Arabic Question Answering

Overview:

The primary objective of this research is to develop Arabic Question Answering systems retrieving answers from an input context in response to user queries.

The challenges inherent in Arabic QA include:

- Semantic Complexity: Arabic language exhibits rich semantics and complex grammatical structures, making it challenging to accurately interpret user queries and retrieve relevant answers.
- Limited Resources: Compared to English, Arabic has fewer annotated datasets and pre-trained language models, posing challenges for training and fine-tuning QA systems.
- Morphological Variability: Arabic words undergo significant morphological changes based on context, which adds complexity to information retrieval and understanding.

Data Set:

Arabic Reading Comprehension Dataset (ARCD)

Methodology:

Machine learning approach

1. Data Preprocessing

Before implementing the machine learning approach for Arabic question answering, a series of preprocessing steps were conducted to clean and prepare the data. The preprocessing steps included:

- Text Cleaning: The raw textual data was cleaned to remove noise, such as special characters, punctuation marks, and non-Arabic characters. This step aimed to ensure that the data is in a clean and standardized format suitable for further processing.

- Tokenization: The cleaned text was tokenized into individual words or tokens, which are the basic units of analysis for the machine learning model.
- Stemming

2. Model Implementation

A simple machine learning model was implemented to find answers to user queries based on the similarity between the question and the sentences in the dataset. The implementation involved the following steps:

- Data Representation: Each sentence in the dataset was represented as a numerical vector using the Bag-of-Words (BoW) technique. These representations capture the semantic content of the sentences and enable similarity comparison with user queries.
- Sentence Tokenization: To facilitate similarity comparison, full sentence tokenization was used. This step ensured that the model could identify and compare individual sentences with the user query.
- Similarity Calculation: The similarity between the user query and each sentence in the dataset was calculated using cosine similarity. This metric quantifies the degree of overlap or similarity between two vectors representing the query and every sentence in the context.
- Answer Retrieval: The sentence with the highest similarity score to the user query was selected as the most relevant answer. This approach assumes that sentences with higher similarity scores are more likely to contain the correct answer to the user query.

3. Evaluation

The performance of the implemented machine learning model was evaluated using evaluation metrics such as Accuracy, Exact Match, and F1-score. Additionally, a qualitative assessment of the model's output was conducted to gauge its effectiveness in retrieving accurate answers to user queries.

Performance Metrics: I implemented the method to evaluate the model and calculate accuracy, F1 score, and exact match. The model's overall accuracy was found to be 2%. While low, it provides a baseline understanding of the model's predictive capability relative to the ground truth answers. The F1 score was also observed to be 2%. The exact match metric, representing the percentage of queries for which the model produced an exact match with the ground truth answer, was found to be 0.0%. This suggests that the model struggled to precisely identify the correct answers in response to user queries, as it depended on only the similarity between the question and every sentence in the context, and returned only one sentence which has the maximum similarity value.

4. Limitations

While the implemented machine learning model offers a straightforward approach to Arabic question answering, it comes with several inherent limitations that warrant consideration such as:

- Single-Sentence Context: The model's reliance on the result from a single sentence limits its ability to capture broader context and nuanced information that may span multiple sentences or paragraphs.
- Lack of Discourse Understanding: By focusing solely on individual sentences, the model may overlook crucial discourse markers, transitions, or implicit connections between sentences within a text. As a result, it may fail to capture the full meaning and coherence of the information presented in longer passages.
- Contextual Ambiguity: Arabic language exhibits rich contextual nuances and ambiguity, which pose challenges for precise question answering. Ambiguities in word meanings, syntactic structures, and cultural references can lead to inaccuracies or misinterpretations, particularly when relying on isolated sentences for answer retrieval.

Transformer approach

1. Pretrained Transformer Model Selection

After conducting a thorough search, an open-source transformer model available on the Hugging Face model hub was identified and selected for Arabic question-answering tasks and it was already trained on my dataset, so I could use it directly. Here is the model I have used. The choice of a pre-trained transformer model offers several advantages, including access to state-of-the-art language representations, fine-tuning capabilities, and compatibility with diverse downstream tasks.

2. Data Preparation

The dataset was prepared by formatting the questions and corresponding passages or documents in a suitable format for input to the transformer model. The data was tokenized into subword units using the same tokenizer associated with the pretrained transformer model to ensure compatibility and consistency in input representations.

3. Evaluation Procedure

The fine-tuned transformer model was evaluated using standard evaluation metrics to assess its performance in Arabic question answering. The evaluation involved the following steps:

- Prediction Generation: The fine-tuned model was used to generate predictions for a test set comprising unseen question-answer pairs. Given a question, the model produced predicted answers.
- Metric Calculation: To assess the model's performance accurately, multiple evaluation metrics were computed, encompassing Accuracy, F1 Score, and Exact Match. These metrics serve to gauge the model's efficacy in accurately identifying and retrieving the correct answers to user queries. Utilizing the "evaluate" library in Python facilitated the computation process. Notably, the F1 score yielded a result of 53.64%, indicating a moderate level of precision and recall balance. Moreover, the exact match metric, denoting the proportion of queries where the model's prediction exactly matched the ground truth answer, reached 26.0%. Notably, this performance surpasses that of the machine learning model previously evaluated.

4. Limitations:

One of the primary constraints encountered in evaluating the Transformer model stemmed from resource limitations. Due to constraints in computational resources, particularly in terms of processing power and memory availability, I was unable to execute the model for evaluation across the entirety of the dataset. As a result, the evaluation process was constrained to a subset of the evaluation data, encompassing only 100 records.

This restricted evaluation scope may have implications for the model's overall performance assessment, potentially limiting the generalizability of the results. Furthermore, the selected subset may not fully represent the diversity and complexity present in the complete dataset, thereby affecting the reliability and comprehensiveness of the evaluation outcomes. because of this limitation, I was forced to use the same subset of the validation dataset to evaluate the model to compare them

LLM Fine-Tuning approach

1. Pre-trained LLM Selection:

A suitable pre-trained Large Language Model (LLM) from the Hugging Face model hub for the Arabic question-answering task. The selection criteria included considerations such as model architecture, pre-training data, language compatibility, and performance on similar tasks. I chose the AceGPT 7b model, AceGPT is a fully fine-tuned generative text model collection based on LlaMA2, particularly in the Arabic language domain.

2. QLORA :

QLoRA represents a more memory-efficient iteration of LoRA. QLoRA takes LoRA a step further by also quantizing the weights of the LoRA adapters (smaller matrices) to lower precision (e.g., 4-bit instead of 8-bit). This further reduces the memory footprint and storage requirements. In QLoRA, the pre-trained model is loaded into GPU memory with quantized 4-bit weights, in contrast to the 8-bit used in LoRA. Despite this reduction in bit precision, QLoRA maintains a comparable level of effectiveness to LoRA.

The QLORA facilitates the fine-tuning of pre-trained LLMs on custom datasets, enabling domain-specific adaptation and optimization for Arabic question answering.

3. Dataset Preparation:

The dataset for Arabic question answering was prepared by curating and preprocessing the raw text data. This involved tasks such as data cleaning, tokenization, and formatting the data into a suitable input format compatible with the QLORA framework.

4. Fine-tuning Process:

Using the QLORA, Fine-tuning involved creating LoRA adapter, After LoRA fine-tuning for our use case, the outcome is an unchanged original LLM and the emergence of a considerably smaller “LoRA adapter,” often representing a single-digit percentage of the original LLM size (in MBs rather than GBs), in our case thee percentage of trainable model parameters: 0.18% of the original model.

During inference, the LoRA adapter must be combined with its original LLM. The advantage lies in the ability of many LoRA adapters to reuse the original LLM, thereby reducing overall memory requirements when handling multiple tasks and use cases.

5. Training and Validation:

The fine-tuning process included training the LLM on a portion of the dataset while reserving another portion for validation. This allowed for monitoring the model's performance during training and tuning hyperparameters to optimize its effectiveness in Arabic question answering. The training loss was 0.91 and the validation loss was 1.033347.

6. Evaluation:

The evaluation phase involved utilizing the fine-tuned Large Language Model (LLM) to predict answers for the evaluation dataset. This process facilitated the

comparison between the true answers from the dataset and the corresponding predicted answers generated by the model (results). The evaluation revealed the performance of the fine-tuned LLM through a direct comparison between the true answers and the model's predictions.

7. Limitations:

Resource constraints posed significant limitations during the fine-tuning process of the Large Language Model (LLM). The primary constraints revolved around computational resources, particularly in terms of processing power, memory availability, and GPU. As a result, several limitations were encountered:

- Limited Training Steps: Due to resource constraints, the fine-tuning process of the LLM was constrained to a maximum of five training steps. This limitation translates to approximately 0.03 epochs, significantly restricting the model's exposure to the training data and potentially limiting its ability to capture complex linguistic patterns and nuances.
- Low Rank for LoRA Adapter: To mitigate the computational burden and reduce the number of trainable parameters, a low rank was assigned to the LoRA (Low-Rank Adapter) adapter during the fine-tuning process. While this approach helped to reduce the number of training parameters and therefore alleviate resource constraints, it may have compromised the model's representational capacity and learning efficiency, particularly for tasks requiring intricate semantic understanding and reasoning.

8. Future Directions:

Based on the evaluation outcomes and insights gained, future directions for research and development were identified. These may include refining the fine-tuning process, exploring additional training data sources, or investigating ensemble techniques to further enhance the LLM's performance, like enhancing the prompt used in the training and addressing specific challenges encountered during evaluation.

conclusion

In conclusion, Large Language Models (LLMs) exhibit significant potential for Arabic question answering, contingent upon adequate computational resources. Despite resource constraints, the fine-tuned LLM demonstrated performance in generating

accurate answers and its power to give better results by simple enhancing like enhancing prompt. The limitations underscore the importance of resource availability for maximizing LLM capabilities. Moving forward, investments in infrastructure and optimization techniques are crucial for harnessing the full potential of LLMs in Arabic language tasks, thereby enhancing communication and information access.