BurhanAI at IslamicEval 2025 Shared Task: Combating Hallucinations in LLMs for Islamic Content; Evaluation, Correction, and Retrieval-Based Solution

Arij Al Adel
arij.aladel@gmail.com

Abu Bakr Soliman
abubakr@rankxy.com

Mohamed Sakher Sawan
me@sakher.co.uk

Rahaf Al-Najjar
rahaf.m.alnajjar@gmail.com

Sameh Amin
Sameh.m.amin@gmail.com

Abstract

In this paper, we describe our submission to the IslamicEval 2025 shared task, covering hallucination detection/correction and closed-world retrieval in Quranic and Hadith. We fine-tuned an LLM for detecting Quran and Hadith text spans, utilizing synthetic augmentation, diacritic variation, and morphological normalization to improve detection robustness (F1 = 87.10%) and used another reasoning model with tools (F1 = 90.06%). For validation, the accuracy is 88.60%, and for correction the accuracy is 66.56% where we employed a layered hierarchical index and search algorithm combining exact, normalized, fuzzy, and semantic matching with prompt-driven repair—to ensure canonical alignment and diacritic fidelity. For the correction stage, we also utilized a reasoning model with access to tools with an accuracy of 61.04%. Regarding the ranked answer-bearing text retrieval task, we implemented a Retrieval-Augmented Generation (RAG) system restricted to the corpora provided by the shared task, with structured output, vector-store grounding, and prompts tuned for "answer-enclosing" citations that achieve MAP@10 of 0.6199 on the development set and 0.2807 on the test set. The results highlight the value of normalization, corpus-restricted search, and reasoning models with tools in mitigating hallucinations and improving retrieval precision in low-resource religious settings and that much smaller fine-tuned models can compete with frontier models (e.g. GPT-5 high) for specialized tasks such as span detection.

1 Introduction

Despite SOTA of large language models (LLMs) in a wide range of natural language processing (NLP) tasks, they frequently hallucinate Li et al. (2024); Hikal et al. (2025); Orgad et al. (2024).

Employing Large Language Models (LLMs) to process religious texts Ganadi et al. (2025); Mohammed et al. (2025) raises different ethical concerns, which makes it a topic of special interest within the Ethics of Natural Language Processing (NLP) Hutchinson (2024). In religious contexts, hallucinations can manifest as misquoted verses, fabricated Hadiths, or distorted interpretations, which pose significant ethical, theological, and social risks. Such errors may undermine public trust in AI systems and contribute to the spread of misinformation, particularly when dealing with sacred texts that have fixed, canonical forms.

Our main contributions to the IslamicEval-2025 Mubarak et al. (2025) shared task are threefold. First, we introduced a data pipeline to generate a synthetic dataset, enabling fine-tuning of a relatively small LLM (gpt-4.1-mini) for detecting spans of religious quotations—both claimed and correct. We benchmarked this approach against large reasoning models with access to a code interpreter, showing that the fine-tuned small model is cheaper and faster while maintaining strong performance. Second, we designed a layered hierarchical index and search algorithm, coupled with a low-cost LLM judge (gpt-4.1-mini), which outperformed a frontier reasoning model (GPT-5 with code interpreter) that is significantly slower and more expensive. Third, we developed a Retrieval-Augmented Generation (RAG) pipeline specialized for Quranic and Hadith question answering, tailored to the unique linguistic and semantic challenges of Islamic texts. We have released our GitHub repository publicly to facilitate transparency and reproducibility of our work [1].

2 Background

We participated in Subtask 1A, which takes a model response as input and detects spans labeled Ayah or Hadith. In addition, we participated in Subtask 1B, which validates the spans identified in Subtask 1A labeling it as correct or incorrect, while Subtask 1C

[1] https://github.com/sakher/IslamicEval-BurhanAI-Public

corrects any spans marked as incorrect by providing their correct form or flagging them as incorrect. Finally, Subtask 2 focuses on retrieving the top 20 answer-bearing citations from the Quran and Sahih Al-Bukhari given an Arabic question.

Many previous works have addressed hallucination in large language models using different approaches. One line of research applies Retrieval-Augmented Generation (RAG) B'echard and Ayala (2024); Alan et al. (2024); Khalila et al. (2025). Other studies focus on instruction tuning and prompt engineering techniques Barkley and van der Merwe (2024); Hikal et al. (2025). Further research highlights verification and fact-checking strategies Sibaee et al. (2024). Additionally, some works emphasize fine-tuning with human feedback Cheng et al. (2025); Lin et al. (2025). Together, these methods enable LLMs to function as more effective tools for factual verification and reliable information use.

3 System Overview

3.1 Subtask 1A – Span Detection:

We used two approaches; we fine-tuned gpt-4.1-mini to output religious text spans. For fine-tuning we constructed a balanced training corpus (460 training examples and 83 validation examples) through multi-stage synthesis combining competition development data (70%) with synthetic examples (30%) generated using gpt-4.1 [2].

Separately, we leveraged a reasoning model with access to a code interpreter, testing both frontier and smaller OpenAI models (see detailed results in Table 1). The model was instructed to detect spans resembling Quran or Hadith. Since LLMs struggle with precise character counting Fu et al. (2024), we enabled the code interpreter tool: whenever the model needed to compute exact offsets, it could generate Python code, which was then executed in a secure sandbox, and the resulting values were fed back into the model. This ensured reliable start and end indices for each span. Outputs were further constrained using the OpenAI API's structured output feature with a JSON schema requiring a list of citations labeled as Ayah or Hadith with character offsets. We then applied heuristic post-processing: checking context within ±64 characters for lexical cues to refine labels, trimming extraneous punctuation or quotations, and merging or disentangling nested spans[3].

3.2 Subtask 1B – Validation and Subtask 1C – Correction:

Our system uses a layered design that combines seven forms of indexing with a six-stage search process. On the indexing side, every Quran verse and Hadith is indexed in multiple ways so the system can quickly switch between exact and approximate lookups. We keep exact MD5 hashes of the raw text, normalized versions without diacritics or punctuation, and character n-grams (3-grams by default) for fuzzy matches. Texts are also grouped into buckets by length to speed up candidate filtering, and we maintain a list for edit-distance checks. When available, we add a Whoosh full-text index for keyword search and a vector index built from Cohere embeddings stored in Qdrant for semantic similarity.

Searching happens in a strict sequence, with early stopping once a confident match is found. It starts with exact and normalized lookups, then falls back to n-gram fuzzy search. If needed, it escalates to semantic retrieval with embeddings and re-ranking. Next, it applies string-level fuzzy scorers such as Levenshtein distance and partial substring matching, followed by token-overlap checks to catch paraphrases. As a last resort, it computes Jaccard similarity on character trigrams. This stepwise design ensures clean matches are resolved instantly, while noisy, partial, or corrupted quotations are still recovered through progressively more flexible methods.

For the 1C correction subtask, we also tested a separate approach using a reasoning model - GPT-5 with high reasoning effort with access to tools. We give the model access to a code interpreter tool and to the corpora as text files. The model could perform multiple text-matching searches in the files to find the right match, then decide whether the matches were found to return them in JSON format.

3.3 Subtask 2

For Subtask 2, we built a Retrieval-Augmented Generation (RAG) system that retrieves passages from the Quran and Sahih Al-Bukhari. The corpora were split into 1,500-token chunks with 400-token overlap and stored as a vector dataset, allowing the reasoning model (GPT-5 with high reasoning) to run multiple searches per query when needed. The

[2] data generation pipeline https://github.com/sakher/IslamicEval-BurhanAI-Public/blob/main/abubakr/taskA/01-index-religion-dataset-for-search.py

[3] Prompts details https://github.com/sakher/IslamicEval-BurhanAI-Public/blob/main/task_a_prompt_engineering/pipeline_task_a.py

model could reformulate queries across iterations and returned ranked citations based on how directly and completely they answered the question. A deterministic post-processing pipeline then mapped Quran ayat to QPC Malhas and Elsayed (2020) passage IDs, validated hadith IDs against the official JSONL, removed duplicate citations.

4 Experimental Setup

All results were against test dataset and as seen on CodaBench. Our systems included a fine-tuned span detector (gpt-4.1-mini, 3 epochs, batch size 1, LR multiplier 2.0, temp 0). Implementation utilizes **Whoosh** for inverted indexing, **FuzzyWuzzy** for edit distance computation, **Qdrant** for vector storage, **Cohere embed-v4.0** for embeddings, **Cohere rerank-v3.5** for neural re-ranking, and **GPT-4.1-mini** for expert-guided validation for subtasks 1B and 1C. For evaluation, we used the proposed shared task evaluation metrics.

5 Results

Our system achieved a macro-averaged F1 score of 87.78 % using Fine-tuned a Span Detection Model approach, and 90.06% using reasoning model with access to tools (o4-mini model with high reasoning setting), see Table 1.

Although we tested larger models like the full-size o3 and GPT-5 three different sizes (full, mini and nano) with all reasoning levels (high, medium and low), none of these made it to the top 3 results, which shows that smaller models and fine-tuned tiny models can outperform larger models for such specialized tasks A.1.2.

Approach	Macro-Averaged F1
Approach-1	90.06%
Approach-2	87.78 %
Approach-3	87.10 %

Table 1: Task 1A evaluation results. Approach 1 is an OpenAI o4-mini with high reasoning effort reasoning model with access to tools. Approach 2 is an OpenAI o3-mini with high reasoning effort reasoning model with access to tools. Approach 3 is a fine-tuned gpt-4.1-mini span-detection model.

As for Subtask 1B, the layered hierarchical index and search algorithm achieves computational efficiency through exact matching optimization (constant-time hash operations) while maintaining comprehensive recall via semantic search for challenging disambiguation cases, yielding validation accuracy of 88.60% Table 2.

Approach	Accuracy
Hierarchical search-1	88.60 %

Table 2: Task 1B evaluation Accuracy results using layered hierarchical index and search algorithm with LLM-based validation.

For the Subtask 1C see Table 3, we used two approaches: layered hierarchical index and search algorithm with 66.56 % accuracy see section 3.2, and reasoning model with tools with 61.04 % accuracy. Table 3.

Approach	Accuracy
Hierarchical search-2	66.56 %
Reasoning model	61.04 %

Table 3: Subtask 1C evaluation Accuracy results. Hierarchical search-2 is a hierarchical search using a layered hierarchical indexing and search algorithm with LLM-based correction, and Reasoning model is a GPT-5 with high reasoning effort model with access to tools and post-processing.

For Subtask 2 see Table 4, the Mean Average Precision (MAP) was used as the main official measure for evaluation. We submitted only one submission. The results show that the model has some ability to find and rank relevant information, but there is significant room for improvement, especially for hypotheses.

Approach	MAP@10	MAP_Q@5	MAP_H@5
RAG-based(benchmark)	0.2807	0.3257	0.2386

Table 4: Subtask 2 results.

6 Conclusion

In this paper, we introduced an overview of our participation in the IslamicEval 2025 shared task Mubarak et al. (2025).

We proposed a layered hierarchical index and search algorithm with fine-tuned model to solve the Subtask 1A, 1B, 1C and reasoning model with tools for tasks 1A and 1C.

Our findings demonstrate that structured tool-assisted reasoning, hierarchical indexing with progressive search strategies, targeted fine-tuning of models, rigorous text normalization, corpus-restricted retrieval, and structured outputs are

highly effective for mitigating hallucinations and ensuring precise retrieval in religious QA contexts. Crucially, our results highlight that compact fine-tuned models (such as GPT-4-mini) and, separately, smaller reasoning models (e.g., o4-mini) with tool access can each achieve comparable or superior performance to large, computationally expensive frontier systems (e.g., GPT-5 with high reasoning), significantly reducing cost and latency—particularly in specialized tasks like span detection and correction (Subtasks 1A and 1C)

In future work, we plan to:

1. Explore vector store ingestion strategies (chunk sizing, overlap) and Arabic-specialized embedding models to improve recall on paraphrastic questions.

2. Add optional query-expansion prompts (synonyms, tafsir-guided paraphrases) while retaining closed-world constraints.

3. Consider shallow re-ranking informed by lightweight heuristics (entity match, directive/answer verbs) only if it demonstrably preserves "answer-enclosing" priority.

4. Evaluate adding auxiliary corpora (e.g., tafsir) as side channels for query reformulation without polluting the scoring universe.

5. Expand the vector store with texts with and without tashkeel (diacritics).

Limitations

Due to the limited time of our submission, we conducted limited experiments to solve the shared task and we were not able to explore more solution spectrum. Consequently, we did not go in depth into the hallucination categories for more fine-grained solutions. The integration of RAG introduces dependencies on retrieval accuracy and system latency, which can constrain its applicability in real-time scenarios or in environments with limited or no connectivity. Although we utilized LLMs to detect hallucinations, we have not yet investigated hallucination occurrences within the generated solutions. Finally, using large frontier models with high reasoning requirements can be both computationally expensive and time-consuming. Therefore, our future work will focus on leveraging lightweight models to improve efficiency.

AI disclaimer

We used ChatGPT and Cursor under author supervision to assist with phrasing and to generate support code for boilerplate and utilities; all research ideas, algorithms, experimental design, and interpretations are the authors' own, and the authors reviewed all outputs and accept full responsibility for the code and text; **no AI system is an author**.

References

Ahmet Yusuf Alan, Enis Karaarslan, and Ömer Aydin. 2024. A rag-based question answering system proposal for understanding islam: Mufassirqas llm. *ArXiv*, abs/2401.15378.

Liam Barkley and Brink van der Merwe. 2024. Investigating the role of prompting and external tools in hallucination rates of large language models. *ArXiv*, abs/2410.19385.

Patrice B'echard and Orlando Marquez Ayala. 2024. Reducing hallucination in structured outputs via retrieval-augmented generation. In *North American Chapter of the Association for Computational Linguistics*.

Xiaoxue Cheng, Junyi Li, Wayne Xin Zhao, and Jiahui Wen. 2025. Think more, hallucinate less: Mitigating hallucinations via dual process of fast and slow thinking. *ArXiv*, abs/2501.01306.

Tairan Fu, Raquel Ferrando, Javier Conde, Carlos Arriaga, and Pedro Reviriego. 2024. Why do large language models (llms) struggle to count letters? *arXiv preprint arXiv:2412.18626*.

Amina El Ganadi, Sania Aftar, Luca Gagliardelli, and Federico Ruozzi. 2025. Generative ai for islamic texts: The eman framework for mitigating gpt hallucinations. In *International Conference on Agents and Artificial Intelligence*.

Baraa Hikal, Ahmed Nasreldin, and Ali Hamdi. 2025. Msa at semeval-2025 task 3: High quality weak labeling and llm ensemble verification for multilingual hallucination detection. *ArXiv*, abs/2505.20880.

Ben Hutchinson. 2024. Modeling the sacred: Considerations when using considerations when using religious texts in natural language processing. In *NAACL-HLT*.

Zahra Khalila, Arbi Haza Nasution, Winda Monika, Aytuğ Onan, Yohei Murakami, Yasir Bin Ismail Radi, and Noor Mohammad Osmani. 2025. Investigating retrieval-augmented generation in quranic studies: A study of 13 open-source large language models. *ArXiv*, abs/2503.16581.

Johnny Li, Saksham Consul, Eda Zhou, James Wong, Naila Farooqui, Yuxin Ye, Nithyashree Manohar, Zhuxiaona Wei, Tian Wu, Ben Echols, Sharon Zhou,

and Gregory Diamos. 2024. Banishing llm hallucinations requires rethinking generalization. *ArXiv*, abs/2406.17642.

Shuyuan Lin, Lei Duan, Philip Hughes, and Yuxuan Sheng. 2025. Harnessing rlhf for robust unanswerability recognition and trustworthy response generation in llms. *arXiv preprint arXiv:2507.16951*.

Rana Malhas and Tamer Elsayed. 2020. Ayatec: building a reusable verse-based test collection for arabic question answering on the holy qur'an. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(6):1–21.

Marryam Yahya Mohammed, Sama Ayman Ali, Salma Khaled Ali, Ayad Abdul Majeed, and Ensaf Hussein Mohamed. 2025. Aftina: enhancing stability and preventing hallucination in ai-based islamic fatwa generation using llms and rag. *Neural Computing and Applications*.

Hamdy Mubarak, Rana Malhas, Watheq Mansour, Abubakr Mohamed, Mahmoud Fawzi, Majd Hawasly, Tamer Elsayed, Kareem Darwish, and Walid Magdy. 2025. IslamicEval 2025: The First Shared Task of Capturing LLMs Hallucination in Islamic Content. In *Proceedings of the Third Arabic Natural Language Processing Conference (ArabicNLP 2025)*, Suzhou, China. Association for Computational Linguistics. Co-located with EMNLP 2025, November 5–9.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. 2024. Llms know more than they show: On the intrinsic representation of llm hallucinations. *ArXiv*, abs/2410.02707.

Serry Sibaee, Abdullah I. Alharbi, Samar Ahmed, Omar Nacar, Lahouri Ghouti, and Anis Koubaa. 2024. Asos at arabic llms hallucinations 2024: Can llms detect their hallucinations :). In *OSACT*.

A Appendix

A.1 Subtask 1A

A.1.1 Fine tune model

The Figure 1 presents the loss curve obtained from the fine-tuning process on the OpenAI platform. It shows the training loss progression for the fine-tuning configuration, illustrating a gradual convergence over the training steps. This plot provides insight into the stability and efficiency of the fine-tuning process.

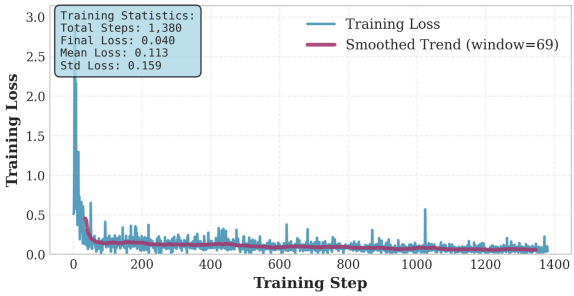

Figure 1: The plot generated from the fine-tuning loss table provided by the OpenAI platform.

A.1.2 Subtask 1A: Details results for reasoning model approach:

Model Name	Reasoning Effort	Score
GPT-5 Nano	low	0.82
O3	high	0.82
GPT-5	low	0.81
GPT-5 Nano	high	0.81
GPT-5 Nano	medium	0.81
O4 Mini	high	0.81
GPT-5	high	0.79
O3 Mini	high	0.79
GPT-5	medium	0.77
GPT-5 Mini	high	0.76
GPT-5	low	0.70
GPT-5 Mini	medium	0.65
GPT-5 Mini	high	0.63

Table 5: Performance of the AI reasoning model with access to tools was evaluated under varying levels of reasoning effort, using models of different sizes

From Table 5, we note that smaller models and tiny models can outperform larger models for such specialized tasks.

A.2 Subtask 2 evaluation on train and evaluation datasets

For Subtask 2 see Table 6, we use the organizers' code unmodified. Because train/dev lack hadith gold, our combined qrels capture Quran supervision only; hadith_sample.qrels remains empty, hence MAP_H@5 is 0 by construction. Evaluation results (merged train + dev Qrels): MAP@10=0.6199, MAP_Q@5=0.5761, MAP_H@5=0.0000 (expected given missing hadith gold).

Approach	MAP@10	MAP_Q@5	MAP_H@5
RAG-based(dev+train datasets)	0.6199	0.5761	0.0000
RAG-based(benchmark)	0.2807	0.3257	0.2386

Table 6: Subtask 2 evaluation results.