Experimental Protocols:
Effect of syntactic features for multi-hop QA

Sasi Kanth Ala
sasikanth.ala@gmail.com

1 Hypotheses

Choosing syntactically related examples for in-context learning and syntactical decomposition of complex question will improve performance of retrieval augmented in-context learning multi-hop QA systems.

2 Data

I am using HotpotQA(Yang et al., 2018) and BeerQA(Qi et al., 2021) to evaluate my hypothesis.

HotpotQA has 113k question-answer pairs on Wikipedia where the questions require multiple supporting passages to answer. HotpotQA is collected by crowd-sourcing where crowd workers are shown multiple supporting context documents and are asked to come up with question which require information from all the documents to answer. The questions usually fall into two categories *bridge questions* and *comparison questions*. To answer bridge questions one needs the passage from the previous hop to retrieve passages required for the next hop. In contrast the passages for comparison questions can be retrieved in parallel. We use train set for selecting examples for few-shot in-context learning and use dev set for evaluation.

BeerQA combines data from 1-hop SquadQA (Rajpurkar et al., 2016), 2-hop HotpotQA manually created 3+ hop questions. The distribution of questions is shown in 1. Unfortunately we don't have 3+ hop questions in train or dev set. Still this is a good benchmark to see if our system handle questions without prior knowledge about question complexity. We use train set for selecting examples for few-shot in-context learning and use dev set for evaluation.

3 Metrics

For evaluating the models I choose Exact Match and F_1. Exact match measures the percentage of predictions that match at last one of expected answer exactly ignoring case, punctuation and articles. F_1 (macro averaged) measures the overlap between the prediction and expected answer. F_1 score for each item is calculated by treating prediction and expected answer as bag of words and taking the maximum of F_1 across all expected answers.

	SQuAD Open	HotpotQA	3+ Hop	Total
Train	59,285	74,758	0	134,043
Dev	8,132	5,989	0	14,121
Test	8,424	5,978	530	14,932
Total	75,841	86,725	530	163,096

Figure 1: Distribution of questions from sources.(Qi et al., 2021)

4 Models

I will be using gpt-3.5-turbo as LM and ColBERTv2 with Wikipedia index.

Vanilla LM (Khattab et al., 2022): This baseline represents the few-shot in-context learning with out retriever. I will randomly sample 16 demonstrations from the train set as examples. The validation question-answers come from Wikipedia and LMs are probably trained on Wikipedia.

multi-hop with random examples: This model uses Demonstrate Search Predict pipeline as specified in DSP paper (Khattab et al., 2022).

multi-hop with syntactically related examples: This model will choose examples for in-context learning based on syntactic similarity to the question. For extracting syntactic information I am planning to parse the sentence using Berkeley Neural Parser(Kitaev and Klein, 2018). For measuring similarity between parse trees I am planning to compute tree edit distance(Zhang and Shasha).

multi-hop with syntactic query decomposition: This model will use Question Decomposition Meaning Representation (QDMR)(Wolfson et al.,

2020) to generate ordered list of steps that are necessary for answering a question. The ordered list encodes dependency information between the step and makes it easy to parallelize interactions between LM and RM.

multi-hop with full syntactic features: This model combines the two previous models. It uses syntactic information for examples and question decomposition.

Table 1 shows the experimental results when finished.

5 General Reasoning

The choice of examples provided for in-context learning has big effect on its performance. Selecting examples based on semantic similarity(Liu et al., 2022) has significant improvements for question answering. My hypothesis is choice of examples based on the question's syntactic form will give similar improvements in question answering performance.

Recently state of the art systems use LM prompting to implement many components of their systems(Gao et al., 2022). LM models are expensive and slow. They are also stateless. Too many back and forth between LM and RM adds to the cost and query latency. It will be useful to investigate specialized components which are faster and cheaper and which will make the whole system competitive with state of the art systems which use LM prompting extensively. Complex question answering requires decomposing the question into constituent parts which can be handled by its own specialized component instead of using LM prompting. BREAKRC(Wolfson et al., 2020) uses high level QMDR structures for answering open-domain multi-hop questions. This semi structured decomposition has potential to be converted to database queries and included as context to LM.

6 Summary of Progress

I played with sample DSP program and am able to make small changes. I was also able to get sample parsing program, tree edit distance and QMDR working. In coming days I will be joining these pieces to implement the models.

References

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2022. Rarr: Researching and revising what language models say, using language models.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Peng Qi, Haejun Lee, Tg Sido, and Christopher Manning. 2021. Answering open-domain questions of varying reasoning steps from text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3599–3614, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Tomer Wolfson, Mor Geva, Ankit Gupta, Matt Gardner, Yoav Goldberg, Daniel Deutch, and Jonathan Berant. 2020. Break it down: A question understanding benchmark. *Transactions of the Association for Computational Linguistics*, 8:183–198.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Kaizhong Zhang and Dennis Shasha. Simple fast algorithms for the editing distance between trees and related problems. 18(6):1245–1262.

	HotpotQA		**BeerQA**	
	EM	F_1	EM	F_1
Vanilla LM	–	–	–	–
multi-hop with random examples	–	–	–	–
multi-hop with syntactically related examples	–	–	–	–
multi-hop with syntactic query decomposition	–	–	–	–
multi-hop with full syntactic features	**–**	**–**	**–**	**–**

Table 1: Development results for syntactically enhanced DSP program against baselines vanilla LM and multi-hop with random examples. It also breaks down the effect of syntactic features in example selection and complex query decomposition.