The file Code contains the markdown and details about the process. This document contains some extra things that might be needed for quick access.

## Dataset Selection and Rationale

The project uses Wikipedia articles about Robert Downey Jr. as the primary dataset, chosen for its:

1. Rich entertainment industry context and relationships
2. Multiple entity types (people, movies, awards, institutions)
3. Complex interconnected relationships
4. Diverse content spanning acting career, personal life, and achievements

## Implementation Steps

1. Environment Setup

2. Model Configuration

* **LLM**: Groq's mixtral-8x7b-32768
* **Embeddings**: HuggingFace sentence-transformers
* **Database**: Neo4j for graph storage

3. Knowledge Graph Creation Process

1. Document Loading:
   * Wikipedia content retrieval
   * Text splitting into manageable chunks
   * Token-based chunking with overlap
2. Entity and Relationship Extraction:
   * LLM-based entity identification
   * Relationship mapping between entities
   * Graph document conversion
3. Graph Structure:
   * Nodes: Person, Movie, Character, Organization
   * Relationships: ACTED\_IN , PRODUCED\_BY, MARRIED\_TO, PORTRAYED

## Graph-RAG Pipeline Components

Vector Store Configuration

* Hybrid search implementation
* Text node properties mapping
* Embedding storage and retrieval

Query Processing

* Natural language query handling
* Context preservation (optional)
* Structured response generation

## Example Usage

*# Process query*

response = chain.invoke({

"question": "When was Robert Downey Jr. born?",

"chat\_history": []

})

## Entity Types in Graph

* Person (Robert Downey Jr., Robert Downey Sr.)
* Movie (Iron Man, Avengers)
* Award (Academy Award, Golden Globe)
* Institution (Marvel Studios, Disney)

## Relationship Types

* Professional (STARRED\_IN, PRODUCED\_BY)
* Personal (CHILD\_OF, MARRIED\_TO)
* Achievement (NOMINATED\_FOR, WON\_AWARD)
* Business (CONTRACTED\_WITH, COLLABORATED\_WITH)

## Notes and Limitations

* Requires valid Groq API key
* HuggingFace embeddings run locally
* Graph visualization requires Neo4j setup

## Dependencies

* langchain-groq
* sentence-transformers
* neo4j
* langchain-experimental