FRI Natural language processing course 2025
Project report

University *of Ljubljana*
Faculty *of Computer and Information Science*

Conversational Agent with Retrieval-Augmented Generation

Blaž Špacapan, Matevž Jecl, Tilen Ožbot

Abstract

The abstract goes here. The abstract goes here.

Keywords
RAG, web-scraping

Advisors: Slavko Žitnik

Introduction

Various fields such as customer service, education, and even general knowledge have shifted greatly with the introduction of sophisticated language models as conversational agents. Even with all the advancements made in the industry, conventional chatbots depend purely on pre-trained static knowledge, which can quickly become outdated, making it impossible to retrieve accurate information. The goal of this project is to address that gap by creating a chatbot that automatically improves its accuracy and responsiveness by web-scraping related news articles from the web, in real-time. The proposed conversational agent utilizes Retrieval-Augmented Generation (RAG) techniques to dynamically access current information, ensuring the responses provided are accurate and up to date.

Master's thesis from Chalmers University of Technology exploring enhancements to conversational agents with Retrieval-Augmented Generation by integrating an autoregressive Large Language Model that generates real-time dense retrieval vectors, significantly improving multi-hop reasoning through synthetic data generation and attention-based relabeling to reduce hallucinations [1].

Research from ESP Journal of Engineering & Technology Advancements introduces a conversational Retrieval-Augmented Generation framework designed for real-time crisis management, integrating structured and unstructured data sources to deliver validated, contextually relevant insights that enhance emergency decision-making and response effectiveness [2].

In this article, the authors propose Retrieval-Augmented Generation (RAG) models that integrate parametric seq2seq generation with non-parametric retrieval mechanisms, demonstrating improved performance in conversational agents through enhanced factual accuracy and response specificity in knowledge-intensive interactions [3].

The paper proposes a misinformation detection framework, FCRV (Full-Context Retrieval and Verification), that integrates claim extraction via Large Language Models (LLMs) with Retrieval-Augmented Generation (RAG) to construct a comprehensive context for news verification, significantly improving detection accuracy, robustness, and scalability against both human and AI-generated fake news [4].

The paper demonstrates how Retrieval-Augmented Generation (RAG) with Large Language Models (LLMs) effectively addresses challenges in extracting precise business event information from diverse and evolving data sources, significantly enhancing adaptability and accuracy in dynamic business environments [5].

Methods

Data

News content will be collected exclusively by web-scraping major outlets (BBC News, CNN, The Guardian, 24ur) as well as regional RSS feeds to ensure broad topical coverage. Each script retrieves full article text and metadata (title, author, publication date, URL) and stores the raw content for processing. A preprocessing pipeline then performs HTML parsing, text normalization, duplicate removal, and formatting. Cleaned articles are converted into vector representations and indexed in a vector database to enable efficient semantic retrieval. This approach produces a continuously updated, semantically organized corpus of both current and historical news articles, supporting rapid access to relevant documents for the conversational agent.

0.1 Database

The system utilizes a PostgreSQL database enhanced with the `pgvector` extension to handle vector embeddings efficiently. Instead of the previous two-table structure, it now employs a single primary table named `content_chunks`. This table is designed to store individual text fragments (`chunk_text`) directly alongside their corresponding vector embeddings (`embedding`, using the `VECTOR` type provided by `pgvector`).

Pipeline

We selected `rtvslo.si` for initial testing because it offers robust filtering by category and date. For this phase, we focus exclusively on sports news. We use the following parameters when querying the `rtvslo.si` API:

- `q`: the search query string
- `s`: section filter (e.g., `3` for sports, `null` for all)
- `sort`: sort order (`1` = newest first, `2` = most popular)
- `a`: time range (`1` = all time, `2` = last 24 h, `3` = last week, `4` = last month, `5` = last year)
- `per_page`: number of results to return per page
- `group`: content type group (`1` = news, `15` = video, `16` = audio)

1. **Receive user query**: The system accepts a natural-language question or request from the user.
2. **Extract keywords from the query**: Three complementary methods identify salient terms:
   - *Stop-Word Filtered Token Extractor*: tokenizes the lowercased question, filters for alphanumeric tokens, and removes Slovene stop words.
   - *Part-of-Speech Keyword Extractor*: uses a spaCy NLP model to collect tokens tagged as PROPN (proper noun), ADJ (adjective), or NOUN.
   - *Named-Entity Extractor*: uses spaCy to extract named entities (e.g., persons, organizations, locations).
3. **Perform web-scraping to discover relevant articles**: Using the extracted keywords, the scraper queries selected news pages to retrieve potentially relevant articles. Since `robots.txt` is not provided by the site, we enforce a 1 s timeout between requests to avoid overwhelming the server.
4. **Parse and collect article metadata and content**: Relevant content from newly found articles is processed, converted into vector representations (embeddings), and stored in a database. The system retrieves a predefined number of the most semantically similar information chunks from the entire database (including newly added ones) based on the user's query embedding.
5. **Chunk Reranking**: The initially retrieved information chunks undergo a refinement step where they are re-evaluated and re-ordered based on their direct relevance to the specific user query, ensuring the most pertinent pieces are prioritized.
6. **LLM-based Answer Synthesis**: Finally, a large language model receives the refined, relevant information chunks and the original user query (potentially along with conversation history). It synthesizes this information to generate a comprehensive and contextually appropriate answer.

Equations

You can write equations inline, e.g. $\cos\pi = -1$, $E = m \cdot c^2$ and α, or you can include them as separate objects. The Bayes's rule is stated mathematically as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \qquad (1)$$

where A and B are some events. You can also reference it – the equation 1 describes the Bayes's rule.

Lists

We can insert numbered and bullet lists:

1. First item in the list.
2. Second item in the list.
3. Third item in the list.

- First item in the list.
- Second item in the list.
- Third item in the list.

We can use the description environment to define or describe key terms and phrases.

Word What is a word?.

Concept What is a concept?

Idea What is an idea?

Random text

This text is inserted only to make this template look more like a proper report. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam blandit dictum facilisis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Interdum et malesuada fames ac ante ipsum primis in faucibus. Etiam convallis tellus velit, quis ornare ipsum aliquam id. Maecenas tempus mauris sit amet libero elementum eleifend. Nulla nunc orci, consectetur non consequat ac, consequat non nisl. Aenean vitae dui nec ex fringilla malesuada. Proin elit libero, faucibus eget neque quis, condimentum laoreet urna. Etiam at nunc quis felis pulvinar dignissim. Phasellus turpis turpis, vestibulum eget imperdiet in, molestie eget neque. Curabitur quis ante sed nunc varius dictum non quis nisl. Donec nec lobortis velit. Ut cursus, libero efficitur dictum imperdiet, odio mi fermentum dui, id vulputate metus velit sit amet risus. Nulla vel volutpat elit. Mauris ex erat, pulvinar ac accumsan sit amet, ultrices sit amet turpis.

Phasellus in ligula nunc. Vivamus sem lorem, malesuada sed pretium quis, varius convallis lectus. Quisque in risus nec lectus lobortis gravida non a sem. Quisque et vestibulum sem, vel mollis dolor. Nullam ante ex, scelerisque ac efficitur vel, rhoncus quis lectus. Pellentesque scelerisque efficitur purus in faucibus. Maecenas vestibulum vulputate nisl sed vestibulum. Nullam varius turpis in hendrerit posuere.

Figures

You can insert figures that span over the whole page, or over just a single column. The first one, Figure 1, is an example of a figure that spans only across one of the two columns in the report.

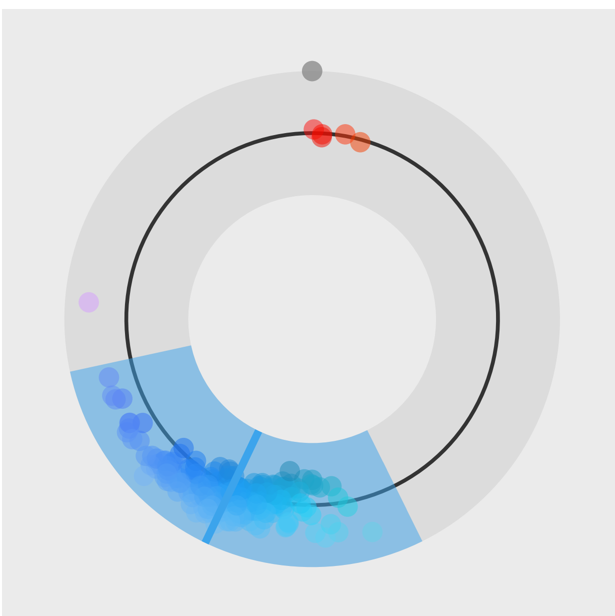

Figure 1. A random visualization. This is an example of a figure that spans only across one of the two columns.

On the other hand, Figure 2 is an example of a figure that spans across the whole page (across both columns) of the report.

Tables

Use the table environment to insert tables.

Table 1. Table of grades.

Name		
First name	Last Name	Grade
John	Doe	7.5
Jane	Doe	10
Mike	Smith	8

Code examples

You can also insert short code examples. You can specify them manually, or insert a whole file with code. Please avoid inserting long code snippets, advisors will have access to your repositories and can take a look at your code there. If necessary, you can use this technique to insert code (or pseudo code) of short algorithms that are crucial for the understanding of the manuscript.

Listing 1. Insert code directly from a file.

```
import os
import time
import random

fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

Listing 2. Write the code you want to insert.

```
import(dplyr)
import(ggplot)

ggplot(diamonds,
          aes(x=carat, y=price, color=cut)) +
  geom_point() +
  geom_smooth()
```

Results

Use the results section to present the final results of your work. Present the results in a objective and scientific fashion. Use visualisations to convey your results in a clear and efficient manner. When comparing results between various techniques use appropriate statistical methodology.

More random text

This text is inserted only to make this template look more like a proper report. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam blandit dictum facilisis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Interdum et malesuada fames ac ante ipsum primis in faucibus. Etiam convallis tellus velit, quis ornare ipsum aliquam id. Maecenas tempus mauris sit amet libero elementum eleifend. Nulla nunc orci, consectetur non consequat ac, consequat non nisl. Aenean vitae dui nec ex fringilla malesuada. Proin elit libero, faucibus eget neque quis, condimentum laoreet urna. Etiam at nunc quis felis pulvinar dignissim. Phasellus turpis turpis, vestibulum eget imperdiet in, molestie eget neque. Curabitur quis ante sed

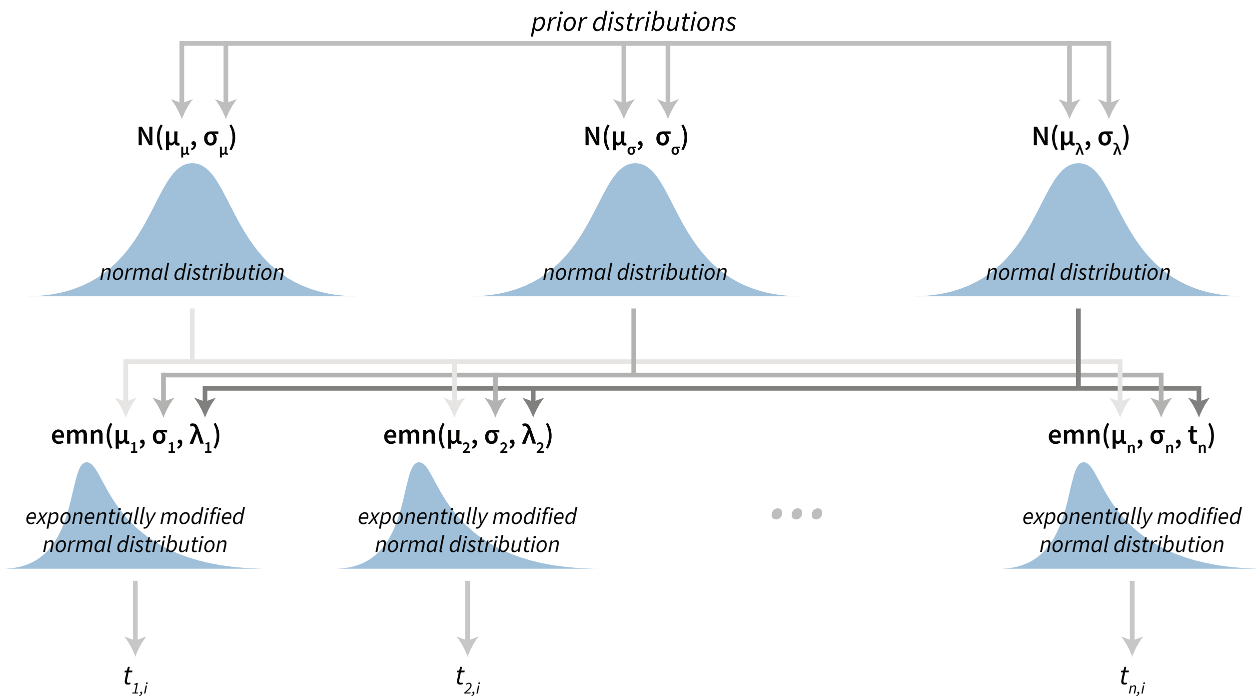

Figure 2. Visualization of a Bayesian hierarchical model. This is an example of a figure that spans the whole width of the report.

nunc varius dictum non quis nisl. Donec nec lobortis velit. Ut cursus, libero efficitur dictum imperdiet, odio mi fermentum dui, id vulputate metus velit sit amet risus. Nulla vel volutpat elit. Mauris ex erat, pulvinar ac accumsan sit amet, ultrices sit amet turpis.

Phasellus in ligula nunc. Vivamus sem lorem, malesuada sed pretium quis, varius convallis lectus. Quisque in risus nec lectus lobortis gravida non a sem. Quisque et vestibulum sem, vel mollis dolor. Nullam ante ex, scelerisque ac efficitur vel, rhoncus quis lectus. Pellentesque scelerisque efficitur purus in faucibus. Maecenas vestibulum vulputate nisl sed vestibulum. Nullam varius turpis in hendrerit posuere.

Nulla rhoncus tortor eget ipsum commodo lacinia sit amet eu urna. Cras maximus leo mauris, ac congue eros sollicitudin ac. Integer vel erat varius, scelerisque orci eu, tristique purus. Proin id leo quis ante pharetra suscipit et non magna. Morbi in volutpat erat. Vivamus sit amet libero eu lacus pulvinar pharetra sed at felis. Vivamus non nibh a orci viverra rhoncus sit amet ullamcorper sem. Ut nec tempor dui. Aliquam convallis vitae nisi ac volutpat. Nam accumsan, erat eget faucibus commodo, ligula dui cursus nisi, at laoreet odio augue id eros. Curabitur quis tellus eget nunc ornare auctor.

Discussion

Use the Discussion section to objectively evaluate your work, do not just put praise on everything you did, be critical and exposes flaws and weaknesses of your solution. You can also explain what you would do differently if you would be able to start again and what upgrades could be done on the project in the future.

Acknowledgments

Here you can thank other persons (advisors, colleagues ...) that contributed to the successful completion of your project.

References

[1] Malte Landgren and Oskar Giljegård. Real-time Relevance: RAG with Dynamic Context for Improved Natural Language Responses. Master's thesis, Chalmers University of Technology, 2024.

[2] Abhinav Balasubramanian. Esp-jeta. *ESP Journal of Engineering & Technology Advancements*, 4(4):118–126, 11 2024.

[3] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020.

[4] Yangxiao Bai and Kaiqun Fu. A large language model-based fake news detection framework with rag fact-

checking. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8617–8619, 2024.

[5] Muhammad Arslan, Saba Munawar, and Christophe Cruz. Exploring business events using multi-source rag. *Procedia Computer Science*, 246:4534–4540, 2024. 28th International Conference on Knowledge Based and Intelligent information and Engineering Systems (KES 2024).