ANLPers at QIAS: CoT for Islamic Inheritance

Serry Sibaee[1*] **Mahmoud Reda** [2] **Omer Nacar**[3] **Yasser Al-Habashi**[1]
Adel Ammar[1] **Wadii Boulila**[1]
[1]Prince Sultan University, Riyadh, Saudi Arabia
[2]Zagazig University
[3]Tuwaiq Academy – Tuwaiq Research and Development Center
{ssibaee, aammar, yalhabashi, wboulila}@psu.edu.sa
{20812017101263}@eng.zu.edu.eg
{o.najar}@tuwaiq.edu.sa
[*]Corresponding author: ssibaee@psu.edu.sa

Abstract

This paper presents a Chain-of-Thought (CoT) prompting approach for Islamic inheritance reasoning in multiple-choice question answering. We address the QIAS 2025 SubTask 1, which requires complex legal reasoning to determine correct inheritance shares according to Islamic jurisprudence. Our system employs two prompting strategies: direct answer extraction and step-by-step reasoning with regex-based answer extraction. We evaluate our approach using Claude 3.7 Sonnet and GPT-4o on Islamic inheritance MCQ tasks. Results demonstrate significant performance improvements when incorporating the thinking step: Claude 3.7 improved from 0.67 to 0.81, and GPT-4o from 0.63 to 0.74. Error analysis reveals that while models perform well in basic reasoning, they struggle with complex correction procedures (**Tasheeh**[1]) in inheritance calculations. Our findings confirm that structured reasoning substantially enhances LLM performance on complex Arabic legal reasoning tasks without requiring additional training or retrieval-augmented generation.

1 Introduction

The task of Islamic Inheritance Reasoning is a significant challenge in natural language processing and algorithmic systems due to the long process and mathematical operations needed to reach the final results. Current state-of-the-art models, such as those by (Sibaee et al., 2025), often struggle with answering a full hard inherence question such as: "Divide this inheritance according to Islamic law: The deceased left behind a father, a mother, the father's mother (paternal grandmother), a full brother, a full sister, a paternal half-brother, a paternal half-sister, a maternal half-brother, a maternal half-sister, and a nephew (son of full brother)." Sonnet-3.5 model did not show full logical thoughts. Sometimes, it reasoned correctly, but the final answer was wrong. This was also shown in (Abdulrahman and Walusimbi, 2024).

To address these limitations, we propose a new system based on showing the thought process of the model before answering the question. Our approach is novel because the type of problem is MCQ and this is an open field to show the logical thinking before answering. We hypothesize that this method will improve the performance obtained by (Wei et al., 2023). Our main contribution is showing the effectiveness of thinking and Chain of thoughts in answering hard and complex Islamic inheritance MCQ and this approach can help in using this SOTA models in these kind of tasks without adding any blocks to the pipeline e.g. RAG or finetuning.

2 Background

This study addresses QIAS 2025 – SubTask 1: *Islamic Inheritance Reasoning* (Bouchekif et al., 2025a), which focuses on answering multiple-choice questions (MCQs) related to the distribution of inheritance according to Islamic jurisprudence. The task inherently requires dual competencies: (1) comprehension of Arabic textual problem statements, and (2) the application of complex legal-mathematical reasoning to determine the precise share for each heir (Bouchekif et al., 2025b).

As an illustrative example, consider the scenario: *"A woman died leaving two sons, three daughters, a husband, a father, and a mother. What is the husband's share?"* with answer choices ranging from (A) Half to (F) Two-thirds. The correct answer, as prescribed by Islamic law, is (E) Quarter.

Previous research has explored various strategies for this domain. For instance, (Abdelazim et al., 2024) investigated the use of *Chain-of-*

[1]Tasheeh is the correction procedure applied when initial fractional shares of heirs do not divide evenly into whole numbers. It ensures integer shares while preserving proportional rights.

Proceedings of The Third Arabic Natural Language Processing Conference, pages 873–877
November 8-9, 2025

Thought (CoT) prompting to enhance performance in complex Arabic question-answering tasks, while (Zouaoui and Rezeg, 2021) employed ontology-based frameworks to model and solve Islamic inheritance problems, and (Sibaee et al., 2025) examined the LLMs on multiple topics including inheretince and showed a very low preformence in all of them.

Building upon these works, our approach integrates CoT reasoning to systematically decompose and solve inheritance problems step-by-step, followed by the application of regular expression (regex) techniques to accurately extract the final answer from the reasoning output. This combination is designed to address both the linguistic and jurisprudential complexity of the task, ensuring logically coherent reasoning and precise answer selection within the MCQ framework.

It is important to note that one of the most error-prone stages for existing models is the *Tasheeh* process, which requires adjusting fractional shares into integer values while maintaining proportional correctness. Our approach is designed to handle this step effectively within the CoT framework, thereby addressing a critical source of error in Islamic inheritance reasoning.

3 System Overview

Our system depends on calling an LLM to answer the given question in two ways: The first is to ask the question as is with the choices and ask the LLM to give the answer letter. The second method is to ask the model to explain the answer before selecting it, which is extracted using formatted regex.

4 Experimental Setup

We conducted the experiments using two SOTA LLMs. The first prompt is shown in Figure 1.

System Role: <You are a strict grader for multiple-choice exams.>

Task: <You must return only the correct choice letter (e.g., A, B, C...) without any explanation.>

Question: <question>

Options: <(options)>

Figure 1: LLM Prompt for Multiple Choice Answering

For the detailed (thinking) answer, the used prompt is displayed in Figure 2. The generic procedure for extracting the final answers from model outputs is described in algorithm 1.

System Role: <You are an expert tutor who solves multiple-choice questions by reasoning step by step.>

Task: <Explain the solution process step-by-step...>

Final Answer: [Letter]

Question: <question>

Options: <chr(10).join(options)>

Figure 2: LLM Prompt for Step-by-Step MCQ Explanation

Algorithm 1: Extracting Final Answers Using Prompt and Regex

Input: A set of multiple-choice questions with their choices

Output: Final answers extracted from model responses

foreach *question in questions* **do**
- **Get** the question text and its corresponding choices
- **Create** the full prompt by embedding the question and choices into the template
- **Send** the prompt to the language model
- **Extract** the final answer using regex (e.g., match `Final Answer: [A-F]`)

In addition to the two main prompting strategies, we experimented with integrating Retrieval-Augmented Generation (RAG) using the available in-dataset context. Specifically, each chunk in the retrieval index consisted of a question–answer pair from the training data. At inference time, for each test question, the most similar chunk was retrieved and appended to the LLM prompt as supporting context.

We also estimated the inference cost of our approach. Running the full QIAS benchmark with step-by-step reasoning required approximately **10 USD** in API costs.

In addition to the CoT prompting strategy, we experimented with a retrieval-augmented generation (RAG) setup. We built the retrieval index directly from the training split, where each chunk was a **concatenation of the original question and an expanded, detailed answer** (i.e., *question + enriched answer*) to provide richer signals. Chunks were stored in a **ChromaDB** vector index. We evaluated several embedding models, including **Omartificial-Intelligence-Space/Arabic-Triplet-Matryoshka-V2**, **sentence-**

transformers/all-MiniLM-L6-v2, and **Begm3**; among these, **Begm3** yielded the strongest retrieval quality in our setting. At inference time, for each test question we **retrieved the top-3 nearest chunks** and appended them to the step-by-step reasoning prompt, then used **Claude 3.7** in CoT mode to select the final answer.

5 Results

The performance of the language models was evaluated with and without the application of a "thinking" step. Without thinking, Claude 3.7 achieved a score of 0.67, while GPT-4o scored slightly lower at 0.6312. When the thinking step was incorporated, performance significantly improved for both models. Claude 3.7 (with thinking) reached a score of 0.81, GPT-4o (with thinking) achieved a score of 0.74, which is similar to the RAG + Claude score. These results suggest that structured reasoning or "thinking" substantially enhances the capabilities of both models.

Model	Score
Claude 3.7	0.67
GPT-4o	0.63
Claude 3.7 (Thinking)	**0.81**
GPT-4o (Thinking)	0.74
RAG (context + Claude 3.7 Thinking)	0.737

Table 1: Performance scores of models with and without the thinking step, on the dev dataset

Model	Score
Claude 3.7	**0.71**
GPT-4o	0.55

Table 2: Performance scores of models with the thinking step, on the test dataset

The RAG+CoT configuration produced **accuracy close to, but lower than**, our CoT-only baseline on the dev set (**0.737** with RAG+CoT vs. **0.81** with CoT-only using Claude 3.7). While competitive, the RAG pipeline introduced additional complexity (sensitivity to embedding choice) and increased hallucination risk from injecting three retrieved passages (occasional reliance on partially relevant context). Given these trade-offs, and since CoT-only already outperformed it, we did **not** adopt RAG in the final system.

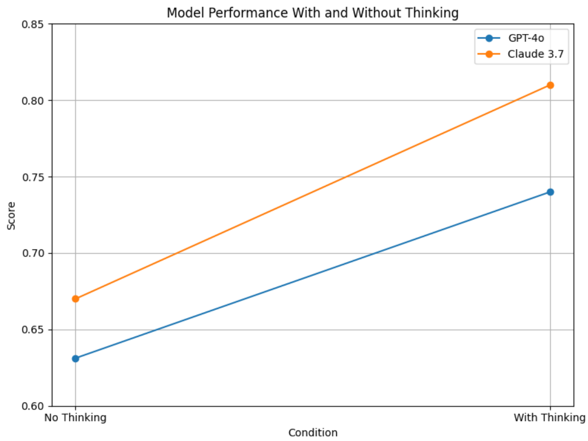

Figure 3: Comparison of model performance with and without thinking.

5.1 Error Analysis

Understanding the reasoning process and underlying thought patterns can reveal valuable insights and promising directions in the study of Chain of Thought reasoning (Wei et al., 2023). The following examples highlight specific instances where errors occur in the reasoning steps:

- First example: A person passed away and was survived by the following heirs: three full brothers, three sons of full brothers (nephews), four daughters, one wife, and five full sisters. The question is: what is the wife's share from the corrected total of the estate (al-tarkeha al-musahhah)?

The model (Claude) answer with thoughts was:

- Let us solve this inheritance problem step by step. First, we identify the heirs and their respective shares: the wife receives one-eighth (1/8) of the estate due to the presence of children; the four daughters collectively receive two-thirds (2/3) because they are multiple; the remaining estate goes to the full brothers and full sisters by way of residuary inheritance (taṣīb); while the sons of the full brothers are excluded due to the presence of closer male heirs. Next, we determine the original estate denominator, which is 24, since it accommodates both one-eighth and two-thirds. The distribution then becomes: 3 shares to the wife (1/8 of 24), 16 shares to the daughters (2/3 of 24), and the remaining 5 shares to be divided among the full brothers and sisters. However, since these 5 shares are to be divided among eight people (3 brothers and 5 sisters), the division is not exact, so we must correct the estate by finding a common multiple. We multiply the original 24 by 8 (the number of residuary heirs), resulting in a corrected estate of 192. The adjusted distribution is as follows: the wife receives 24 shares (3 × 8), the daughters receive 128 shares (16 × 8), and the full siblings receive 40 shares (5 × 8). Finally, since the question asks about the wife's share from the original estate before correction, her share is 3 out of 24. Thus, the correct answer is: A.

Analysis: The model answer was completly correct until reaching the 'Tasheeh' part where it did not write the shares after 'Tasheeh' so it chooses the before 'Tasheeh' answer.

- Second Example: A man died leaving: 3 sons, a mother, and 3 daughters. How many shares does each son receive?

The model (Claude) answer with thoughts was:

- Okay, let's solve this problem step by step: (1) Identify the heirs and their shares: the mother — one-sixth (because there are descendant heirs); the sons — the residue by ta'sīb (residuary), males receive twice the share of females; the daughters — together with the sons they inherit by ta'sīb with the male receiving twice the female's share. (2) Base of the calculation = 6 (the denominator for the sixth). (3) Distribution of shares: mother: 1 share (the sixth); the remainder 5 shares go to the sons and daughters by ta'sīb. (4) Calculate the number of portions: each son = 2 portions, each daughter = 1 portion, total = (3×2) + (3×1) = 9 portions. (5) Value of one portion: 5 (remaining shares) × 6 = 30 shares; 30 ÷ 9 = 10 shares per portion. (6) Each son's share: 2 × 10 = 20 shares. Looking at the available choices, we find each son's share is 20 shares. Final answer: F.

Analysis: Here the model did multiple errors from the first step of explaining the shares on this simpler question.

6 Future Work

Our RAG pilot with in-dataset QA context showed only limited improvements over CoT-only prompting, mainly due to coverage and retrieval granularity issues. To strengthen system robustness, future work will focus on three directions: (i) building a domain-specific inheritance corpus from authoritative sources to reduce retrieval sparsity, (ii) exploring structure-aware chunking and hybrid retrieval strategies to better capture fiqh-specific rules, and (iii) integrating RAG with CoT under context constraints to improve reasoning quality without incurring the overhead of fine-tuning.

7 Conclusion

In this paper, we presented our Chain-of-Thought prompting system for Islamic inheritance reasoning in the QIAS 2025 shared task. We introduced a dual-prompting approach that effectively addresses the complexity of Islamic jurisprudence reasoning through step-by-step explanation before answer selection. Our experiments demonstrate that incorporating a "thinking" step significantly improves model performance, with Claude 3.7 and GPT-4o achieving 21% and 17% relative improvements respectively. Through error analysis, we identified inheritance correction procedures (Tasheeh) as a primary area for future improvement, where models correctly perform initial calculations but fail to apply final correction steps. Our work confirms the difficulty of Islamic inheritance reasoning tasks but also shows that structured prompting can substantially enhance SOTA language models' performance on complex Arabic legal reasoning without additional model modifications or training.

References

Hazem Abdelazim, Tony Begemy, Ahmed Galal, Hala Sedki, and Ali Mohamed. 2024. Multi-hop arabic llm reasoning in complex qa. *Procedia Computer Science*, 244:66–75. 6th International Conference on AI in Computational Linguistics.

Manswab Mahsen Abdulrahman and Abdul Hafiz Musa Walusimbi. 2024. Evaluating the use of artificial intelligence for issuing fatwas in islamic inheritance cases: A juristic study with a comparison to gpt-3.5. *Asy-Syari'ah*, 26(2):121–146.

Ahmet Yusuf Alan, Enis Karaarslan, and Ömer Aydın. 2025. Improving llm reliability with rag in religious question-answering: Mufassirqas. *Turkish Journal of Engineering*, 9(3):544–559.

Abdessalam Bouchekif, Samer Rashwani, Mohammed Ghaly, Mutaz Al-Khatib, Emad Mohamed, Wajdi Zaghouani, Heba Sbahi, Shahd Gaben, and Aiman Erbad. 2025a. Qias 2025: Overview of the shared task on islamic inheritance reasoning and knowledge assessment. In *Proceedings of The Second Arabic*

Natural Language Processing Conference, ArabicNLP 2025, Suzhou, China, November 5–9, 2025. Association for Computational Linguistics.

Abdessalam Bouchekif, Samer Rashwani, Heba Sbahi, Shahd Gaben, Mutaz Al-Khatib, and Mohammed Ghaly. 2025b. Assessing large language models on islamic legal reasoning: Evidence from inheritance law evaluation. In *Proceedings of The Second Arabic Natural Language Processing Conference (ArabicNLP 2025)*, Suzhou, China. Association for Computational Linguistics.

Serry Sibaee, Omer Nacar, Adel Ammar, Yasser Al-Habashi, Abdulrahman Al-Batati, and Wadii Boulila. 2025. From guidelines to practice: A new paradigm for arabic language model evaluation. *Preprint*, arXiv:2506.01920.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Samia Zouaoui and Khaled Rezeg. 2021. Islamic inheritance calculation system based on arabic ontology (arafamonto). *Journal of King Saud University - Computer and Information Sciences*, 33(1):68–76.