



Mineração de Dados

# LLM como guia turístico para o Norte de Portugal

Relatório de Desenvolvimento

Pedro Pereira Sousa PG54721  
Mateus Medeiros Pereira PG54089  
Nuno Miguel Leite da Costa PG54121  
Elione Culeca Cossengue PG51634

**Grupo 2**  
MINERAÇÃO DE DADOS 2023/2024  
UNIVERSIDADE DO MINHO  
JUNE 14, 2024

## Contents

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Métodos</b>	<b>2</b>
2.1	Dados . . . . .	3
2.2	LLM . . . . .	4
2.3	Funcionamento . . . . .	4
2.4	Interface . . . . .	6
2.5	Exemplos . . . . .	6
<b>3</b>	<b>Resultados e Discussão</b>	<b>6</b>
3.1	Benchmark . . . . .	6
3.2	Discussão dos Resultados . . . . .	9
<b>4</b>	<b>Conclusão e Trabalho Futuro</b>	<b>9</b>

## List of Figures

1	Etapas do sistema . . . . .	3
2	Resposta do TravelAgent à prompt presente na imagem seguinte . . . . .	6
3	Exemplo de um prompt na nossa interface . . . . .	6
4	<i>Chunk</i> recuperado da RAG relacionado com "some activities" . . . . .	6

## List of Tables

# 1 Introdução

No âmbito do trabalho prático da unidade curricular de Mineração de Dados, foi-nos proposto o desenvolvimento de uma aplicação ou estudo que implementasse a integração de um modelo de linguagem (LLM), para que este fosse refinado através da recolha e processamento de dados de fontes públicas. Desta forma, o grupo optou por basear o seu trabalho num LLM que servisse como guia turístico para várias localizações no Norte de Portugal.

Esta decisão surgiu quando estávamos a testar o ChatGPT com várias perguntas acerca de localizações em Braga, e apercebemo-nos das suas respostas de baixa qualidade. Começamos a pensar em várias maneiras de integrar o conceito de RAG (*Retrieval-Augmented Generation*) com um guia turístico, e, após uma discussão entre os membros do grupo, o tópico ficou decidido.

Os nossos objetivos durante o desenvolvimento deste projeto foram criar um guia turístico que fornecesse informações precisas sobre diversas localizações do Norte de Portugal, adaptando-se às preferências do utilizador, sejam elas locais ou atividades específicas.

## Estado da Arte

Atualmente, o estado da arte no que toca ao desenvolvimento de guias turísticos inteligentes aborda modelos que fornecem informações de alta qualidade e relevância sobre diversas localidades. Uma abordagem chave é a RAG (*Retrieval-Augmented Generation*), que recupera documentos relevantes de uma grande base de dados vetorial para informar o processo de geração, garantindo informações precisas e atualizadas sobre pontos turísticos e atividades.

Modelos de recuperação baseados em BERT [3], como BM25, DPR e ColBERT, são amplamente utilizados para melhorar a precisão da recuperação, gerando embeddings com BERT. Modelos híbridos, que combinam métodos de recuperação baseados em redes neurais e tradicionais baseados em termos, equilibram precisão e abrangência, garantindo uma cobertura abrangente das informações turísticas.

Grafos de conhecimento estruturam e representam informações sobre locais turísticos, aprimorando a compreensão semântica e a relevância contextual das respostas geradas. Sistemas avançados de perguntas e respostas, utilizando modelos pré-treinados como o T5 ou o GPT-4 ajustados para domínios específicos, fornecem respostas precisas a consultas dos utilizadores.

Algoritmos de personalização, como filtragem colaborativa e filtragem baseada em conteúdo, são usados para adaptar recomendações com base nas preferências dos utilizadores, criando um itinerário personalizado. Modelos de IA contextual, como o ChatGPT, ajustados com dados específicos do domínio, também geram respostas contextualmente relevantes e coerentes.

Além disso, a IA multimodal, que combina texto com imagens, mapas e possivelmente áudio, oferece uma experiência de utilizador mais rica e envolvente. Essas técnicas de ponta coletivamente contribuem para o desenvolvimento de guias turísticos inteligentes e adaptativos, que fornecem informações precisas e relevantes sobre várias localidades.

Adicionalmente, algumas empresas têm usado LLMs comercialmente para gerar guias turísticos, como é o exemplo do website *llmtemplates.ai* [2].

## 2 Métodos

O programa é dividido em duas fases. A primeira consiste no povoamento da RAG que vai ser utilizada pelo Agente para dar informação adicional à LLM, de maneira a produzir itinerários customizados às preferências do utilizador. A segunda consiste no processamento da query do utilizador, obtendo os dados relevantes da RAG e comunicando com a LLM para produzir o itinerário. A imagem 1 demonstra o funcionamento das etapas.

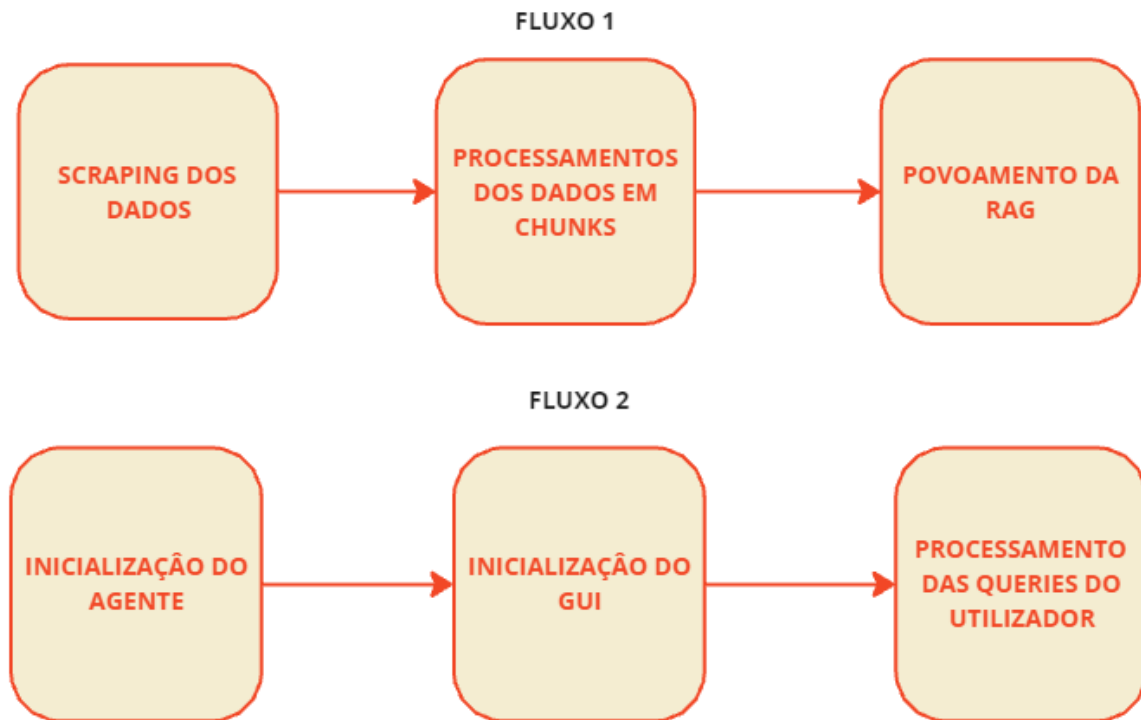


Figure 1: Etapas do sistema

## 2.1 Dados

Antes de explicar a origem dos dados usados na nossa aplicação, temos de referir que o ChatGPT e o modelo que usamos através da API, o GPT3.5, já contém dados sobre várias localidades do Norte de Portugal. No entanto, esses dados estão geralmente incompletos e, como podemos ver na secção das Benchmarks, há uma grande tendência para o modelo alucinar localidades e não ser preciso geograficamente, criando um trajeto impossível de ser realizado pelo utilizador.

Assim, decidimos criar uma RAG com dados extraídos do website VisitPortugal . Estes dados foram descarregados na forma de vários pdfs, através da funcionalidade existente no website. Devido ao tempo que demorava a criação do pdf, tivemos que dividir os alojamentos em vários ficheiros diferentes.

Através da biblioteca *fitz* e *langchain*, a informação das entidades em cada um dos ficheiros foi posteriormente processada e convertida em *chunks* para serem introduzidos na RAG. Os *chunks* foram introduzidos nas bases de dados correspondentes a cada entidade. As entidades que criamos foram acomodações, atividades, praias, casinos, jardins, museis, parques temáticos, e zoos ou aquários.

Inicialmente as entidades eram guardadas na mesma base de dados, mas decidimos subdividir em bases de dados diferentes porque tivemos problemas relativamente a devolver chunks relevantes à query do utilizador. Por exemplo, poderia acontecer que o utilizador pedisse um itinerário que envolvesse zoos, praias e casinos, e os chunks mais parecidos com a query do utilizador fossem todos da entidade praia. Assim, sendo, depois de passar os chunks à LLM, esta não teria informação suficiente sobre zoos e casinos para criar um itinerário adequado ao cliente.

Desta forma, através de um passo adicional de interação com a LLM, foi-nos possível identificar quais entidades o utilizador pede no itinerário, e portanto devolver os chunks correspondentes às entidades pedidas.

## 2.2 LLM

Como mencionado anteriormente, a LLM que usamos ao longo de todo o projeto foi o **gpt-3.5-turbo**. Utilizamos este modelo através da API da OpenAI [4], que conectamos ao nosso programa através da biblioteca da mesma empresa, incluída na *Langchain*. Esta integração permitiu-nos aceder de forma eficiente às capacidades avançadas de processamento de linguagem natural do GPT-3.5, facilitando a implementação de funcionalidades complexas.

Para melhorar a precisão e a relevância das respostas fornecidas pelo modelo, utilizamos os dados da RAG. Esta RAG utiliza uma função de similaridade para comparar a prompt do utilizador com as várias entradas presentes nas bases de dados, garantindo que as informações mais pertinentes sejam utilizadas na geração das respostas. Para além disso, implementamos as bases de dados vetoriais necessárias para o RAG utilizando a biblioteca *ChromaDb*, que consiste numa ferramenta eficaz para criar e gerir bases de dados vetoriais, permitindo-nos indexar e consultar grandes volumes de dados de forma eficiente. Com a *ChromaDb*, conseguimos armazenar e recuperar informações com base em similaridades de alto nível, o que melhorou significativamente a capacidade do nosso sistema em fornecer respostas relevantes e precisas.

## 2.3 Funcionamento

Quanto ao funcionamento da app, começamos por criar um agente que realiza três importantes ações, após a prompt do utilizador. Cada uma dessas ações é introduzida na LLM sequencialmente através de uma system prompt que dá as informações necessárias à LLM para obter a resposta desejada, com um formato específico.

### 1º Passo

O 1º passo no funcionamento da App consiste em **validar** a prompt do utilizador. Ou seja, o agente verifica se é possível criar o percurso desejado pelo utilizador no tempo que lhe foi fornecido. Fazemos isto através da seguinte system prompt:

---

```
You are a travel agent who helps users make exciting travel plans.

The user's request will be denoted by four hashtags. Determine if the user's
request is reasonable and achievable within the constraints they set.

A valid request should contain the following:
- A start and end location
- A trip duration that is reasonable given the start and end location
- Some other details, like the user's interests and/or preferred mode of transport

If the request is not valid, set
plan_is_valid = 0 and use your travel expertise to update the request to make it valid,
keeping your revised request shorter than 100 words.

If the request seems reasonable, then set plan_is_valid = 1 and
don't revise the request.
```

---

### 2º Passo

Já no 2º passo, queremos identificar todas as entidades que estão presentes na query do utilizador. Com entidades estamos-nos a referir essencialmente aos vários pontos de interesse que temos guardados na base de dados, como os hotéis, praias, ou monumentos, como anteriormente referido. Isto tudo é feito através da prompt:

---

You are an agent who detects the presence of certain entities in user requests.  
The user's request will be denoted by four hashtags.  
Determine which entities are present in the user's request.  
The variable of each entity group followed by their entities are the following:

- var1: any type of accomodation
- var2: beaches
- var3: activities
- var4: gardens, parks and forests
- var5: thematic parks
- var6: zoos or aquariums
- var7: casinos
- var8: museums, monuments, and touristic sites

For each group of entities above, if the user's request mentions any of the entities, set the respective variable to True, otherwise, set the variable to False.

For example:

####

Give me an itinerary in Braga. Also suggest hotels, monuments, and aquariums that we can go to.

###

In this example, you set var1 = True, var6 = True, and var8 = True. The other entities you set to False, because the user didn't request them.

---

De seguida, chamamos um função de comparação que compara a query do utilizador com os vários chunks das bases de dados correspondentes às entidades identificadas, e retorna os que forem mais semelhantes.

### 3º Passo

No último passo, o nosso objetivo consiste agora em finalizar o processo de criação do itinerário. Ao utilizar a query do utilizador bem como os chunks retirados através do RAG, a LLM gera uma série de locais para o utilizador visitar, assim como as entidades pedidas pelo utilizador. Desta forma, as recomendações são personalizadas e relevantes, proporcionando uma experiência satisfatória e adaptada às preferências individuais do utilizador. Isto é realizado através da prompt:

---

You are a travel agent who helps users make exciting travel plans.

The user's request will be denoted by four hashtags. Convert the user's request into a detailed itinerary describing the places they should visit and the things they should do.

Try to include the specific address of each location.

Remember to take the user's preferences and timeframe into account, and give them an itinerary that would be fun and realistic given their constraints.

Try to make sure the user doesn't need to travel for more than 8 hours on any one day during their trip.

Return the itinerary as a bulleted list with clear start and end locations and mention the type of transit for the trip.

If specific start and end locations are not given, choose ones that you think are suitable and give specific addresses.

Your output must be the list and nothing else.

---

## 2.4 Interface

Para comunicar com a LLM de uma forma fácil e intuitiva implementamos uma interface com o Tkinter, uma biblioteca de Python para criar GUIs. O nosso objetivo ao criar a GUI foi, de uma certa maneira, imitar o estilo do ChatGPT, apesar de parecer rudimentar.

## 2.5 Exemplos

Nesta secção, serão apresentados alguns exemplos que demonstram o funcionamento da nossa solução desenvolvida.

```
TravelAgent:

- Day 1:
  - Start your day in Aveiro.
  - Travel by bus to Parque da Devesa in Vila Nova de Famalicão.
  - Address: Rua Fernando Mesquita, nº 2453 – D, Antas 4760-034 Vila Nova de Famalicão
  - Enjoy a leisurely walk in the park, explore the green spaces, and learn about the biodiversity.
  - Visit the lake, centenary oaks, and the educational buildings.
  - Have lunch at the park's restaurant.
  - Travel by bus to Bragança.
  - Address: R. N. Sra da Lapa, 27 - Lagomar 5300-514 Bragança
  - Check-in to your accommodation.
  - Relax and prepare for the next day.

- Day 2:
  - Start your day with a guided tour by Bétula Tours.
  - Address: R. N. Sra da Lapa, 27 - Lagomar 5300-514 Bragança
  - Engage in activities like boat trips, walking, nature watching, and birdwatching in Montesinho Natural Park or Douro Internacional Natural Park.
  - Explore remote locations and enjoy the natural beauty of the region.
  - Have a picnic lunch in a picturesque spot.
  - Return to Bragança in the evening and end your trip.
```

Figure 2: Resposta do TravelAgent à prompt presente na imagem seguinte

```
User:
I want to do a 2 day trip from Aveiro to Bragança.
I want to visit remote locations, gardens, and do some activities.
```

Figure 3: Exemplo de um prompt na nossa interface

De seguida, é apresentada uma imagem que diz respeito a um **chunk** da RAG em resposta à prompt presente na figura anterior.

```
Bétula Tours
Address: R. N. Sra da Lapa, 27 - Lagomar 5300-514
Bragança
Telephone: +351 273 326 290 / 960 237 459
E-mail: info@antoniosa.com Website:
http://www.antoniosa.com/tours
Other informations:
Photo tours
```

Figure 4: *Chunk* recuperado da RAG relacionado com "some activities"

## 3 Resultados e Discussão

### 3.1 Benchmark

Para avaliar a qualidade do nosso guia turístico baseado em um Modelo de LLM e identificar oportunidades de melhoria, realizamos um benchmarking comparativo. Este benchmark envolveu a

comparação do desempenho do nosso sistema com outras soluções similares disponíveis no mercado, bem como uma análise das respostas do nosso LLM antes e depois da integração com a nossa RAG. A metodologia adotada para o benchmarking consistiu em selecionar perguntas típicas que um turista poderia fazer sobre locais no Norte de Portugal. Testar essas perguntas em três sistemas diferentes:

- Nosso guia turístico com integração RAG.
- ChatGPT (versão padrão, sem integração com a nossa RAG).
- A LLM Mistral Small.

**Os parâmetros utilizados para comparar os sistemas foram:**

- Precisão das Respostas: Avaliada pela exatidão e completude das informações fornecidas.
- Relevância das Respostas: Medida pela pertinência das informações em relação à pergunta feita.

**Nossa query:**

**User:** I want to do a 3 day roadtrip starting in Porto and finishing in Braga, Portugal. In the first day I want to visit beaches and touristic sites. In the second day I want to go to a zoo in the afternoon, and a casino in the evening. In the third day I want to spend my day in the hotel. Build me an itinerary.

**Resultado do nosso Guia de viagens:**

- 
- Day 1:
    - Start your road trip in Porto.
    - Visit Praia da Foz in Porto to enjoy the beach and the ocean view.
      - Address: Praia da Foz, Porto.
    - Explore the touristic sites in Porto.
    - Drive towards Braga.
    - Stay overnight at Casa da Ponte do Porto in Braga.
      - Address: Av. Ponte do Porto, 47, 4710-730 Braga.
  - Day 2:
    - Visit Zoo Santo Inácio in Vila Nova de Gaia, Porto.
      - Address: Rua 5 de Outubro, 4503, 4430-809 Vila Nova de Gaia.
    - In the evening, head to Casino da Póvoa in Póvoa de Varzim.
      - Address: Edifício do Casino da Póvoa de Varzim, 4490-403 Póvoa de Varzim.
    - Stay overnight at Casa da Ponte do Porto in Braga.
  - Day 3:
    - Spend the day relaxing at Casa da Ponte do Porto in Braga.
      - Address: Av. Ponte do Porto, 47, 4710-730 Braga.
    - End your road trip in Braga.
- 

**Resultado do GPT3.5:**

---

### Day 1: Porto to Viana do Castelo

**\*\*Morning:\*\***



- Visit Matosinhos Beach and Praia da Luz in Porto.
- Explore Ribeira Square and Porto Cathedral.

**\*\*Afternoon:\*\***

- Drive to Viana do Castelo (approx. 1.5 hours).
- Visit Santa Luzia Sanctuary and Praia do Cabedelo.

**\*\*Evening:\*\***

- Check into your hotel in Viana do Castelo.
- Dinner at a local seafood restaurant.

**### Day 2: Viana do Castelo to Braga**

**\*\*Morning:\*\***

- Depart for Vila Nova de Gaia.
- Visit Zoo de Santo Inácio (approx. 1.5 hours).

**\*\*Afternoon:\*\***

- Drive to Braga (approx. 1 hour).
- Check into your hotel and have lunch.

**\*\*Evening:\*\***

- Visit Casino de Braga.
- Enjoy dinner and entertainment at the casino.

**### Day 3: Braga**

**\*\*Morning & Afternoon:\*\***

- Relax at the hotel spa or pool.
- Visit Bom Jesus do Monte and Braga Cathedral.

**\*\*Evening:\*\***

- Dinner at a traditional Portuguese restaurant in Braga.
- 

## Resultado do Mistral Small:

---

Sure, here's a suggested itinerary for your 3-day road trip:

**\*\*Day 1: Beaches and Touristic Sites\*\***

- Start your day in Porto and head towards Matosinhos (10 minutes drive), where you can enjoy a morning walk along the beach and try some traditional Portuguese seafood for breakfast.
- After breakfast, drive to Miramar Beach (20 minutes from Matosinhos), known for its beautiful views and calm waters.
- Next, head to Póvoa de Varzim (10 minutes from Miramar Beach), a charming coastal town with a historic center and a beautiful beach. You can have lunch here.
- In the afternoon, visit the São Félix Hill Fortress in Vila do Conde (20 minutes from Póvoa de Varzim), offering panoramic views of the city and the Atlantic Ocean.
- End your day with a visit to the Guimarães Castle and the Historical Centre of Guimarães (30 minutes from Vila do Conde), a UNESCO World Heritage site.
- Drive to Braga for the night (30 minutes from Guimarães).

**\*\*Day 2: Zoo and Casino\*\***

- In the morning, explore the city of Braga. You can visit the Bom Jesus do Monte, a famous pilgrimage site, and the Braga Cathedral.

- After lunch, drive to the Paços de Ferreira Zoo (20 minutes from Braga). It's a small zoo, but it's a great place to spend the afternoon.
- In the evening, head to the Casino da Póvoa (30 minutes from Paços de Ferreira). Don't forget to dress up as there's a dress code.
- Return to your hotel in Braga.

#### **\*\*Day 3: Relaxation Day\*\***

- Spend your day relaxing at your hotel in Braga. You can use the hotel facilities, such as the spa, pool, or gym.
  - You could also explore the local area around your hotel, do some shopping, or try some local cuisine.
  - In the evening, you could have a nice dinner at one of the restaurants in Braga.
- 

### **3.2 Discussão dos Resultados**

Os resultados indicam que a integração de uma RAG utilizando a ChromaDB e a LangChain melhorou substancialmente a qualidade das respostas fornecidas pelo nosso guia turístico. Nomeadamente, o GPT3.5 alucinou, recomendando uma viagem ao casino de Braga (que, como sabemos, não existe), e o Mistral igualmente produziu informações erradas, como por exemplo afirmando tanto a existência do "Zoo de Passos de Ferreira", como também mencionando que demora 20 minutos uma viagem entre Varzim e Vila do Conde.

Adicionalmente, podemos observar que o nosso modelo foi bastante preciso nas localizações enunciadas, informando-nos das moradas, ruas, e códigos postais, enquanto que os outros modelos não conseguiram ter tal nível de precisão. Com isto podemos concluir que a capacidade de pesquisar e fornecer informações relevantes e precisas contribuiu para uma *user experience* superior em comparação com outras LLMS como o GPT3.5 usado no ChatGPT, ou até o Mixtral, da Mistral.

## **4 Conclusão e Trabalho Futuro**

Consideramos que o trabalho realizado cumpriu as nossas expectativas e com os objetivos estabelecidos, servindo como método eficaz para consolidar as temáticas abordadas na UC de Mineração de Dados. Podemos confiantemente afirmar que concluímos a cadeira com muito mais conhecimento sobre LLMs, bem como das bibliotecas que se usam para interagir com elas.

Relativamente ao trabalho futuro, há um reconhecimento da necessidade de aprimorar a usabilidade e a estética da interface do utilizador, para garantir uma experiência mais intuitiva e agradável aos utilizadores. Outra hipótese poderá passar em utilizar bibliotecas que já criam uma interface automaticamente, ou ainda usar uma aplicação de desktop como LM Studio ou open-webui, entre outras. Finalmente, há a intenção de adicionar novos dados às entidades existentes, visando enriquecer e expandir as capacidades da solução, tornando-a ainda mais robusta e abrangente.

## **References**

- [1] VisitPortugal. Website do VisitPortugal que contém os nossos dados. <https://www.visitportugal.com/pt-pt/search/site?regioes=285&context=490>
- [2] Website do llmtemplates.ai <https://www.llmtemplates.ai/>
- [3] BERT da Google <https://arxiv.org/abs/1810.04805>
- [4] API da OpenAI <https://openai.com/index/openai-api/>