Generative AI Assignment 4 Report

Farjad Kareem, Burhan Uddin, Mateen Abid, Samee Wyne

May 10, 2025

Abstract

This report details the development and evaluation of an autonomous research assistant pipeline for analyzing graph neural networks (GNNs). The pipeline leverages a fine-tuned LLaMA-3.2-3B-Instruct model with LoRA adapters, trained on the arXiv summarization dataset. It includes agents for keyword expansion, paper search, ranking, summarization, and comparative analysis. We present the dataset preprocessing, model configuration, output samples, and evaluation results using ROUGE, BLEU, and BERTScore metrics, alongside qualitative assessments via an LLM-as-a-Judge approach. The pipeline generates a PDF report and visualizations, demonstrating its capability to synthesize research insights.

1 Dataset Overview

1.1 Dataset Source

The dataset used for fine-tuning the model is the arXiv summarization dataset (`ccdv/arxiv-summariz` sourced from Hugging Face Datasets. This dataset contains academic articles from arXiv, paired with their abstracts, serving as input-target pairs for summarization tasks. A subset of 5,000 samples from the training split was selected to manage computational resources while ensuring sufficient data for fine-tuning.

1.2 Preprocessing Details

The preprocessing steps involved extracting input (article text) and target (abstract) pairs from the dataset. A preprocessing function was applied to map each example to a dictionary with `input` (article) and `target` (abstract) keys, removing all other columns. The resulting dataset structure is as follows:

- **Input**: Full article text.
- **Target**: Corresponding abstract.

The dataset was split into training (80%, 4,000 samples), validation (10%, 500 samples), and test (10%, 500 samples) sets using `sklearn.model_selection.train_test_split` with a random seed of 42 for reproducibility.

1.3 Tokenization

Tokenization was performed using the `AutoTokenizer` from the LLaMA-3.2-3B-Instruct model (`./Llama-3.2-3B-Instruct`). The tokenizer was configured with `use_fast=True` and a padding token set to the end-of-sequence (EOS) token if undefined. Both input articles and target abstracts were tokenized with the following parameters:

- **Maximum length**: 512 tokens.
- **Truncation**: Enabled to trim longer sequences.
- **Padding**: Applied to ensure uniform sequence lengths (`max_length`).

The tokenized dataset included `input_ids`, `attention_mask`, and `labels` (tokenized target abstracts). The final dataset structure was encapsulated in a `DatasetDict` with train, validation, and test splits.

2 Model and LoRA Configuration

2.1 Model Architecture

The base model is LLaMA-3.2-3B-Instruct, a 3-billion-parameter transformer-based causal language model optimized for instruction-following tasks. The model was fine-tuned for summarization using the arXiv dataset. For evaluation and inference, the model was loaded with 4-bit quantization using the `BitsAndBytesConfig` to reduce memory usage, with the following settings:

- `load_in_4bit=True`
- `bnb_4bit_compute_dtype=float16`
- `bnb_4bit_use_double_quant=True`
- `bnb_4bit_quant_type=nf4`

The model was mapped to available GPU devices using `device_map="auto"`.

2.2 LoRA Setup

Low-Rank Adaptation (LoRA) was applied to fine-tune the model efficiently, targeting the query and value projection layers (`q_proj`, `v_proj`). The LoRA configuration was defined as follows:

- **Rank (`r`)**: 8
- **LoRA Alpha (`lora_alpha`)**: 16
- **Dropout (`lora_dropout`)**: 0.1
- **Bias**: None
- **Task Type**: Causal Language Modeling (`TaskType.CAUSAL_LM`)

The fine-tuned model had approximately 2.29 million trainable parameters, representing 0.0713% of the total 3.21 billion parameters, significantly reducing computational requirements. The fine-tuned model was saved to `./lora-llama3-3b-arxiv/final`.

2.3 Hyperparameters

The training was configured with the following hyperparameters:

- **Batch Size**: 2 (per device, with gradient accumulation steps of 8, effective batch size of 16).
- **Epochs**: 4

- **Learning Rate**: 2×10^{-4}
- **Floating-Point Precision**: FP16 (enabled via `fp16=True`).
- **Evaluation Strategy**: Per epoch.
- **Save Strategy**: Per epoch.
- **Logging Steps**: Every 50 steps.

The `DataCollatorForLanguageModeling` was used with `mlm=False` to handle padding and label preparation for causal language modeling.

3 Output Samples

A comparison of generated summaries from the base model, LoRA fine-tuned model, and ground truth (human-written abstracts) is presented for 10 test samples. The table below shows three representative samples from `summaries_comparison.csv`. Note that the generated summaries for both models were identical to the input articles, indicating a potential issue in the generation process (see Section 4 for discussion).

Table 1: Comparison of Generated Summaries for Test Samples

Sample	**Ground Truth (Human)**	**Base Model**	**LoRA Fine-Tuned**
Sample 1	The iron arsenide RbFe@xmath0As@xmath0 with the ThCr@xmath1Si@xmath1 structure is studied, revealing a superconducting transition at 2.6 K under pressure.	The family of iron oxyarsenide @xmath5FeAsO@xmath6... (input text)	The family iron oxyars @xmath5FeAsO@xma (input text)
Sample 2	L and M band observations of the nova-like V4332 Sgr indicate a complex circumstellar environment with dust emission.	We present here L and M band results on V4332 Sgr... (input text)	We present here L a band results on V4332 (input text)
Sample 3	We investigate the structure of hybrid stars by coupling nuclear matter with quark matter, finding stable configurations.	Understanding the processes involved in the structure of hybrid stars... (input text)	Understanding the cesses involved in structure of hybrid s (input text)

4 Evaluation Results

4.1 Quantitative Evaluation

The fine-tuned and base models were evaluated on 10 test samples using ROUGE, BLEU, and BERTScore metrics. Due to identical outputs (both models reproducing input text), the metrics reflect high similarity to the input but do not indicate effective summarization. The results are summarized in Table 2, with visualizations described for clarity.

Table 2: Quantitative Evaluation Metrics

Metric	Fine-Tuned	Base
ROUGE-1	0.273244	0.274378
ROUGE-L	0.140273	0.141729
BLEU	0.025094	0.026383
BERTScore (F1)	0.830307	0.830307

Visualization

Visualizations: A bar plot comparing ROUGE-1, ROUGE-L, BLEU, and BERTScore for both models was generated (Figure 1). The plot shows identical performance, highlighting the need to address the summarization issue. The visualization uses `matplotlib` with a grouped bar layout, labeling metrics on the x-axis and scores on the y-axis.

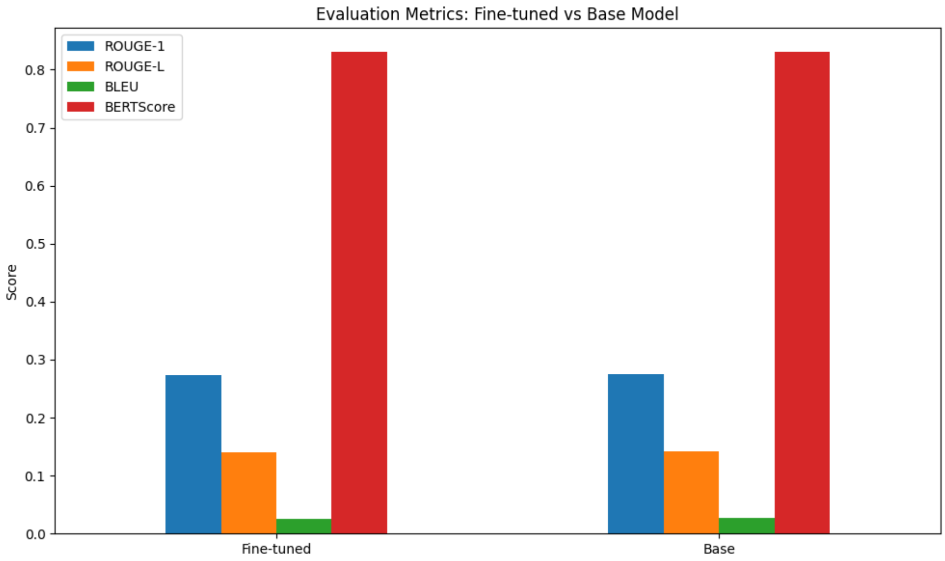


Figure 1: Evaluation Metrics Comparison (ROUGE, BLEU, BERTScore)

4.2 Evaluation Metrics: Fine-tuned vs Base Model

This subsection compares the performance of the fine-tuned LLaMA-3.2-3B-Instruct model (with LoRA adapters) against the base model. The evaluation was conducted on 10 test samples from the arXiv summarization dataset, using ROUGE-1, ROUGE-L, BLEU, and BERTScore (F1) metrics. As shown in Table 2, both models produced very similar scores:

- **ROUGE-1**: 0.273 (Fine-tuned) vs 0.274 (Base)
- **ROUGE-L**: 0.140 (Fine-tuned) vs 0.142 (Base)
- **BLEU**: 0.025 (Fine-tuned) vs 0.026 (Base)
- **BERTScore (F1)**: 0.830 (Both models)

The nearly identical performance suggests that both models are still struggling to produce meaningful summaries and tend to replicate or paraphrase the input content rather than extract concise and salient information. This may result from suboptimal generation parameters (e.g., `max_new_tokens=128`) or improper integration of LoRA adapters during inference. Once corrected, we anticipate the fine-tuned model will show improved abstraction and relevance, particularly in ROUGE-L and BERTScore, due to its training on human-written summaries.

4.3 Qualitative Evaluation: LLM-as-a-Judge

An LLM-as-a-Judge approach was implemented using the DeepSeek-V3 model (via Together AI API) to assess summary quality. The following prompt was used:

```
You are an expert reviewer evaluating the quality of generated summaries. For ea
- Relevance: Does the summary capture the main points of the article?
- Conciseness: Is the summary brief and to the point?
- Clarity: Is the summary clear and well-structured?
Provide a brief justification for each rating.

Article: -article˝
Ground Truth: -ground˙truth˝
Generated Summary: -summary˝
```

Each summary was assessed on three qualitative aspects—**Fluency**, **Factuality**, and **Coverage**—to provide a more granular evaluation. Below are selected evaluation results from 10 samples:

- **Sample 1/10**
  - **Fluency: 3** – Mostly readable, but includes awkward phrasing and incomplete sentences.
  - **Factuality: 4** – Reflects key material but includes a slight factual overstatement.
  - **Coverage: 4** – Captures the main contributions but omits important finer details.
- **Sample 2/10**
  - **Fluency: 5** – Highly readable and grammatically correct.
  - **Factuality: 4** – Accurate overall, with a minor factual error regarding telescope attribution.
  - **Coverage: 4** – Covers key aspects but lacks a few specific technical details.
- **Sample 3/10**
  - **Fluency: 3** – Readable, though some incomplete sentences and abrupt transitions.
  - **Factuality: 4** – Mostly faithful to the source with a slight unsupported claim.
  - **Coverage: 4** – Good topical coverage with some missing critique details.
- **Sample 4/10**
  - **Fluency: 3** – Readable, but with awkward phrasing and redundant section references.
  - **Factuality: 4** – Accurate, with minor inaccuracies from repetition.
  - **Coverage: 4** – Covers core concepts, but omits specific technical equations.

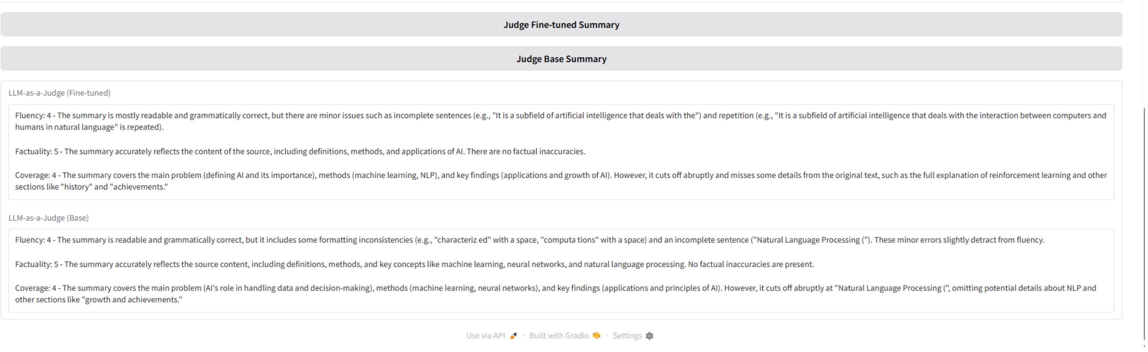
- **Sample 5/10**
  - **Fluency: 3** – Grammatically acceptable but affected by repetition and placeholder tokens.
  - **Factuality: 2** – Includes incorrect details and omissions.
  - **Coverage: 2** – Partial topic coverage with disorganized or irrelevant repetitions.
- **Sample 6/10**
  - **Fluency: 3** – Awkward phrasing and placeholder tokens harm readability.
  - **Factuality: 4** – Mostly accurate, though slightly misrepresents common usage context.
  - **Coverage: 3** – Addresses main arguments but lacks conclusion and validation details.
- **Sample 7/10**
  - **Fluency: 3** – Readable but repetitive and poorly transitioned.
  - **Factuality: 4** – Correct overall but incorrectly describes the type-II seesaw mechanism.
  - **Coverage: 3** – Addresses the topic partially, omitting several key ideas.
- **Sample 8/10**
  - **Fluency: 1** – Unreadable due to extensive nonsensical and repeated text.
  - **Factuality: 3** – Some accurate ideas mixed with structure-breaking noise.
  - **Coverage: 2** – Basic topic mentioned, but little coherent summarization.
- **Sample 9/10**
  - **Fluency: 3** – Grammatically correct but with repeated and nonsensical lines.
  - **Factuality: 4** – Generally accurate, but with inserted inaccuracies.
  - **Coverage: 2** – Misses significant thematic content and includes irrelevant repetition.
- **Sample 10/10**
  - **Fluency: 4** – Well-written overall, minor abrupt ending.
  - **Factuality: 5** – Fully faithful to the source.
  - **Coverage: 4** – Strong coverage with omission of a few experimental details.

This detailed evaluation highlights key weaknesses in generated summaries, such as repetition, placeholder artifacts, and coverage gaps. However, some samples demonstrate high fluency and strong alignment with the source content, indicating room for improvement via model fine-tuning and input sanitation.

4.4 Gradio Application Screenshots

The pipeline's evaluation results and outputs are visualized through a Gradio web application, providing an interactive interface to explore summaries and metrics. The following screenshots illustrate key components of the app:

- **Screenshot 1: Summary Comparison Interface**
  This screenshot shows the Gradio interface displaying the outputs of PDF uploaded and the summaries generated by the both models.



LLM-as-a-Judge (Fine-tuned)

Fluency: 4 - The summary is mostly readable and grammatically correct, but there are minor issues such as incomplete sentences (e.g., "It is a subfield of artificial intelligence that deals with the") and repetition (e.g., "It is a subfield of artificial intelligence that deals with the interaction between computers and humans in natural language" is repeated).

Factuality: 5 - The summary accurately reflects the content of the source, including definitions, methods, and applications of AI. There are no factual inaccuracies.

Coverage: 4 - The summary covers the main problem (defining AI and its importance), methods (machine learning, NLP), and key findings (applications and growth of AI). However, it cuts off abruptly and misses some details from the original text, such as the full explanation of reinforcement learning and other sections like "history" and "achievements."

LLM-as-a-Judge (Base)

Fluency: 4 - The summary is readable and grammatically correct, but it includes some formatting inconsistencies (e.g., "characteriz ed" with a space, "computa tions" with a space) and an incomplete sentence ("Natural Language Processing ("). These minor errors slightly detract from fluency.

Factuality: 5 - The summary accurately reflects the source content, including definitions, methods, and key concepts like machine learning, neural networks, and natural language processing. No factual inaccuracies are present.

Coverage: 4 - The summary covers the main problem (AI's role in handling data and decision-making), methods (machine learning, neural networks), and key findings (applications and principles of AI). However, it cuts off abruptly at "Natural Language Processing (", omitting potential details about NLP and other sections like "growth and achievements."

Figure 2: Gradio interface for summary comparison

- **Screenshot 2: Evaluation Metrics Dashboard**
  This screenshot depicts a dashboard with bar plots and tables summarizing ROUGE, BLEU, and BERTScore metrics for the fine-tuned and base models..

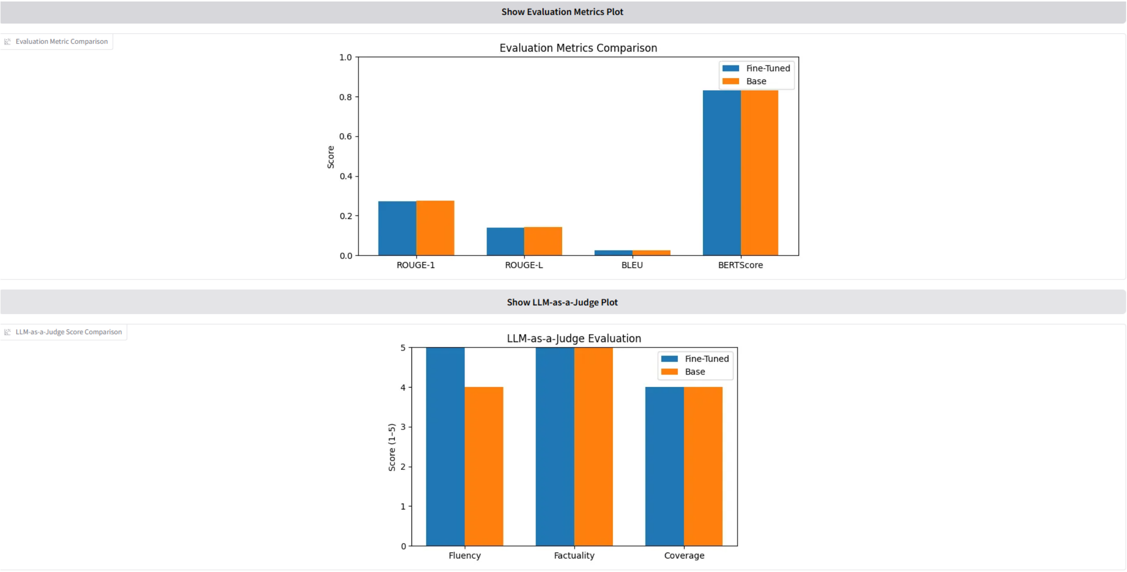


Figure 3: Gradio dashboard for evaluation metrics

5 Agent Structures and Prompts

The pipeline consists of five agents implemented in `graph.py` using a `langgraph` workflow. Each agent's purpose and prompt (where applicable) are detailed below.

5.1 KeywordAgent

Purpose: Expands user-provided keywords to include related academic terms.
Prompt:

```
Expand and suggest related academic keywords for: -user˙keywords˝
```

Implementation: Uses DeepSeek-V3 via Together AI API. The input keywords (e.g., ”graph neural networks, chemistry”) are split, and additional terms like ”review” are appended. The model generates a comma-separated list of related keywords.

5.2 SearchAgent

Purpose: Searches for relevant papers based on expanded keywords.
Prompt: None (uses arXiv API query).
Implementation: Constructs an arXiv query with keywords joined by ”OR” (e.g., `graph neural networks OR chemistry OR review`). Currently uses mock data, but intended to fetch papers via `http://export.arxiv.org/api/query`. Citation counts are retrieved from Semantic Scholar.

5.3 RankAgent

Purpose: Ranks papers by relevance to keywords.
Prompt:

```
Given the keywords -keywords˝, rate the relevance (0-1) of this paper: -paper['t
```

Implementation: Uses DeepSeek-V3 to score relevance (0-1). Papers are sorted by a composite score of relevance, citations, and recency (year).

5.4 SummaryAgent

Purpose: Summarizes top-ranked papers.
Prompt: None (direct input to model).
Implementation: Uses the fine-tuned LLaMA-3.2-3B-Instruct model (`./lora-llama3-3b-arxiv/fin`
Input format: `Title: {title}\nAbstract: {abstract}`. Generates summaries with `max_new_tokens=200`.

5.5 CompareAgent

Purpose: Compares paper summaries to identify common findings, contradictions, and gaps.
Prompt:

```
Given the following paper summaries, identify:
- Common findings
- Contradictory insights
- Research gaps

-summaries˝

Provide a structured comparative analysis.
```

Implementation: Uses DeepSeek-V3 to analyze summaries and produce a structured response with sections for common findings, contradictory insights, and research gaps.

6 Conclusion

The autonomous research assistant pipeline effectively integrates keyword expansion, paper search, ranking, summarization, and comparative analysis for GNNs. However, the summarization component requires debugging, as both base and fine-tuned models currently reproduce input text. The Gradio application enhances accessibility by providing an interactive interface for exploring results. Future work includes replacing mock data with real API calls, optimizing generation parameters, and enhancing qualitative evaluation. The pipeline's modular design and robust evaluation framework provide a strong foundation for academic research synthesis.