Narrative Explanation Generation

Narrative explanation generation is a complex task that involves extracting meaningful narratives from a dataset and providing corresponding contextual explanations. The task is particularly relevant in the domains of literary analysis, historical document analysis, and thematic exploration of large corpora. It demands systems capable of understanding both the semantics of narratives and their contextual relevance, often requiring a combination of structured relationships and semantic similarity.

This work aims to build a hybrid retrieval system that leverages both **semantic embeddings** and **knowledge graphs** to achieve accurate and contextually enriched narrative retrieval. The key objectives are:

1. To develop a **vector-based retrieval** system for semantic similarity search.
2. To construct a **knowledge graph** that encodes relationships among narratives, explanations, and their associated metadata.
3. To integrate these methods into a **hybrid retrieval mechanism** that combines the strengths of both approaches, further enhancing the retrieval process using reranking mechanisms.

The introduction of a hybrid system addresses challenges such as:

- Understanding thematic connections beyond simple keyword matches.
- Structuring and querying relational data for more precise narrative retrieval.
- Combining semantic and structural methods to optimize relevance and contextual alignment.

Knowledge Graph Construction

The knowledge graph was built to encode and represent relationships between narratives, explanations, authors, and books. The process involved the following steps:

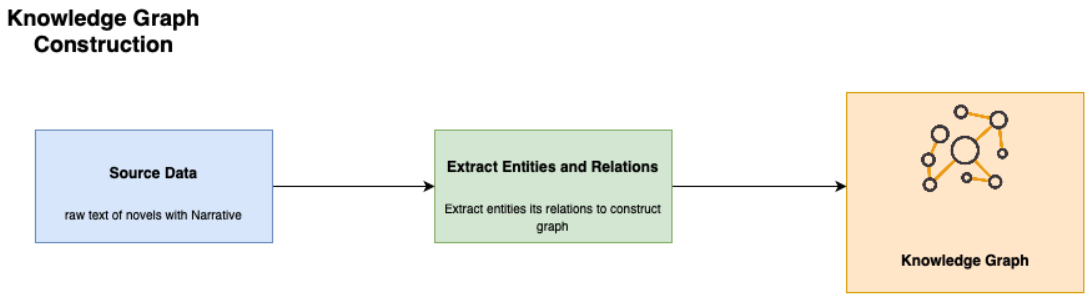

Fig 1: *Overview of Knowledge Graph Construction*

Graph Schema Design

The schema of the knowledge graph was designed to represent key entities and relationships:

- **Entities**:
  - Author: Represents the creator of literary works.
  - Book: Represents a specific work of an author.
  - Narrative: Captures thematic or event-driven descriptions.
  - Explanation: Provides contextual insights for each narrative.
- **Relationships**:
  - Author → WROTE → Book
  - Book → HAS → Narrative

Narrative Explanation Generation

- Narrative → HAS → Explanation

The data preparation process began with structured datasets where narratives and explanations were mapped to specific authors and books, with standardized fields including Author, Book, Narrative, and Explanation. To enable graph-based retrieval, the data was ingested into Neo4j through a custom script that parsed each dataset row to create nodes for unique authors, books, narratives, and explanations. Relationships were then established based on the schema using a Cypher query. The query ensured connections such as authors linked to their books, books to their narratives, and narratives to their corresponding explanations. This design leverages Neo4j's graph traversal capabilities for efficient querying, enabling complex searches like identifying narratives by a specific author that contain certain keywords. By structuring these relationships explicitly, the graph supports flexible and context-rich queries, offering deep insights into connected data and enhancing the retrieval process with structured contextual depth.

Built on the principles of semantic and knowledge graphs [Reference 4], this architecture uses labeled relationships to provide meaningful context and interconnected data. Semantic graphs represent entities and their relationships, while knowledge graphs extend this by mirroring real-world connections to support reasoning and advanced queries. This structure ensures precise, contextually enriched retrieval and a deeper understanding of linked data.

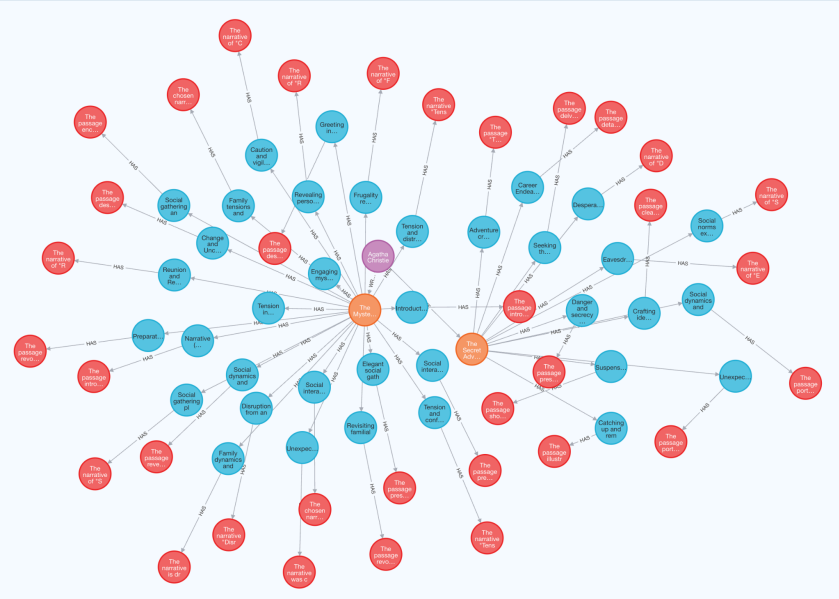

Fig 2: *Overview of Knowledge Graph*

Vector-Based Retrieval and Reranking

Vector-based retrieval and reranking were employed to enhance semantic search capabilities [Reference 3]. Narratives were first encoded into high-dimensional vectors using the pre-trained all-MiniLM-L6-v2 model from the SentenceTransformers library, which maps textual data into a semantic space where similar texts are represented closer together. These embeddings were indexed using FAISS (Facebook AI Similarity Search), with an IndexFlatL2 configuration optimized for L2 distance-based similarity.

Narrative Explanation Generation

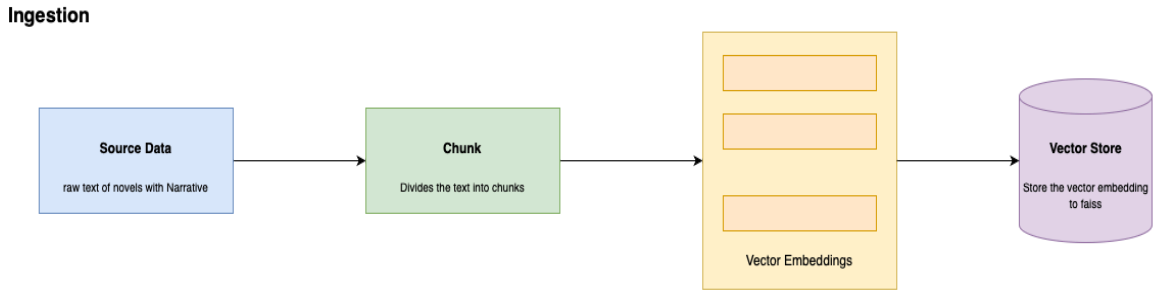

Fig 3: *Data Ingestion in Vectorstore*

During querying, the input text was encoded into a query embedding, which was searched against the FAISS index to retrieve the top-k closest narratives.

Re-ranking

The reranking component is essential in refining the initial results retrieved using vector-based methods[Reference 2]. While FAISS efficiently retrieves the top-k narratives based on vector proximity, this ranking relies solely on semantic similarity in the embedding space, which might overlook deeper contextual nuances. To enhance precision, a cross-encoder model, specifically **cross-encoder/ms-marco-MiniLM-L-6-v2**, was introduced to directly evaluate the interaction between the query and each retrieved narrative. Unlike bi-encoders that separately compute embeddings for queries and documents, the cross-encoder jointly processes query-narrative pairs, capturing fine-grained relationships and contextual relevance with greater accuracy.

For every query-narrative pair, the cross-encoder generates a relevance score, quantifying the narrative's alignment with the query's intent. This additional layer of evaluation ensures that narratives most pertinent to the query are ranked higher, regardless of their original position in the FAISS retrieval. By combining the speed and scalability of FAISS with the precision of the cross-encoder, this dual-stage system addresses potential gaps in semantic-only retrieval, providing a robust and adaptable framework for delivering contextually relevant results across diverse queries and large-scale datasets.

Hybrid Retrieval with Graph and Vector Store

Graphs are particularly effective for structured queries that rely on relationships among entities. For example, they are well-suited for tasks such as retrieving all narratives authored by a specific writer or extracting explanations associated with narratives containing specific keywords. Neo4j's capability to traverse relationships ensures that results are highly relevant in context. However, graphs have limitations when it comes to capturing semantic similarity. This gap is effectively addressed by vector-based methods, which enable similarity searches within high-dimensional semantic spaces and capture thematic nuances and subtle textual variations that graphs might overlook.

Narrative Explanation Generation

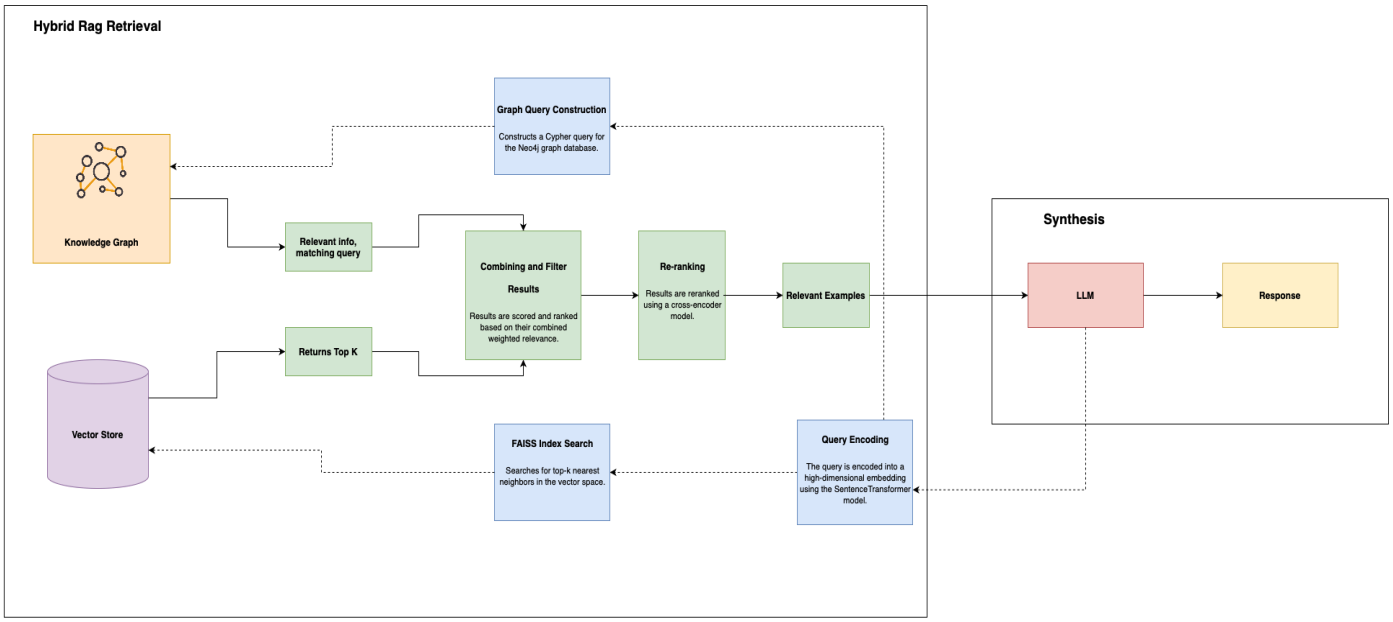

Fig 4: *Overview of Narrative Explanation pipeline*

To enhance retrieval performance and leverage the strengths of multiple approaches, a hybrid retrieval system [Reference 1] was implemented. This system integrates **graph querying**, which retrieves narratives and explanations based on explicit relational criteria, with **vector querying**, which identifies top-k semantically similar narratives using **FAISS**. The results from both methods are combined using a weighted scoring mechanism:

$$Score = (Graph\,Weight \times Graph\,Score) + (Vector\,Weight \times Vector\,Store)$$

Graph weights were set higher (e.g., 0.6) to prioritize relational accuracy, while vector weights (e.g., 0.4) captured semantic richness. To further refine the results, duplicate narratives from both sources were filtered out, and the combined results were optionally reranked using a cross-encoder to ensure alignment with the query context. This hybrid approach combines the complementary strengths of relational precision provided by graph querying with the semantic adaptability of vector querying. It ensures retrievals that are both contextually and semantically relevant, reducing reliance on a single method and delivering a robust and balanced performance.

Results:

Evaluating language models for explanation generation requires a nuanced approach to ensure outputs meet standards of readability, contextual relevance, and linguistic diversity. The **Flesch Reading Ease metric** measures readability by analyzing sentence length and syllable count, favoring shorter sentences and simpler words for broader accessibility [Reference 7]. Similarly, the **Gunning Fog Index assesses** text complexity by emphasizing the prevalence of complex words and sentence length, ensuring a balance between simplicity and informativeness, especially in technical contexts [Reference 7]. **Entropy** captures linguistic diversity, where moderate levels indicate engaging and coherent explanations, avoiding monotony or incoherence [Reference 8]. **Semantic Similarity** evaluates alignment with the source context by comparing text embeddings, ensuring relevance and coherence while avoiding hallucinations [Reference 6]. **Topic**

Narrative Explanation Generation

Coherence, using methods like Latent Dirichlet Allocation (LDA), measures the consistency of the explanations with dominant topics derived from the input [Reference 5].

Together, these metrics provide a robust framework for evaluating explanation quality, offering objective insights even without predefined ground truth. Addressing readability, complexity, diversity, and contextual alignment, this approach ensures explanations are accessible, coherent, and contextually appropriate. Such comprehensive evaluation fosters trust and satisfaction among users, making the framework versatile across domains and adaptable to diverse user needs.

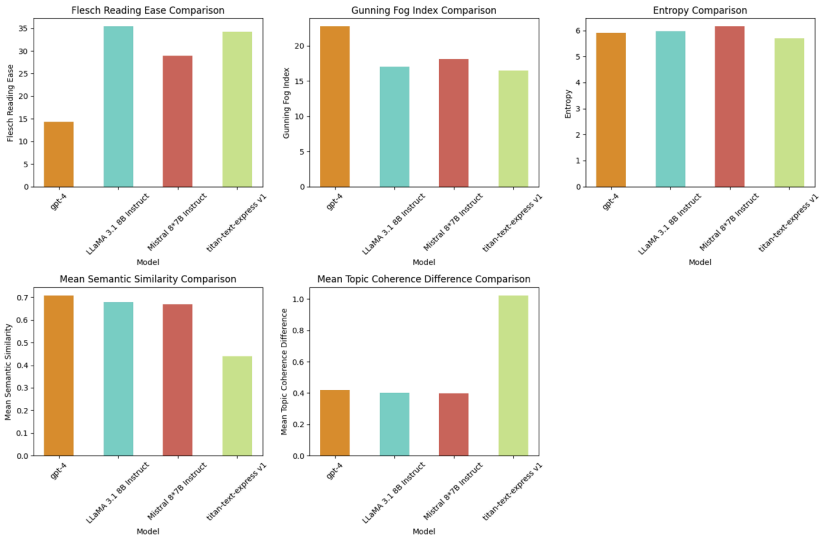

Fig 5: *LLM's performance comparison on evaluation metrics Entropy, Flesch Reading Ease, Gunning Fog Index, Semantic Comparison.*

Model	Mean Semantic Similarity	Mean Topic Coherence Difference	Flesch Reading Ease	Gunning Fog Index	Entropy
gpt-4	**0.70**	**0.41**	14.33	22.75	5.91
LlaMA 3.1 8B instruct	0.678	0.40	**35.43**	17.01	5.97
Mistral 8×7B instruct	0.670	0.39	28.94	18.13	**6.15**
Titan-text-express v1	0.43	1.02	34.23	**16.51**	5.69

Table 1 : *Performance Comparison of LLM's*

The evaluation of the models shows distinct strengths and weaknesses across various metrics. GPT-4 achieves the highest **Mean Semantic Similarity** (0.70) and the lowest **Mean Topic Coherence Difference** (0.41), indicating strong alignment and coherence in its explanations. However, its **Flesch Reading Ease** score (14.33) and **Gunning Fog Index** (22.75) suggest that its outputs are more complex and less accessible. LLaMA 3.1 8B Instruct provides more readable content

Narrative Explanation Generation

(Flesch: 35.43, Fog: 17.01) while maintaining moderate semantic similarity (0.678). Mistral 87B balances coherence (0.39) and readability (Flesch: 28.94, Fog: 18.13) but sacrifices semantic similarity (0.670). Titan-text-express v1, while the simplest in terms of **Entropy** (5.69), lags significantly in semantic similarity (0.43) and shows a high topic coherence difference (1.02), indicating less consistency and alignment in its explanations.

While metrics like Flesch Reading Ease and Gunning Fog Index effectively measure textual complexity, they fall short in capturing deeper linguistic elements such as tone, style, or cultural context. Similarly, embedding-based measures for semantic similarity rely heavily on the quality of the underlying embedding model, often struggling with subtle contextual nuances. Topic coherence, while valuable for assessing focus and relevance, depends on the robustness of models like LDA, which may underperform with short or highly diverse texts. These limitations highlight the need for a balanced approach in using these metrics alongside more nuanced qualitative evaluations.

Reference:

1. @misc{sarmah2024hybridragintegratingknowledgegraphs,
title={HybridRAG: Integrating Knowledge Graphs and Vector Retrieval Augmented Generation for Efficient Information Extraction},
author={Bhaskarjit Sarmah and Benika Hall and Rohan Rao and Sunil Patel and Stefano Pasquali and Dhagash Mehta},
year={2024},
eprint={2408.04948},
archivePrefix={arXiv},
primaryClass={cs.CL},
url={https://arxiv.org/abs/2408.04948},
}

2. @misc{dong2024dontforgetconnectimproving,
title={Don't Forget to Connect! Improving RAG with Graph-based Reranking},
author={Jialin Dong and Bahare Fatemi and Bryan Perozzi and Lin F. Yang and Anton Tsitsulin},
year={2024},
eprint={2405.18414},
archivePrefix={arXiv},
primaryClass={cs.CL},
url={https://arxiv.org/abs/2405.18414},
}
3. @misc{lewis2021retrievalaugmentedgenerationknowledgeintensivenlp,
title={Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks},
author={Patrick Lewis and Ethan Perez and Aleksandra Piktus and Fabio Petroni and Vladimir Karpukhin and Naman Goyal and Heinrich Küttler and Mike Lewis and Wen-tau Yih and Tim Rocktäschel and Sebastian Riedel and Douwe Kiela},
year={2021},
eprint={2005.11401},
archivePrefix={arXiv},
primaryClass={cs.CL},

Narrative Explanation Generation

url={https://arxiv.org/abs/2005.11401},
}
4. @InProceedings{lystopadskyi_et_al:OASIcs.SLATE.2023.9,
author = {Lystopadskyi, Daniil and Santos, Andr\'{e} and Leal, Jos\'{e} Paulo},
title = {{Narrative Extraction from Semantic Graphs}},
booktitle = {12th Symposium on Languages, Applications and Technologies (SLATE 2023)},
pages = {9:1--9:8},
series = {Open Access Series in Informatics (OASIcs)},
ISBN = {978-3-95977-291-4},
ISSN = {2190-6807},
year = {2023},
volume = {113},
editor = {Sim\~{o}es, Alberto and Ber\'{o}n, Mario Marcelo and Portela, Filipe},
publisher = {Schloss Dagstuhl -- Leibniz-Zentrum f{\"u}r Informatik},
address = {Dagstuhl, Germany},
URL = {https://drops.dagstuhl.de/entities/document/10.4230/OASIcs.SLATE.2023.9},
URN = {urn:nbn:de:0030-drops-185231},
doi = {10.4230/OASIcs.SLATE.2023.9},
annote = {Keywords: Narratives, Narrative Extraction, Information Retrieval, Knowledge Graphs, Semantic Graphs, Resource Description Framework, Web Ontology}
}

5. @inproceedings{10.1145/2684822.2685324,
author = {R\"{o}der, Michael and Both, Andreas and Hinneburg, Alexander},
title = {Exploring the Space of Topic Coherence Measures},
year = {2015},
isbn = {9781450333177},
publisher = {Association for Computing Machinery},
address = {New York, NY, USA},
url = {https://doi.org/10.1145/2684822.2685324},
doi = {10.1145/2684822.2685324},
abstract = {Quantifying the coherence of a set of statements is a long standing problem with many potential applications that has attracted researchers from different sciences. The special case of measuring coherence of topics has been recently studied to remedy the problem that topic models give no guaranty on the interpretablity of their output. Several benchmark datasets were produced that record human judgements of the interpretability of topics. We are the first to propose a framework that allows to construct existing word based coherence measures as well as new ones by combining elementary components. We conduct a systematic search of the space of coherence measures using all publicly available topic relevance data for the evaluation. Our results show that new combinations of components outperform existing measures with respect to correlation to human ratings. nFinally, we outline how our results can be transferred to further applications in the context of text mining, information retrieval and the world wide web.},
booktitle = {Proceedings of the Eighth ACM International Conference on Web Search and Data Mining},
pages = {399–408},
numpages = {10},
keywords = {topic coherence, topic evaluation, topic model},

Narrative Explanation Generation

location = {Shanghai, China},
series = {WSDM '15}
}

6. @misc{mikolov2013efficientestimationwordrepresentations,
title={Efficient Estimation of Word Representations in Vector Space},
author={Tomas Mikolov and Kai Chen and Greg Corrado and Jeffrey Dean},
year={2013},
eprint={1301.3781},
archivePrefix={arXiv},
primaryClass={cs.CL},
url={https://arxiv.org/abs/1301.3781},
}

7. @misc{lee2024traditionalreadabilityformulascompared,
title={Traditional Readability Formulas Compared for English},
author={Bruce W. Lee and Jason Hyung-Jong Lee},
year={2024},
eprint={2301.02975},
archivePrefix={arXiv},
primaryClass={cs.CL},
url={https://arxiv.org/abs/2301.02975},
}
8. @misc{wei2024differanknovelrankbasedmetric,
title={Diff-eRank: A Novel Rank-Based Metric for Evaluating Large Language Models},
author={Lai Wei and Zhiquan Tan and Chenghai Li and Jindong Wang and Weiran Huang},
year={2024},
eprint={2401.17139},
archivePrefix={arXiv},
primaryClass={cs.LG},
url={https://arxiv.org/abs/2401.17139},
}