Title

QUESTION ANSWERING USING DEEP LEARNING MODEL

Introduction

Introduction: Deep Learning Approaches for Question Answering System
Question Answering (QA) systems have greatly improved with deep learning, allowing machines to understand and answer natural language queries. In this study, we focus on two powerful models: BERT (Bidirectional Encoder Representations from Transformers) and LSTM (Long Short-Term Memory).
BERT, a transformer-based model, processes text bidirectionally, making it effective for extractive QA tasks by directly retrieving answers from the context. LSTM, a type of recurrent neural network, captures the sequence of words, which is useful for understanding longer passages

Method

1. BERT Model for Extractive QA
   - Preprocessing: Tokenize the context (passage) and question, adding special tokens.
   - Model Setup: Fine-tune a pre-trained BERT model on a QA dataset like SQuAD.
   - Answer Extraction: BERT predicts the start and end positions of the answer within the context.
   - Training: Fine-tune using the loss between predicted and true answer spans.
2. LSTM Model for Sequential QA
   - Preprocessing: Convert words into embeddings and pad sequences.
   - Model Setup: Use LSTM layers to capture the sequence of words, with attention to focus on relevant parts of the text.
   - Answer Generation: Predict the answer span or generate an answer from the context.
   - Training: Train using a loss function like cross-entropy to optimize the model.

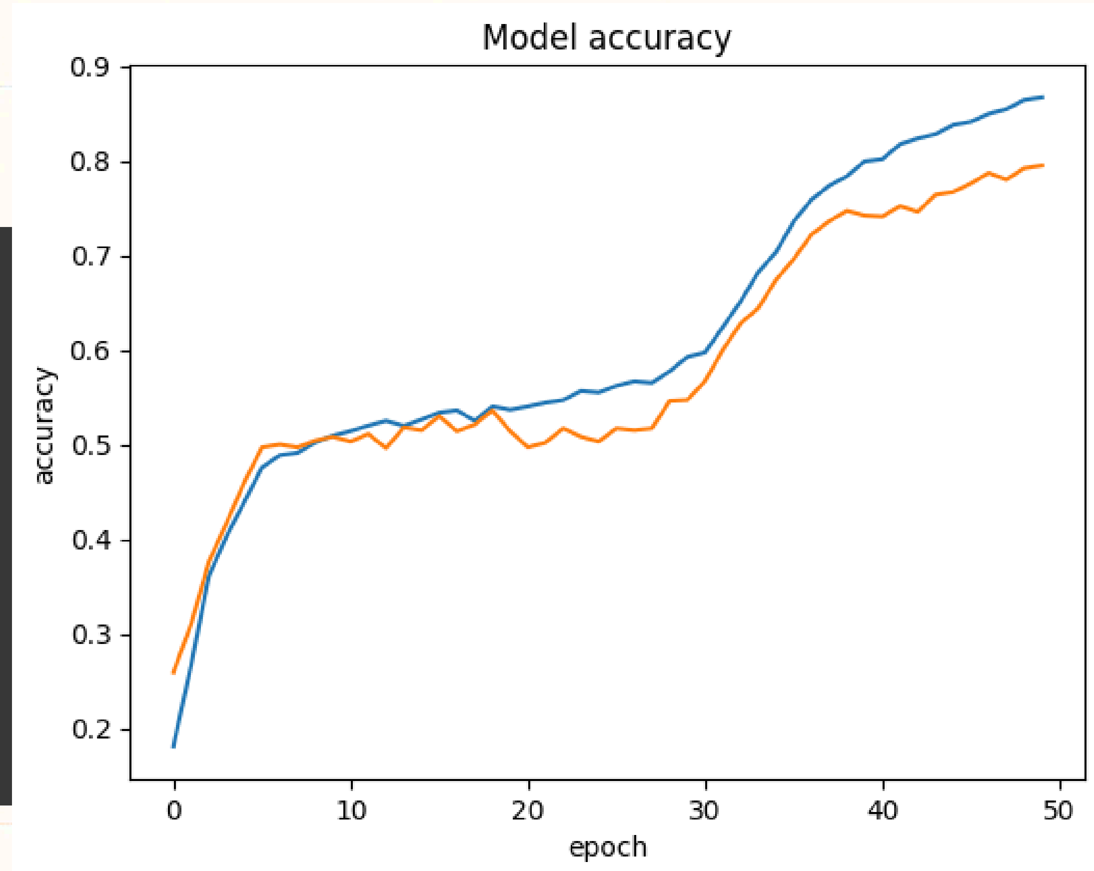
Result

```
# Step 4: Load the trained model into the question-answering pipeline
question_answerer = pipeline("question-answering", model=model_checkpoint, tokenizer=model_checkpoint)

# Step 5: Provide inputs (context and question) and get the answer
context = input("Enter the context: ")
question = input("Enter the question: ")

try:
    # Step 6: Generate the answer
    answer = question_answerer(question=question, context=context)['answer']
    print("Answer:", answer)
except Exception as e:
    print("Error:", e)
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", for
Enter the context: BeyoncÃ© Giselle Knowles-Carter (/biËˈjÉ'nseÉª/ bee-YON-say) (born September 4, 1981) is a
Enter the question: When did Beyonce start becoming popular?
Answer: late 1990s
```



References

[1]J. Weston, A. Bordes, S. Chopra, and T. Mikolov. Towards AI-complete question answering: A set of prerequisite toy tasks. arXiv preprint: 1502.05698, 2015.
[2]Tait Larson, Johnson(Heng) Gong, Josh Daniel "Providing a Simple Question Answering System By Mapping Questions to Questions.",Technical report, Department of Computer Science,Stanford University, 2006.
[3]Leon Derczynski, Alan Ritter, Sam Clark, Kalina Bontcheva,(2013) "Twitter Part-of-Speech Tagging for All: Overcoming Sparse and Noisy Data " Proceedings of Recent Advances in Natural Language Processing, pages 198–206 .
[4]J. Ramos(2003) "Using TF-IDF to Determine Word Relevance in Document Queries"Technical report, Department of Computer Science, Rutgers University, 2003.
[5]Sanglap Sarkar, Venkateshwar Rao, Baala Mithra SM, Subrahmanya VRK Rao(2015)"NLP Algorithm Based Question and Answering System", Proceedings of 2015 Seventh International Conference on Computational Intelligence, Modelling and SimulationPages 97-101 .

Conclusion

In question answering, LSTM models with attention work better than regular LSTMs and even BERT in some cases. Regular LSTMs struggle to focus on key parts of long text, but attention helps by highlighting important words, making the model more accurate. While BERT is powerful, it's more complex and resource-heavy. LSTM with attention strikes a balance—it performs well by focusing on relevant information, while being faster and less resource-demanding than BERT, especially for tasks with long or structured text.