266\_Project\_Proposal

**Project Proposal - Improving Fairytale Comprehension Through Question Specific Experts**

Dylan Sivori & Danielle Yoseloff

Natural Language Processing - 266  
Summer 2025

{dylan.sivori , dyoseloff} @berkeley.edu

For this project, we plan to create a new question answering algorithm that combines Retrieval-Augmented Generation (RAG), Retrieval-Augmented Thinking (RAT) and Mixture of Experts (MoE) architectures to the FairyTaleQA dataset. This novel approach will allow for different expert models to handle different question types while also leveraging these retrieval methods to have improved reasoning and reduced hallucination.

This is an important area of study as hallucination and complex reasoning are two areas that language models continue to struggle with today and will need to be improved upon. This will be a challenging study because the FairyTaleQA dataset contains seven different question types (character, setting, action, feeling, casual relationships, outcome resolution, and prediction) with answers that can be categorized as explicit or implicit requiring different forms of reasoning. This is also what makes this dataset an ideal candidate for experimenting with the MoE structure in a new domain, allowing for expert models in different question types, and improved reasoning combining RAT and RAG retrieval methods.

The primary dataset that we plan to use is the newly developed FairyTaleQA dataset which consists of 10,580 questions across 278 children stories. This dataset was uniquely developed by real educational experts rather than generated to ensure authentic grade school narrative comprehension. Another dataset that we are considering using is the NarrativeQA dataset since it has significantly longer story contexts and could provide an additional challenge in question answering for our architecture to handle.

We are planning to leverage Llama, GPT-2, BERT, and BART based models as well as a domain specific model fine-tuned on the NarrativeQA dataset in conjunction with our new architecture to enable the answer generation. We plan to evaluate the performance of this architecture using the ROUGE-L F1 score that is reported in the FairytaleQA dataset paper.

References:

1. Fantastic Questions and Where to Find Them: FairytaleQA -- An Authentic Dataset for Narrative Comprehension (2022) <https://arxiv.org/abs/2203.13947>
   1. This paper introduces the [FairytaleQA](https://huggingface.co/datasets/WorkInTheDark/FairytaleQA)dataset as an authentic reading comprehension evaluation to the popular [NarrativeQA](https://huggingface.co/datasets/deepmind/narrativeqa)
2. SyllabusQA: A Course Logistics Question Answering Dataset (2024) <https://arxiv.org/abs/2403.14666>
   1. This paper develops a qa dataset with distinct question types focusing on pursuing factual accuracy. They find success with a LLaMA with RAG and CoT which supports our idea of using RAT on a qa dataset with distinct question types
3. A Survey on Mixture of Experts in Large Language Models (2024) <https://arxiv.org/abs/2407.06204>
   1. This paper thoroughly details the MoE methodology and applications
4. RAT: Retrieval Augmented Thoughts Elicit Context-Aware Reasoning in Long-Horizon Generation (2024) <https://arxiv.org/abs/2403.05313v1>
   1. This paper proposes RAT as improvement to qa problems comparing against chain-of-thought (CoT) and RAG models

Overall I like the choice of task, I think story reading comprehension question answering is a good challenging type of problem for this class, and the dataset you've chosen seems like a good one.

Just in the organization of your write-up here, you've put the cart before the horse a bit. It's no problem for the proposal. But for the final paper, make sure that you start with an actual statement of what problem you're focusing on and why. You'll want a background section that actually talks about why it's valuable in some specific real world use case(s) to automate story-based question answering, who would be the intended user of this type of model. You also want to talk broadly about why this is challenging as a language problem, what aspects are difficult for models to learn. That's what motivates you to then think about different potential improvements that you could make.

Regarding both this and the previous paragraph: I like some of your ideas here, but I'm not sure about others. I'm not actually sure if any retrieval-augmented approaches make sense for this task, I don't actually know what else you'd be trying to retrieve. Retrieval means that there's some additional supplemental information that you think would help your model do better, than what's already included with each example.

In your case, you want the model to focus on answering the given question based on the given story. You don't want to add some other less-related stories in as well, that would only confuse the model. I think you probably also don't need some other factual info about fairytales, in order to answer each question, or if you do I'm not sure that you have separately labeled excerpts from some wiki of fairytale narrative devices or tropes that paired with the types of questions in your dataset, which is what you'd need to train a model to retrieve that additional reference info.

However, the mixture of experts idea could be interesting. It sounds like you're suggesting that you might train a separate "expert" for each different type of question, since your dataset is annotated with different categories of questions. That's an interesting idea, and it could help. For that experiment, I assume you would also need a triage classifier that first predicts which type of question each one is, in order to send it to the appropriate question-type model? Unless you intend to have the different question-type models vote on their answers, and then you need a way to decide how to weight or aggregate them.

That's a lot of methodology, and you'd want to compare it to a more standard model in which you don't have that added machinery (i.e. one model for all types of questions combined). You'd probably want to do both of those variants for several pre-trained models as well, to compare different options you have for the starting model architecture and pre-training, plus whether/how much added value you get from using the different types of questions to train variants of each model or not.

So I wouldn't try to do some sort of RAG in addition to all of that (which is good, since I'm not seeing how it'll work here, though I could be missing something). With all of this, you'll want to carefully compare each methodological option, starting with the different pre-trained models, fine-tuning each in a standard way for all question types, then fine-tuning different versions of them for your mixture-of-experts variant, and evaluating and comparing each of those options to see what specific choices work best in what ways and on what types of questions.

This sounds like a good choice of dataset, I gather this is the one that has different types of questions annotated? That will be useful both if you choose to experiment with ensemble model approaches that involve splitting the training data by question type. Either way, it will also help to have those different question types in your test set, because it will let you break down your results by question type to see which models do better on which types of questions (which you should discuss and interpret as to why you think that might be happening, based on the specific model differences).

If you do that, you'll want to provide a reason for doing so. (In general we always want to see some justification for the choices you made, because that's an important part of the process; you can't try everything.) Citing some limitation in your first dataset can be a reason, e.g. if you have a fairly small number of examples in certain particularly challenging question types, and this second dataset has more of those difficult reasoning type questions. That way you have a hypothesis: you can see if the scores go up most in the part of the test set that represents those harder reasoning types of questions, when you added the second training set, than when you didn't.

These seem like reasonable pre-trained models to choose, though they have important differences that you'll want to make sure are all appropriate for your task. In particular, BERT can only be used for classification (categorical predictions, extractive question answering) but not for text generation. I'm not sure if your main dataset can be approached extractive, since it sounds like at least some of the answers require reasoning, and you were concerned about hallucination (which only happens when a text generation model is really being used abstractively).

As noted in the last thread above, whatever models you choose, we want to see a justification for each. You can cite other papers that did very similar tasks, if they found certain models to work well, that's a good reason to start with those models yourselves. It's also good to mention any differences in architecture (decoder-only vs encoder-decoder) and in pre-training (pre-training task structure or the corpus/types of text the model was pre-trained on) and why you think each one might be relevant to your particular problem and domain in different ways. Those also represent hypotheses that will help you interpret your results and see if what you think might make a model work well plays out in how well they do.

Along those lines, you mentioned domain-specific models. Those would also be good choices. Like everything you choose, it's good to compare doing it vs not, in as controlled a way as you can. So it will be good to have e.g. a general domain Llama or BART model compared to one that has already been trained on fiction or fairytales (or whatever similar corpora you might find a model version trained on).

A few more notes about specific pre-trained models: the Llama models are good but tend to be quite large (though you can try the smallest ones available). Computing resources will be a big constraint, so try to stick to the smallest versions of options that you think will be a good fit. You didn't mention T5, but those are also very reasonable choices for this type of task (that includes FLAN-T5, which is trained on instruction prompts like Llama is).

In the other direction, GPT-2 is actually a fairly old model, relative to the others in the list. You might try OPT instead, which was designed to be similar to GPT-3 (so the next generation, though made by Facebook because OpenAI stopped opens-sourcing their models at that point).

ROUGE-L F1 is a good choice, but you will actually want more than one evaluation metric. It's quite common to include all of the standard ROUGE metrics (ROUGE-1, 2, L and skipgram), because they each capture slightly different forms of loosely matching word order. All of them are highly imperfect, so it'll be hard to get a good picture of which text is a good match to the reference answer without using multiple metrics.

You'll also want to include some semantic similarity metric (i.e. a BLEURT model or a score like BERTScore or cosine similarity between sentence-level vectors using a sentence transformer), since all of the ROUGE metrics only match literal ngrams and don't account for synonyms, word sense, or other more latent meaning.

That can be especially important if your key concerns about the task involve things like hallucination, which the standard evaluation metrics won't tell you if that's happening, but you can review a sample to see how often it appears to be happening yourselves. One other way that people try to at least loosely measure hallucination, is to take a separate model trained on a task called Natural Language Inference (NLI) or Textual Entailment, which involves pairs of sentences labeled with whether they are logically consistent or inconsistent.

You won't want to have to train that model yourselves, because it's a separate task that'll take away from your main one. But there are off-the-shelf models already trained for that, that you can use as a judge to predict how often your model's answers have factually inconsistent info with the reference answers, as another form of evaluation. (You don't have to do that if you end up with too many things to tackle in the time you have, it's just one more suggestion. Prioritize the model variants/experiments you want to run, then the evaluation methods you can get done. Human evaluation is also an entirely valid way to check for hallucinations in the time you have.)

