Fact-Checking System using GraphRAG

Preyali Dave, Aditya Tanna, Kirtan Soni and Krish Rupapara

Abstract

Given the widespread dissemination of misinformation on social media, implementing fact-checking mechanisms for online claims is essential. Manually checking every claim is not efficient. Here, we have presented a fact-checking system based on GraphRAG. We utilize the Fever dataset to assess the performance of our system.

Keywords

Fact-checking system, GraphRAG, Large Language Models,

1. Introduction

The total amount of fake news and misinformation has increased enormously in recent years. False claims have, in some cases, created riots and even caused the loss of life. This problem is particularly prevalent in times like the elections. Given the vast volume of online content, manually fact-checking every claim is not possible. Therefore, we have implemented a fact-checking system with the help of GraphRAG [1].

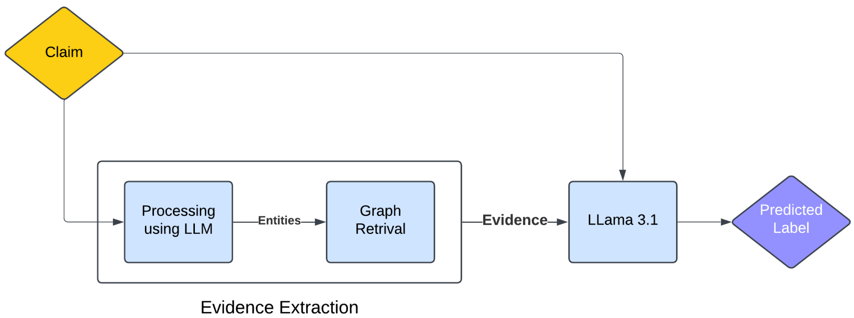

Figure 1: Overview diagram of our system

In our approach, we predict if the claim is True or False with the help of external knowledge. We have also leveraged several recent advances, such as knowledge graphs and large language models (LLMs). In figure 1, we have shown the high-level overview of our system.

2. Dataset

In our system, we used two datasets. We have used Wikidata5m dataset for the knowledge graph. In this dataset, there are around 5 million entities and 822 relations which helped to create our knowledge graph. For our evaluation purposes, we have used the Fever dataset [2]. In the fever data set, the label was given for each claim, from Supports, Refutes, and Not Enough Info.

Introduction to Information Retrieval (IT550), Autumn 2024, DAIICT, India
202318028 (P. Dave); 202103023 (A. Tanna); 202101197 (K. Soni); 202101198 (K. Rupapara)

3. Methodology

Given a knowledge, our system is comprised of three components: entity extraction, evidence extraction from knowledge graph, and veracity prediction based on the extracted evidence. The first two components form our GraphRAG pipeline. In figure 2, we have shown the entire methodology in detail with the example.

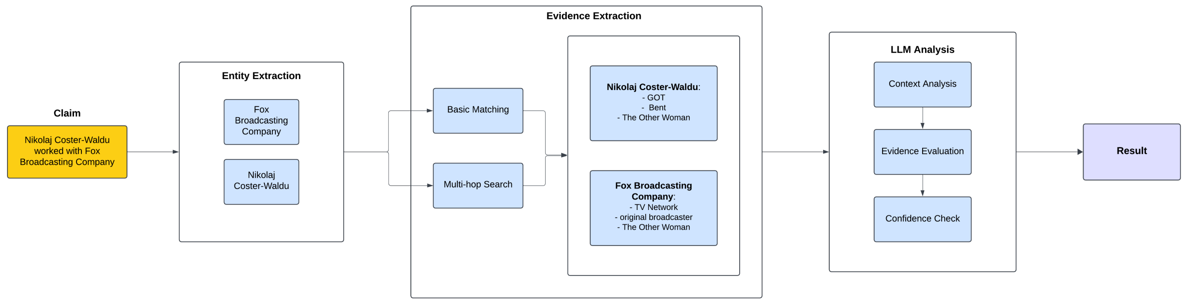

Figure 2: The methodology of our system with an example

3.1. Entity Extraction using LLM

Usually, claim contains unnecessary words like stop words and some redundant information. So, we have used the Large Language Model, LLama 3.1:8b to extract entity from the claim. So, that we can easily get the evidence based on those entities from the knowledge graph.

3.2. Evidence extraction from Knowledge Graph

After the entity extraction, we try to find the relations between two entities from our knowledge graph. If there is no direct relationship between two entities, we have used multi-hoping to get the indirect relationship between them. Since we are using a knowledge graph, we can get these indirect relationship, which would be difficult to get in the regular RAG method. In our system, we have limited ourselves to 2 hops, due to dense relationships in our knowledge graph.

3.3. Fact-checking based on the evidence

The relationships extracted from the previous method is taken as evidence of each pair of entities. From this evidence, we find if the given claim is true or not, using the LLM model, Llama 3.1:8b. For this, we give the LLM model, the original claim and these relationships. From the given input, LLM identifies if the given claim is true or not, without any external knowledge.

4. Evaluation methods and Result

Since, we are predicting whether the given claim is true or not, we have used evaluation methods like precision, recall and accuracy. We have done our experiment on both with only basic matching and with basic matching with multi-hoping case. We have shown our result in table 1.

5. Challenges and Learning

In our approach, one of the biggest challenge was lack of resources. Due to that, we couldn't experiment with more advanced LLMs. We also struggled with finding the exact reason of decrease in recall on the multi-hoping approach. While making this system, we had chance to learn the knowledge graph and neo4j. We also learned more about RAGs and how to implement them.

Table 1
Result of our system for both approaches

Evaulation Metrics	Basic Matching	Multi-hoping
Precision	0.91	0.82
Recall	0.82	0.68
Accuracy	0.92	0.94

6. Further Improvements

This system can have some few improvements to get much better results. Our system suffers, if the extracted entity doesn't exist in the knowledge graph. We can also try using embedding rather than direct matching to get the similar entities rather than the exact ones. We can also try enhanced LLMs to further get the better results.

References

[1] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, J. Larson, From local to global: A graph rag approach to query-focused summarization, 2024. URL: https://arxiv.org/abs/2404.16130. arXiv:2404.16130.

[2] J. Thorne, A. Vlachos, C. Christodoulopoulos, A. Mittal, Fever: a large-scale dataset for fact extraction and verification, 2018. URL: https://arxiv.org/abs/1803.05355. arXiv:1803.05355.