deepest	ain’t it? Well, death came uncommon handy you,...	shut. I’ll get lodge	t it? Well, death came uncommon handy you, would	t it? Well, death came uncommon handy you, would	“Spare money,” said dead, even tone. “	then, driven back	t it? Well, death came uncommon handy you, would	Holmes?” friend sat head upon hands, sunk deepest	Holmes.” “Then, permission, leave that

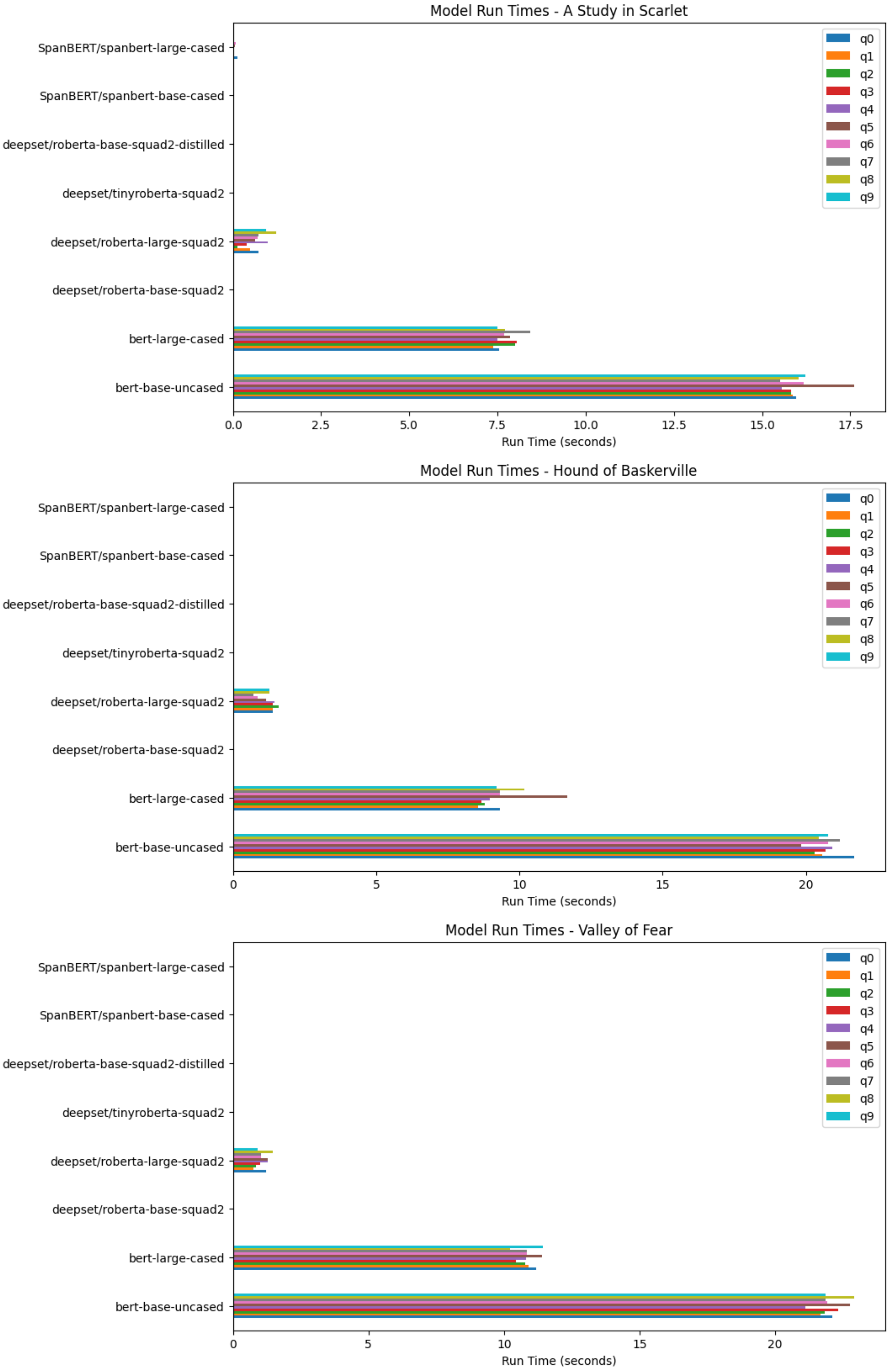


Fig. 10. Inference run times for each model to generate an answer to ten questions. No times are shown for five models due to negligible run times. For all three books, the models have similar run times. See Figs. 7, 8, and 9.