

# Uso de LLM e RAG para construção de um chatbot voltado a potenciais turistas e moradores de João Pessoa

Filipe Trindade de Oliveira



CENTRO DE INFORMÁTICA  
UNIVERSIDADE FEDERAL DA PARAÍBA

João Pessoa, 2025



Filipe Trindade de Oliveira

Uso de LLM e RAG para construção de um chatbot  
voltado a potenciais turistas e moradores de João  
Pessoa

Monografia apresentada ao curso Ciência da Computação  
do Centro de Informática, da Universidade Federal da Paraíba,  
como requisito para a obtenção do grau de Bacharel em Ciência da Computação

Orientador: Moisés Dantas dos Santos

Maio de 2025

**Catálogo na publicação**  
**Seção de Catalogação e Classificação**

O48u Oliveira, Filipe Trindade de.  
    Uso de LLM e RAG para construção de um chatbot voltado a potenciais turistas e moradores de João Pessoa / Filipe Trindade de Oliveira. - João Pessoa, 2025.  
    70 f. : il.

Orientação: Moisés dos Santos.  
TCC (Graduação) - UFPB/CI.

1. Chatbot. 2. Turismo. 3. João Pessoa. 4. Grandes modelos de linguagem. 5. Grafo de conhecimento. I. Santos, Moisés dos. II. Título.

UFPB/CI

CDU 004.8



CENTRO DE INFORMÁTICA  
UNIVERSIDADE FEDERAL DA PARAÍBA

Trabalho de Conclusão de Curso de Ciência da Computação intitulado ***Uso de LLM e RAG para construção de um chatbot voltado a potenciais turistas e moradores de João Pessoa*** de autoria de Filipe Trindade de Oliveira, aprovada pela banca examinadora constituída pelos seguintes professores:

---

Prof. Dr. Moisés Dantas dos Santos  
Universidade Federal da Paraíba

---

Prof. Dra. Tatiana Araújo Simões  
Universidade Federal da Paraíba

---

Prof. Mestre. Manoel de Pontes Fernandes  
Secretaria de Estado da Educação da Paraíba

---

Coordenador(a) do Departamento Centro de Informática  
Giorgia de Oliveira Mattos  
CI/UFPB

João Pessoa, 24 de maio de 2025



*Os limites da minha linguagem significam os limites do meu mundo.*



## **DEDICATÓRIA**

Dedico esse trabalho a todos aqueles que estão presentes na minha vida e me apoiam: meus pais, Marcos e Luciana, meu irmão Tiago, meus outros familiares, meus amigos íntimos e minha namorada Aline. Que eu sempre possa compartilhar meu sucesso com vocês. Sou e sempre serei grato por tudo.

## **AGRADECIMENTOS**

Agradeço a todos aqueles que contribuíram, de alguma forma, em minha jornada acadêmica, seja na escola ou na faculdade. Colegas, professores, coordenadores, todos que me influenciaram positivamente para que eu chegasse aqui. Em especial, ao professor Moisés e aos meus colegas do TRIL, que me deram suporte e ajudaram em meu caminho de aprendizado recente.

## RESUMO

Este trabalho apresenta o desenvolvimento de um protótipo de agente conversacional baseado em um grande modelo de linguagem (Large Language Model – LLM), incorporando técnicas avançadas de RAG (Retrieval-Augmented Generation, ou geração aumentada de recuperação) para aprimorar seu desempenho. O agente foi concebido para atuar no contexto da cidade de João Pessoa, na Paraíba, escolhida por seu crescente destaque no cenário nacional em termos de turismo e qualidade de vida. Essa escolha torna a cidade um excelente estudo de caso para a implementação de um agente conversacional com base em LLM. Este demonstrou boa capacidade conversacional baseada nas métricas de avaliação propostas, indicando seu potencial como meio de divulgação da cidade.

**Palavras-chave:** chatbot; turismo; João Pessoa; grandes modelos de linguagem; grafo de conhecimento.

## ABSTRACT

This work presents the development of a prototype conversational agent (chatbot) based on a Large Language Model (LLM), incorporating advanced Retrieval-Augmented Generation (RAG) techniques to enhance its performance. The agent was designed to operate within the context of João Pessoa, a city in the state of Paraíba, Brazil, which was chosen due to its growing national prominence in tourism and quality of life. This makes the city an excellent case study for implementing an LLM-based conversational agent. The chatbot demonstrated strong conversational abilities according to the proposed evaluation metrics, indicating its potential as a means of promoting the city.

**Key-words:** chatbot; tourism; João Pessoa; large language models; knowledge graph.

## LISTA DE FIGURAS

1	Esquema do <i>pipeline</i> do agente, dividido em subagentes. . . . .	40
2	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério. . . . .	44
3	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema bairro. . . . .	45
4	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema imóveis. . . . .	46
5	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema informações gerais da cidade. . . .	47
6	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema locais na cidade. . . . .	48
7	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema notícias. . . . .	49
8	Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema transporte público. . . . .	50
9	Exemplo de conversa simples, onde inicialmente se pede para que o agente construa um plano de viagem curta (3 dias). . . . .	53
10	Exemplo de conversa simples, onde inicialmente pergunta-se sobre o bairro Mangabeira. . . . .	54
11	Exemplo de conversa simples, onde inicialmente pergunta-se ao agente sobre bairros acessíveis para morar. . . . .	55
12	Exemplo de conversa simples, onde inicialmente pede-se sugestões de praia para serem visitadas. . . . .	55

## LISTA DE TABELAS

1	Critérios de avaliação da corretude das respostas . . . . .	42
2	Critérios de avaliação da relevância das respostas . . . . .	42
3	Critérios de avaliação da compreensibilidade das respostas . . . . .	43
4	Resultados da avaliação das respostas do agente . . . . .	44
5	Resultados da avaliação das respostas do agente para perguntas relaciona- das a bairros . . . . .	45
6	Resultados da avaliação das respostas do agente para perguntas relaciona- das a imóveis . . . . .	46
7	Resultados da avaliação das respostas do agente para perguntas relaciona- das a informações gerais da cidade . . . . .	47
8	Resultados da avaliação das respostas do agente para perguntas relaciona- das a locais e tipos de locais na cidade . . . . .	48
9	Resultados da avaliação das respostas do agente para perguntas relaciona- das a notícias na cidade . . . . .	49
10	Resultados da avaliação das respostas do agente para perguntas relaciona- das ao transporte público na cidade . . . . .	50
11	Sites utilizados como base de conhecimento do agente . . . . .	63
12	Notícias, artigos e reportagens utilizados na base de conhecimento do agente, com suas respectivas URLs . . . . .	65
13	Formato de perguntas feitas ao agente . . . . .	68

# LISTA DE ABREVIATURAS E TERMOS

## SIGLA – NOME COMPLETO E OU SIGNIFICADO

*LLM*: Large Language Models, que podem ser traduzidos como Grandes Modelos de Linguagem. Inteligências artificiais, treinadas em grandes bases de dados, capazes de processar e produzir texto humano.

*NLP*: Natural Language Processing, ou processamento de linguagem natural, ramo da computação voltado estudo de processamento e manipulação de linguagem humana.

*RAG*: Retrieval-Augmented Generation, que pode ser traduzido como geração aumentada de recuperação. Técnica para elevar o desempenho de LLM, fornecendo-lhes informações externas, possivelmente não presentes em sua base de conhecimento

*Inferência*: Chamada a uma LLM com um *prompt* (texto de entrada), para que ela gere um retorno.

*Engenharia de Prompt*: Processo de criação de *prompts*, de forma que se busque moldar o comportamento da LLM de forma adequada para o contexto de uso da aplicação.

*Agente*: Entidade de software capaz de executar tarefas específicas de forma autônoma ou com pouca intervenção humana, utilizando inteligência artificial.

*API*: Sigla para Application Programming Interface, ou Interface de Programação de Aplicação. Um conjunto de funcionalidades que um software disponibiliza a terceiros.

*Pipeline*: No contexto de uma aplicação com LLM, um pipeline seria um conjunto de funções utilizadas na composição do agente, que influenciam a produção de uma saída a partir de uma entrada. Essas funções podem incluir chamadas à própria LLM (inferências), chamadas a alguma base de dados, tratamento de dados, entre outras funções.

*JSON*: JavaScript Object Notation, um formato de dado feito para ser simples e compacto, utilizado por diversos sistemas. Possui formato atributo-valor.

*Grafo*: Estrutura matemática composta de vértices (ou nós) e arestas que conectam pares desses vértices. Grafos podem ser utilizados para modelar relações e conexões em diversos contextos.

*Knowledge-graph*: Em tradução “grafo de conhecimento”. No contexto de LLM, corresponde a implementação de um grafo que permite a LLM acessar múltiplas bases de dados, buscar informações e fazer múltiplas inferências para a geração de uma saída.

*Embedding textual*: Processo de representação de palavras, frases ou textos como vetores numéricos em um espaço multidimensional, de forma que relações semânticas entre os elementos sejam preservadas. Esses vetores podem ser usados, por exemplo, para alimentar mecanismos de busca semântica ou sistemas de similaridade textual.

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>19</b>
1.1	Definição do Problema . . . . .	20
1.2	Premissas e Hipóteses . . . . .	21
1.2.1	Objetivo geral . . . . .	21
1.2.2	Objetivos específicos . . . . .	21
1.3	Estrutura da monografia . . . . .	22
<b>2</b>	<b>CONCEITOS GERAIS E REVISÃO DA LITERATURA</b>	<b>23</b>
2.1	LLM e o desafio de informação em João Pessoa . . . . .	23
2.1.1	Grandes Modelos de Linguagem . . . . .	23
2.1.2	Aplicações com uso de LLMs . . . . .	23
2.2	Técnicas de uso e limitações . . . . .	24
2.2.1	Técnicas de uso de LLMs . . . . .	24
2.2.2	Limitações de recursos . . . . .	24
2.3	Considerações Éticas e Legais . . . . .	25
2.3.1	Considerações Éticas e Legais . . . . .	25
2.3.2	Proteção de Dados e Direitos Autorais . . . . .	25
2.4	Trabalhos Relacionados . . . . .	25
2.4.1	Dificuldades associadas a LLM . . . . .	25
2.4.2	Uso de RAG em aplicações de LLM . . . . .	26
2.4.3	Uso prático de LLM . . . . .	27
<b>3</b>	<b>METODOLOGIA</b>	<b>28</b>
3.1	Coleta de Dados . . . . .	28
3.2	Metodologia da Coleta de Dados . . . . .	29
3.2.1	Demografia e geografia . . . . .	30
3.2.2	Transporte público . . . . .	30
3.2.3	Localidades segmentadas por tipo e bairro . . . . .	30
3.2.4	Informações de notícias na cidade . . . . .	31

3.2.5	Informações de clima em tempo real . . . . .	32
3.2.6	Informações gerais sobre a cidade . . . . .	32
3.2.7	Informações de imóveis . . . . .	32
3.3	Agrupamento e tratamento de dados . . . . .	33
3.3.1	Informações de demografia e geografia de cada bairro . . . . .	33
3.3.2	Informações de transporte público na cidade . . . . .	33
3.3.3	Informações de localidades na cidade segmentadas por tipo e bairro . . . . .	33
3.3.4	Informações de notícias na cidade . . . . .	33
3.3.5	Informações gerais sobre a cidade . . . . .	34
3.3.6	Informações de imóveis . . . . .	34
3.3.7	Geração de dados adaptados . . . . .	35
3.4	Algoritmo escolhido . . . . .	35
3.5	Concepção do Algoritmo . . . . .	37
3.6	Avaliação de Resultado . . . . .	40
<b>4</b>	<b>APRESENTAÇÃO E ANÁLISE DOS RESULTADOS</b>	<b>44</b>
4.1	Resultados e análise . . . . .	44
4.1.1	Resultados gerais . . . . .	44
4.1.2	Perguntas relacionadas a bairros . . . . .	45
4.1.3	Perguntas relacionadas a imóveis . . . . .	46
4.1.4	Perguntas relacionadas a informações gerais da cidade . . . . .	47
4.1.5	Perguntas relacionadas a locais . . . . .	48
4.1.6	Perguntas relacionadas a notícias . . . . .	49
4.1.7	Perguntas relacionadas ao transporte público na cidade . . . . .	50
4.2	Limitações observadas . . . . .	50
4.3	Possíveis pontos de melhora . . . . .	51
4.3.1	Erro ao executar função . . . . .	51
4.3.2	Tempo de resposta . . . . .	52
4.3.3	Respostas com conteúdo insuficiente . . . . .	52
4.3.4	Dificuldade em obter certas informações . . . . .	52

4.4 Exemplos de interações . . . . .	53
<b>5 CONCLUSÃO E TRABALHOS FUTUROS</b>	<b>56</b>
5.1 Trabalhos Futuros . . . . .	57
<b>REFERÊNCIAS</b>	<b>57</b>
<b>ANEXO A - ANEXOS E APÊNDICES 1</b>	<b>62</b>

# 1 INTRODUÇÃO

A cidade de João Pessoa vem ganhando grande destaque nacional nos últimos anos devido ao seu potencial turístico e de qualidade de vida, tendo sido eleita como um dos destinos em alta para 2025 pela plataforma estadunidense Booking (2024). Dentro desse cenário, um fenômeno que se demonstra em crescimento na localidade é o Turismo Imobiliário, consiste na vinda de indivíduos de outras regiões ou países, que, após uma experiência turística positiva, demonstram interesse em tomar residência naquela cidade (FERREIRA e SILVA, 2008).

Tal fenômeno já vem sendo estudado em outras cidades de grande porte, como a capital do Rio Grande do Norte, Natal, em especial na década de 80. Esta localização, dado seu momento histórico anterior, continha características geográficas, culturais e climáticas similares às da capital Paraibana atualmente. Com isso, pode-se ficar claro o motivo da busca de informações acerca de João Pessoa, tanto para potenciais turistas quanto para potenciais residentes, ter aumentado (BBC News Brasil, 2025), se tornando razoável assumir que o turismo imobiliário é um fenômeno presente na realidade pessoense.

O aumento relacionado à popularidade na cidade também justifica a necessidade de meios de divulgar informações acerca da mesma. Um exemplar dessa demanda é o agente conversacional “Fala Turista”, lançado em 2024 pela Secretaria de Turismo de João Pessoa (Cardoso, 2024), uma tecnologia, que funciona através do uso de Grandes Modelos de Linguagem (Large Language Models - LLMs).

As LLMs são uma tecnologia cuja popularidade teve um aumento considerável nos últimos anos, de forma que diversos exemplares se consolidaram no mercado. Elas podem ser definidas como sistemas de inteligência artificial capazes de processar e gerar texto de qualidade humana (NAVEED *et al.*, 2024).

Como exemplo de sua variedade, podem ser destacados marcas como: o ChatGPT, da Open AI; o Claude, da Anthropic; o Gemini, da Google; entre outros (Scherkabov *et al.*, 2024). Suas existências, aliado a acessibilidade e tecnologias adjacentes, abrem margem para o desenvolvimento de diversas aplicações das tais numa grande variedade de nichos. Aplicações com fins criativos, voltadas para a área de saúde, ou simplesmente agentes conversacionais personalizados (KADDOUR *et al.*, 2024) são algumas ocorrências notáveis e presentes no mercado.

Apesar de utilizarem o mesmo princípio e tecnologia, o que difere seus produtos dentro de diferentes nichos é a forma como LLMs são utilizadas, sendo o desenvolvimento deles um processo amplamente interconectado à habilidade dos desenvolvedores de “personalizar” o comportamento do modelo para o contexto de uso desejado. Nesse sentido, algumas delas compartilham uma mesma qualidade: essas aplicações podem fazer uso de

IA para produzir um *chatbot* com capacidade de conversação similar à humana, além da possibilidade da automação de diversas funcionalidades.

Embora seja possível observar aplicações com LLM voltadas a turistas, percebe-se que o mesmo não é verdade para outras que possam também ser úteis a possíveis residentes de uma cidade, sendo esse um nicho mais limitado. Entretanto, esta área poderia ser uma das mais beneficiadas, já que a tecnologia utilizada permite que usuários possam obter informações e tirar suas possíveis dúvidas de forma mais autônoma e intuitiva, o que gera economia de tempo e eleva o padrão de serviço, tornando a experiência mais agradável (BENADDI *et al.*, 2024). Com isso, é proposto a criação de um agente, que possui como base de conhecimento diversas bases de dados contendo informações acerca de João Pessoa, que possa ser útil tanto para potenciais turistas quanto para eventuais residentes.

## 1.1 Definição do Problema

Recentemente, João Pessoa vem observando um exponencial crescimento de sua popularidade, resultando em diversos novos residentes; foi-se notado um aumento de 15,3% em sua população desde 2010 (BBC News Brasil, 2025). Uma outra consequência é a valorização imobiliária: segundo uma reportagem do G1 (2025), o índice FipeZap (2024), mostra que João Pessoa teve a segunda maior valorização imobiliária entre as capitais do país. Portanto, a criação de uma aplicação com LLM acerca de João Pessoa poderia ter finalidade de disponibilizar um meio que permitisse potenciais turistas e moradores de obter informações acerca da cidade, similar ao visto no assistente virtual “Fala Turista”.

Embora já exista um agente voltado para turismo, percebe-se que existe um potencial nicho específico que poderia ser atendido: o de possíveis novos moradores. O seguinte trabalho estuda a aplicação de um protótipo utilizando LLMs visando atingir este público, explorando a intersecção entre fatores que determinam a capacidade de uma cidade de atrair turistas e também novos moradores para abranger também o nicho turístico.

Pode-se definir um “agente” como uma entidade de *software* capaz de executar tarefas específicas de forma autônoma ou com pouca intervenção humana, utilizando inteligência artificial. No contexto de um agente com LLMs, ele entidade seria capaz de processar e interpretar linguagem natural, fornecendo um retorno do mesmo modo. Logo, torna-se intrínseco que tal agente seja personalizado para o contexto de seu uso.

Neste cenário, encontramos uma das maiores finalidades deste estudo: criar uma ferramenta com uso de LLM de qualidade, aplicando a tecnologia e adaptando seu funcionamento ao contexto desejado. Para tal, torna-se necessário fazer uma análise das dificuldades associadas quanto a criação de ferramentas do tipo. Esses impasses variam amplamente, impactando diversas áreas, desde o desenvolvimento até a usabilidade de ferramentas do tipo (KADDOUR *et al.*, 2023).

Dentre tais adversidades, podem ser citadas a incapacidade de agentes de lidar com grandes quantidades de informação, os erros na produção de saídas textuais, e a dificuldade inerente do desenvolvedor para a criação e formação de entradas textuais que gerem os resultados esperados no contexto da aplicação (KADDOUR *et al.*, 2023). Porém, algumas citadas podem ser remediadas ou evitadas através do uso de técnicas específicas, que minimizem as limitações de LLMs. Nesse sentido, o estudo se propõe a explorar e utilizar técnicas para melhorar o desempenho do agente, tornando assim uma demonstração prática de como tais permitem um resultado melhor para um agente.

Tais técnicas devem ser utilizadas juntos a uma aplicação que contenha uma extensa base de dados acerca da cidade de João Pessoa, que sejam devidamente tratados e permitam ao agente desempenhar seu papel de forma eficiente. Portanto, o estudo também se propõe a filtrar e selecionar tais dados, e aplicá-los dentro do contexto apresentado.

## 1.2 Premissas e Hipóteses

Observando o cenário atual em relação a cidade de João Pessoa, junto de seu crescente volume de turistas e de novos moradores, é lógico abordar a seguinte problemática: é possível criar um agente capaz de conversar com esses indivíduos e fornecer informações precisas e relevantes sobre a cidade, mesmo diante das dificuldades existentes em criar exemplares do tipo? Partindo dessa premissa, o presente trabalho tem como objetivo explorar e aplicar a criação de um protótipo de um agente que faça uso de dados acerca da cidade de João Pessoa, explorando e fazendo uso de técnicas de voltadas para LLM para melhorar o seu desempenho, de forma que o mesmo seja funcional e capaz.

### 1.2.1 Objetivo geral

Propor a criação de um protótipo de um agente que faz uso de LLM. Possuindo uma base de dados de informações relacionadas à cidade de João Pessoa, este será capaz de dialogar e responder dúvidas de potenciais turistas e moradores da mesma. Ele será um meio acessível às informações relacionadas ao município em relação a aspectos geográficos, turísticos e imobiliários, dentre outros. Ademais, a metodologia utilizada no protótipo será estruturada de forma a permitir a sua aplicação em diferentes cidades e contextos.

### 1.2.2 Objetivos específicos

Os seguintes itens detalham objetivos específicos a serem alcançados ao longo do desenvolvimento do trabalho.

- Listar tipos de dados que sejam relevantes para potenciais turistas e moradores de uma cidade;

- Realizar coleta de dados que possam ser considerados relevantes para a cidade de João Pessoa, buscando fontes que possam ser consideradas confiáveis;
- Buscar quais são as limitações existentes na criação de aplicações com LLM, e quais as possíveis estratégias que possam existir para lidar com elas;
- Aplicar tais estratégias, observando se aplicação das mesmas facilita o processo de desenvolvimento, e se também gera resultados;
- Avaliar a capacidade do agente quanto a conversação e também a responder perguntas, a partir das informações que foram fornecidas para o mesmo;
- Respeitar as limitações econômicas que forem apresentadas ao longo do desenvolvimento do trabalho;
- Sugerir ideias adequadas para pesquisas futuras que abordem a mesma temática, assim como possíveis modificações no protótipo criado, de acordo com os fatores observados ao longo do projeto;
- Apontar o potencial de adaptação da metodologia para aplicação em outras cidades, ampliando o escopo de uso da solução proposta.

### 1.3 Estrutura da monografia

A monografia está separada nas seguintes seções:

- "Introdução", onde é apresentada uma contextualização da proposta do projeto e a sua metodologia idealizada.
- "Conceitos Gerais e Revisão da Literatura", onde são abordados termos chaves e trabalhos relacionados essenciais para a contextualização, concepção e estruturação do problema.
- "Metodologia", onde é detalhado cada decisão tomada acerca de tratamento de dados, criação do protótipo do agente e procedimentos afins para atingir os objetivos.
- "Apresentação e Análise dos Resultados", seção onde, após aplicação da metodologia, observam-se os resultados obtidos nos experimentos e sua análise qualitativa.
- "Conclusão e Trabalhos Futuros", finalizando a discussão e dando premissas e conselhos para futuros trabalhos no nicho em que este se encaixa.

Ademais, "Referências" e "Anexos" contém fontes de informação e detalhes mais intrínsecos à metodologia e resultados.

## 2 CONCEITOS GERAIS E REVISÃO DA LITERATURA

### 2.1 LLM e o desafio de informação em João Pessoa

Neste trabalho, é proposto a criação de um protótipo de um agente *chatbot* voltado a fornecer informações sobre a cidade de João Pessoa de forma aprofundada, útil, confiável e com boa comunicação, tanto para turistas quanto para possíveis moradores. Tal fará, na prática, uso de técnicas que permitam que ele extraia informações de bases de dados; ele deve conter conhecimento relevante quanto à convivência urbana: infraestrutura (transporte público, estabelecimentos de conveniência, etc.), custo de vida, dados demográficos, entre outras informações. A partir disso, deve ser construído com objetivo de se tornar uma plataforma referencial acerca da cidade.

Dada a complexidade dos desafios da coleta de dados e da criação do agente, torna-se necessário a redução do escopo a que o trabalho se refere, particularmente quanto a limitações econômicas e de obtenção de informações.

Os conceitos, artigos e trabalhos apresentados a seguir foram levados em consideração por terem relevância quanto ao contexto deste trabalho.

#### 2.1.1 Grandes Modelos de Linguagem

As LLMs são inteligências artificiais treinadas para receber e produzir texto humano. Tais modelos são desenvolvidos através de técnicas de *Natural Language Processing* (ou apenas NLP) otimização e também o uso de bases de dados textuais.

Estes exemplares necessitam de *hardware* de grande poder computacional para funcionar, visto que comumente operam com milhares de parâmetros. *Hardware* de qualidade, somado a evolução das técnicas de *software* e arquiteturas mais robustas, permitiram que tenha-se observado um grande avanços na produção de modelos na área, tais como os GPTs, da OpenAI, e o BERT, do Google (RAIAAN *et al.*, 2024). Aqueles que são posteriores aos citados apresentam notável capacidade de geração de texto natural, tendo potencial para uso em diversas áreas.

#### 2.1.2 Aplicações com uso de LLMs

O uso de LLMs ocorre em aplicações onde existe uma potencial utilidade na produção de texto natural. Existem diversas áreas onde observa-se o uso de LLMs, tais como com trabalhos criativos, área de saúde e turismo. Um dos exemplos mais comuns é o uso de *chatbots*, agentes voltados ao diálogo, que combinam recuperação de informação e interações em turnos para geração de texto (KADDOUR *et al.*, 2023).

## 2.2 Técnicas de uso e limitações

### 2.2.1 Técnicas de uso de LLMs

Embora LLMs demonstrem considerável qualidade de escrita, elas ainda apresentam limitações em sua produção textual. Nesse sentido, observa-se o uso de técnicas em aplicações que façam uso desses modelos que visem permitir que os mesmos produzam saídas em um formato desejado, com o conteúdo que faça sentido para seu contexto de uso.

Uma das técnicas a se comentar é o uso de RAG , proposta por Lewis *et al.* (2021). Tal método visa permitir que o modelo possa produzir conteúdo textual acerca de um domínio específico, provendo a ele as informações necessárias. Outras táticas a se destacarem são as de *Fine-Tuning*, adaptação de um modelo pré-treinado a um certo domínio ou contexto, e o uso de *embeddings*, que permite salvar e obter termos semânticos a partir de uma base de dados vetorial.

### 2.2.2 Limitações de recursos

Ao produzir uma aplicação que utiliza LLMs, existem duas principais opções de desenvolvimento: a implementação de um modelo de código aberto em *hardware* dedicado; ou a adoção de um Modelo Proprietário. A primeira opção exige um investimento significativo em componentes físicos especializado, dada a demanda computacional intensiva dos tais. Já a segunda opção baseia-se no uso da API de um Modelo Proprietário, na qual os custos se baseiam por taxa de uso.

Para este trabalho, foi escolhida a opção de menor custo. Esta preferência não apenas simplifica o processo de desenvolvimento, mas também se justifica pela sua qualidade em comparação com alternativas de código aberto, particularmente demonstrada em Modelos Proprietários consolidados no mercado. Esta abordagem assegura uma implementação eficaz, alavancando a capacidade dos Modelos Proprietários sem os encargos logísticos e financeiros associados à gestão de infraestrutura de *hardware* própria.

Outros fatores a se considerar são quanto a obtenção de dados relacionados a localidades, e a aquisição de notícias em tempo real. Em ambos os casos, se faz necessário o uso de serviços de terceiros, gerando implicações econômicas significativas. Com esta perspectiva, é importante que seja considerada uma abordagem que minimize o uso de recursos financeiros.

## 2.3 Considerações Éticas e Legais

### 2.3.1 Considerações Éticas e Legais

No desenvolvimento de aplicações que utilizem bases de dados, é importante respeitar a Lei Geral de Proteção de Dados (também chamada de LGPD, sancionada em 2018), a qual determina que informações pessoais só podem ser utilizadas com consentimento explícito. Nesse sentido, ao longo deste trabalho, durante a construção do agente, serão adotadas práticas que ocorram conforme a LGPD, tais como uso exclusivo de conhecimento público, evitando o uso de conteúdo pessoais de terceiros.

No contexto de grandes modelos de linguagem proprietários, considerando que os mesmos necessitam de extensas bases de dados para serem treinados, é essencial evitar o fornecimento de informações de terceiros aos mesmos - visto que ferramentas como o ChatGPT, da OpenAI, utilizam o conteúdo das interações para aprimoramento dos modelos (OpenAI, 2024). Discussões acerca de tópicos como consentimento, direito ao esquecimento de dados, segurança da informação, entre outros, podem ser relevantes tratando-se de LLMs, mas fogem do escopo deste trabalho.

Ademais, é importante destacar que o trabalho desenvolvido apresenta propósito exclusivamente acadêmico, de forma que não haja interesse comercial nos dados de outras fontes utilizadas.

### 2.3.2 Proteção de Dados e Direitos Autorais

Todos e quaisquer dados coletados para aplicação são escolhidos com o critério de terem natureza pública, como *sites* e *blogs*. Tais não devem apresentar qualquer informação que possa ser considerada pessoal. Ademais, eles têm finalidade prática puramente acadêmica, visto que, para o protótipo desenvolvido, não há um interesse genuíno em lançá-lo de forma comercial. A intenção será apenas demonstrar a viabilidade de uma utilidade do tipo.

Por fim, considerações éticas devem se tornar relevantes, de modo que os dados obtidos não tenham caráter de ofender ou denegrir seus autores ou a quem eles dizem respeito. As fontes utilizadas para todas as informações utilizadas serão devidamente referenciadas ao longo do trabalho.

## 2.4 Trabalhos Relacionados

### 2.4.1 Dificuldades associadas a LLM

Existem diversas áreas de uso de aplicações práticas com LLM. No artigo “Challenges and Applications of Large Language Models”, Kaddour *et al.* (2023) apresenta

algumas delas, a partir de exemplos. Os autores exploram a utilização deste tipo de tecnologia em diversos nichos, incluindo o do presente estudo (como, por exemplo, *chatbots*). A discussão utilitária e qualitativa desbrava os desafios e peculiaridades da ferramenta em diferentes ambientes de teste, sendo útil para o desenvolvimento de novas técnicas.

O artigo também demonstra muito das limitações existentes quanto aos modelos atuais e o seu uso, de forma que torna-se relevante obter noção delas perante a criação de uma aplicação que faça uso dos modelos. Algumas destes impedimentos estão relacionados à produção de texto com informações imprecisas, e à dificuldade dos modelos em lidar com contextos maiores. Entender tais empecilhos é um passo essencial para saná-las e construir um projeto de sucesso, algo que é discutido no trabalho a ser produzido.

Outras barreiras comuns também são citadas no trabalho, como seu alto custo necessário para atingir melhores qualidades de saída. Porém, dada a pluralidade de seu estudo, Kaddour *et al.* (2023) também torna possível perceber melhorias possíveis dado contextos específicos através de suas diferentes aplicações para nichos direcionados.

#### 2.4.2 Uso de RAG em aplicações de LLM

O artigo "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks" de Lewis *et al.* (2021) discute a metodologia Retrieval-Augmented Generation (RAG), que busca melhorar o desempenho quanto ao conhecimento no uso de Processamento de Linguagem Natural (NLP) combinando memórias paramétricas pré-treinadas e não-paramétricas (ou seja, baseadas em recuperação) em modelos de geração de linguagem. O objetivo é permitir que modelos pré-treinados possam atualizar seu conhecimento usando dados externos.

No artigo, faz-se uso de memória não-paramétrica através de *embeddings* - representações vetoriais de palavras ou textos. Os resultados mostram que a abordagem apresentada consegue superar modelos paramétricos em contextos de domínio aberto. Tal metodologia permite combinar modelos proprietários pré-treinados com novas bases de conhecimento, aumentando seu desempenho.

Outro trabalho a se destacar é o trabalho "A Survey on RAG with LLMs" de Arslan *et al.* (2024), que apresenta um estudo acerca de aplicações que façam uso de RAG e LLMs, investigando como essas tecnologias têm sido aplicadas em tarefas específicas e áreas variadas, com o intuito de identificar direções futuras de pesquisa e melhor entender o estado atual do campo.

O estudo revelou uma ampla gama de aplicações de RAG em diversos campos, dentre eles medicina, finanças, educação e desenvolvimento de *software*. Notavelmente, a integração de RAG com LLMs tem mostrado melhoria significativamente à precisão e a relevância das respostas geradas por modelos de linguagem, ao incorporar dinamicamente

informações de fontes de dados externas.

No trabalho "Retrieval Augmented Generation (RAG) and Beyond: A Comprehensive Survey on How to Make your LLMs use External Data More Wisely", Zhao *et al.* (2024) apresenta outra análise de técnicas de RAG no contexto de aplicações com LLM, mostrando o potencial de tais técnicas. O diferencial do trabalho está em avaliar o uso de técnicas de categorização das consultas que podem ser feitas a um agente de uma aplicação, de forma que, para cada tipo de consulta, é apresentada uma abordagem eficiente, que permita ao modelo obter as informações corretas para elaborar sua resposta.

Ao categorizar as consultas e apresentar as técnicas possíveis, o trabalho demonstra de forma concreta abordagens que possam ter um impacto positivo no desenvolvimento de aplicações com LLMs, particularmente quanto ao tratamento dos dados disponíveis. Dentre as diversas técnicas apresentadas, estão o "ajuste-fino" (adaptação de um modelo pré-treinado via um novo treinamento para um contexto específico), uso de *embeddings*, construções de algoritmos complexos para recuperação de informações, entre outros.

### 2.4.3 Uso prático de LLM

LLMs se demonstram ser ferramentas extremamente úteis ao produzir conteúdo textual acerca de cidades, segundo o estudo "What can LLM tell us about cities?" de Li *et al.* (2024). Modelos proprietários, treinados com bases de dados extensas, demonstram capacidade de reter conhecimentos específicos acerca de diversas cidades e regiões. Ademais, essas entidades são capazes de serem utilizadas como ferramentas preditivas para análise, utilizando informações conhecidas para predição de variáveis-alvo.

Desta forma, mostra-se o potencial de tais ferramentas para o contexto de cidades. Contudo, o trabalho também evidencia limitações dos modelos, como a tendência a gerar conteúdo genérico ou aleatório diante de falta de informação concreta. Tal fato evidencia a necessidade de que um possível agente tenha acesso a informações que estejam além do seu conhecimento próprio.

O trabalho "Real Estate with AI: An agent based on LangChain" apresenta o desenvolvimento de um agente, criado com uso do *framework* LangChain, voltado ao setor imobiliário. Tomando como referência o mercado imobiliário da Dinamarca, o agente criado tem como proposta auxiliar indivíduos na tomada de decisões, particularmente diante da dificuldade inerente associada a tal área (HAURUM, MA e LONG, 2024). O agente criado mostrou-se capaz de superar outras LLMs nos testes de desempenho apresentados.

O trabalho demonstra o potencial do uso de LLMs no contexto de ramo imobiliário, para auxílio na tomada de decisão e ação do consumidor. Características relacionadas ao imóvel em si, como localização, além também de detalhes burocráticos associados à compra de imóveis, são informações fornecidas ao agente para auxílio de usuários.

### 3 METODOLOGIA

O procedimento metodológico deve propor um direcionamento para as partes vitais do trabalho: a obtenção dos dados, construção do agente e o teste de seu desempenho.

No que se refere a obtenção dos dados, foi estabelecido um plano para sua coleta e tratamento, de forma que os mesmos fossem confiáveis, condizentes com a realidade da cidade de João Pessoa, e possíveis de serem manipulados pelo agente a se construir. Já no que se refere à construção do tal agente, ele deveria levar em consideração uma investigação de quais seriam as maiores problemáticas existentes durante o processo; em seguida, foram pesquisadas técnicas de como lidar com tais. A partir disso, sua estruturação foi feita considerando as informações obtidas e a forma mais eficiente de permitir que o agente lide com levantamentos.

Para a manutenção e posse dos dados, propôs-se o uso de arquivos JSON, que possuem um formato simples e ideal para acessar informação a partir de chaves ou índices referentes ao valor. Como a aplicação desenvolvida tem como foco uso de LLM, assumiu-se que os conhecimentos utilizados são textuais; logo, acessá-los via índices se tornou mais simples. Foi também utilizado um banco de palavras vetorial, que permitiu salvar *tokens* (pedaços de texto) e acessar os mesmos via termos de busca.

Para a construção do agente, portanto, foi proposto a implementação de um grafo de conhecimento, consistindo num conjunto de funcionalidades que podem ser utilizadas pela LLM, como chamadas às bases de dados e a ela própria, respeitando um fluxo semi-sequencial para geração de resposta; tal fará uso do *framework* DSPy (Stanford NLP Group, [s.d.]). Para a sua avaliação, foi empregado um *framework* para análise qualitativa de diálogos com *chatbots*. Um *framework* pode ser definido como uma abstração de código e funcionalidade(s).

O teste do desempenho do agente considerou técnicas específicas para avaliação de *chatbots*, abordando também o contexto do trabalho e as limitações existentes. Com isso, avaliou-se, sobretudo, sua capacidade demonstrar conhecimento acerca da base de dados que lhe foi fornecida apropriadamente e sua capacidade conversacional.

Logo, para este trabalho, conclui-se que a metodologia utilizada é de natureza aplicada e exploratória. O trabalho também baseia-se no uso de pesquisa experimental (através do uso de prototipagem) e análise qualitativa quanto a capacidade do agente.

#### 3.1 Coleta de Dados

Para tornar esse trabalho viável, foi necessário coletar dados referentes a fatores relevantes à vivência na cidade de João Pessoa. Inicialmente, para essa pesquisa, foram estabelecidos os seguintes tipos de dados como “relevantes”, e que, portanto serão

utilizados na criação da aplicação:

- Informações acerca do transporte público da cidade.
- Informações relacionadas a demografia de cada bairro. A demografia deve incluir noções como atividades econômicas comuns no bairro, renda média da população e segurança.
- Informações geográficas de cada bairro, o que inclui sua localização na cidade e também pontos geográficos relevantes.
- Informações relacionadas a localidades de variados tipos na cidade, segmentados por bairro. Os escolhidos foram: bares, restaurantes, lanchonetes, escolas, faculdades, academias, farmácias, hospitais, igrejas, mercados, praças, parques e praias. Esses locais podem ser relevantes tanto para potenciais turistas quanto potenciais moradores, pois são locais que se relacionam ao lazer, necessidades básicas e de conveniência.
- Informações de notícias da cidade atualizadas em tempo real.
- Informações de imóveis na cidade.
- Informações de atrações turísticas locais turísticas considerados relevantes no geral pela população.
- Informações gerais sobre a cidade, como clima, história, eventos comuns ao longo do ano.

Desta forma, a aplicação terá posse de uma extensa base de dados, que possuam caráter descritivo e informativo da cidade e de seus bairros.

### **3.2 Metodologia da Coleta de Dados**

Considerando os tipos de amostras necessárias e os fatores limitantes estabelecidos, cada um de seus tipos foram coletados com uma abordagem única e bem planejada. Por se tratar de uma aplicação com LLM, assume-se que as informações coletadas possuem caráter textual. Os dados cuja obtenção era essencial foram categorizados em 6: Dados de bairros (demografia e geografia dos mesmos), dados de localidades na cidade (relacionados também a subtipos), dados de transporte público na cidade, notícias relacionadas a cidade, informações gerais da cidade e, finalmente, informações de imóveis na cidade.

### 3.2.1 Demografia e geografia

Para coletar dados demográficos, foram utilizadas duas fontes principais; a primeira foi o Atlas Municipal, disponível no *site* do governo Filipeia. Este endereço virtual contém informações de cada bairro, tal como uma descrição breve de sua população e geografia, além de indicar os limites de tais índices. Isso inclui, por exemplo, as fronteiras de cada local e sua posição na cidade. A lista completa de bairros pode ser encontrada no Anexo A, na Lista de bairros de João Pessoa.

Outra fonte relevante para o projeto foi o IBGE, que disponibiliza informações segmentadas em setores censitários. Tais foram agrupados com base nos bairros obtidos no Atlas Municipal. A partir disso, foi possível estabelecer uma renda média para a população de cada um. Como referência, foi tomado o censo de 2010, assumindo-se que o perfil socioeconômico da população não demonstrou mudanças significativas até o presente ano atual (2025).

### 3.2.2 Transporte público

Examinando que João Pessoa possui como transporte público apenas uma frota de ônibus, foram utilizados como base as informações disponibilizadas pela Superintendência Executiva de Mobilidade Urbana, a SEMOB, em seu *site*. Para cada linha, levantou-se o seu nome e seu itinerário, destacando-se o número de paradas em seu trajeto de ida e volta, e também o bairro em que se localiza cada uma.

### 3.2.3 Localidades segmentadas por tipo e bairro

Os tipos escolhidos de localidades a se obter dados sobre foram: bares, restaurantes, lanchonetes, escolas, faculdades, academias, farmácias, hospitais, igrejas, mercados, praças, parques e praias, além de locais turísticos considerados relevantes.

Inicialmente, foi estabelecido que seria utilizado o Google Maps como fonte principal para a busca dessas informações. O *software* disponibiliza um serviço chamado Places API, que pode ser utilizado para fazer buscas de locais com base em um identificador próprio (GOOGLE). Dada sua aplicabilidade, foram escolhidos como dados a serem usados: nome, descrição, nota, faixa de preço, localização, mídias sociais da localidade, “análises” disponibilizadas por usuários do Google e telefone para contato.

Para parques e praças contidos na cidade, o *site* Filipeia disponibiliza um arquivo contendo informações de segmentadas por bairro, com o nome de cada instância. Portanto, seria necessário utilizar o Google Maps apenas para obter uma descrição de cada considerado.

Para praias e pontos turísticos relevantes, foi utilizado o *site* do governo Turismo João Pessoa, que contém diversas informações sobre a cidade, fornecendo para cada localidade uma descrição e seu endereço.

Para informações de hospitais e faculdades, foram selecionados aqueles considerados mais relevantes no contexto da cidade. A partir disso, obteve-se para essas localidades seu nome completo, endereço e descrição. Embora relevantes, esses estabelecimentos não são numerosos se comparados aos outros tipos. A seleção dos mesmos ocorreu de forma manual.

Para as demais localidades - bares, restaurantes, lanchonetes, escolas, farmácias, academias, hospitais, igrejas e mercados - foi utilizado uma abordagem automatizada. A partir de cada tipo, no contexto dos bairros selecionados, foi criado um termo de busca: "[tipo de localidade] no bairro [nome do bairro]".

Os termos formatados foram enviados para a API Places, que retornaria uma lista de localidades do tipo; para cada uma, filtram-se os 10 elementos com melhor nota. Em seguida, ocorre uma nova busca, desta vez com base no identificador único de cada uma das instâncias, para que fossem obtidos os seus respectivos dados. As últimas mencionadas possuem uma faixa mais limitada, de modo que nem todos os bairros possuem pontos geográficos daquele tipo.

Para considerar um contexto cronológico, é preciso dar importância que a busca dessas informações ocorreu no dia 25 de janeiro de 2025. Os dados obtidos foram salvos em arquivos JSON.

Dois fatores devem ser considerados nessa abordagem:

- Em uma aplicação real, seria plenamente possível fazer essas buscas simultaneamente na API Places. Contudo, por se tratar de um protótipo, a intenção é apenas mostrar a viabilidade do processo. Ademais, deve-se considerar que, para cada ação, há um custo envolvido. Dadas as limitações financeiras presentes ao autor, salvar os dados para que sejam acessados localmente torna-se mais vantajoso.
- A natureza das buscas na API, embora não sejam não-determinísticas, apresentam um certo grau de imprevisibilidade nos resultados (algo que já é previsto na própria documentação), o que torna difícil a sua reprodução.

### 3.2.4 Informações de notícias na cidade

Dada a variedade de opções para obtenção de notícias, foi escolhida a API Serper, que permite a extração de dados de retorno de buscas no *site* Google (SERPER.DEV). Tal endereço eletrônico é referência global como plataforma de busca, e o Serper foi selecionado

com base na facilidade de uso e custos mais acessíveis. O uso consiste em, a partir de uma simples query, como “Notícias em João Pessoa”, obter-se diversas informações em formato JSON.

### 3.2.5 Informações de clima em tempo real

Escolheu-se a Open-Meteo API para obtenção de informações climáticas em tempo real, como temperatura, umidade e possibilidade de chuva. A API, que é gratuita e de código aberto, permite obter dados para um intervalo de até 16 dias (OPEN-METEO).

### 3.2.6 Informações gerais sobre a cidade

Informações relacionadas a história da cidade, passeios e atrações turísticas foram retiradas do *site* Turismo João Pessoa, e de outros artigos, *blogs* e reportagens encontradas no Google. A lista completa se encontra no Anexo A. A seleção dos textos a se utilizarem ocorreu de forma manual, com base na utilidade julgada para cada, e a verossimilhança de seu conteúdo com a realidade.

Foram pesquisados informações relacionadas aos bairros da cidade, custos de vida, entre outros fatores. Sites de imobiliárias e *blogs* de turismo, como o Jornal da Paraíba e o G1 foram utilizados como fontes, assim como reportagens acerca de custo de vida na cidade e problemas encontrados na mesma. A lista completa de fontes utilizadas para alimentar a base de conhecimento do agente poderá ser encontrada no Anexo A.

A Segurança Pública também foi um fator considerado para obtenção de dados. Contudo, após consulta com a Secretária de Estado da Segurança e da Defesa Social, foi constatado que informações dessa natureza são sensíveis e, portanto, devem ser mantidas fora do escopo do trabalho.

### 3.2.7 Informações de imóveis

Inicialmente, planejou-se que, para cada bairro, fosse feito um estudo acerca de informações como preço médio de terrenos e imóveis para compra e aluguel, de forma que fosse viável traçar um perfil descritivo do mercado imobiliário de cada bairro, permitindo que o agente pudesse filtrar ofertas com base em uma explanação disponibilizada pelo usuário.

Foi retirado a partir do *site* Agente Imóvel informações de preço de metro quadrado de imóvel para compra, segmentados por bairro, na cidade de João Pessoa (Agente Imóvel, 2025). A escolha da fonte de dados com valores monetários foi feita após uma busca extensiva, onde esta se demonstrou como a mais adequada para o presente estudo.

Contudo, dadas as limitações estabelecidas quanto a obtenção de dados de imóveis individuais, foi proposta uma abordagem diferente: a intenção do protótipo é não só demonstrar a viabilidade de prospecção de informação, mas também da capacidade do agente de interpretar os mesmos, podendo ser criado conhecimento “falso” (com um grau alto de verossimilhança) demonstrado ao tal como verídico. Espera-se que o agente mostre-se capaz de utilizar os dados da forma esperada.

### **3.3 Agrupamento e tratamento de dados**

Uma vez que os dados foram obtidos, todos passaram devidamente por um pré-processamento para que estivessem na codificação UTF-8, tendo essa sido escolhida pela sua praticidade e uso comum.

#### **3.3.1 Informações de demografia e geografia de cada bairro**

Foi criado um JSON contendo informações dos bairros. Cada um, identificado pelo seu nome, possui uma descrição, uma lista de similares com que faz fronteira, e sua posição na cidade

#### **3.3.2 Informações de transporte público na cidade**

Foi criado um JSON contendo cada linha de ônibus presente na cidade. Para cada, seria indicado os terminais de ida e de volta, o número de paradas, e quais bairros estavam incluídos na mesma. Este foi organizado de forma que o processo de acessar informações acerca de uma linha e de um bairro específico (ambos a partir de seus nomes) fosse simplificado.

#### **3.3.3 Informações de localidades na cidade segmentadas por tipo e bairro**

No JSON de informações de bairros, foi adicionado para cada um campo que corresponde às localidades existentes no tal, agrupados por tipo. Dessa forma, torna-se possível delimitar os pontos de interesse existentes em um específico, assim como acessar tais zonas.

#### **3.3.4 Informações de notícias na cidade**

Para cada dia em que a aplicação for executada, no início do dia, será feita uma busca através do Serper. O resultado será salvo como um JSON. Todos aqueles de notícias se encontram em uma pasta específica. Caso o agente necessite acessar informações de maneira simultânea, ele obterá as notícias referentes ao dia.

### 3.3.5 Informações gerais sobre a cidade

As informações textuais obtidas foram colocadas em um único arquivo de texto; após isso, ele passou pelo processo de tokenização, dividindo o texto em frases de até 1024 caracteres, e depois sofrendo *embedding*. A intenção era que fossem feitas solicitações de busca (as chamadas “queries”) para esse banco de dados vetorial, e tais retornassem frases semanticamente parecidas com a original.

### 3.3.6 Informações de imóveis

Estabeleceu-se a criação de um JSON contendo as informações de preço de imóveis por metro quadrado na cidade para compra, segmentados por bairro, além dos imóveis do respectivo bairro.

Como os imóveis a serem utilizados são irrealis, foi estabelecido um algoritmo para geração dos mesmos. Cada um teve como informações o seu endereço, descrição, bairro, tipo de moradia (casa ou apartamento) e de negócio (aluguel ou aquisição de propriedade). O objetivo, como comentado, foi produzir exemplares que alimentem a base de dados da aplicação, simulando a realidade, para testar a capacidade do agente de falar sobre os mesmos.

Para cada bairro, foram geradas 4 casas e 4 apartamentos. Dentre todo quarteto, 2 são negócios de aluguel, e 2 negócios de aquisição de propriedade. Os endereços e as descrições foram sorteados aleatoriamente de uma lista de endereços fictícios e descrições pré-definidas. As listas pode ser encontrada no Anexo A, sendo essas listas: Lista de ruas fictícias para geração de imóveis irrealis e lista de pontos de referência para geração de imóveis irrealis.

Para a geração dos preços, os bairros de João Pessoa foram agrupados em 3 grupos. Os 5 com maior renda média com base no censo de 2010 foram colocados no grupo 1, os 10 seguintes no 2, e o resto no 3.

Esses grupos foram criados numa tentativa de retratar uma das características da cidade: os bairros com população de maior renda média apresentam um determinado perfil de visitantes, geralmente sendo eles com melhor infraestrutura e mais próximos à praia, assim sendo mais atrativos e afetados pela especulação imobiliária.

O algoritmo define um preço base e um multiplicador. O multiplicador é influenciado tanto pelo tipo do imóvel quanto a moradia, o bairro em que se localiza, o tipo de compra e seu número de quartos. No fim da operação, o resultado deve ser o produto do deste multiplicador com o preço base. O pseudocódigo do algoritmo pode ser encontrado no Anexo A, em Algoritmo para geração de imóveis irrealis e seus preços.

### 3.3.7 Geração de dados adaptados

Uma estratégia comum a se utilizar com aplicações que fazem uso de LLM diante de diferentes bases de dados é a implementação de métodos para adaptar as informações já existentes em uma fonte de conhecimento para outro formato, ação feita por parte dos próprios modelos de linguagem (MIN *et al.*, 2024).

Uma das possíveis vantagens, que com esse fim ocorrerá a implementação dessa estratégia, é a possibilidade de permitir a existência de um maior escopo de conhecimento que possa ser obtido em buscas em bases de dados, como por exemplo, aquelas contendo *tokens* de textos. Com base nisso, a estratégia foi implementada no trabalho com a intenção de gerar texto informativo acerca da cidade de João Pessoa pela LLM, a partir de referências já existentes. Isso inclui descrições de bairros, com base nas características demográficas e geográficas obtidas através da coleta de registros, e também com elementos incluídos em artigos e reportagens.

Em ambos os casos, pede-se à LLM que reescreva, com suas próprias palavras, o material apresentado. Além disso, é solicitado também que ela use sentenças curtas e objetivas, as chamadas “frases com fatos explícitos”, de modo que a busca no banco vetorial possa encontrar enunciados semelhantes e/ou relevantes com maior facilidade. Ademais, construções com “frases com fatos explícitos”, como citado por Zhao *et al.* (2024), podem ser mais facilmente interpretadas por LLMs, o que facilita, por exemplo, a geração de respostas simples para contextos de pergunta-resposta.

## 3.4 Algoritmo escolhido

O algoritmo escolhido para criação do agente é o de um fluxo estruturado com LLM. Um *pipeline*, no contexto de uma aplicação com modelo de linguagem, consiste num conjunto de etapas a serem executadas na geração de uma resposta final, como chamadas a funções externas, bases de dados ou a realização de inferências por parte da própria LLM. Nesse cenário, a própria construção desse fluxo já se demonstra desafiadora. Como comentado no artigo de Kaddour *et al.* (2023), muitas das dificuldades inatas do desenvolvimento de um agente com modelo de linguagem, mesmo utilizando uma versão proprietária, acabam por impactar o andamento da implementação.

A primeira dificuldade que será abordada neste trabalho está relacionada à limitação do tamanho da janela do contexto. Ao fornecer informações para uma LLM, é preciso considerar que, quanto maior o comando de entrada, maior será a probabilidade de que o modelo acabe por cometer imprecisões na geração de sua saída. Portanto, para um fluxo baseado em modelo de linguagem, torna-se fundamental que cada etapa use, em suas inferências, o menor *input* possível; em outras palavras, é importante que a LLM receba apenas os dados essenciais para gerar uma resposta.

Outra limitação é a “fragilidade de *prompt*”: alterações sutis em um comando podem gerar mudanças consideráveis na saída. Portanto, cabe ao desenvolvedor refinar e testar diferentes instruções até que se obtenha a resposta em um formato desejável — a chamada “engenharia de *prompt*”. Esse procedimento, que pode ser longo e desgastante, representa um empecilho para o desenvolvimento de aplicações com LLM e, idealmente, deveria ser minimizado.

Uma última questão a ser considerada são as chamadas “alucinações”: produções de textos por parte da LLM contendo dados imprecisos — e que, em alguns casos, parecem tão verossímeis que podem passar despercebidos. Esse fenômeno, que pode ser atribuído a uma falha da arquitetura do modelo, normalmente pode ser remediado com o uso de RAG — ao fornecer o conhecimento necessário ao modelo para a realização de uma certa inferência, aumenta-se consideravelmente a probabilidade de que uma resposta adequada seja produzida.

Diversos *frameworks* e bibliotecas foram propostos para a criação de fluxos estruturados com modelos de linguagem; o escolhido para a realização deste trabalho será o DSPy. Ele é uma ferramenta proposta por Khattab *et al.* (2023), e tem como objetivo sugerir uma abordagem sistemática para desenvolver e otimizar processos baseados em LLM.

A ideia do DSPy é permitir que os fluxos sejam abstraídos como nós de transformação textual em um grafo, podendo ser chamados via módulos declarativos. A ferramenta também propõe a criação de um compilador capaz de maximizar a performance do processo a partir de uma determinada métrica. A inspiração do projeto é que ele possa ser visto como uma representação de rede neural, onde existem camadas modularizadas e os pesos do sistema podem ser treinados com otimizadores em vez de serem manualmente ajustados.

Dentre os componentes disponibilizados pela biblioteca, estão os módulos “Chain of Thought” (“Cadeia de pensamento”) e “ReAct” (junção das palavras “*Reason*” e “*Act*”, equivalente a algo como “refletir e atuar”). Esses dois elementos são empregados devido à sua capacidade de viabilizar o uso de estruturas de *prompt* que proporcionem um melhor desempenho dos modelos de linguagem no que diz respeito à geração de respostas.

O componente Chain of Thought permite que seja estabelecida uma “cadeia de raciocínio” para que o modelo de linguagem elabore sua resposta. Ele é particularmente útil quando se deseja obter uma solução para um problema mais complexo (WEI *et al.*, 2022). Já o recurso ReAct funciona por meio de múltiplas interações, nas quais o sistema, a partir de uma requisição inicial, formula uma sequência de pensamentos e ações, como, por exemplo, a execução de uma função externa (YAO *et al.*, 2022). Este é indicado para casos em que se deseja que, no fluxo do agente, ele acesse múltiplas bases de dados. Na

prática, tal recurso acaba gerando várias chamadas à LLM, o que implica em um tempo de resposta mais elevado.

A ideia é que, ao se utilizar a ferramenta, o responsável pela implementação precise apenas criar esses componentes, tratando as chamadas ao modelo como simples funções capazes de gerar texto — representadas como nós em um grafo. O “*prompt*” criado, no contexto do DSPy, é chamado de “Signature” (em tradução livre, “assinatura”), que apenas indica que, para aquele bloco, dadas as entradas e o formato de saída desejado, qual tarefa deve ser realizada, e não como ela deve ser executada. A intenção aqui é que o DSPy otimize o fluxo, de forma que parte do esforço do desenvolvedor ao testar cada etapa do processo seja abstraída.

Desta forma, assume-se que as dificuldades ligadas às “alucinações” e às “fragilidades de *prompt*” dos modelos sejam reduzidas e, idealmente, eliminadas. A divisão das funcionalidades do fluxo, inclusive aquelas referentes ao acesso às bases de dados da aplicação, pode garantir que a questão da limitação da janela de contexto do sistema gerador de texto também seja uma condição passível de ser remediada.

### 3.5 Concepção do Algoritmo

O protótipo da aplicação foi um agente que implementa um *pipeline* com um grande modelo de linguagem. Ele fez uso do ChatGPT-4o mini, da OpenAI. Lançado em julho de 2024, o GPT-4o mini apresenta resultados sólidos em diversas referências, segundo a própria empresa, sendo uma opção com melhor custo-benefício e qualidade satisfatória. Ele possui uma janela de contexto de até 128 mil *tokens*, aceitando até 16 mil por solicitação, e com base de conhecimento atualizada até outubro de 2023 (OpenAI, 2024).

A OpenAI é considerada uma das empresas mais relevantes na área devido a popularidade e qualidade de seus grandes modelos de linguagem, e o ChatGPT-4o mini apresenta-se como sendo uma opção mais econômica dentre os mesmos. Considerado um modelo “pequeno”, ele também destaca-se comparativamente em relação a outros modelos do tipo (OpenAI, 2024). A qualidade do modelo, somado a sua acessibilidade quanto ao custo, tornam-lhe uma excelente escolha para uso no contexto deste trabalho.

Quanto ao *framework* DSPy, foram empregados os módulos Chain of Thought, ReAct e Predict (módulo de inferência simples). Para a implementação da base vetorial, foi utilizada a biblioteca ChromaDB, responsável por armazenar textos após passarem pelo processo de *embedding* e por realizar consultas de busca. A escolha do ChromaDB se justifica por ser uma solução de código aberto e de uso simplificado (CHROMA).

Foram estabelecidas cinco fontes de dados distintas para alimentar o agente: um arquivo JSON com as linhas de ônibus da cidade; outro com informações sobre os bairros

e suas localizações; uma pasta com notícias diárias também em formato JSON; um aditivo contendo imóveis fictícios; e uma base vetorial com conteúdos sobre João Pessoa, extraídos de artigos e textos criados pelo próprio modelo de linguagem sobre locais e regiões da cidade.

Cada uma dessas bases de dados poderá ser acessada por um conjunto de funções que obterão todo o seu conteúdo ou similares filtrados de diferentes formas:

- Para a base de dados contendo informações acerca de linhas de ônibus na cidade, as funções disponíveis são: consultar uma linha específica através de seu nome e obter suas respectivas informações; verificar aquelas disponíveis em um bairro; e identificar um trajeto possível entre dois bairros, a partir do nome dos mesmos, mostrando quais são possíveis para fazer o percurso.
- Para a base de dados contendo detalhes acerca dos bairros na cidade, as funções disponíveis são: consultar todas as informações relacionadas ao bairro; busca por um tipo de localidade (os citados incluem bares, restaurantes, escolas, academias, farmácias, igrejas, praias, parques, faculdades, hospitais, mercados) na cidade; identificar as melhores zonas para cada tipo, ranqueadas por sua nota respectiva; localizar um exemplar em um bairro específico (por exemplo, restaurantes no bairro Bancários); e acessar dados relacionados à região, exceto por seus pontos de interesse.
- Para a base de dados contendo notícias, será possível obter notícias referentes ao dia atual ou alguma matéria em um dia específico anterior. Paralelamente, as informações climáticas em tempo real e previsão do tempo semanal também estarão disponíveis para quando for feita uma chamada às funções que contém informações de notícias.
- Para a base de dados contendo imóveis fictícios na cidade, será possível pesquisar por: unidades em um bairro específico; propriedades de uma categoria relacionada à finalidade de uso (aluguel ou aquisição permanente); moradias de um tipo definido (casa ou apartamento); e imóveis conforme os critérios já citados, mas filtrados por localização específica. Ademais, será possível também fazer uma pesquisa por preço do metro quadrado de imóveis por bairro para compra.
- Para a base de dados vetorial, considerando uma query de busca, serão retornadas um certo número de segmentos “n”, ou seja, os n trechos mais semelhantes ao termo pesquisado. Pode ser escolhido um valor arbitrário e, no caso, será utilizado o número 7.

Desta forma, será viável que, durante a execução do *pipeline* de geração de resposta, o agente consiga acessar as informações disponíveis.

A existência de múltiplas fontes de informação acaba por gerar também a necessidade de diversas chamadas inferenciais em alguns contextos de resposta. A abordagem escolhida para se lidar com a complexidade da criação de um *pipeline* do tipo é a de abstrair o acesso a cada repositório através de um agente. Na prática, existem múltiplos módulos autônomos, cada um responsável por receber uma requisição de busca e retornar os conteúdos que encontra em sua respectiva base. Deste modo, dada uma requisição inicial para geração de resposta, um componente em nível superior terá consigo uma lista de operações que pode executar, com nome e descrição. Algumas dessas ações são, na prática, instruções que resultam na chamada a outro agente.

Por exemplo, diante da mensagem “As casas no bairro Bancários são baratas?”, o agente de geração de resposta poderá chamar os métodos que possuem acesso às bases de informações de bairro e de imóveis, passando, por exemplo, o termo de busca “preço casas Bancários”. Cada um desses métodos, por sua vez, corresponde a um agente próprio, que possui acesso à sua respectiva base e a um conjunto de operações relacionadas ao domínio em questão. Individualmente, essas rotinas também podem, caso seja suficientemente complexa, ter uma unidade dedicada para lidar com ela.

A implementação desses agentes é feita por meio do uso do módulo ReAct do DSPy. Já para as funções que necessitam de uma inferência menos complexas para serem utilizadas, e que não disponibilizam muitas ações, emprega-se o módulo Chain of Thought ou o módulo Predict, destinado a inferências simples.

Conforme a imagem abaixo, torna-se possível visualizar o *pipeline* do agente de forma conceitual, observando-se o esquema de como circula a sua informação. Esse *pipeline* é sempre executado durante a geração de uma resposta.

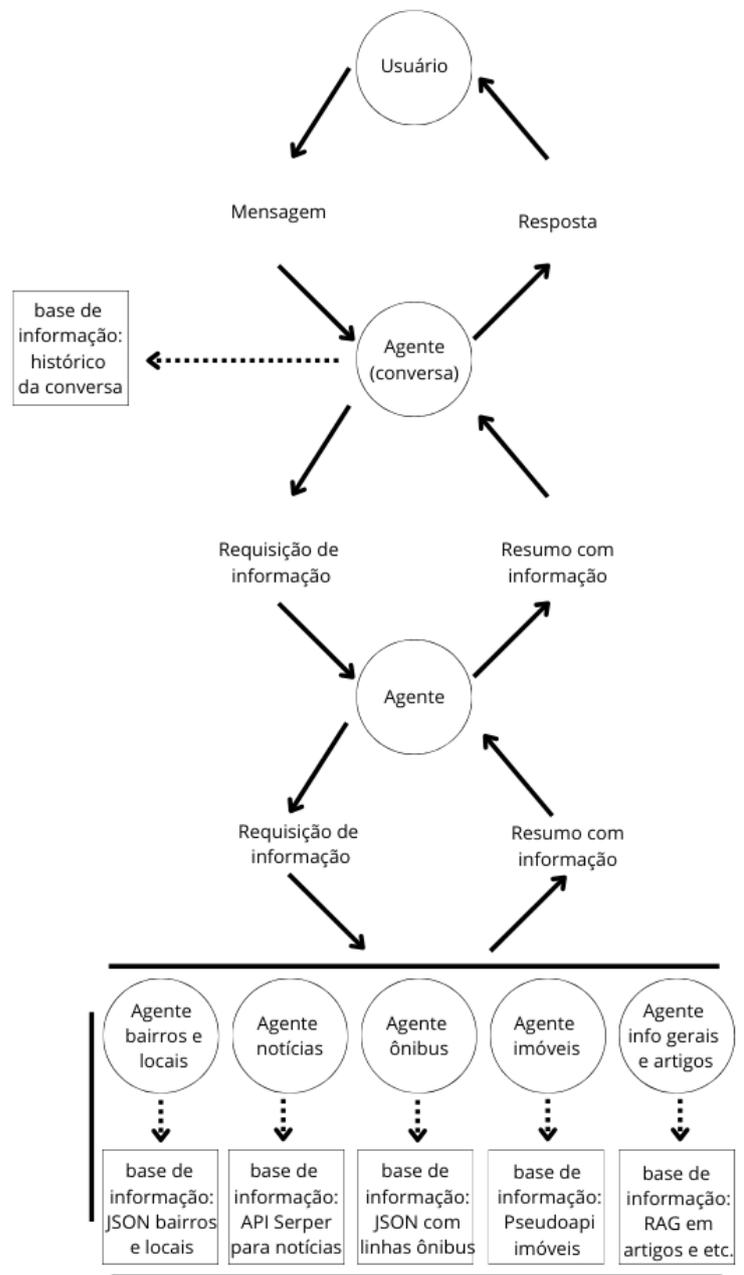


Figura 1: Esquema do *pipeline* do agente, dividido em subagentes.

Fonte: Elaborado pelo autor.

### 3.6 Avaliação de Resultado

Para avaliação do resultado, foi necessário adotar uma abordagem que examinasse a qualidade do *chatbot* quanto a sua qualidade conversacional. Um *framework* proposto

por Følstad e Taylor (2021) tem como meta analisar a capacidade da ferramenta no que se refere a serviço para cliente, considerando sua habilidade de executar ações e sua coerência no diálogo. Ao fim da conversa, procura-se avaliar o seu resultado pela perspectiva do usuário e a eficiência da ferramenta conversacional.

Embora o protótipo de agente proposto neste trabalho tenha intenção de ser usado por usuários reais, na prática, torna-se necessário fazê-lo com casos de testes representativos. Para tal, seriam requeridos participantes com perfil semelhante ao pensado para a aplicação. Ademais, realizar testes contínuos com o sistema também exige uso intensivo de chamadas aos serviços de que ele depende, como as respectivas APIs.

Dadas as limitações já estabelecidas de recursos, percebe-se que fazê-lo na prática torna-se inviável. Portanto, foi proposta uma adaptação do *framework*. Como ainda se deseja verificar a capacidade do *chatbot* em oferecer respostas de forma adequada, foi criado um banco de perguntas consideradas relevantes para turistas e possíveis moradores no contexto de João Pessoa. Em seguida, o agente gerou conteúdos com base nessas solicitações.

Para cada pergunta respondida pelo agente, haverá uma “resposta ideal”, que reflete a melhor formulação possível com base nas informações disponíveis. A partir da saída gerada pelo agente, comparada também à resposta ideal, são avaliados três aspectos:

- Corretude da resposta: Considera o quão próxima esteve a resposta do agente em relação à ideal. No melhor cenário, o sistema é capaz de acessar todas as informações necessárias para responder a determinada pergunta, fornecendo um conteúdo similar ao da referência adequada.
- Relevância da resposta: Considera se o agente conseguiu fornecer uma saída pertinente no contexto da pergunta. Embora a corretude seja um aspecto importante de avaliação, ela ainda é bastante rígida, pois assume que uma pergunta tem apenas uma resposta ideal. Num cenário de conversa real, pode-se imaginar que existe mais de um tipo de resposta que possa ser considerada satisfatória em uma conversa.
- Compreensibilidade da resposta: O quão compreensível é a resposta. Uma resposta clara consegue usar linguagem objetiva quando possível, utiliza a quantidade adequada de informação e apresenta uma formatação textual visualmente agradável.

Para cada aspecto, está estabelecida uma nota, que vai de 1 (menor) até 5 (maior). São analisadas as notas do agente, a média para cada aspecto, a mediana de cada, e também a sua capacidade de responder perguntas relacionadas a determinados tópicos.

Para Corretude, que compara uma resposta ideal com a do agente:

**Tabela 1: Critérios de avaliação da corretude das respostas**

<b>Nota</b>	<b>Descrição do critério para nota</b>
5	Resposta totalmente correta, apresenta todas as informações esperadas.
4	Resposta correta, mas carece de algumas informações não tão relevantes.
3	Resposta parcialmente correta, carece de alguns detalhes importantes.
2	Resposta parcialmente incorreta, carece de vários detalhes importantes.
1	Resposta incorreta, não apresenta nenhuma informação útil ou relacionada ao tópico da pergunta.

**Fonte: Elaborado pelo autor.**

Por exemplo, ao perguntar-se ao agente uma descrição completa acerca do bairro Bessa, espera-se uma resposta apresentado uma série de informações. Uma resposta tal como "O Bessa é um bairro ao norte da cidade, próximo a Cabedelo" apresenta informações corretas, mas mais informações poderiam ter sido fornecidas (como por exemplo, o bairro ser conhecido pela sua qualidade de vida). Logo, a nota na corretude não seria baixa, mas também não seria alta.

Para Relevância, que julga uma resposta com base na temática da pergunta:

**Tabela 2: Critérios de avaliação da relevância das respostas**

<b>Nota</b>	<b>Descrição do critério para nota</b>
5	Resposta totalmente relevante, aborda todos os aspectos da pergunta
4	Resposta relevante, aborda a maioria dos aspectos relacionados a pergunta
3	Resposta parcialmente relevante, aborda os principais pontos da pergunta, mas deixa informações importantes de lado.
2	Resposta parcialmente irrelevante, tangência vários elementos da pergunta.
1	Resposta irrelevante, tangência totalmente o tema da pergunta.

**Fonte: Elaborado pelo autor.**

Embora os critérios "Corretude" e "Relevância" sejam semelhantes, existem, na prática, algumas diferenças a serem consideradas. Por exemplo, ao perguntar o agente "me sugira restaurantes no bairro de Mangabeira", uma resposta com corretude alta iria

conter detalhes acerca de diversos restaurantes da base de dados referentes aquele bairro. Uma resposta com relevância alta poderia apenas conter a descrição de alguns restaurantes, visto que para tal critério de avaliação, menos detalhes na resposta são necessários.

Para Compreensibilidade e Objetividade (chamada só de "Compreensibilidade"), que julga uma resposta com base na sua objetividade e o quão compreensível ela é.

**Tabela 3: Critérios de avaliação da compreensibilidade das respostas**

Nota	Descrição do critério para nota
5	Resposta totalmente clara e objetiva, apresenta as informações com linguagem simples, de forma clara e direta.
4	Resposta clara e objetiva: é clara e objetiva, mas poderia ser mais bem estruturada em algum ponto.
3	Resposta parcialmente clara e objetiva, possui trechos que podem parecer confusos ou não claros, mas no geral entendível.
2	Resposta parcialmente complexa ou confusa.
1	Resposta totalmente confusa ou mal-estruturada.

**Fonte: Elaborado pelo autor.**

Uma resposta desnecessariamente complexa e pouco objetiva, ou simplesmente mal-estruturada, obteria uma nota ruim quanto a esse critério de avaliação.

É também feita uma análise de em quais tópicos o agente apresenta desempenho bom ou insatisfatório, os motivos por trás desses resultados e como ele raciocina para chegar às respostas que fornece, de modo que seja possível entender plenamente em quais aspectos o sistema carece de mudanças.

Para a criação das perguntas do seu referido banco, foi definida como estratégia inicial agrupar as questões por tópicos, onde cada uma e sua respectiva "resposta ideal" fossem baseadas diretamente, para o seu tema, em alguma das bases de conhecimento fornecidas ao agente. Portanto, as perguntas foram organizadas da seguinte forma: questões sobre informação de bairro, e perguntas sobre locais, sobre transporte público (uso de ônibus), sobre dados gerais da cidade, sobre notícias locais e sobre o mercado imobiliário em João Pessoa.

É importante considerar que o que está sendo avaliado é a capacidade do agente de responder às perguntas com os dados que lhe foram fornecidos através de sua base de conhecimento. Portanto, não vem ao caso a criação de questões que abranjam tópicos além das fontes de informação apresentadas.

A lista de perguntas está nos Anexo A, em Perguntas feitas ao agente para teste.

## 4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Os resultados apresentados nas seguintes seções foram obtidos com base na aplicação da metodologia proposta previamente. Serão mostradas as notas das respostas, as médias e as medianas para cada critério (corretude, relevância, compreensibilidade e objetividade). Além disso, serão destacadas quantas respostas obtiveram notas 4 e 5, que podem ser considerados valores que indicam bom desempenho.

Os resultados também serão apresentados organizados por tópico, de forma que torne-se mais simples observar a performance do agente em relação a tópicos específicos.

### 4.1 Resultados e análise

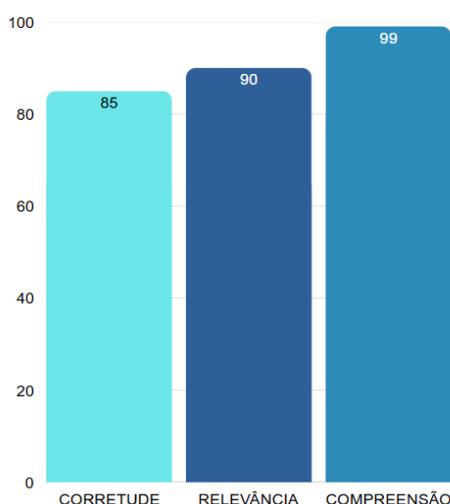
#### 4.1.1 Resultados gerais

A tabela a seguir contém o compilado das notas das respostas do agente no que se refere aos critérios de avaliação propostos.

**Tabela 4: Resultados da avaliação das respostas do agente**

Indicador	Média	Mediana
Corretude - nota	4.11	4
Relevância - nota	4.60	5
Compreensibilidade e Objetividade - nota	4.66	5
Tempo de Resposta (segundos)	24.35	21

**Porcentagem de notas 4 ou 5 (%):**



**Figura 2: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério.**

No geral, o agente demonstrou boa capacidade de resposta, conquistando boas médias para todos os critérios, com desempenho notável para os critérios compreensibilidade e objetividade e também relevância. Estes resultados indicam que, a partir da base de dados disponibilizada, o agente mostrou-se capaz de obter as informações necessárias para fornecer respostas suficientemente boas.

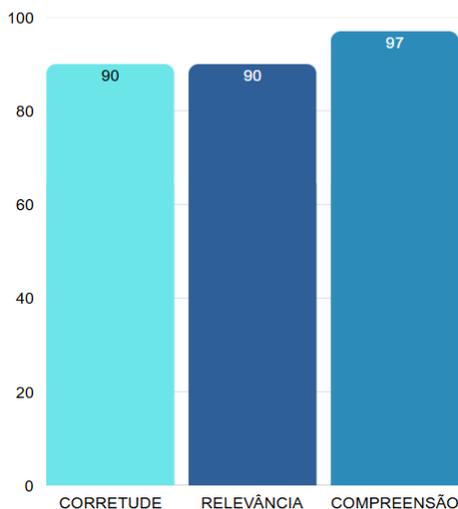
Os resultados das seções seguintes indicam o desempenho do agente em tópicos específicos.

#### 4.1.2 Perguntas relacionadas a bairros

**Tabela 5: Resultados da avaliação das respostas do agente para perguntas relacionadas a bairros**

Indicador	Média	Mediana
Corretude - nota	4.05	4
Relevância - nota	4.60	5
Compreensibilidade e Objetividade - nota	4.97	5
Tempo de Resposta (segundos)	30	18

Porcentagem de notas 4 ou 5 (%):



**Figura 3: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema bairro.**

As respostas do agente obtiveram média boa para relevância e compreensibilidade. A nota menor de média para corretude indica dificuldade em encontrar todas as informações disponíveis na base de dados. O tempo médio de resposta alto indica maior procura na base de dados em relação a algumas informações, e possíveis dificuldades.

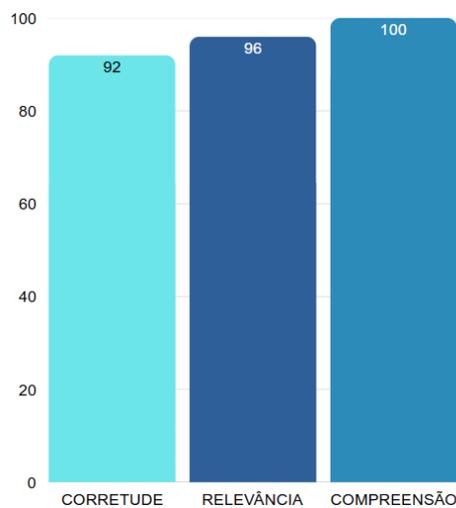
Como os dados de bairros estão disponibilizados em diversas bases de dados, e não apenas uma, seria razoável assumir inicialmente que o agente mostrasse dificuldade em acessar e obter toda a informação possível.

#### 4.1.3 Perguntas relacionadas a imóveis

**Tabela 6: Resultados da avaliação das respostas do agente para perguntas relacionadas a imóveis**

Indicador	Média	Mediana
Corretude - nota	4.33	4
Relevância - nota	4.88	5
Compreensibilidade e Objetividade - nota	4.92	5
Tempo de Resposta (segundos)	14	15

Porcentagem de notas 4 ou 5 (%):



**Figura 4: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema imóveis.**

As respostas do agente para esse tópico e suas respectivas notas indicam uma boa capacidade de obter e fazer uso das informações necessárias e gerar uma resposta em um formato ideal. Além disso, notou-se um tempo de resposta abaixo da média geral, o que indica a qualidade da parte do pipeline relacionada ao acesso a essa parte da base de dados.

Considerando que os dados relacionados a imóveis estão quase todos em uma base de dados única, percebe-se que esse fator foi determinante no bom desempenho do agente.

#### 4.1.4 Perguntas relacionadas a informações gerais da cidade

Tabela 7: Resultados da avaliação das respostas do agente para perguntas relacionadas a informações gerais da cidade

Indicador	Média	Mediana
Corretude - nota	4.07	4
Relevância - nota	4.57	5
Compreensibilidade e Objetividade - nota	4.79	5
Tempo de Resposta (segundos)	29	30

Porcentagem de notas 4 ou 5 (%):

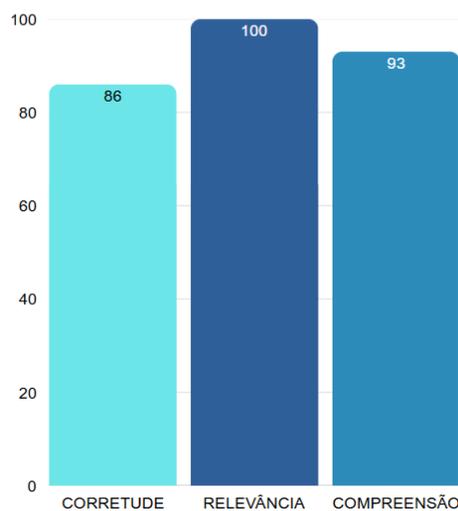


Figura 5: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema informações gerais da cidade.

Nota-se um tempo de resposta acima da média, assim como a mediana, o que indica uma maior busca por informação. Uma nota de corretude abaixo da média corrobora com o mesmo. Esses fatores indicam uma possível dificuldade do agente em acessar todas as informações necessárias para responder as perguntas feitas, ou simplesmente que o agente não julga necessitar de tanta informação ao gerar sua resposta, pois considera respostas mais simples boas o suficiente.

Outro fator a se considerar é que diversas bases de dados possuem informações consideradas relevantes para a maioria das perguntas. Desta forma, justifica-se uma dificuldade maior do agente em obter certos tipos de informação.

#### 4.1.5 Perguntas relacionadas a locais

Tabela 8: Resultados da avaliação das respostas do agente para perguntas relacionadas a locais e tipos de locais na cidade

Indicador	Média	Mediana
Corretude - nota	3.87	4
Relevância - nota	4.91	5
Compreensibilidade e Objetividade - nota	4.65	5
Tempo de Resposta (segundos)	28	30

Porcentagem de notas 4 ou 5 (%):

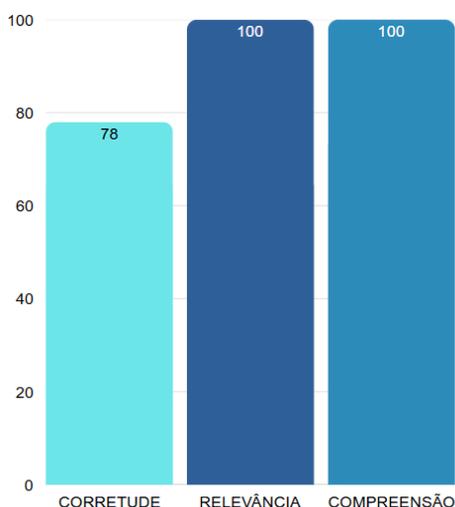


Figura 6: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema locais na cidade.

A menor média já observada para corretude, aliada a nota alta para relevância, indicam que o agente é capaz de trazer informações consideradas relevantes para o contexto em suas respostas, mas possivelmente produzindo respostas menos completas ou complexas do que as respostas esperadas de referência. A nota observada para compreensibilidade e objetividade, abaixo da média geral, indica que o agente também produz respostas em um formato considerado não ideal. De fato, as respostas observadas muitas vezes mostraram-se prolixas.

Pode-se concluir, portanto, que o agente esteja recebendo muita informação quando vai gerar sua resposta ao acessar a base de dados relacionada a locais, o que impacta a qualidade das mesmas. Como comentado anteriormente, janelas de contexto maiores tendem a dificultar a geração de respostas por parte de LLMs. Considerando também um cenário de conversação, onde respostas longas são desencorajadas, torna-se nítido que o

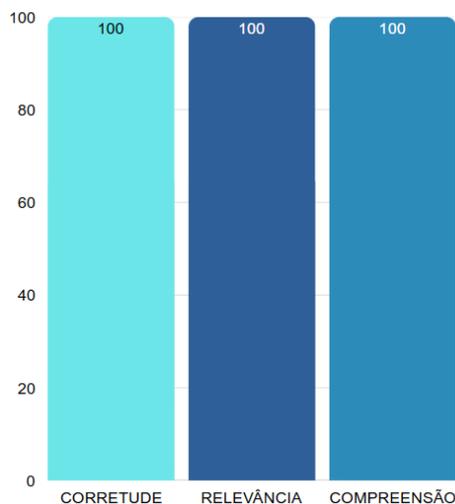
agente teve dificuldade em conciliar a quantidade de informação a sua disposição e o tipo de resposta que deveria produzir

#### 4.1.6 Perguntas relacionadas a notícias

**Tabela 9: Resultados da avaliação das respostas do agente para perguntas relacionadas a notícias na cidade**

Indicador	Média	Mediana
Corretude - nota	5	5
Relevância - nota	5	5
Compreensibilidade e Objetividade - nota	4	4
Tempo de Resposta (segundos)	20	20

Porcentagem de notas 4 ou 5 (%):



**Figura 7: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema notícias.**

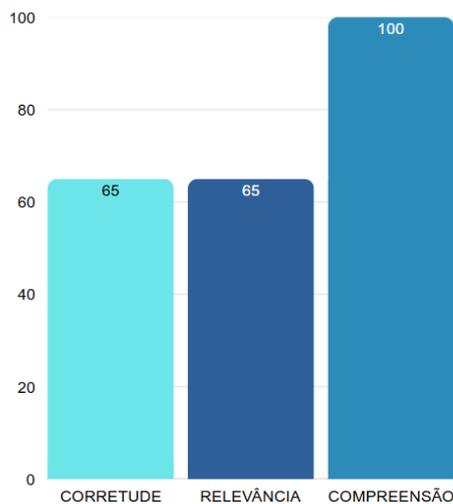
O agente apresentou respostas com conteúdo como esperado. Contudo, apresentou baixa compreensibilidade relativa aos outros resultados. Um fator determinante a isso foi o fato da produção de respostas com uso de língua inglesa, mesmo em um contexto onde a língua portuguesa foi usada para envio da mensagem inicial. Uma hipótese é a de que, diante das informações de notícias recebidas em um JSON de conteúdo majoritariamente em inglês, o agente considerou que devesse produzir uma resposta no mesmo idioma, o que não deveria ser o caso.

#### 4.1.7 Perguntas relacionadas ao transporte público na cidade

**Tabela 10: Resultados da avaliação das respostas do agente para perguntas relacionadas ao transporte público na cidade**

Indicador	Média	Mediana
Corretude - nota	3.35	4
Relevância - nota	3.6	4
Compreensibilidade e Objetividade - nota	4.65	5
Tempo de Resposta (segundos)	12	11

Porcentagem de notas 4 ou 5 (%):



**Figura 8: Gráfico mostrando quantas respostas obtiveram notas 4 ou 5 para cada critério em relação as perguntas de tema transporte público.**

O agente demonstrou dificuldade em acessar e utilizar certas funções relacionadas a base de dados de transporte público, o que resultou em diversos erros e, consequentemente, respostas de pouco conteúdo ou valor. Por exemplo, ao tentar invocar uma função relacionada a uma linha de ônibus específica, o agente demonstrou dificuldade em chamar a função passando o nome correspondente da linha de ônibus devida. Isso indica que as funções apresentadas ou fornecidas necessitam de uma edição ou reformulação, de forma que torne-se mais claro ao agente de como usá-las de forma mais efetiva, para que o mesmo gere respostas de maior qualidade.

#### 4.2 Limitações observadas

No que pode-se observar para cada critério de nota e do raciocínio do agente para geração de resposta, pode ser observadas situações que justifiquem uma boa ou má resposta para o agente:

- Para “corretude”, observa-se duas situações: o agente pode não encontrar toda a informação necessária, em seu raciocínio, gerando assim uma resposta aquém do esperado; o agente encontra toda a informação que precisa, mas em alguns casos, quando um módulo de agente acaba por criar uma resposta para outro módulo de agente, a resposta criada não apresenta todas as informações necessárias (o sub-agente emite informações). Isso gera uma ”perca” de informação durante a geração da resposta.
- Para “relevância”, as condições são semelhantes ao critério de corretude. Contudo, como uma resposta extremamente relevante não necessariamente terá máxima corretude, os fatores limitantes causam um impacto menor.
- Para compreensibilidade e objetividade, pode-se perceber que, em alguns casos, o agente gera respostas com formatação textual confusa, particularmente no que se refere às situações onde há muita informação que ele considera necessária para gerar uma resposta. Esse é o critério de melhor desempenho do agente

### 4.3 Possíveis pontos de melhora

Os resultados sugerem que, embora o agente apresente um bom desempenho, existem pontos em que há uma possível melhora, e tais pontos devem ser diretamente afetados em futuras versões.

#### 4.3.1 Erro ao executar função

Durante a chamada ou execução de alguma função, caso ocorra uma exceção que seja devidamente tratada, é comportamento padrão dos módulos ReAct e Chain of Thought do DSPy tentar novamente a execução da função em questão. No ambiente em que o agente está inserido, os erros podem surgir devido a falhas internas ou chamadas incorretas das funções (como uso inadequado de parâmetros).

Uma situação possível é o próprio agente não conseguir identificar corretamente quais argumentos utilizar em determinada função, gerando assim um erro. Nesses casos, é comum que diversas tentativas sejam feitas sem sucesso, impedindo o progresso pelas etapas do pipeline e comprometendo a qualidade da resposta. Uma forma de lidar com tal problema é demonstrar ao agente, de forma mais eficiente, como utilizar uma determinada função.

Erros internos de funções, causados por outros fatores, indicam que as funções devem ser refeitas.

### 4.3.2 Tempo de resposta

A inconsistência nos tempos de respostas, com médias maiores para alguns tópicos, indica que o tempo de resposta é um fator a se considerar numa futura versão do agente, considerando também que o agente tem como intuito ser uma entidade conversacional. Tempos altos para certas respostas indicam, por exemplo, múltiplas inferências e chamadas a funções variadas, o que o agente está buscando alguma informação, talvez sem sucesso;

Uma possível melhora deveria considerar o seguinte equilíbrio: usar uma quantidade menor de inferências poderia reduzir o tempo de resposta, ao custo de que, para cada inferência a menos realizada, as inferências restantes teriam uma maior responsabilidade, pois provavelmente iriam ter que receber *prompts* maiores e gerar saídas menores, o que geraria mais erros.

Outro fator a se considerar é em relação aos casos de erro durante a chamada ou execução de alguma função. Visto que espera-se que o programa faça novas tentativas até atingir um certo limite ou obtenção de sucesso, o tempo de resposta sofre aumento nessas situações.

### 4.3.3 Respostas com conteúdo insuficiente

Em alguns casos, percebe-se que o agente acaba por produzir respostas com menos conteúdo do que a informação que ele teoricamente teria obtido. Por exemplo, nas perguntas acerca de localidade, o agente não apresenta certos detalhes, como descrição do local, embora outros detalhes como endereço ou avaliação apareçam.

Isso sugere que, durante o processo de chamadas internas entre agentes e subagentes, há informação “perdida” ou não emitida. Isso pode ocorrer, por exemplo, caso um subagente produza uma resposta excessivamente curta a outro subagente. Esse fenômeno pode ser prejudicial à geração de resposta por parte do agente, e poderia explicar respostas contendo menos informação do que o esperado.

Outro possível motivo é que, durante a geração de sua resposta, o agente simplesmente considera que não precisa de tantos dados para gerar sua resposta. Embora idealmente a autonomia seja algo desejável, talvez nesse caso isso indique que haja uma necessidade de mudança nos módulos que constituem o agente, para que seu comportamento se aproxime do ideal.

### 4.3.4 Dificuldade em obter certas informações

Percebe-se que, para algumas perguntas, o agente indicou em sua resposta não ter o conhecimento necessário para responder a pergunta. Isso indica que o agente não

conseguiu obter as informações necessárias na base de dados. Tal fenômeno pode ser explicado por algumas razões: incapacidade de um agente ou subagente de usar alguma das funções que lhe foi disponibilizada ou procurar nas bases de dados disponíveis; Erro de comunicação entre os subagentes, de forma que em algum ponto dessa comunicação haja uma mensagem que induz uma das partes ao erro;

Ademais, observou-se que, para perguntas complexas, o agente muitas vezes acaba por responder a pergunta apenas parcialmente.

#### 4.4 Exemplos de interações

As imagens a seguir contém exemplos de conversas feitas com o agente, de forma que torne-se possível visualizar as respostas do mesmo. As conversas mostradas não foram incluídas na avaliação de resultados. Para facilitar a visualização, foi criada uma interface no *framework* Streamlit. O Streamlit é um *framework* destinado a criação e prototipação de interfaces de forma simplificada (STREAMLIT).



oi. me sugira um roteiro para 3 dias em joao pessoa

Para um roteiro de 3 dias em João Pessoa, você pode considerar as seguintes atividades e atrações: **Dia 1: Exploração Cultural e Histórica - Manhã:** Visite o Centro Histórico de João Pessoa, onde você pode explorar o Centro Cultural São Francisco e a Basílica Nossa Senhora das Neves. Não perca o Mercado de Artesanato Paraibano (MAP) para comprar souvenirs. - **Almoço:** Experimente a gastronomia local em um dos restaurantes do centro. - **Tarde:** Passeie pelo Parque Solon de Lucena, conhecido como Parque da Lagoa, e aproveite a área verde. - **Noite:** Assista a uma apresentação no Teatro Santa Roza ou no Casarão 34, que oferecem programação cultural. **Dia 2: Praias e Natureza - Manhã:** Dirija-se à Praia de Cabo Branco e aproveite a praia e o calçadão. Você pode alugar uma bicicleta para explorar a área. - **Almoço:** Almoce em um dos muitos restaurantes à beira-mar. - **Tarde:** Faça uma trilha na Mata do Buraquinho ou nas trilhas da praia do Coqueirinho, onde você pode ver cânions de argila colorida. - **Noite:** Jante na Praia de Tambaú e visite a Feirinha de Tambaú para artesanato e comidas típicas. **Dia 3: Aventura e Relaxamento - Manhã:** Faça um passeio de quadriciclo pelas trilhas do Conde, visitando os mirantes Dedo de Deus e das Tartarugas. - **Almoço:** Almoce em um restaurante local no Conde. - **Tarde:** Participe de um passeio de barco para as piscinas naturais de Picãozinho ou do Seixas, onde você pode mergulhar e ver a vida marinha. - **Noite:** Retorne a João Pessoa e aproveite um jantar em um dos shoppings, como o Manaíra Shopping, que também oferece opções de entretenimento.

quais restaurantes vc sugere no cabo branco

Os restaurantes que eu sugiro na Praia de Cabo Branco são: 1. Gulliver Mar Restaurante Cabo Branco - Avaliação: 4.7. 2. Fabio Beach Bar e Restaurante - Avaliação: 4.7. 3. HG Petiscaria - Avaliação: 5. 4. Don Germoglio - Avaliação: 4.9. 5. Restaurante Peixada do Amor - Avaliação: 4.7. Esses são alguns dos melhores lugares para comer na Praia de Cabo Branco.

Figura 9: Exemplo de conversa simples, onde inicialmente se pede para que o agente construa um plano de viagem curta (3 dias).

Após a elaboração do plano pelo agente, pergunta-se a ele uma sugestão de res-

taurante para visitar em um bairro específico (Cabo Branco).

## Teste local do chatbot



vou me mudar para João Pessoa, irei trabalhar em uma empresa no bairro Mangabeira. Como é o custo de vida na região?



O custo de vida em Mangabeira é moderado, com uma renda média de aproximadamente R 943,56. *Os preços de aluguel variam, com apartamentos de 3 quartos custando cerca de R 806,4.* O bairro oferece uma boa variedade de opções de comércio e lazer, tornando-o uma boa escolha para quem se muda para João Pessoa.



quanto é o custo de metro quadrado no bairro? vc sabe de casas a venda na região?



O custo do metro quadrado em Mangabeira é de R 3.315. *As casas à venda na região incluem uma casa com 2 quartos, próxima à Praça do Parque, por R 168.000, e outra casa com 2 quartos, próxima à Parada de Ônibus, também por R 168.000. Além disso, há uma casa para alugar com 1 quarto, próxima à Igreja, por R 1.344.*



vc sugere morar em Mangabeira ou em algum bairro próximo?



Após comparar Mangabeira com os bairros próximos, posso sugerir que Mangabeira é uma boa opção devido ao seu custo de vida moderado e infraestrutura, mas bairros como o Bairro dos Estados ou Tambaú também podem ser considerados, dependendo das preferências pessoais do usuário, como proximidade ao trabalho, opções de lazer e comércio.

Figura 10: Exemplo de conversa simples, onde inicialmente pergunta-se sobre o bairro Mangabeira.

É possível notar a existência de erros em uma das respostas fornecidas (bairros Tambaú e dos Estados não se localizam próximos a Mangabeira). A má formatação textual observada ocorre devido a interface do Streamlit.

## Teste local do chatbot

 ola. estou pensando em me mudar para João Pessoa e ter uma casa lá. Quais bairros tem metro quadrado mais barato?

 bairros com metro quadrado mais barato em João Pessoa: Rangel (R\$2.219), Varadouro (R\$2.177), Emani Sátyro (R\$1.998), BairrodasIndústrias (R\$2.550), Roger (R\$2.492), AltodoMateus (R\$2.687), Grotão (R\$2.655), Valentina (R\$2.715), Planalto da Boa Esperança (R\$2.728), Colibris (R\$2.749).

 como é morar no Rangel?

 O Rangel é um bairro residencial de baixa renda em João Pessoa, conhecido por ser próximo ao centro da cidade. A renda média é de aproximadamente R\$ 616,54, o que é consideravelmente mais baixo em comparação com outros bairros. O bairro possui uma dinâmica de atividades econômicas, mas é importante considerar que a infraestrutura e os serviços podem ser limitados. Se você busca um lugar mais acessível, o Rangel pode ser uma opção, mas é bom visitar e conhecer a área pessoalmente antes de decidir.

Figura 11: Exemplo de conversa simples, onde inicialmente pergunta-se ao agente sobre bairros acessíveis para morar.

Em seguida, dada uma lista de sugestões, pergunta-se ao mesmo sobre um desses bairros (Rangel). O agente apresenta uma breve descrição do mesmo.

## Teste local do chatbot

 ola! queria visitar praias em joao pessoa. me sugira quais ir ver

 As melhores praias para visitar em João Pessoa incluem: Praia de Cabo Branco, Praia de Tambaú, Praia do Seixas, Praia do Coqueirinho, Praia Naturista de Tambaba, Praia do Amor e Barra de Gramame. Essas praias oferecem uma combinação de beleza natural, opções de lazer e uma rica cultura local.

 como vai estar o clima nessa semana? devo me preocupar com chuva?

 Para esta semana, a previsão do tempo em João Pessoa indica uma mínima de 24.9°C e máxima de 29.9°C na segunda-feira, com uma pequena chance de chuva. Nos dias seguintes, não há previsão de chuva, com temperaturas variando entre 25.3°C e 31.4°C. Portanto, não há necessidade de se preocupar excessivamente com a chuva, exceto para o início da semana.

Figura 12: Exemplo de conversa simples, onde inicialmente pede-se sugestões de praia para serem visitadas.

Em seguida, pergunta-se acerca da previsão do tempo para a semana.

## 5 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho apresentou o desenvolvimento e avaliação de um protótipo de agente baseado em Grandes Modelos de Linguagem (LLMs) com foco em fornecer informações e assistência relacionadas à cidade de João Pessoa, tanto para turistas quanto para potenciais residentes.

A implementação do protótipo, tomando como base os objetivos citados na introdução, mostrou que a criação de um agente real desse tipo é viável e também capaz de reproduzir bons resultados, considerando o desempenho obtidos nos testes iniciais. Este, embora tenha apresentado algumas limitações e possíveis pontos de melhoria, demonstrou potencial de aplicabilidade em contextos reais.

A metodologia adotada, incluindo a seleção, coleta e tratamento dos dados utilizados na base de conhecimento, pode ser adaptada e reaplicada a diferentes contextos. Os tipos de dados escolhidos mostraram-se relevantes quanto a vivência da população da cidade. As fontes escolhidas, como *blogs* e *sites* locais, mostraram-se condizentes com a realidade da cidade. Ademais, tais fontes possivelmente possuem versões análogas para outras cidades ou regiões, tornando viável a replicação do processo de coleta e uso de dados.

Além disso, algumas dessas fontes, como o IBGE e o Google Maps, oferecem informações amplas sobre diversas localidades do país, o que facilita a extensão da abordagem. Naturalmente, ajustes na escolha e no tratamento dos dados podem ser necessários, a depender das particularidades de cada cidade ou região abordada pela aplicação.

O desenvolvimento do algoritmo com uso de LLM foi feito considerando algumas das diversas limitações observadas (via revisão literária) quanto a essa tecnologia. Após pesquisa e consideração de limitações específicas relevantes ao contexto do trabalho, o algoritmo proposto mostrou-se eficiente em contornar ou reduzir o impacto das mesmas.

A avaliação do agente se deu de forma a fazer uso de meios de avaliação qualitativa da sua capacidade conversacional, considerando métodos relevantes para a área de *chatbots*, adaptando tais métodos para as especificidades do contexto do trabalho. Os resultados obtidos pelo agente nos testes indicam bom desempenho quanto aos métodos de avaliação propostos. É importante considerar que, no trabalho executado, foi avaliada a capacidade do agente de responder às perguntas com os dados que lhe foram fornecidos através de sua base de conhecimento. A criação de questões que abranjam tópicos além das fontes de informação apresentadas não foi, portanto, avaliada.

Um dos fatores que foi possível observar nesse trabalho é a necessidade de dados mais acessíveis e abrangentes sobre João Pessoa. Atualmente, existe uma carência de informações estruturadas e facilmente acessíveis, especialmente no que diz respeito a certos

bairros da cidade e sua população. Essa limitação de dados afeta diretamente a capacidade do agente em fornecer respostas completas e complexas para certos tópicos. Tal fator demonstra a importância de melhorar a infraestrutura de dados locais, para enriquecer as fontes de informação disponíveis.

Vale-se destacar, também, a abordagem utilizada para uso dos dados fornecidos ao agente. Em uma aplicação real, espera-se que haja atualização com uma frequência considerável ao banco de dados disponível, se não em tempo real. O protótipo desenvolvido no trabalho apresenta os dados acessados, particularmente aqueles advindos da Places API, como estáticos. Ademais, a natureza das buscas realizadas na API, embora não sejam não-determinísticas, apresentam um certo grau de imprevisibilidade nos resultados (algo que já é previsto na própria documentação), o que torna difícil a sua reprodução.

Concluindo, demonstrou-se nesse trabalho um exemplo de construção de uma aplicação com LLM. Essa aplicação fez uso de técnicas que melhoram o desempenho dessa tecnologia. O desenvolvimento de um *chatbot* com conhecimentos sobre João Pessoa têm o potencial de melhorar a experiência de turistas e novos residentes, além de servir como exemplo para outras iniciativas similares em diferentes contextos.

## 5.1 Trabalhos Futuros

Em trabalhos ou pesquisas futuras similares, podem ser exploradas diversas abordagens. Caso decida-se fazer um agente aos moldes do apresentado neste trabalho, é necessário melhorar suas limitações, tais como os aspectos em que ele apresenta dificuldade em criar respostas com qualidade, ou apresenta maior demora. Essas melhorias poderiam derivar, por exemplo, do uso de outros módulos ou ferramentas para construção do agente.

Testes mais amplos são necessários para uma melhor validação do protótipo, seja em conversas reais ou de tópicos não abordados na avaliação do agente feita. Outros tópicos a se analisar são o tempo de resposta e o custo de mensagem, de forma que seja possível verificar que o agente é eficaz e economicamente viável para uso contínuo e escalável. Esses testes, somados ao feedback de possíveis usuários e também o uso de uma base de dados mais robusta, poderão ser um caminho para uma aplicação de sucesso.

Ademais, possíveis novas funcionalidades, inseridas na aplicação do *chatbot*, poderão tornar a utilização do mesmo uma experiência mais interessante aos usuários. Diversas ideias apresentam potencial: disponibilizar a localização do usuário ao agente em tempo real, para facilitar as sugestões de localidades; o uso de imagens e outras mídias, retirados de uma base de dados de imagens; um sistema que permitisse traçar o perfil do usuário, para tornar o comportamento do agente mais adequado. Tais ideias são apenas meras sugestões, que necessitam análise de viabilidade, implementação e validação.

## REFERÊNCIAS

- [1] AGENTE IMÓVEL. Mercado Imobiliário 2025 – Índice Imóveis João Pessoa – PB. s.l.: Agente Imóvel, s.d. Disponível em: <<https://www.agenteimovel.com.br/mercado-imobiliario/a-venda/pb/joaopessoa/>>. Acesso em: 10 abr. 2025.
- [2] ARSLAN, Muhammad; GHANEM, Hussam; MUNAWAR, Saba; CRUZ, Christophe. A survey on RAG with LLMs. *Procedia Computer Science*, v. 246, p. 3781–3790, 2024. 28th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2024). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1877050924021860>>. Acesso em: 10 abr. 2025. DOI: 10.1016/j.procs.2024.09.178.
- [3] BENADDI, L.; OUADDI, C.; JAKIMI, A.; OUCHAO, B. A systematic review of chatbots: classification, development, and their impact on tourism. *IEEE Access*, v. 12, p. 78799–78810, 2024. Disponível em: <<https://ieeexplore.ieee.org/abstract/document/10542991>>. Acesso em: 10 abr. 2025. DOI: 10.1109/ACCESS.2024.3408108.
- [4] BOOKING.COM. Travel reinvented: Booking.com’s 2025 travel predictions. s.l.: Booking.com, 16 out. 2024. Disponível em: <<https://www.booking.com/articles/travelpredictions2025.en-gb.html>>. Acesso em: 10 abr. 2025.
- [5] BRASIL. Lei nº 13.709, de 14 de agosto de 2018. Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet). *Diário Oficial da União: seção 1, Brasília, DF, 15 ago. 2018*. Disponível em: <[https://www.planalto.gov.br/ccivil\\_03/\\_ato2015-2018/2018/lei/l13709.htm](https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm)>. Acesso em: 18 abr. 2025.
- [6] CARDOSO, Fábio. Site promocional da Secretaria de Turismo utiliza inteligência artificial nas informações. Prefeitura Municipal de João Pessoa, João Pessoa, 19 jun. 2024. Disponível em: <<https://www.joaopessoa.pb.gov.br/noticias/secretarias-e-orgaos/setur-noticias/site-promocional-da-secretaria-de-turismo-utiliza-inteligencia-artificial-nas-informacoes/>>. Acesso em: 10 abr. 2025.
- [7] CHROMA. Documentação oficial do ChromaDB. s.l.: Chroma, s.d. Disponível em: <<https://docs.trychroma.com/>>. Acesso em: 10 abr. 2025.
- [8] FØLSTAD, A.; TAYLOR, C. Investigating the user experience of customer service chatbot interaction: a framework for qualitative analysis of chatbot dialogues. *Qualitative User Experience*, v. 6, n. 6, 2021. Disponível em:

- <<https://link.springer.com/article/10.1007/s41233-021-00046-5>>. Acesso em: 10 abr. 2025. DOI: 10.1007/s41233-021-00046-5.
- [9] FERREIRA, Angela Lúcia; SILVA, Alexsandro F. C. Para além do muro alto: “Turismo Imobiliário” e novas configurações sócio-espaciais na Região Metropolitana de Natal - RN/Brasil. In: VALENÇA, Márcio Moraes; BONATES, Mariana Fialho (org.). Globalização e marginalidade: o Rio Grande do Norte em foco. Natal: EDUFRN, 2008. v. 2, p. 457-468.
- [10] G1 PB. Preço do aluguel residencial sobe 10% em 2024 em João Pessoa, aponta pesquisa. G1, s.l., 14 jan. 2025. Disponível em: <<https://g1.globo.com/pb/paraiba/noticia/2025/01/14/preco-do-aluguel-residencial-sobe-10percent-em-2024-em-joao-pessoa-aponta-pesquisa.ghtml>>. Acesso em: 10 abr. 2025.
- [11] GOOGLE. Places API: visão geral — Documentação da API Places — Google for Developers. s.l.: Google, s.d. Disponível em: <<https://developers.google.com/maps/documentation/places/web-service/overview?hl=pt-br>>. Acesso em: 10 abr. 2025.
- [12] HAURUM, Kasper Raupach; MA, Ruiqi; LONG, Wen. Real estate with AI: an agent based on LangChain. *Procedia Computer Science*, v. 242, p. 1082–1088, 2024. 11th International Conference on Information Technology and Quantitative Management (ITQM 2024). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1877050924019185>>. Acesso em: 10 abr. 2025. DOI: 10.1016/j.procs.2024.08.199.
- [13] INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. Agregados por setores censitários. Rio de Janeiro: IBGE, s.d. Disponível em: <[https://ftp.ibge.gov.br/Censos/Censo\\_Demografico\\_2010/Resultados\\_do\\_Universo/Agregados](https://ftp.ibge.gov.br/Censos/Censo_Demografico_2010/Resultados_do_Universo/Agregados)>. Acesso em: 10 abr. 2025.
- [14] KADDOUR, Jean; HARRIS, Joshua; MOZES, Maximilian; BRADLEY, Herbie; RAILEANU, Roberta; MCHARDY, Robert. Challenges and applications of large language models. arXiv preprint arXiv:2307.10169 [cs.CL], 19 jul. 2023. Disponível em: <<https://arxiv.org/abs/2307.10169>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2307.10169.
- [15] KHATTAB, Omar et al. DSPy: compiling declarative language model calls into self-improving pipelines. arXiv preprint arXiv:2310.03714 [cs.CL], 5 out. 2023. Disponível em: <<https://arxiv.org/abs/2310.03714>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2310.03714.

- [16] LEWIS, Patrick et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. arXiv preprint arXiv:2005.11401 [cs.CL], 12 abr. 2021. Disponível em: <<https://arxiv.org/abs/2005.11401>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2005.11401.
- [17] LI, Zhuoheng et al. What can LLM tell us about cities? arXiv preprint arXiv:2411.16791 [cs.CL], 25 nov. 2024. Disponível em: <<https://arxiv.org/abs/2411.16791>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2411.16791.
- [18] MIN, Dehai et al. Exploring the impact of table-to-text methods on augmenting LLM-based question answering with domain hybrid data. arXiv preprint arXiv:2402.12869 [cs.CL], 9 abr. 2024. Disponível em: <<https://arxiv.org/abs/2402.12869>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2402.12869.
- [19] NAVEED, Humza; KHAN, Asad Ullah; QIU, Shi; SAQIB, Muhammad; ANWAR, Saeed; USMAN, Muhammad; AKHTAR, Naveed; BARNES, Nick; MIAN, Ajmal. A comprehensive overview of large language models. arXiv preprint arXiv:2307.06435 [cs.CL], 17 out. 2024. Disponível em: <<https://arxiv.org/abs/2307.06435>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2307.06435.
- [20] OPENAI. GPT-4o mini: advancing cost-efficient intelligence. 18 jul. 2024. Disponível em: <<https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/>>. Acesso em: 10 abr. 2025.
- [21] OPENAI. Política de Privacidade. s.l.: OpenAI, 4 nov. 2024. Disponível em: <<https://openai.com/pt-BR/policies/privacy-policy/>>. Acesso em: 10 abr. 2025.
- [22] OPEN METEO. Free open-source weather API. s.l.: Open-Meteo, s.d. Disponível em: <<https://open-meteo.com/>>. Acesso em: 10 abr. 2025.
- [23] RAIAN, M. A. K. et al. A review on large language models: architectures, applications, taxonomies, open issues and challenges. IEEE Access, v. 12, p. 26839–26874, 2024. DOI: 10.1109/ACCESS.2024.3365742.
- [24] SCHERBAKOV, Dmitry et al. The emergence of Large Language Models (LLM) as a tool in literature reviews: an LLM automated systematic review. \*arXiv preprint\* arXiv:2409.04600 [cs.DL], 6 set. 2024. Disponível em: <<https://arxiv.org/abs/2409.04600>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2409.04600.
- [25] SERPER.DEV. The world’s fastest and cheapest Google Search API. s.l.: Serper, s.d. Disponível em: <<https://serper.dev/>>. Acesso em: 10 abr. 2025.

- [26] STANFORD NLP GROUP. DSPy. s.l.: Stanford NLP Group, s.d. Disponível em: <<https://dspy.ai/>>. Acesso em: 10 abr. 2025.
- [27] STREAMLIT. Streamlit documentation. s.l.: Streamlit, s.d. Disponível em: <<https://docs.streamlit.io/>>. Acesso em: 10 abr. 2025.
- [28] SUPERINTENDÊNCIA EXECUTIVA DE MOBILIDADE URBANA DE JOÃO PESSOA – SEMOB/PMJP. Linhas de ônibus. João Pessoa: SEMOB/PMJP, s.d. Disponível em: <<https://servicos.semobjp.pb.gov.br/linhas-de-onibus/>>. Acesso em: 10 abr. 2025.
- [29] TAVARES, Vitor. Como João Pessoa foi de capital 'esquecida' a nova 'queridinha' do verão do Nordeste. BBC News Brasil, Recife, 4 jan. 2025. Disponível em: <<https://www.bbc.com/portuguese/articles/c878w1140evo>>. Acesso em: 10 abr. 2025.
- [30] WEI, Jason et al. Chain-of-thought prompting elicits reasoning in large language models. arXiv preprint arXiv:2201.11903 [cs.CL], 10 jan. 2023. Disponível em: <<https://arxiv.org/abs/2201.11903>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2201.11903.
- [31] YAO, Shunyu et al. ReAct: synergizing reasoning and acting in language models. arXiv preprint arXiv:2210.03629 [cs.CL], 10 mar. 2023. Disponível em: <<https://arxiv.org/abs/2210.03629>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2210.03629.
- [32] ZHAO, Siyun; YANG, Yuqing; WANG, Zilong; HE, Zhiyuan; QIU, Luna K.; QIU, Lili. Retrieval augmented generation (RAG) and beyond: a comprehensive survey on how to make your LLMs use external data more wisely. arXiv preprint arXiv:2409.14924 [cs.CL], 23 set. 2024. Disponível em: <<https://arxiv.org/abs/2409.14924>>. Acesso em: 10 abr. 2025. DOI: 10.48550/arXiv.2409.14924.

## ANEXO A – ANEXOS E APÊNDICES 1

### Lista de bairros de João Pessoa

Aeroclube, Água Fria, Altiplano, Alto do Céu, Alto do Mateus, Anatólia, Bairro das Indústrias, Bairro dos Estados, Bairro dos Ipês, Bancários, Barra de Gramame, Bessa, Brisamar, Cabo Branco, Castelo Branco, Centro, Colibris, Costa do Sol, Costa e Silva, Cristo Redentor, Cruz das Armas, Cuiá, Distrito Industrial, Ernani Sátyro, Ernesto Geisel, Expedicionários, Funcionários, Gramame, Grotão, Ilha do Bispo, Jaguaribe, Jardim Cidade Universitária, Jardim Oceania, Jardim São Paulo, Jardim Veneza, José Américo, João Agripino, João Paulo II, Manaíra, Mandacaru, Mangabeira, Miramar, Mumbaba, Muçumagro, Oitizeiro, Padre Zé, Paratibe, Pedro Gondim, Penha, Planalto da Boa Esperança, Ponta do Seixas, Portal do Sol, Rangel (também chamado de Varjão), Roger, São José, Tambauzinho, Tambaú, Tambiá, Torre, Treze de Maio, Trincheiras, Valentina, Varadouro.

### Lista de tipos de localidades considerados para coleta de informação

- Academias
- Bares
- Escolas
- Faculdades
- Farmácias
- Hospitais
- Igrejas
- Lanchonetes
- Mercados
- Outros (inclui pontos turísticos e outras localidades relevantes)
- Parques
- Praças
- Praias
- Restaurantes

## Tabela de reportagens, notícias e artigos sobre João Pessoa

São artigos e reportagens utilizados para a base de dados da aplicação. Tais textos contém informações utilizadas na base de conhecimento do agente. Na seção de referências, os artigos, notícias ou reportagens são citados de forma mais detalhada, com autor, data de publicação, acesso e link respectivos.

**Tabela 11: Sites utilizados como base de conhecimento do agente**

Nome do Site	Nome do Artigo, Reportagem ou Postagem	Informações Retiradas	Domínio
Filipeia	Filipeia Mapas da Cidade	Contém o Atlas Municipal, além de arquivos com informações de praças e parques.	< <a href="https://filipeia.joaopessoa.pb.gov.br">https://filipeia.joaopessoa.pb.gov.br</a> >
IBGE	Censo Demográfico 2010, Agregados por Setores Censitários	Informações de João Pessoa de acordo com o Censo do IBGE de 2010.	< <a href="https://ftp.ibge.gov.br">https://ftp.ibge.gov.br</a> >
SEMOB-JP	Linhas de Ônibus	Informações das linhas de ônibus na cidade.	< <a href="https://servicos.semobjp.pb.gov.br/">https://servicos.semobjp.pb.gov.br/</a> >
G1	Diversas notícias e reportagens	Reportagens e notícias sobre demografia e turismo da cidade.	< <a href="https://g1.globo.com/l">https://g1.globo.com/l</a> >
Jampa Imóveis	Melhores bairros para morar em João Pessoa, conheça o ranking	Informações relacionadas a João Pessoa e seus bairros.	< <a href="https://jampaimoveis.com/">https://jampaimoveis.com/</a> >

Continua na próxima página

---

**Continuação da Tabela – Sites utilizados como base de conhecimento do agente**

---

<b>Nome do Site</b>	<b>Nome do Artigo, Reportagem ou Postagem</b>	<b>Informações ou Retiradas</b>	<b>Domínio</b>
Turismo João Pessoa	Bem-vindo a João Pessoa	Informações sobre localidades e pontos turísticos da cidade.	< <a href="https://turismo.joaopessoa.pb.gov.br/">https://turismo.joaopessoa.pb.gov.br/</a> >
Sonhar e Morar MRV	Bairros de João Pessoa: onde qualidade de vida e beleza se encontram	Informações sobre a cidade e seus bairros.	< <a href="https://sonharemorar.mrv.com.br/">https://sonharemorar.mrv.com.br/</a> >
TWS Empreendimentos	Onde Morar em João Pessoa: Melhores Bairros para Investir e Viver	Informações sobre bairros com boa qualidade de vida.	< <a href="https://tws-empresendimentos.com.br/">https://tws-empresendimentos.com.br/</a> >
Viva Real	Bairros de João Pessoa: conheça as melhores opções para morar	Informações sobre a cidade e alguns de seus bairros.	< <a href="https://www.vivareal.com.br/">https://www.vivareal.com.br/</a> >
Portal Zuk	João Pessoa: Confira os 10 melhores bairros para morar	Informações sobre bairros da cidade.	< <a href="https://www.portalzuk.com.br/">https://www.portalzuk.com.br/</a> >
Top 10 da Construção Brasil	Descubra os Melhores Bairros Para Morar em João Pessoa – Qualidade de Vida à Sua Porta	Informações sobre bairros da cidade.	< <a href="https://top10daconstrucaobrasil.com.br/">https://top10daconstrucaobrasil.com.br/</a> >
Massai	Descubra quais os melhores bairros para morar na grande João Pessoa	Apresenta informações sobre bairros da cidade.	< <a href="https://massai.com.br/">https://massai.com.br/</a> >

---

Continua na próxima página

---

**Continuação da Tabela – Sites utilizados como base de conhecimento do agente**

---

<b>Nome do Site</b>	<b>Nome do Artigo, Reportagem ou Postagem</b>	<b>Informações ou Retiradas</b>	<b>Domínio</b>
Jornal da Paraíba	Vale a pena morar em João Pessoa? Veja principais indicadores	Reportagem contendo indicado- res sobre a cidade.	< <a href="https://jornaldaparaiba.com.br/">https://jornaldaparaiba.com.br/</a> >

**Fonte: Elaborado pelo autor.**

---

A tabela a seguir contém os links de artigos, reportagens ou postagens da seção anterior de forma mais detalhada.

**Tabela 12: Notícias, artigos e reportagens utilizados na base de conhecimento do agente, com suas respectivas URLs**

<b>Autor</b>	<b>Data de publicação</b>	<b>URL</b>
FILIPEIA/SEPLAN/PMJP	s.d.	< <a href="https://filipeia.joaopessoa.pb.gov.br/">https://filipeia.joaopessoa.pb.gov.br/</a> >
FILIPEIA/SEPLAN/PMJP	s.d.	< <a href="https://filipeia.joaopessoa.pb.gov.br/files/atlas/Perfil_de_bairro.pdf/">https://filipeia.joaopessoa.pb.gov.br/files/atlas/Perfil_de_bairro.pdf/</a> >
G1 PB	6 out. 2020	< <a href="https://g1.globo.com/pb/paraiba/eleicoes/2020/noticia/2020/10/06/pesquisa-ibope-saude-e-principal-problema-de-joao-pessoa.ghtml">https://g1.globo.com/pb/paraiba/eleicoes/2020/noticia/2020/10/06/pesquisa-ibope-saude-e-principal-problema-de-joao-pessoa.ghtml</a> >
G1 PB	5 ago. 2023	< <a href="https://g1.globo.com/pb/paraiba/noticia/2023/08/05/turismo-de-aventura-em-joao-pessoa-trilhas-esportes-aquaticos-passeios-de-barco-e-mergulho-sao-atracoes.ghtml">https://g1.globo.com/pb/paraiba/noticia/2023/08/05/turismo-de-aventura-em-joao-pessoa-trilhas-esportes-aquaticos-passeios-de-barco-e-mergulho-sao-atracoes.ghtml</a> >
G1 PB	14 nov. 2024	< <a href="https://g1.globo.com/pb/paraiba/noticia/2024/11/14/mangabeira-permanece-como-o-bairro-mais-populoso-da-paraiba-gramame-assume-2o-lugar.ghtml">https://g1.globo.com/pb/paraiba/noticia/2024/11/14/mangabeira-permanece-como-o-bairro-mais-populoso-da-paraiba-gramame-assume-2o-lugar.ghtml</a> >

*Continua na próxima página*

Continuação da Tabela – Referências

Autor	Data de publicação	URL
G1 PB	24 nov. 2024	< <a href="https://g1.globo.com/pb/paraiba/noticia/2024/11/24/conheca-os-principais-pontos-turisticos-de-joao-pessoa-destino-em-alta-para-2025.ghtml">https://g1.globo.com/pb/paraiba/noticia/2024/11/24/conheca-os-principais-pontos-turisticos-de-joao-pessoa-destino-em-alta-para-2025.ghtml</a> >
JORNAL DA PARAÍBA	5 ago. 2023	< <a href="https://jornaldaparaiba.com.br/economia/vale-a-pena-se-mudar-para-joao-pessoa">https://jornaldaparaiba.com.br/economia/vale-a-pena-se-mudar-para-joao-pessoa</a> >
MASSAI CONSTRUÇÕES INCORPORAÇÕES E PARTICIPAÇÕES LTDA	13 out. 2023	< <a href="https://massai.com.br/noticias/descubra-que-qual-eh-os-melhores-bairros-para-morar-na-grande-joao-pessoa">https://massai.com.br/noticias/descubra-que-qual-eh-os-melhores-bairros-para-morar-na-grande-joao-pessoa</a> >
PORTAL ZUK	s.d.	< <a href="https://www.portalzuk.com.br/blog/joao-pessoa-confira-os-10-melhores-bairros-para-morar">https://www.portalzuk.com.br/blog/joao-pessoa-confira-os-10-melhores-bairros-para-morar</a> >
PREFEITURA MUNICIPAL DE JOÃO PESSOA	s.d.	< <a href="https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/pracas-e-parques/">https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/pracas-e-parques/</a> >
PREFEITURA MUNICIPAL DE JOÃO PESSOA	s.d.	< <a href="https://turismo.joaopessoa.pb.gov.br/um-pouco-da-sua-historia/">https://turismo.joaopessoa.pb.gov.br/um-pouco-da-sua-historia/</a> >
PREFEITURA MUNICIPAL DE JOÃO PESSOA	s.d.	< <a href="https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/artesanato/">https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/artesanato/</a> >
PREFEITURA MUNICIPAL DE JOÃO PESSOA	s.d.	< <a href="https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/monumentos-historicos/">https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/monumentos-historicos/</a> >
PREFEITURA MUNICIPAL DE JOÃO PESSOA	s.d.	< <a href="https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/praias-e-piscinas-naturais/">https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/praias-e-piscinas-naturais/</a> >
PREFEITURA MUNICIPAL DE JOÃO PESSOA	s.d.	< <a href="https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/igrejas/">https://turismo.joaopessoa.pb.gov.br/o-que-fazer/pontos-turisticos/igrejas/</a> >

*Continua na próxima página*

---

### Continuação da Tabela – Referências

---

Autor	Data de publicação	URL
SILVA, Luana	5 dez. 2024	< <a href="https://g1.globo.com/pb/paraiba/noticia/2024/12/05/joao-pessoa-capital-com-maior-desigualdade-de-renda-do-pais.ghml">https://g1.globo.com/pb/paraiba/noticia/2024/12/05/joao-pessoa-capital-com-maior-desigualdade-de-renda-do-pais.ghml</a> >
SONHAR E MORAR	15 fev. 2024	< <a href="https://sonharemorar.mrv.com.br/bairros-de-joao-pessoa-onde-qualidade-de-vida-e-beleza-se-encontram/">https://sonharemorar.mrv.com.br/bairros-de-joao-pessoa-onde-qualidade-de-vida-e-beleza-se-encontram/</a> >
TOP 10 DA CONSTRUÇÃO	3 nov. 2023	< <a href="https://top10daconstrucaobrasil.com.br/os-melhores-bairros-para-morar-em-joao-pessoa/">https://top10daconstrucaobrasil.com.br/os-melhores-bairros-para-morar-em-joao-pessoa/</a> >
TWS EMPREENDIMENTOS	s.d.	< <a href="https://tws-empresendimentos.com.br/onde-morar-em-joao-pessoa/">https://tws-empresendimentos.com.br/onde-morar-em-joao-pessoa/</a> >
UMBERTO SANTOS IMÓVEIS	18 maio 2024	< <a href="https://jampaimoveis.com/melhores-bairros-para-morar-em-joao-pessoa-conheca-o-ranking/">https://jampaimoveis.com/melhores-bairros-para-morar-em-joao-pessoa-conheca-o-ranking/</a> >
VIVA REAL	6 mar. 2024	< <a href="https://www.vivareal.com.br/blog/cidades/bairros-de-joao-pessoa/">https://www.vivareal.com.br/blog/cidades/bairros-de-joao-pessoa/</a> >

---

### Perguntas feitas ao agente para testes

A lista a seguir contém parte das perguntas feitas ao agente para testes.

- Fale sobre a história de João Pessoa
- Fale sobre eventos comuns em João Pessoa
- Fale sobre passeios turísticos em João Pessoa
- Fale sobre os melhores bairros de João Pessoa para se turistar
- Fale sobre os melhores bairros de João Pessoa para se morar
- Fale sobre os bairros mais valorizados de João Pessoa
- Fale sobre bairros mais baratos de se viver em João Pessoa
- Fale sobre bairros centrais em João Pessoa
- Fale sobre bairros próximos à praia em João Pessoa

- Fale sobre o transporte público em João Pessoa
- Fale sobre o sistema de saúde de João Pessoa
- Fale sobre a situação da segurança em João Pessoa
- Como é o custo de vida em João Pessoa, comparado a outras capitais
- Como é o clima ao longo do ano em João Pessoa

**Tabela 13: Formato de perguntas feitas ao agente**

<b>Formato da pergunta</b>	<b>Variável da pergunta</b>
Como é morar no bairro x1	"x1" corresponde a um nome de bairro da cidade
Me fale sobre x2 no bairro y2	"x2" corresponde a um tipo de localidade da lista de localidades, e "y2" aos bairros da lista de Bancários e Centro.
Me sugira os(as) melhores x3 em João Pessoa	"x3" correspondente a um tipo de localidade da lista de localidades citadas.
Como posso ir do bairro x4 ao bairro y4 utilizando ônibus?	"x4" e "y4" correspondem a nomes de bairros escolhidos de forma arbitrária.
Quais linhas de ônibus passam no bairro x5?	"x5" corresponde a nomes de bairros da lista de Bessa, Centro, Bancários, Gramame e Mumbaba.
Por onde passa a linha x6?	"x6" corresponde a linhas de ônibus escolhidas de forma arbitrária. A lista é 002, 101, Tambaú, Circular e T001.
Me fale sobre x7 para y7 no bairro z7?	"x7" correspondete a "casas" e "apartamentos", "y7" corresponde a "comprar" e "alugar" e "z7" corresponde a um nome de bairro dentre a lista Bancários, Centro, Bessa, Gramame ou Mangabeira.

#### **Lista de ruas fictícias para geração de imóveis irreais**

- Rua das Acácias
- Avenida dos Flamboyants
- Travessa dos Jasmins
- Alameda dos Pinheiros

- Rua dos Jacarandás
- Avenida das Camélias
- Travessa das Magnólias
- Alameda dos Eucaliptos
- Rua dos Cedros
- Avenida das Palmeiras

#### **Lista de pontos de referência para geração de imóveis irreais**

- Próximo a Supermercado
- Próximo a Praça com Parque
- Próximo a Igreja
- Próximo a Parada de Ônibus
- Próximo a Escola

#### **Algoritmo para geração de imóveis irreais e seus preços**

---

**Algoritmo 1:** Cálculo de preço de imóvel com base em características do bairro e da moradia

---

**Entrada:** Bairro, tipo de moradia, tipo de compra, renda do bairro, número de quartos

**Resultado:** Preço final estimado do imóvel ou aluguel

```
1 precoBase ← 120000 ; // Valor arbitrário para compra
2 multiplicador ← 1 ; // Fator de ajuste do preço
  /* // */
  /* Ajuste conforme o grupo de renda do bairro */
3 if bairro ∈ grupo1 then
4 | multiplicador ← multiplicador × 1.8 ; // Aumento de 80%
5 else
6 end
7 if bairro ∈ grupo2 then
8 | multiplicador ← multiplicador × 1.4 ; // Aumento de 40%
9 else
10 end
  /* Ajuste conforme o tipo de moradia */
11 if tipoMoradia = "Apartamento" then
12 | precoBase ← precoBase × 0.8 ; // Reduz valor base
13 else
14 | precoBase ← precoBase × 2 ; // Aumenta valor base
15 end
  /* // */
  /* Ajuste conforme o tipo de compra */
16 if tipoCompra = "Aluguel" then
17 | precoBase ← precoBase/100 ; // Aluguel é fração do preço
18 else
19 | precoBase ← precoBase × 2 ; // Valor final de compra
20 end
  /* // */
  /* Ajuste conforme o número de quartos */
21 if nQuartos = 1 then
22 | multiplicador ← multiplicador × 0.8
23 else
24 | if nQuartos = 2 then
25 | | multiplicador ← multiplicador × 1.0
26 | end
27 | if nQuartos = 3 then
28 | | multiplicador ← multiplicador × 1.2
29 | end
30 end
31 precoFinal ← Arredondar(precoBase × multiplicador, 2);
32 retorna precoFinal
```

---