**SciGuru**

**Application Design Document**

**1. Introduction**

**1.1 Summary**

This research project focuses on training a large language models (LLMs) to give better scientific explanations. By critically analyzing existing methodologies and leveraging stateof-the-art advancements, our goal is to fine-tune LLMs for improved precision and contextuality in scientific discourse.

Our approach integrates a blend of Supervised Fine Tuning (SFT) and different Reinforcement Learning methods, aiming to enhance the process of language modeling using advanced deep learning techniques.

We are creating and implementing an evaluation metric designed to measure the quality of scientific explanations. This metric enables us to gauge the success of our methods for enhancing the abilities of LLMs. Our goal is to fill a critical void regarding how LLMs function today, making them more useful for educational and research purposes. We're working towards enhancing how AI can help us understand and communicate scientific ideas, which could transform learning and discovery in significant ways.

In addition to enhancing accuracy, we're placing significant emphasis on the quality of scientific explanations provided by LLMs. We aim to train a model that conveys complex scientific concepts in an engaging, clear, and contextually rich manner. Through this, we aspire to make science explanations more approachable and enjoyable for all.

**1.2 Background**

Our research aims to enhance Large Language Models' (LLMs) capability to generate high-quality scientific explanations. While LLMs have demonstrated remarkable abilities across various tasks, their proficiency in providing clear, accurate, and pedagogically sound scientific explanations remains an area with significant room for improvement.

The field of scientific explanation assessment has been extensively researched, providing us with established frameworks to evaluate and enhance explanation quality. We reviewed several key papers to ensure our approach aligns with state-of-the-art methodologies in this domain.

A foundational framework in this field is Toulmin's Argumentation Pattern (TAP), introduced by philosopher Stephen Toulmin in "The Uses of Argument" (1958). TAP provides a structured approach to analyzing explanations through six key elements:

* ***Claim****: The main conclusion being presented*
* ***Data****: Supporting evidence and facts*
* ***Warrant****: Logical connection between data and claim*
* ***Backing****: Additional support for the warrant*
* ***Qualifier****: Conditions under which the claim holds true*
* ***Rebuttal****: Potential counter-arguments or exceptions*

Building on this foundation, "An Instrument for Assessing Scientists' Written Skills in Public Communication of Science" presents a comprehensive evaluation framework that identifies five critical components of effective scientific communication:

1. **Content Features**: Focuses on the selection and organization of scientific information, including decisions about what to include or omit.
2. **Knowledge Organization**: Examines how information is structured and reinforced through repetition and logical flow.
3. **Analogical Approaches**: Evaluates the use of analogies and metaphors to explain complex concepts through familiar references.
4. **Narrative Elements**: Assesses the incorporation of storytelling techniques to make scientific concepts more engaging and memorable.
5. **Dialogic Communication**: Measures how well the explanation facilitates understanding and engagement, particularly for complex or controversial topics.

Further insights come from "Analyzing how Scientists Explain their Research," which presents a rubric specifically designed for evaluating scientific explanations. This rubric emphasizes three key dimensions:

1. **Pedagogical Knowledge**:
   * Structure and organization
   * Audience awareness
   * Language choice
   * Technical presentation skills
2. **Content Knowledge**:
   * Factual accuracy
   * Information organization
   * Knowledge transfer capabilities
3. **Integration**:
   * Contextual placement
   * Visual and mental imagery
   * Scaffolded learning approach

These frameworks provide our project with robust criteria for:

* Evaluating LLM-generated scientific explanations
* Developing effective training methodologies
* Creating comprehensive assessment metrics
* Guiding the fine-tuning process

By incorporating these established evaluation frameworks into our LLM fine-tuning process, we aim to create a model that not only provides accurate scientific information but also presents it in a way that is pedagogically sound and accessible to diverse audiences. This approach ensures our work builds upon proven methodologies while advancing the state-of-the-art in AI-generated scientific explanations.

Use-cases? Talk to Nir

Chapter 2 System Architecture – Yuval

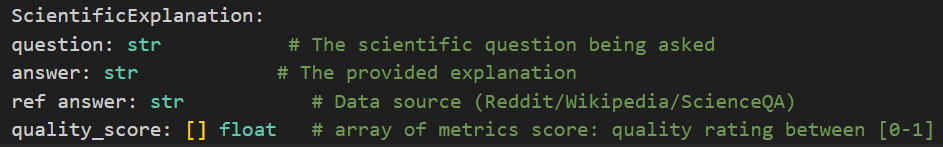
**3. Data Model**

* 1. **Description of Data Objects**

SciGuru consists of several core data objects, each serving a distinct role in the fine-tuning and evaluation pipeline. Below, we detail each object and its key attributes.

1. **ScientificExplanation**

This object represents a single scientific explanation, serving as the fundamental unit of our training data. We used 100 Q&A and answers to test our current implementation, the model responded to each question and it's answer was compared to a suitable reference answer using predetermined "best-practice" metrics.



To test our model's responses, we used the following metrics:

תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטי

1. **QualityMetrics**

* **BLEU (Bilingual Evaluation Understudy):** 
  + Measures how similar a generated text is to a reference text using n-gram (word sequence) overlap.
  + Higher BLEU means better similarity to the reference text.
  + Often used in machine translation.
* **ROUGE (Recall-Oriented Understudy for Gisting Evaluation)**
* Measures text overlap, especially useful for summarization tasks.
* ROUGE-1: Measures overlap of single words (unigrams).
* ROUGE-2: Measures overlap of word pairs (bigrams).
* ROUGE-L: Measures longest common subsequence (captures fluency and coherence).
* Higher scores mean the generated text captures more key parts of the reference text.
* **METEOR (Metric for Evaluation of Translation with Explicit Ordering)**
* Improves on BLEU by considering synonyms, stemming (e.g., "run" vs. "running"), and word order.
* A higher METEOR score means better alignment with the reference text.
* **BERTScore**
* Uses deep learning (BERT model) to compare similarity at the meaning level, not just word overlap.
* Captures semantic similarity between words in context.
* Higher BERTScore means the generated text conveys a similar meaning to the reference.

**III. Model Configuration**

Configuring the model to the correct settings became more and more potent as we examined larger and larger LLMs. Our University cluster runs RTX 3090/4090 - which can support large language models - if configured correctly. Below is a breakdown of the best performing configurations for SciGuru:

**base\_model:** We experimented with multiple models:

* **T5\_small\_70m** (70 million parameters)
* **T5\_Large\_700m** (700 million parameters)
* **LLAMA3.1\_7b** (7 billion parameters)
* **Falcon\_mamba\_7b** (7 billion parameters, Mamba variant) – our best-performing design. The model can generate human-like text based on a given prompt or input, engage in conversations and respond to questions and statements,  has a deep understanding of language and can comprehend complex texts and conversations.

**adapter\_path**: Path to the **LoRA (Low-Rank Adaptation) adapter**, which is a lightweight fine-tuning technique that efficiently updates a small subset of model parameters while keeping the rest frozen. This helps when fine-tuning large LLMs as most of the billions of weights doesn’t need to be adjusted every iteration.

**tokenizer\_path:** Specifies the path to the tokenizer used with the base model. Tokenizers transform raw text into numerical representations (tokens) that the model can process. For example, a sentence like *"The cat is sitting on the sofa"* has no inherent meaning to a computer. However, after tokenization, it might be converted into a sequence of numerical tokens like [13, -211, 312, 499]. These tokens allow the model to perform mathematical operations, such as dot product calculations, to determine relationships between words and infer meaning within the context of a paragraph. This step is crucial for enabling deep learning models to understand and generate human-like text.

**quant\_config:** Defines the quantization settings, which optimize model efficiency by reducing numerical precision while preserving performance. Essentially, this process determines how many decimal places to retain before rounding off values. By lowering precision (e.g., we used 4-bit instead of 16-bit floating-point numbers), we significantly reduce memory usage and computational demands. This step is critical for running large language models efficiently on our cluster, where both storage and processing power are limited.

* 1. **Databases**

Our project utilizes multiple data collections to support the training and evaluation of the language model. While we don't use a traditional relational database, we maintain several structured datasets that are essential to our system's operation.

**Training Data Collection**

**Collection Name:** scientific\_explanations

{

"question": String, # The scientific question

" ref answer": String, # The expert explanation

" model answer": String, # model generated reply to the question

"metadata": {

"timestamp": DateTime,

"category": String # Scientific domain

}

**Model Evaluation Collection**

**Collection Name:** model\_evaluations

{

"base\_model ": String, // Multiple base model tested.

"model\_version": String,

"timestamp": DateTime,

"test\_cases": [{

"question": String,

"generated\_answer": String,

"reference\_answer": String,

"metrics": {

"bleu": Float,

"rouge\_scores": {

"rouge1": Float,

"rouge2": Float,

"rougeL": Float

},

"meteor": Float,

"bertscore": Float

}

}],

"aggregate\_scores": {

"avg\_bleu": Float,

"avg\_rouge1": Float,

"avg\_rouge2": Float,

"avg\_rougeL": Float,

"avg\_meteor": Float,

"avg\_bertscore": Float

}

}

**Main Transactions**

1. **Training Data Operations**
   * insert\_explanation: Adds new scientific explanations to the training dataset
   * update\_quality\_scores: Updates evaluation metrics for existing explanations
   * batch\_retrieve: Fetches training batches for model fine-tuning
2. **Evaluation Operations**
   * log\_evaluation: Records model performance metrics
   * compute\_aggregate\_metrics: Updates aggregate performance statistics

**Key Relationships:**

* ScientificExplanation objects are stored directly in the scientific\_explanations collection
* QualityMetrics are embedded within each explanation document
* ModelConfiguration metadata is tracked in the model\_evaluations collection

**Bibliography**

Osborne, J., Erduran, S., & Simon, S. (2004). Enhancing the quality of argumentation in school science. Journal of Research in Science Teaching, 41(10), 994–1020. <https://doi.org/10.1002/tea.20035>

Toulmin, S. E. (2003). The Uses of Argument (2nd ed.). Cambridge: Cambridge University Press.

Baram-Tsabari, A., & Lewenstein, B. V. (2013). An Instrument for Assessing Scientists’ Written Skills in Public Communication of Science. Science Communication, 35(1), 56–85. <https://doi.org/10.1177/1075547012440634>

Sevian, H., & Gonsalves, L. (2008). Analysing how Scientists Explain their Research: A rubric for measuring the effectiveness of scientific explanations. International Journal of Science Education, 30(11), 1441–1467. <https://doi.org/10.1080/09500690802267579>

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems. <https://arxiv.org/abs/2307.09288>

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35, 27730-27744. [arXiv:2203.02155](https://www.google.com/search?q=Constitutional+ai%3A+Harmlessness+from+ai+feedback&oq=Constitutional+ai%3A+Harmlessness+from+ai+feedback&gs_lcrp=EgZjaHJvbWUyBggAEEUYOTIGCAEQRRg7MgYIAhBFGD3SAQgxNDQ4ajBqNKgCALACAA&sourceid=chrome&ie=UTF-8)

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., ... & Kaplan, J. (2022). Constitutional ai: Harmlessness from ai feedback. arXiv preprint [arXiv:2212.08073](https://arxiv.org/abs/2204.05862).

Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint [arXiv:1707.06347](https://arxiv.org/abs/1707.06347) (2017)

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. Advances in neural information processing systems, 35, 22199-22213. [arXiv:2205.11916](https://arxiv.org/abs/2205.11916)