Rimal AI Technical Report

Prepared by: Raneem Alshehri, Wejdan Majid **Date: May 5, 2025**

1. Project Overview

AI-Rimal is a cutting-edge AI system engineered to process, analyze, and deliver insights on Saudi Arabian cultural and geographical data. It integrates structured and unstructured datasets and responds interactively via text, voice, image, and map-based interfaces. The system serves as both an educational tool and a cultural preservation platform.

Core Value Proposition:

- Cultural Conservation: Digitally preserves endangered aspects of Saudi heritage
- Educational Platform: Serves as an interactive learning resource for students and researchers
- Tourism Enhancement: Provides engaging, multilingual information for visitors
- Vision 2030 Alignment: Supports Saudi Arabia's digital transformation goals

2. Core Components

2.1 Data Processing

- **Dataset Integration:**
  - **Rimal AI Dataset:** 40 entries × 32 columns, includes metadata about landmarks, historical sites, traditions, and key cultural symbols.
  - **Arabic Poems Dataset:** 60 entries × 7 columns, containing poets' names, poem texts, eras, and themes.
- **Capabilities:**
  - Parses nested JSON → Pandas DataFrames.
  - Normalizes data across sources (e.g., links poems to landmarks).
  - Arabic text preprocessing (diacritics removal, script normalization).

2.2 AI/ML Components

- **Embedding Generation:**
  - Uses all-MiniLM-L6-v2 (Sentence Transformers) for light, efficient vectorization + OpenAI embeddings for multilingual support.
- **Vector Search:** FAISS index for low-latency similarity matching**.**
  - **Retrieval-Augmented Generation (RAG):**
  - Hybrid retrieval: Semantic search + GPT-4 Turbo for dynamic responses.
  - Fine-tuned prompts for:
  - Cultural context (traditions, rituals).
  - Poetry analysis (themes, historical background).
  - Landmark details (Mapbox integration).
- **LangChain Agent:**
  - Orchestrates multi-step workflows (e.g., voice → text → search → response).
  - Supports complex queries like: "Explain Janadriyah Festival and show me nearby landmarks."

2.3 Language Models

- **GPT-4 Integration:**
  - **Primary LLM for response generation**
    - Custom system messages ensure cultural sensitivity.
    - Bilingual switching within a session.
    - **HuggingFace Models**: Zero-shot intent classification.

3. Key Features

3.1 Multi-Modal Interaction

Interface	Technologies Used	Functionality
Text	Streamlit UI, LangChain	NLP intent detection, rich media responses
Voice	SpeechRecognition, gTTS/Whisper	Real-time STT/TTS with language detection
Maps	Mapbox API	Interactive maps with custom markers
Images	OpenAI DALL·E, Pillow	Poem artwork generation, media galleries

3.2 Intelligent Processing

- **Intent Classification:**
  - Supports 15+ distinct intents: from *"find me a landmark"* to *"analyze this poem"*.
- **Hybrid Search:**
  - Combines vector similarity + keyword matching.
- **Failure Handling:**
  - Structured logging + fallback responses.

3.3 Data Categories

1. **Cultural Information**
   - Traditions: weddings, festivals (e.g., Janadriyah Festival)
   - Heritage: UNESCO World Heritage Sites
   - Values & beliefs: honor codes, religious practices
2. **Geographical Data**
   - Landmarks: Al-Ula, Masmak Fortress
   - Cities: Riyadh, Jeddah, Dammam
   - Climate
   - Coordinates
3. **Literary Content**
   - Arabic poetry: Jahiliyyah era to modern works
   - Classical literature: prose, chronicles
   - Modern works: contemporary Saudi authors
   - Translations: Arabic-English equivalents for study purposes

4. Technical Implementation

4.1 Data Processing Pipeline

1. **Data Loading**
   - Parses nested JSON.
   - Converts to **Pandas DataFrames** with multi-index support.
   - Cleans and validates against predefined schemas.
2. **Embedding Generation**
   - Arabic script preprocessing (diacritics removal, normalization).
   - Vectorization via Sentence Transformers and OpenAI embeddings.
   - FAISS index creation.
3. **Query Processing**
   - **Intent classification** using zero-shot classification (Hugging Face models).
   - Semantic similarity search for retrieving context.
   - GPT-4 based **response generation**, formatted for target modality (text, voice, image, map).

4.2 API Integrations

- **OpenAI**: GPT-4, embeddings, image generation.
- **Mapbox**: Geospatial visualization.
- **Google Search**: Supplemental data retrieval.

4.3 Tools and Utilities

- **Media Retrieval**: Fetches images, audio files via pre-indexed URLs.
- **Coordinate Processing**: Converts address queries into geo-coordinates.
- **Image Generation**: Uses text-to-image models for cultural visualizations.
- **Voice Synthesis**: Outputs bilingual synthesized speech.

5. Performance Features

5.1 Search Capabilities

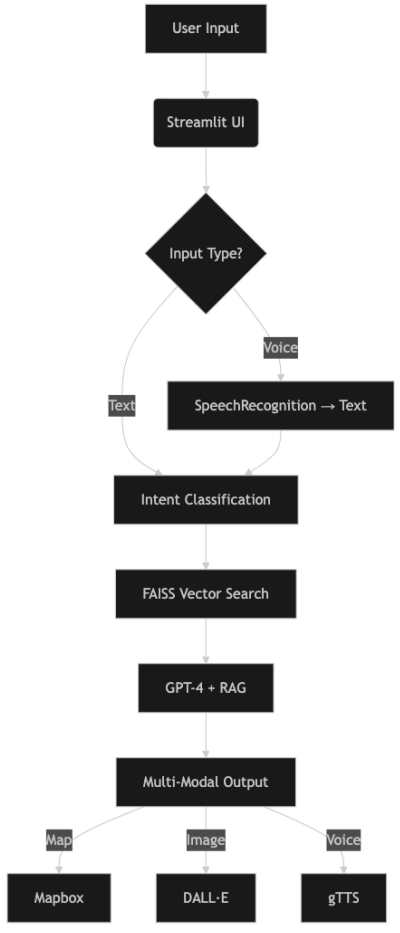
- **Semantic Search**: Finds culturally or geographically relevant results.
- **Similarity Matching**: Matches poems or landmarks by vector distance.
- **Multi-Modal Results**: Receives voice/text inputs, and returns mixed text, image, map outputs.

5.2 Response Generation

- Supports **Arabic-English switching** in one session.
- Contextual answers with source citations.
- Embeds **media links** (images/audio/video) in answers.
- Converts answers into **spoken voice output**.

6. System Architecture

This architecture merges robust AI processing with an engaging user interface while maintaining cultural accuracy and technical scalability



7. System Performance Evaluation

7.1 Evaluation Objective

The primary goal of this evaluation was to assess the performance of the system under different conditions, focusing on its accuracy and ability to handle multiple user queries simultaneously.

7.2 Evaluation Metrics

- **Accuracy Assessment**:
  The accuracy of the system was evaluated by comparing the predicted answers to the reference answers using semantic similarity techniques. The system was expected to generate answers that closely align with the reference answers, with the threshold for success set at a cosine similarity of 0.75 or higher.

- **System Load Handling (Concurrency Test):**
  The system's performance under concurrent user queries was tested to determine if it could manage multiple users without significant degradation in response time. The performance was assessed based on the average response time with 10 simultaneous users.

- **Failure Rate (Availability Test):**
  An availability test was conducted to measure the reliability of the system by simulating 100 queries. The failure rate was calculated to assess the stability and robustness of the service.

- **Cosine Similarity Calculation using Langsmith and SentenceTransformer:**
  Langsmith was used in conjunction with the SentenceTransformer model (paraphrase-MiniLM-L6-v2) to calculate the cosine similarity between the predicted answer and the reference answer. The degree of similarity was calculated based on the cosine similarity score, with a score greater than 0.75 indicating a successful match.You can find more details about Langsmith:
  - View the evaluation results for project 'ample-cook-60'
  - View all tests for Dataset rimalai-qa-test-v4

7.3 Results Summary

- **Accuracy:**
  The system demonstrated a high level of accuracy, consistently generating responses that closely matched the reference answers, with cosine similarity values typically exceeding the threshold of 0.75. This indicates that the system reliably understands and responds to queries with relevant and contextually accurate information.

- **Concurrency Test:**
  Under the load of 10 concurrent users, the system showed a slight increase in response time, but it was still able to handle the requests without major slowdowns. However, the average response time was higher than optimal, suggesting that the system's performance slightly degrades under load.

- **Availability:**
  The failure rate was minimal, with the system showing no significant failures during the test, indicating that it is highly available and reliable.

- **Cosine Similarity Calculation:**
  The system accurately assessed the similarity between the predicted and reference answers using cosine similarity. Most of the responses achieved similarity scores above the 0.75 threshold, confirming the system's ability to generate relevant and accurate answers.

7.4 Conclusions and Recommendations

- **Strengths:**
  - The system excels in providing accurate responses with a high degree of relevance.
  - It demonstrates good availability with no significant downtime during extensive query testing.
  - It can handle multiple user queries simultaneously, maintaining reasonable performance.
  - The cosine similarity approach effectively measures the relevance of generated responses.

- **Areas for Improvement:**
  - Although the system can manage multiple users, its response time increases under heavier loads. Optimizing the system to maintain consistent performance with a higher number of concurrent users is recommended.
  - Although the accuracy is high, some improvements can be made in fine-tuning the response generation to ensure more precise matches in edge cases.

- **Next Steps**
  - Focus on improving the system's response time during high load conditions.
  - Continue monitoring the performance to ensure scalability and reliability as the system grows and handles more user interactions.

8. Future Development

8.1 Planned Enhancements

- **Dataset Expansion**: Adding archaeological data, oral histories.
- **Voice Interface**: Natural conversation flow (wake words, barge-in).
- **Visualization**: Timelines, historical maps.
- **Analytics**: Track cultural trends, user behavior.
- **LangGraph**: Replace LangChain for complex workflows.

8.1 Technical Improvements

- Implement **Redis caching** for frequently accessed data.
- Optimize FAISS with **HNSW index** for faster large-scale search.
- Improve error handling with structured logging and fallback routines.

8.2 Feature Enhancements

- Add **interactive data visualizations** (e.g., D3.js timelines, map heatmaps).
- Expand **voice processing** elevenlabs.io TTS for more natural speech.
- Enable **real-time data updates** from external cultural repositories.

8.2 Integration Opportunities

- **Mobile Apps**: Native iOS/Android cultural guide.
- **Web Interfaces**: Bilingual educational portals.
- **Educational Platforms**: Integration with Saudi school curricula.
- **Research Tools**: Exportable datasets for academic work.

9. Conclusion

Rimal-AI represents a sophisticated, modular AI system. By combining semantic search, large language models, and multi-modal interfaces, it offers deep and accessible insights into Saudi Arabian culture, geography, and literature. This project is well-suited for use in cultural preservation, education, and tourism.

Future enhancements should focus on improving data quality, expanding multimodal capabilities, optimizing cost efficiency, and further personalizing the user experience. The project serves as a robust foundation for building intelligent, culturally aware AI systems for the Arabic-speaking world and beyond.

10. Recommendations

10.1 Technical Improvements

- Implement **Redis caching** for frequently accessed data.
- Optimize FAISS with **HNSW index** for faster large-scale search.
- Improve error handling with structured logging and fallback routines.

10.2 Feature Enhancements

- Add **interactive data visualizations** (e.g., D3.js timelines, map heatmaps).
- Expand **voice processing** elevenlabs.io TTS for more natural speech.
- Enable **real-time data updates** from external cultural repositories.

11. Appendix: Sample Data Structures

11.1 Rimal AI Dataset (Sample Entry)


```
{
   "type": "culture",
   "name": "Saudi Coffee (Gahwa)",
   "description": "Saudi coffee is a traditional beverage symbolizing
hospitality, often served with dates during social gatherings.",
   "rituals": [
     "Preparation with cardamom and saffron",
     "Serving in small cups called 'finjan'",
     "Ceremonial importance in welcoming guests"
   ],
   "vision2030": "Promoted as part of intangible cultural heritage to
enhance cultural tourism.",
   "values": [
     "Hospitality",
     "Generosity"
   ],
   "media": {
     "images": [
       "https://example.com/saudi_coffee.jpg"
     ],
     "audio": [],
     "videos": [
       "https://www.youtube.com/watch?v=d7qROEWIJVY",
       "https://www.youtube.com/watch?v=TT8EIySiUFs"
```

```
    ]

  },

  "id": 5
```

11.2 Arabic Poems Dataset (Sample Entry)

```
  {
      "title": "لو كنت أعلم أن الحب يقتلني",
      "poet": "جميل بثينة",
      "era": "أموي",
      "region": "الجزيرة العربية",
      "category": "غزل",
      "text": "لو كنت أعلم أن الحب يقتلني ... أعددت لي قبل أن أهوى مقابرًا",
      "translation": "Had I known love would slay me, I would have prepared my
grave before falling in love."
  },
```

12. References

- HuggingFace Sentence Transformers
- FAISS: Facebook AI Similarity Search
- OpenAI GPT-4o
- DALL-E Image Generation
- gTTS: Google Text-to-Speech
- SpeechRecognition Library
- https://elevenlabs.io/
- LangSmith