

## **General Questions**

1. **Q:** What does this application do?  
**A:** It transforms unstructured movie data into structured insights using AI, enabling conversational queries with a graph and vector database.
2. **Q:** Who can use this application?  
**A:** Movie analysts, researchers, studios, OTT platforms, and anyone interested in extracting insights from scripts, reviews, or character dialogues.
3. **Q:** How do I interact with the agent?  
**A:** Through a web interface built with React, where you can chat with the AI agent that uses Neo4j and ChromaDB to answer queries.
4. **Q:** Can I upload my own scripts?  
**A:** No, not as of right now. One of our future aspects include scripts or dialogues in text format can be ingested, chunked, embedded, and queried.
5. **Q:** Is this real-time?  
**A:** The system provides near real-time responses to queries, but ingestion of large datasets happens offline.

## **Data Processing & Storage**

6. **Q:** How are scripts processed?  
**A:** Scripts are split into semantic chunks, embedded into vector space, and linked with metadata such as characters, movies, and genres.
7. **Q:** What database is used?  
**A:** Neo4j for graph-based relationships and ChromaDB for vector similarity search.
8. **Q:** Why both Neo4j and ChromaDB?  
**A:** Neo4j captures structured relationships (movies → actors → awards), while ChromaDB captures semantic similarity of dialogues and text.
9. **Q:** How are reviews handled?  
**A:** Reviews are ingested, analyzed for sentiment using RoBERTa, and stored as Review nodes linked to Movie nodes.
10. **Q:** How are character dialogues stored?  
**A:** Each character's dialogue is chunked into smaller segments, embedded, and stored in ChromaDB with metadata like movie, character, and scene.

## **Features**

11. **Q:** Can I track sentiment arcs for characters?  
**A:** Yes, the system computes character sentiment over time, showing emotional highs and lows across the movie.
12. **Q:** Can I query by genres?  
**A:** Yes, movies are linked to genres in Neo4j, so you can ask things like “*Suggest comedies with romance.*”
13. **Q:** Does it recommend movies?  
**A:** Yes, the system can suggest similar movies based on embeddings, genres, or graph connections.
14. **Q:** Can I ask award-related queries?  
**A:** Yes, awards are stored as nodes connected to movies, allowing queries like “*Which movies won an Oscar after 2000?*”
15. **Q:** What about companies and producers?  
**A:** The system models production companies and producers as nodes linked to movies, enabling organization-level insights.

## **Technical (Backend)**

16. **Q:** Which LLM is used?  
**A:** We use Groq’s hosted LLMs like deepseek-r1-distill-llama-70b integrated via LangChain.
17. **Q:** What’s the backend framework?  
**A:** Flask serves as the backend API that communicates with Neo4j, ChromaDB, and the LLM.
18. **Q:** How are embeddings generated?  
**A:** Using all-MiniLM-L6-v2 from SentenceTransformers for compact yet powerful sentence embeddings.
19. **Q:** How is semantic chunking done?  
**A:** Using LangChain’s SemanticChunker which splits text based on embedding similarity rather than fixed length.
20. **Q:** Is GPU acceleration used?  
**A:** Yes, CUDA-enabled GPUs accelerate embedding generation and sentiment analysis.

## **Technical (Frontend)**

21. **Q:** What frontend framework is used?  
**A:** React (via Vite) is used for a fast and modular frontend.
22. **Q:** How is chat displayed?  
**A:** The chat uses a custom React component with markdown support and feedback options (   ).
23. **Q:** Can markdown responses render?  
**A:** Yes, markdown is rendered so bold, italics, and lists display properly.
24. **Q:** Can I give feedback?  
**A:** Yes, thumbs up/down allow users to signal satisfaction, and the agent can retry based on thumbs down.
25. **Q:** How does routing work?  
**A:** React Router handles navigation between pages like Home, Features, and Chat.

## **ML Models**

26. **Q:** What models are used for regression tasks?  
**A:** XGBoost Regressor and Random Forest Regressor predict Metacritic scores.
27. **Q:** What models are used for classification tasks?  
**A:** Logistic Regression (OvR), XGBoost Classifier, and Random Forest Classifier predict awards.
28. **Q:** What is binary prediction used for?  
**A:** Classifying whether a movie is award-winning or not based on metadata and reviews.
29. **Q:** Which NLP techniques are used?  
**A:** TF-IDF, BERT embeddings, and SentenceTransformers embeddings.
30. **Q:** What metrics are used?  
**A:** RMSE, MAE, Precision, Recall, F1, Accuracy, and ROC-AUC depending on the task.

## **Benefits**

31. **Q:** What insights can I gain?  
**A:** Character arcs, sentiment trends, movie similarities, award likelihood, and metadata-driven queries.

32. **Q:** How does this save time?

**A:** Analysts no longer need to manually parse through text; insights are generated instantly.

33. **Q:** Can SMEs interact with the data?

**A:** Yes, subject matter experts can query naturally in plain English.

34. **Q:** Does it help storytelling?

**A:** Yes, by analyzing sentiment arcs and dialogue themes, writers can improve narrative flow.

35. **Q:** How does it help researchers?

**A:** Provides a structured way to analyze large script datasets for academic or industrial research.

## ***Challenges & Solutions***

36. **Q:** How do you handle large datasets?

**A:** By parallel processing and batching embeddings with GPU acceleration.

37. **Q:** How is speed optimized?

**A:** We batch embeddings, use Chroma's persistent client, and cache results where possible.

38. **Q:** How do you ensure graph consistency?

**A:** We merge nodes instead of overwriting, ensuring no data loss.

39. **Q:** How do you handle data quality?

**A:** By cleaning, normalizing, and validating metadata before ingestion.

40. **Q:** How do you avoid LLM hallucinations?

**A:** By grounding responses in ChromaDB context and Neo4j queries via RAG.

## ***Business Use Cases***

41. **Q:** How can studios use this?

**A:** To analyze scripts pre-production, track audience sentiment, and plan award campaigns.

42. **Q:** How can OTT platforms use this?

**A:** To recommend movies or analyze their catalog for gaps and trends.

43. **Q:** How can educators use this?

**A:** For teaching narrative structure, sentiment analysis, or computational storytelling.

44. **Q:** How does it help SMEs?

**A:** SMEs can query insights conversationally instead of relying on SQL or Cypher skills.

45. **Q:** Can this be commercialized?

**A:** Yes, as a SaaS platform, APIs, or enterprise solution for content analytics.

## **Meta**

46. **Q:** What makes this solution unique?

**A:** The integration of graph + vector + LLM pipelines for both structured and unstructured insights.

47. **Q:** How scalable is it?

**A:** Very — embeddings scale horizontally, and Neo4j handles billions of relationships.

48. **Q:** Is it extensible?

**A:** Yes, more metadata (like screenplay annotations) can be integrated.

49. **Q:** What if I don't solve this problem?

**A:** You lose valuable insights, recommendations, and automation, and rely on manual analysis.

50. **Q:** What's the long-term vision?

**A:** To build an AI-powered knowledge graph for the entertainment industry that powers research, recommendations, and creative tools.

## **Team Information:**

51. **Q:** Who are the team members?

**A:** The team was led by Anurag Panda alongside his teammates Shreyasi Dey, Asish, Ayush, Abha, Chandrika, Sainee and Ashit.