Chatbot Multimodal com LLMs para Atendimento de Ocorrências em Serviços Públicos de Segurança

Luis Henrique Miranda Queiroz[1], **Raimundo Santos Moura**[1]

[1]Universidade Federal do Piauí (UFPI) – Departamento de Computação

{luis.hmq, rsm}@ufpi.edu.br

Abstract. *This paper presents the development of an intelligent conversational agent, based on large-scale language models (LLMs), to assist in the registration of police reports and emergency services through digital platforms such as WhatsApp. The prototype allows multimodal communication (text, audio, image and location), automatically classifying occurrences, identifying gaps in reports and performing validations via external services. The solution uses technologies such as FastAPI, LangChain, Google Cloud Platform and Twilio, and is integrated with transcription, computer vision and data verification tools. The system is already functional in a controlled environment, with initial tests showing positive results. This work is part of the intersection between Artificial Intelligence, Human-Computer Interaction and Public Informatics, promoting accessibility, agility and security in communication between citizens and public security services.*

Resumo. *Este artigo apresenta o desenvolvimento de um agente conversacional inteligente, baseado em modelos de linguagem de grande escala (LLMs), para auxiliar no registro de boletins de ocorrência policiais e atendimento de urgência por meio de plataformas digitais como o WhatsApp. O protótipo permite comunicação multimodal (texto, áudio, imagem e localização), classificando automaticamente ocorrências, identificando lacunas nas denúncias e realizando validações via serviços externos. A solução utiliza tecnologias como FastAPI, LangChain, Google Cloud Platform e Twilio, sendo integrada com ferramentas de transcrição, visão computacional e verificação de dados. O sistema já encontra-se funcional em ambiente controlado, com testes iniciais mostrando resultados positivos. Este trabalho se insere na interseção entre Inteligência Artificial, Interação Humano-Computador e Informática Pública, promovendo acessibilidade, agilidade e segurança na comunicação entre cidadão e serviços públicos de segurança.*

1. Introdução

A eficiência e a qualidade do atendimento na segurança pública brasileira ainda esbarram em limitações humanas, sobrecarga de chamadas e interfaces digitais pouco flexíveis. Sistemas convencionais — o telefone 190, delegacias físicas e a Delegacia Virtual — raramente validam dados em tempo real, classificam mal ocorrências e exigem longos tempos de espera. Frente a esse cenário, modelos de linguagem de grande escala (LLMs) associados a técnicas de Processamento de Linguagem Natural oferecem um meio de automatizar triagens, interpretar linguagem espontânea e operar 24 h × 7.

Este artigo descreve o desenvolvimento de um chatbot multimodal para a Secretaria de Segurança Pública do Piauí. Integrado ao WhatsApp, ele processa texto, áudio, imagens e localização; valida CPF instantaneamente; consulta serviços externos (por exemplo, BigQuery, GIS) e encaminha registros para as unidades competentes. A solução combina LLMs, infraestrutura em nuvem e APIs de mensageria para criar uma interface acessível, rastreável e ágil.

O restante do texto está estruturado da seguinte forma: Seção 2 apresenta os fundamentos teóricos; Seção 3, os trabalhos relacionados; Seção 4 detalha a metodologia; Seção 5 descreve a arquitetura proposta; Seção 6 traz resultados preliminares; Seção 7 discute análise visual e testes; Seção 8 conclui e aponta trabalhos futuros; Seção 9 lista as referências.

2. Fundamentação Teórica

O Processamento de Linguagem Natural (PLN) é uma subárea da Inteligência Artificial (IA) que estuda a interação entre computadores e linguagem humana. Seu objetivo é permitir que sistemas computacionais compreendam, interpretem e gerem textos de maneira semelhante à comunicação humana. Essa área tem sido amplamente aplicada em tarefas como tradução automática, análise de sentimentos, classificação de textos, reconhecimento de fala e, mais recentemente, na construção de agentes conversacionais com capacidade de raciocínio complexo.

Com o avanço dos Modelos de Linguagem de Grande Escala (LLMs – Large Language Models), como o *GPT*, *Claude* e *DeepSeek*, tornou-se possível gerar textos altamente coerentes e contextuais, abrindo caminho para aplicações mais sofisticadas, como assistentes virtuais e chatbots inteligentes. Esses modelos são treinados com grandes volumes de dados textuais e são capazes de realizar inferências complexas, responder perguntas, sintetizar informações e interagir de maneira fluida com usuários humanos.

Nesse contexto, surgem os agentes baseados em LLMs, que combinam os modelos de linguagem com estruturas de decisão, ferramentas externas e fluxos de diálogo. Esses agentes são capazes de decompor problemas em etapas, consultar APIs externas, tomar decisões condicionais e adaptar respostas com base em histórico de interação. Uma arquitetura comum envolve o uso de frameworks como o *LangChain*, que permite a criação de agentes com "tools", ou ferramentas, capazes de executar tarefas como busca de dados, verificação de identidade e análise de conteúdo multimodal. Cada ferramenta é acionada por meio de uma técnica chamada de *function calling*, um mecanismo que permite que o modelo de linguagem invoque dinamicamente funções externas com base nas intenções do usuário.

3. Trabalhos Relacionados

A literatura recente mostra um movimento consistente rumo ao uso de LLMs para modernizar serviços públicos. (Dam et al. 2024) traçam essa trajetória e ressaltam duas dificuldades ainda em aberto — manter o contexto em diálogos extensos e exercer controle fino sobre o conteúdo gerado —, mas concluem que a capacidade dos modelos de interpretar instruções em linguagem natural os torna particularmente atraentes para cenários de atendimento de massa. Diversos componentes tecnológicos vêm sendo combinados a esses

modelos. No backend, o framework *FastAPI* possibilita a construção de APIs REST leves, enquanto a integração com o WhatsApp ocorre via Twilio, permitindo o recebimento de texto, áudio, imagem e localização em um canal popular. Para suportar mídias ricas, recorre-se ao *Google Speech-to-Text* na transcrição de mensagens sonoras e ao Gemini Vision, do Vertex AI, na análise de imagens. Uma API estadual específica valida CPF em tempo real, reforçando a autenticidade das denúncias, ao passo que o BigQuery armazena o histórico de conversas para fins de auditoria e análise estatística; o Redis mantém o contexto entre turnos, assegurando continuidade.

A qualidade percebida pelo usuário continua sendo fator crítico. Estudo de (Oliveira 2024) mostra que clareza, coerência e tempo de resposta impactam diretamente a satisfação; por isso, o presente sistema incorpora a Escala de Likert como métrica formal de avaliação. Em termos de infraestrutura, trabalhos como (Gunnam et al. 2024) defendem serviços serverless pela elasticidade e segurança oferecidas; em consonância, a nossa aplicação reside no Google Cloud Run, que dimensiona recursos de forma automática. Para orquestrar múltiplos agentes especializados, investigamos ainda o LangGraph, biblioteca que permite compor fluxos paralelos de conversação. A síntese desses avanços — LLMs, microserviços em nuvem, validação automática de dados e métricas de usabilidade — fundamenta a solução proposta neste artigo, voltada a um registro de ocorrências emergenciais ágil, seguro e acessível.

4. Metodologia

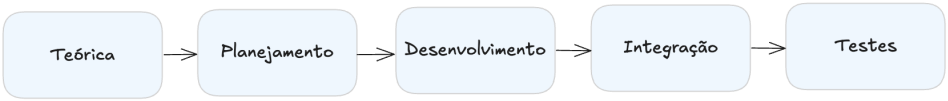

Figura 1. Metodologia Geral. Fonte: Autor

A execução do projeto foi dividida em cinco fases principais. A primeira delas, chamada Fase Teórica, consistiu em um levantamento detalhado das bibliotecas e ferramentas utilizadas em Processamento de Linguagem Natural (PLN) e no desenvolvimento de agentes baseados em LLMs. Foram exploradas bibliotecas como *NLTK*, *spaCy*, *Scikit-Learn* e frameworks como *LangChain*, além de APIs para transcrição de áudio, como *Google Cloud Speech-to-Text* e *IBM Watson*. Também se analisaram aspectos legais relacionados ao uso de dados sensíveis, uma vez que o sistema lida com informações pessoais como CPF e localização.

Na Fase de Planejamento, foram definidos os requisitos técnicos e funcionais do sistema, além do fluxo de interação entre o cidadão e o agente. Também foi estabelecida a estrutura das rotas da API, os modelos de validação e o controle de sessões para garantir a consistência da conversa. Para avaliar a experiência do usuário, optou-se pela adoção da Escala de Likert, que permitirá análises qualitativas e quantitativas nos testes com usuários.

A Fase de Desenvolvimento concentrou-se na implementação efetiva do sistema. As principais tecnologias utilizadas foram selecionadas com base em critérios de desem-

penho, escalabilidade, integração com serviços de terceiros e suporte à multimodalidade. A seguir, detalham-se os critérios de escolha de cada tecnologia:

- **FastAPI**: escolhida para construção do backend devido à sua alta performance, tipagem forte via Pydantic, suporte nativo à documentação automática (OpenAPI/Swagger) e excelente integração com aplicações assíncronas, superando alternativas como Flask em produtividade e escalabilidade.
- **Twilio**: adotado como gateway para comunicação via WhatsApp por oferecer suporte robusto e oficial à API do WhatsApp Business, documentação clara e ampla compatibilidade com webhooks. Foi preferido em relação a Zenvia e Gupshup por sua confiabilidade em ambientes de produção e integração facilitada com serviços em nuvem.
- **Google Cloud Speech-to-Text**: utilizado para transcrição de áudio em tempo real, foi escolhido por sua acurácia em português brasileiro, suporte a sotaques regionais e facilidade de integração com outros serviços da Google Cloud Platform.
- **LangChain**: selecionado para orquestrar os agentes LLM e ferramentas externas, devido à sua arquitetura modular que facilita a implementação de agentes personalizados, suporte a múltiplos modelos de linguagem e facilidade de integração com APIs externas via "tools".
- **OpenAI GPT, Claude (Anthropic), DeepSeek, Gemini (Google) e Sabiá (Maritaca AI)**: utilizados como modelos de linguagem centrais, cada um foi testado em diferentes etapas com o objetivo de avaliar variações de custo, latência, domínio e qualidade de resposta. O uso de múltiplos LLMs aumentou a flexibilidade e robustez do sistema, além de permitir testes com modelos open-source (como Sabiá) desenvolvidos para o contexto brasileiro.
- **BigQuery**: plataforma de armazenamento e análise de dados, escolhida por sua capacidade de processar grandes volumes de dados estruturados, integração com GCP e suporte nativo a SQL para consultas analíticas.
- **Redis**: utilizado para gerenciamento de estado das conversas e cache de sessões, foi escolhido por sua baixa latência e capacidade de operar como banco de dados em memória altamente eficiente, essencial para manter o contexto da interação com o agente de IA.
- **LangSmith**: adotado como ferramenta de monitoramento e logging das execuções do agente, permitindo rastreamento detalhado das conversas, análise de falhas e avaliação de desempenho dos modelos de linguagem ao longo do tempo.

Em seguida, na Fase de Integração, todas as funcionalidades foram unificadas em um protótipo funcional hospedado no serviço serverless *Cloud Run* da *Google Cloud Platform*, o que garantiu escalabilidade automática, gerenciamento de containers e alta disponibilidade. A arquitetura foi projetada de maneira modular, de forma que cada componente do sistema possa ser atualizado ou substituído de forma independente, promovendo flexibilidade e manutenção facilitada.

Por fim, na Fase Final, que se encontra em andamento, foram conduzidos testes locais em ambiente controlado. A etapa de testes em larga escala ainda será realizada, juntamente com a aplicação da Escala de Likert para avaliar a experiência dos usuários em aspectos como clareza das respostas, tempo de atendimento e satisfação geral. Os dados coletados também servirão como base para ajustes nos modelos de linguagem, parâmetros da infraestrutura e melhorias na usabilidade do sistema.

5. Solução Proposta

Nesta seção, apresenta-se a solução proposta para o atendimento automatizado de ocorrências em serviços públicos de segurança. O sistema foi concebido com foco em escalabilidade, segurança, integração com múltiplas fontes de entrada de dados e orquestração inteligente de respostas utilizando agentes baseados em LLMs. A seguir, descreve-se a arquitetura geral da solução e suas principais camadas tecnológicas.

5.1. Interação com API e Agente de IA

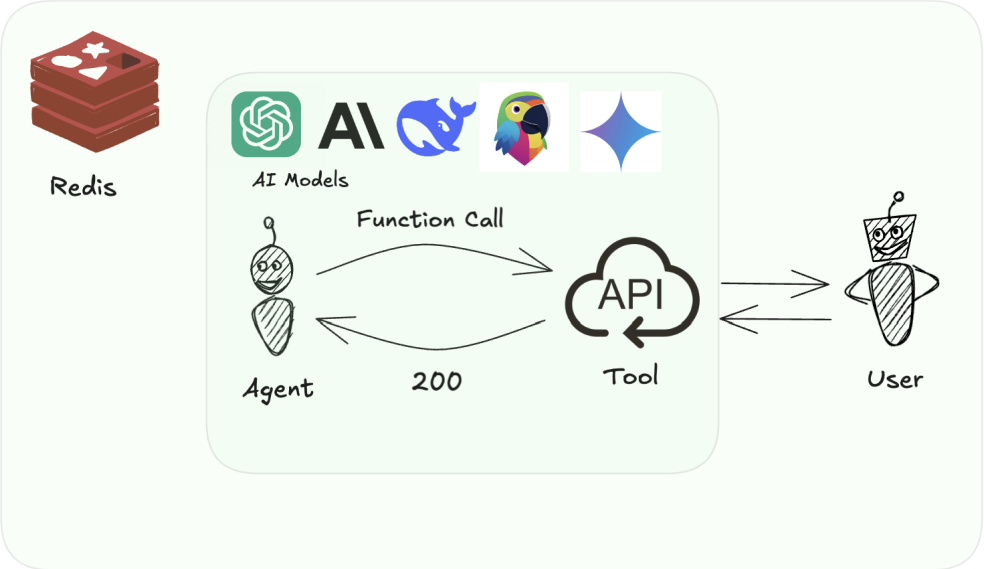

Figura 2. Arquitetura geral do Agente de Inteligência Artificial Generativa. Fonte: Autor

A Figura 2 representa o ecossistema interno voltado à execução das funções do agente de IA, integrando modelos de linguagem, ferramentas externas e um sistema de persistência com Redis. No centro do sistema está o agente baseado em LLMs, utilizando tecnologias como *OpenAI GPT*, *Claude* (Anthropic), *DeepSeek*, *Gemini* (Google) e *Sabiá* (Maritaca AI). Esse agente é gerenciado por um orquestrador (*LangChain*) que realiza *function calls* com as ferramentas.

As ferramentas acionadas incluem sistemas de validação de dados (ex: validação de CPF), módulos de classificação e enriquecimento de dados. Toda interação passa por um processo de interpretação e construção de resposta textual ou multimodal, com suporte de um mecanismo de memória temporária via Redis.

O desempenho do sistema é continuamente monitorado por ferramentas como *LangSmith*, que permitem rastrear logs, latência, falhas e métricas de interação.

5.2. Integração com Google Cloud, Twilio e Multimodalidade

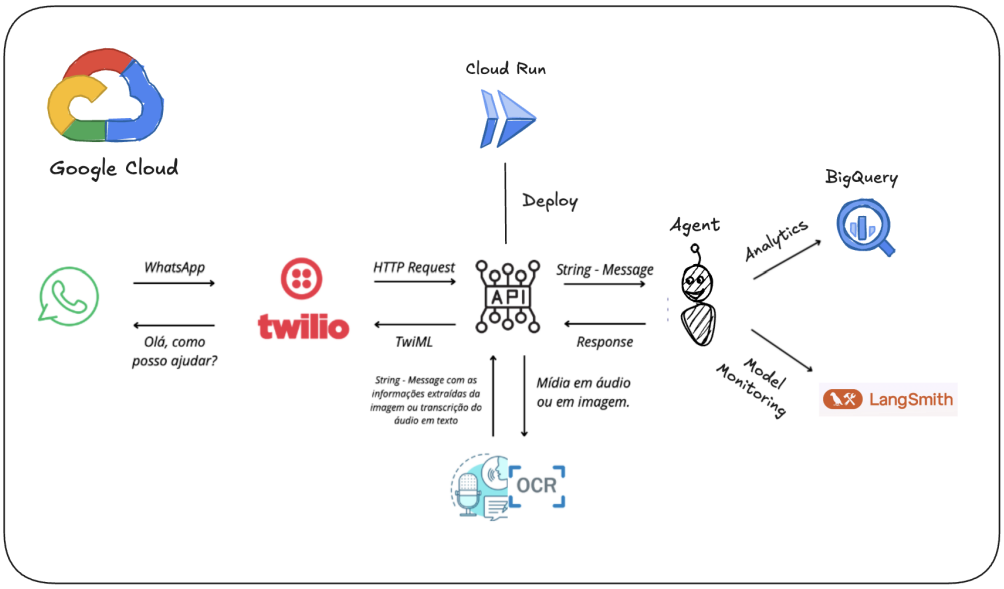

Figura 3. Arquitetura geral do sistema integrando *Twilio*, API e *Google Cloud*. Fonte: Autor

A Figura 3 mostra a arquitetura de ponta a ponta, desde o recebimento das mensagens via WhatsApp até o retorno da resposta ao cidadão. O fluxo inicia com a entrada da mensagem via WhatsApp, que é repassada pelo Twilio para a API desenvolvida em *FastAPI*, hospedada no *Google Cloud Run*.

As mensagens recebidas passam por uma etapa de classificação conforme seu tipo: se texto, seguem diretamente para o agente; se áudio, são transcritas via *Google Cloud Speech-to-Text*; se imagem, são analisadas por *Gemini Vision*. Todas as mensagens são tratadas de forma unificada como entrada textual para posterior classificação e resposta pelo agente LLM.

Além de emitir respostas ao usuário, o sistema também realiza: Armazenamento das conversas no *BigQuery*, Registro de métricas de desempenho e logs no *LangSmith*, Validação de informações pessoais como CPF por meio de API estadual, Gerenciamento de contexto de conversa por meio do *Redis*.

A arquitetura modular permite substituição de componentes, atualizações independentes e escalabilidade. Essa abordagem garante não apenas robustez e resiliência, mas também conformidade com a LGPD e segurança dos dados sensíveis tratados no atendimento digitalizado de ocorrências.

6. Resultados Preliminares

Até o momento, o projeto obteve progressos relevantes na implementação do chatbot, englobando aspectos como a integração com o WhatsApp via *Twilio*, que possibilita a comunicação multimodal (texto, áudio e imagens) e, consequentemente, facilita a interação com o usuário. Além disso, a classificação automática de ocorrências está em operação,

permitindo que o sistema identifique e categorize diferentes tipos de incidentes com base no detalhamento das informações fornecidas. Outro avanço notável é a inclusão de um módulo dedicado ao atendimento de urgência (190), que prioriza ocorrências críticas e agiliza o encaminhamento das demandas mais graves. A verificação de dados, por sua vez, ganhou robustez com a utilização da API responsável por validar o CPF do cidadão, o que contribui para a redução de fraudes e inconsistências.

Em relação ao estado de testes, observa-se que o protótipo já se encontra funcional, embora ainda não tenham sido realizados ensaios em larga escala com grande número de usuários; nos testes restritos em ambiente controlado, os resultados têm sido positivos, mas questões de escalabilidade e robustez precisarão ser avaliadas em uma fase posterior.

6.1. Limitações Identificadas

Apesar dos avanços, o sistema ainda apresenta algumas limitações importantes. Uma delas está relacionada ao tempo de processamento de mensagens de áudio. A plataforma *Twilio*, utilizada como gateway para o WhatsApp, impõe um limite de timeout para a entrega e manipulação de mídias, o que inviabiliza a transcrição de áudios muito longos (acima de 75 segundos, por exemplo). Esse limite técnico pode comprometer a experiência do usuário em casos em que ele deseja relatar um incidente detalhado por meio de uma única mensagem extensa.

Além disso, a transcrição automática de áudio, embora eficiente na maioria dos testes, apresenta variações de desempenho em situações com ruído de fundo, sotaques regionais acentuados ou vocabulário informal. Isso pode impactar a compreensão correta da ocorrência e a tomada de decisão automatizada. Da mesma forma, o módulo de visão computacional para imagens ainda depende da clareza e qualidade da mídia enviada. Imagens desfocadas, escuras ou com baixa resolução comprometem a extração de informações úteis.

Outro ponto crítico está na dependência de serviços externos para validação de dados (como a API de CPF). Falhas de disponibilidade, lentidão ou instabilidade desses serviços impactam diretamente o fluxo de atendimento, podendo gerar atrasos ou bloqueios temporários no processo de registro de ocorrência. Além disso, nem todos os CPFs consultados retornam como válidos, especialmente em casos de cidadãos não cadastrados ou com inconsistências nos bancos de dados estaduais. Isso pode gerar frustração no usuário e necessidade de atendimento manual ou verificação alternativa por operadores humanos.

Uma limitação estratégica observada é a ausência de uma opção clara para transferência do atendimento para operadores humanos. Em contextos críticos ou quando o sistema não é capaz de resolver adequadamente a solicitação do usuário, a ausência de um handover para atendimento humano pode comprometer a eficácia e a confiança na solução. Estudos apontam que sistemas híbridos, que combinam automação com suporte humano, oferecem maior satisfação ao usuário e garantem respostas mais empáticas e contextualizadas.

Por fim, é importante destacar questões relacionadas à privacidade e à ética. Como o sistema trata dados pessoais sensíveis, como nome, CPF, endereço e arquivos de mídia, foi necessário garantir o cumprimento da Lei Geral de Proteção de Dados (LGPD). O sistema adota práticas como: Transmissão criptografada de dados, Retenção mínima das informações, com descarte automático de sessões expiradas, Uso de armazenamento se-

guro (Google Cloud e BigQuery com controles de acesso), Consentimento do usuário no início da interação.

No entanto, mesmo com essas precauções, o uso de modelos de linguagem de grande escala (LLMs) ainda apresenta riscos relacionados a vieses algorítmicos. Esses modelos são treinados com grandes volumes de dados coletados da internet e, por isso, podem refletir estereótipos, preconceitos e inferências imprecisas. Isso pode afetar o atendimento, especialmente em casos sensíveis ou que envolvam linguagem ambígua. Como medida de mitigação, o sistema inclui filtros semânticos e logs de auditoria que permitem revisar as decisões automatizadas e ajustar o comportamento do agente quando necessário.

Essas limitações estão sendo mapeadas e analisadas com o objetivo de orientar futuras melhorias. Possíveis soluções incluem o fracionamento de áudios longos, mecanismos de confirmação para transcrições ambíguas, otimização da infraestrutura de integração com APIs externas, estratégias de fallback para ausência de cadastro de CPF, inclusão de um fluxo de escalonamento para operadores humanos, e o refinamento contínuo dos modelos com base em análises de qualidade e diversidade de atendimento.

7. Análise Visual e Testes Iniciais

Realizou-se um ensaio controlado com o chatbot integrado ao WhatsApp/Twilio. Avaliou-se: (i) recepção de localização, (ii) transcrição de áudio, (iii) validação de CPF em tempo real, (iv) classificação da ocorrência e (v) envio de instruções de segurança.

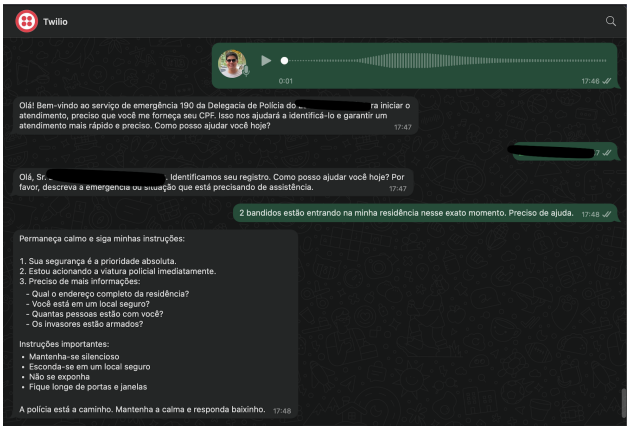

Figura 4. Solicitação de CPF e descrição da invasão.

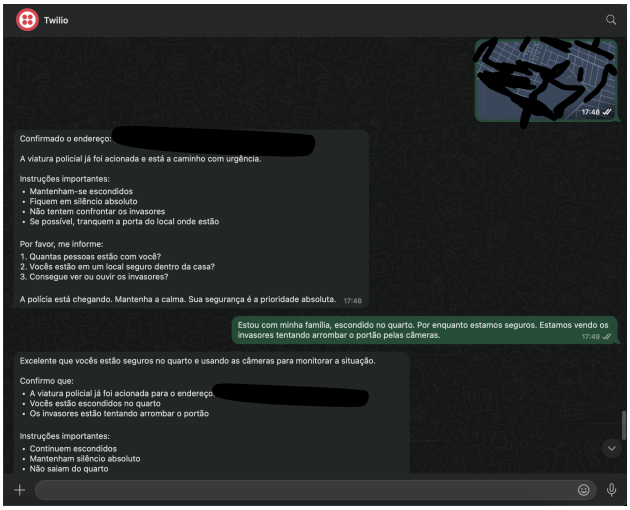

Figura 5. Confirmação automática de endereço e orientações.

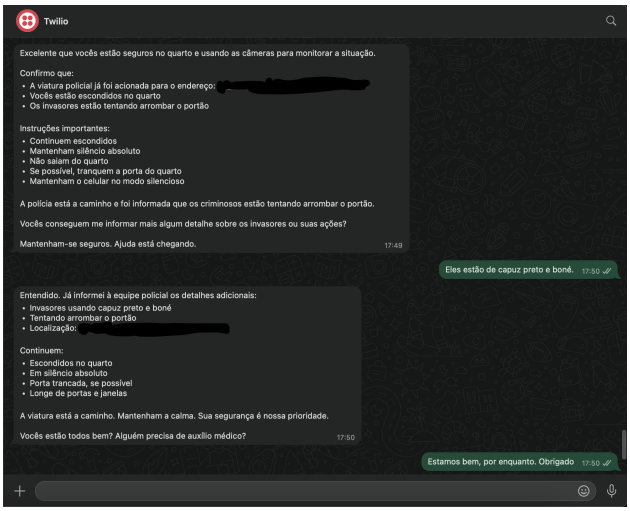

Figura 6. Atualizações do usuário e monitoramento contínuo.

As Figuras 4–6 ilustram o fluxo completo: identificação, coleta de dados adicionais e acompanhamento até o despacho da viatura. O sistema respondeu em tempo real, manteve coerência e validou todos os campos obrigatórios.

8. Conclusão e Trabalhos Futuros

Os resultados obtidos até aqui indicam que o uso de chatbots baseados em LLMs, aliados a tecnologias de verificação de dados, pode otimizar significativamente tanto o registro de boletins de ocorrência quanto o atendimento de urgência (190), promovendo mais agilidade e precisão nos serviços de segurança pública do estado Piauí. O protótipo funcional desenvolvido representa um avanço importante na modernização dos processos de atendimento, embora sua consolidação final ainda dependa de testes em larga escala.

Como próximos passos, estão previstos testes com novos modelos de linguagem, com o objetivo de comparar desempenho, custo e capacidade de contextualização. Além

disso, a infraestrutura atual será aprimorada para reduzir a latência e garantir maior escalabilidade do sistema, especialmente em cenários de alta demanda. Pretende-se também adotar a biblioteca LangGraph, o que permitirá a orquestração de múltiplos agentes colaborativos, possibilitando fluxos de atendimento mais complexos e especializados. Paralelamente, serão feitas melhorias na robustez dos módulos de reconhecimento de fala e processamento de imagens, com foco na adaptação a sotaques regionais, ruído de fundo e variedade de qualidade de mídia.

Por fim, almeja-se concluir a integração completa com os sistemas da SSP/PI, observando rigorosamente os protocolos de segurança e privacidade. Com isso, espera-se oferecer uma solução escalável, confiável e eficaz, beneficiando tanto os profissionais da segurança pública quanto os cidadãos, ao proporcionar uma experiência de atendimento mais acessível, fluida e precisa.

Agradecimentos

O presente trabalho foi financiado parcialmente pela Secretaria de Segurança Pública do Estado do Piauí (SSP/PI), com apoio da Fundação Cultural e de Fomento à Pesquisa, Ensino, Extensão e Inovação (FADEX).

Agradecemos também à equipe técnica da SSP/PI — Venceslau Felipe, Joaquim Carvalho e os demais colegas — pelo apoio e disponibilidade constante em conversar com os pesquisadores do Departamento de Computação da UFPI.

Referências

[Dam et al. 2024] Dam, S. K., Hong, C. S., Qiao, Y., and Zhang, C. (2024). A complete survey on llm-based ai chatbots. *arXiv preprint*.

[Gunnam et al. 2024] Gunnam, G. R., Inupakutika, D., Mundlamuri, R., Kaghyan, S., and Akopian, D. (2024). Assessing performance of cloud-based heterogeneous chatbot systems and a case study. *IEEE Access*.

[Oliveira 2024] Oliveira, E. G. (2024). Chatbots: a importância do processamento de língua natural para a experiência do usuário. Master's thesis, Universidade Federal do Rio Grande do Norte (UFRN).