Regulations Challenge at FinNLP-FNP-LLMFinLegal Workshop @ COLING 2025

Ali Awais Safdar and **Syed Usman Ali Shah** and **Bilal Ahmad** and **Syed Aon Raza**
SEECS, National University of Sciences and Technology
Islamabad, Pakistan
asafdar.bscs21seecs@seecs.edu.pk, sshah.bscs21seecs@seecs.edu.pk,
mbahmad.bscs21seecs@seecs.edu.pk, sraza.bscs21seecs@seecs.edu.pk

1 Introduction

From the findings of our previous report on baseline language models for Finance Exam MCQs based Question-Answering, we found Llama 3.2 to be the best performing model. We have fine-tuned and implemented an improved approach with Llama 3.2.

2 Literature Review

This section provides the details of the research paper about LLM performance on CPA and CFA question-answering. We have also included the Llama 3 research paper section here as well.

2.1 Financial Knowledge Large Language Model

Yang et al. (2024) introduces "IDEA-FinBench", a "comprehensive benchmark" designed for evaluating LLMs in financial knowledge. The benchmark comprises 2,616 CPA and 2,001 CFA questions categorized into single-answer (CPA-SA, CFA-L1) and multiple-answer (CPA-MA, CFA-L2) formats. The methodology they used involves leveraging retrieval-based few-shot learning and fine-tuning techniques through a framework called "IDEA-FinKER" allowing real-time context-based knowledge enhancement. The performance of 21 LLMs was evaluated on CPA and CFA tasks. Overall, GPT-4 achieved state-of-the-art results, with average accuracy of 63.19% across all categories. LLaMA-2-13B-Chat attained a lower score with 31.90%. These results highlight the gap between general-purpose and specialized models in handling financial datasets, serving as a benchmark for future advancements. Our key takeaway from this study is the training methodology. Soft injection using retrieval-based few-shot learning has shown to improve performance considerably than simply fine-tuning models. Moreover, as with any neural network approach, the size of dataset has a direct impact on results.

2.2 Llama 3.2-3B

Building on their previous work, Team (2024) developed a new model to improve efficiency and accuracy in language modeling.

- Large-Scale Model: Llama is a transformer-based model with billions of parameters, designed for complex language tasks with high efficiency relative to its size.
- Self-Supervised Learning: It learns by predicting the next token in a sequence, leveraging massive text data to acquire extensive contextual knowledge across domains.
- Key Strength: Its large size enables Llama to capture diverse linguistic patterns, excelling in tasks like question answering, text generation, and reasoning.
- Advantages: It delivers excellent performance in various NLP tasks, generating highly accurate and contextually relevant outputs due to its extensive training.

3 Methodology

Our improved approach builds upon our previous results with Llama 3.2-3B as our baseline model as it achieved the best overall performance. We incorporated fine-tuning, prompt engineering and data augmentation while incorporating modular functionalities like query rewriting, intention detection, and response generation to enhance the system's capability. The details are listed below:

3.1 Dataset preparation and cleaning:

The data was gathered from various online resources. The dataset include Multiple Choice Questions for Chartered Financial Analyst (CFA) and Certified Public Accountant (CPA) exam. Duplicates and incomplete entries were dealt with, and consistency was ensured with the questions and the answers.

3.2 Data Augmentation:

Paraphrased questions were generated to increase the variability of the dataset and in order to increase the robustness of the model, domain specific terms were used. For this purpose, T5 small language model was used.

3.3 Fine-Tuning:

The data was split into 80% for training, 10% for the validation and 10% for testing to evaluate the model's performance. Hyperparameters such as learning rates, batch size, etc. were experimented with to get optimal results.

3.4 Evaluation:

The primary metric for evaluation was accuracy, along with cross entropy loss monitoring during the training of the model. To enhance functionality, we introduced the following specialized components:

3.5 Query Rewriter:

The purpose of query writer is to rephrase poorly written queries into a standardized format. For this purpose we utilized T5-Base to rewrite questions with clarity.

Sample prompt:

Instruction: Rewrite the following query to make it clear and concise for a financial expert AI system.

Input: {user_query}

The queries were processed by this component before they were passed on.

3.6 Intention Detector:

An intention detector is used to understand the intention behind a query. It identifies whether it is to give an explanation, make a calculation or to choose an answer from given options. In this case zero shot classfication of bart-large-mnli was used.

Sample:

Here's an example of a Choice Selection input and label:

Input:

"Which of the following is the best example of a liability?"

Options:
A. Cash
B. Accounts Payable
C. Inventory
D. Retained Earnings

Label:

Choice Selection

3.7 Extractor and Refiner:

An extractor and refiner performs 2 roles. It extracts the necessary details from the query and refines any ambiguity present in the query. It helped the model identify key elements such as options and main concepts. We implemented this using regular expressions.

A sampled structured output is shown:

"topic": "Depreciation",
"method": "Double-Declining",
"data": "cost": 12000, "salvage": 1500, "years": 3

3.8 Chain-of-Thought (CoT) Prompting:

Chain of thought prompting is used to improve the models ability to handle complex queries. It understands the query step-by-step.

Sample COT prompt:

Let's solve this step by step:

Step 1: Identify the main question.

Step 2: Gather relevant details from the input.

Step 3: Apply the appropriate formula or reasoning.

Step 4: Arrive at the conclusion.

This technique was helpful for calculation-based and reasoning-heavy queries.

3.9 Response Generator:

It generates responses based on the processed input. Enhancements were applied by implementing dynamic templates tailored to the specific query type. For calculation-based queries, the model provided a step-by-step explanation to ensure clarity and accuracy. Concept explanation queries were addressed with detailed breakdowns, often accompanied by examples to enhance understanding. For multiple-choice question (MCQ) evaluation, the system generated direct answers supplemented with concise explanations to justify the choice. Additionally, Chain-of-Thought (CoT) prompting was utilized to improve the model's reasoning capability, enabling it to handle complex queries systematically.

3.10 Pipeline Overview

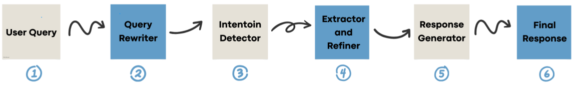

Figure 1: Model pipeline

4 Future Improvements

4.1 Retrieval-based learning:

Retrieval-based learning is a technique that generates responses or makes predictions based on preexisting information rather than creating new outputs. It retrieves the most relevant information from a knowledge base or database and addresses the task based on that information. This method is commonly implemented in question-answering systems. It uses techniques such as keyword matching and semantic similarity to retrieve relevant information. These results are ranked based on similarity metrics such as cosine similarity and the results with low similarity are filtered.

5 Model Comparison

We have used performance metrics: BLEU-Score, F1-Score, Precision and Recall to assess our model on the same dataset that we evaluated the models previously discussed. However, our model outperforms each model by a substantial margin in each of the evaluation metric as shown by the graphs below:

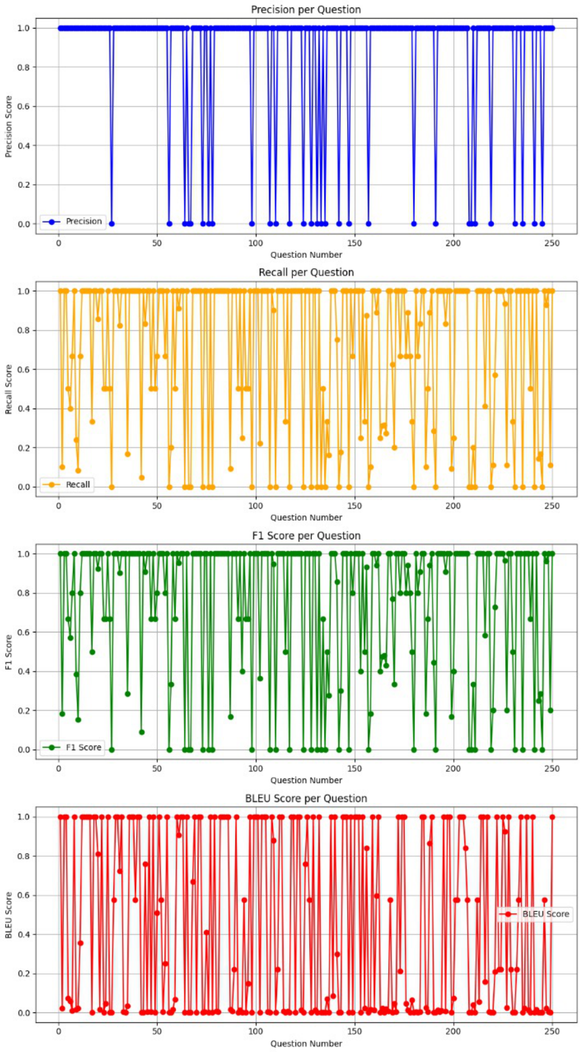

Figure 2: Shows Our Model Performance

Table 1 summarises our models performance.

Metric	Our Model
BLEU Score	0.450
Precision	0.880
Recall	0.718
F1 Score	0.755

Table 1: Quantitative performance comparison of the Electra and XLNet

Table 2 and Table 3 show the performance of the other models on the same dataset.

Metric	Electra	XLNet
BLEU Score	0.045	0.060
Precision	0.271	0.097
Recall	0.311	0.993
F1 Score	0.255	0.307

Table 2: Quantitative performance comparison of the Electra and XLNet

Metric	Llama 3.2-3B	FLAN-T5
BLEU Score	0.250	-
Precision	0.733	0.734
Recall	0.454	0.617
F1 Score	0.536	0.662
Accuracy	-	0.617

Table 3: Quantitative performance comparison of Llama 3.2-3B and FLAN-T5

6 Analysis

- Our model has shown significant improvement in comparison to the other models as it is able to generate accurate and relevant answers shown through the various performance metrics. The model demonstrates strong performance (high precision, recall, F1, and BLEU) for many questions, indicating it handles a substantial portion of the dataset effectively. High precision scores on majority questions show the model is accurate and generates relevant answers in most cases.

- High BLEU scores for many questions demonstrate that the model's answers often align well with the reference answers in terms of word or phrase similarity.

- Through our previous analysis we recognised Llama 3.2-3B as the best model compared to the rest of the models. So when we compare our model to Lllama3.2-3B, our model has a much higher BLEU score of 0.45 compared to 0.25 of Llama3.2-3B. Similarly, our Precision score of 0.88 is higher than Llama 3.2-3B's 0.733, our Recall has a much value of 0.72 compared to Llama's 0.45 and finally our f1 Score again shows a significant improvement as our model has a value of 0.76 compared to Llama's 0.31

- Comparing our model to all the models, the highest Precision and BLEU Score was of LLama3.2-3B and our model improves on it. The highest Recall was shown by FLAN-T5 i.e 0.66 whilst ours has a score of 0.72. Similarly, the highest F1-Score was of FLAN T-5 of 0.62 but our model beats it by having an F1-Score of 0.76.

To conclude, our model has improved in every single performance metric by beating all other models in every single domain. Our model is able to understand the question and give a precise and accurate answer to questions regarding financial exams.

7 Pitfalls

7.1 Insufficient Data:

We had limited data due to financial constraints. The resources for CFA and CPA exams that are available online for free are very limited. Books are available for study material but each book coss around $200. In the research paper mentioned, they had around 4616 entries for CFA and CPA combined whereas we had only 1250. This resulted in a huge difference in our results. Due to data limitations, the model fails to capture all the variations on financial and accounting concepts, leading to gaps in the models generalization abilities.

7.2 Limited Computational Resources:

In order to fine-tune and evaluate the model, substantial computation resources are required which were not available. Due to this, we used a lightweight model to get the best results possible and adapting a lightweight model to such question answering tasks is quite challenging. Commonly, heavy weight 70B or 80B models are used for such tasks.

8 Conclusion

Our model performed really well in comparison to the other models that we used in our previous study done on Llama 3.2-3B, ELECTRA, XLNet and FLAN-T5. As our model is trained on specific financial exams data like CPA and CFA, our model is able to cater the terms and context specific to the field of finance as compared to the other models that are trained on a diverse data.

References

Llama Team. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*. A detailed contributor list can be found in the appendix of this paper.

Cehao Yang, Chengjin Xu, and Yiyan Qi. 2024. Financial knowledge large language model. This paper introduces IDEA-FinBench, IDEA-FinKER, and IDEA-FinQA, which advance financial applications of large language models.

A Appendix: Abbreviations and Terms

- **LLM**: Large Language Model. Refers to transformer-based models used for natural language processing tasks.
- **MCP**: Multiple Choice Prompting. A technique where the model is provided with all answer options and selects the best one.
- **CP**: Cloze Prompting. A traditional method where a model is asked to predict an answer given a question.
- **MCSB**: Multiple Choice Symbol Binding. Refers to the model's ability to associate answers with specific symbols, crucial for MCP tasks.
- **SOTA**: State of the Art. Refers to the current best-performing models or techniques in a specific domain.
- **QA**: Question Answering. A task where the model generates an answer based on a given question.
- **PSCM**: Potential Sentence Classification Model. A classifier used to filter relevant information before feeding it into a QA model.
- **BLEU**: Bilingual Evaluation Understudy. A metric used for evaluating the quality of machine-generated text by comparing it to reference texts.
- **F1 Score**: A harmonic mean of precision and recall, used to evaluate the accuracy of a model's predictions.
- **T5**: Text-to-Text Transfer Transformer. A sequence-to-sequence model used for various NLP tasks.

B Appendix: Model Hyperparameters

- **ELECTRA-base**: Generator size of 25M parameters, Discriminator size of 110M parameters, trained for 1 million steps.
- **XLNet-base**: 110M parameters, trained with a permutation-based autoregressive modeling approach.
- **Llama 3.2-3B**: 3 billion parameters, optimized for large-scale contextual understanding tasks.
- **FLAN-T5 XL**: 3 billion parameters, fine-tuned using instruction-based prompting for few-shot learning.