UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

LARA ESQUIVEL DE BRITO SANTOS

RECUPERAÇÃO AUMENTADA POR GERAÇÃO EM MODELOS DE LINGUAGEM DE LARGA ESCALA

FEIRA DE SANTANA
2025

LARA ESQUIVEL DE BRITO SANTOS

RECUPERAÇÃO AUMENTADA POR GERAÇÃO EM MODELOS DE LINGUAGEM DE LARGA ESCALA

Trabalho de Conclusão de Curso apresentado ao Colegiado do curso de Engenharia de Computação como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação pela Universidade Estadual de Feira de Santana.

Orientador: João Batista da Rocha Junior

FEIRA DE SANTANA
2025

Dedico todo este trabalho à minha querida mãe Tatiara, que sempre esteve ao meu lado e me apoiou em todos os meus sonhos. Aos meus professores, em especial ao meu orientador, e amigos e familiares que me guiaram e acompanharam nesta jornada.

AGRADECIMENTOS

Gostaria de expressar minha sincera gratidão a todas as pessoas que, de alguma forma, contribuíram para a realização deste trabalho. Agradeço primeiramente a minha família, em especial a minha mãe, por todo o apoio, paciência e incentivo durante essa jornada.

Agradeço especialmente ao Professor João Batista Rocha Junior, por sua orientação, disponibilidade e ensinamentos fundamentais ao longo deste processo. Sua dedicação foi essencial para que este trabalho se concretizasse.

Também estendo meus agradecimentos a todos que colaboraram direta ou indiretamente, com palavras de incentivo, apoio emocional ou mesmo com pequenos gestos que fizeram grande diferença.

Muito obrigada!

RESUMO

O trabalho apresenta uma investigação sobre a capacidade de modelos de linguagem de larga escala (MLLE), utilizando o Gemini Flash 2.0, em responder perguntas usando diferentes contextos: o contexto completo, resumido e filtrado a partir da remoção das *stopwords*. O estudo busca equilibrar qualidade e desempenho envolvendo uma análise quantitativa e qualitativa das respostas retornadas pela MLLE. Os resultados indicam que embora a MLLE apresenta bons resultados em tarefas diretas, possuem limitações quanto a precisão quantitativa e a distinção de ruídos.

Palavras-chave: Modelos de Linguagem de Larga Escala. Processamento de Linguagem Natural. Recuperação Aumenta por Geração

ABSTRACT

This work presents an investigation into the capabilities of large language models, using Gemini Flash 2.0, to answer questions based on different input contexts: the full context, a summarized version, and a filtered version obtained by removing stopwords. The study aims to balance response quality and performance through both quantitative and qualitative analyses of the answers generated by the LLMs. The results indicate that although the LLMs performs well in direct tasks, it shows limitations in terms of quantitative precision and the ability to distinguish noise.

Keywords: Large Language Models. Natural Language Process. Retrival-Augmented Generation

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Nos últimos anos, os avanços nas tecnologias de inteligência artificial (IA), especialmente no desenvolvimento de modelos pré-treinados para diferentes domínios, como imagem, texto e áudio, têm transformado diversos setores, incluindo saúde e educação. Esses modelos oferecem soluções inovadoras e suporte especializado, ampliando as possibilidades de aplicação da IA. Em particular, os progressos no Processamento de Linguagem Natural com foco no desenvolvimento de métodos e sistemas capazes de processar a linguagem humana de forma computacional (CASELI, NUNES, 2024). Esse progresso impulsionou o surgimento de modelos de linguagem de larga escala (MLLE), projetados para generalizar uma ampla gama de tarefas por meio de instruções em linguagem natural (MIZRAHI et al., 2024), como exemplificado por ferramentas como ChatGPT, Copilot, Deepseek e Gemini, entre outros.

Essas ferramentas são serviços que pegam uma cadeia de texto e retornam uma outra cadeia de texto, ou seja texto entra e sai. A entrada é chamada de *prompt* enquanto a saída, resposta ou *completion* (BERRYMAN; ZIEGLER, 2024). Todavia esses modelos pré-treinados são construídos com bilhares de parâmetros que embora possam possuir um conhecimento rico em aprendizado de texto, e beneficiar diversas tarefas, eles tem algumas limitações e desafios a serem superados: como o alto custo computacional, desafios éticos envolvendo coleta e uso de dados em treinamento, a qualidade de dados de treinamento com informações sem viés e claro a complexidade do processo de ajuste fino (RAIAAN *et al.*, 2024).

O ajuste fino é uma técnica de ajuste de modelos de aprendizado de máquina para realizar tarefas especificas, refinando suas capacidades com dados adicionais. Ao invés de treinar o modelo do zero, ajustamos os parâmetros para uma nova tarefa (MALLADI *et al.*, 2023). Por exemplo a coleção MedAlpaca, que são versões do LLaMA treinada com dados médicos para se especializar em questionamento e diálogos médicos (HAN *et al.*, 2023).

Esse processo envolve muito trabalho humano no processo de anotação de dados para a tarefa, o que demanda o custo tanto da mão de obra quanto do tempo, o que consequentemente dificulta a atualização dos modelos com dados recentes (RAIAAN *et al.*, 2024). Entre outras técnicas utilizadas há a Recuperação Aumentada por Geração, onde se recupera pedaços de informação as quais são passadas pelo *prompt* e usadas para gerar informação, sendo uma forma mais barata de atualizar modelos e eficaz para diminuir a alucinação (CHEN *et al.*, 2024).

Todavia a Recuperação Aumentada por Geração faz com que se tenha uma dependência de uma base de documentos e complexidade arquitetural para a integração desses sistemas.(CHEN *et al.*, 2024).

Diante desse cenário, surge a seguinte questão: até que ponto é possível obter respostas de qualidade apenas por meio da inserção direta de informações no *prompt*, explorando a capacidade de contexto expandido e recuperação aumentada por geração implícita, realizada internamente nos serviços que operam modelos recentes?

Ademais ao reduzir o contexto — seja por meio da remoção de stopwords ou da geração automática de amostras — é possível melhorar o tempo de resposta sem comprometer a qualidade das respostas geradas?

Este trabalho consiste na avaliação qualitativa e analítica das respostas do Gemini-Flash 2.0 quando submetido a contextos longos e amostrados de mensagens provindas do grupo de WhastApp da Câmera de Graduação da Universidade Estadual de Feira de Santana.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Avaliar as respostas de um modelo de linguagem de grande escala em diferentes representações do mesmo contexto.

1.1.2 Objetivos Específicos

- Avaliar as respostas para as diferentes representações do contexto;
- Avaliar o tempo de resposta para contextos de diferentes tamanhos e números de tokens;
- Definir um conjunto de perguntas para serem respondidas para um contexto;
- Processar contexto, para obter representações com menor tamanho e número de tokens.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capitulo são apresentados aspectos teóricos necessários para a construção desta pesquisa. Na Seção 2.1 é abordado sobre Modelos de linguagem e a evolução de abordagens estatísticas para MLLE capazes de compreender e gerar texto com alto grau de coerência, sendo fundamentais para tarefas complexas em Processamento de Linguagem Natural (PLN) (ZHAO *et al.*, 2023). Na Seção 2.2 é abordado mais sobre os Modelos de linguagem de larga escala (MLLE), explicando que são redes neurais com bilhões de parâmetros, capazes de realizar tarefas complexas de linguagem natural com alto desempenho, embora exijam grande poder computacional (RAIAAN *et al.*, 2024).

A Seção 2.3 aborda conceito gerais de Processamento de Linguagem Natural que serão essenciais para a compreensão do texto. A Seção 2.4 explica tarefas comuns ao utilizar um sistema de PLN, elicitamos: envolvendo a extração de informações, resposta automática a perguntas e geração de texto.

2.1 MODELOS DE LINGUAGEM

O termo linguagem refere-se ao uso da língua para expressar pensamentos e sentimentos por meio de palavras (FERREIRA *et al.*, 2010). Uma máquina capaz de processar a linguagem da mesma forma que os seres humanos representa um indicativo claro da verdadeira inteligência artificial (JURAFSKY; MARTIN, 2014). Quando uma máquina consegue se comunicar com eficácia na linguagem humana, demonstra a capacidade de agir de forma semelhante a um ser humano.

Um modelo de linguagem é um tipo de modelo que atribui probabilidades a palavras ou sequências de palavras. Ele usa probabilidade para estimar qual será a próxima palavra em uma sequência, com base em palavras anteriores. Esses modelos são fundamentais para a compreensão de linguagem natural, pois podem prever e gerar texto coerente ao aprender padrões linguísticos a partir de grandes volumes de dados (JURAFSKY; MARTIN, 2014).

Entre os modelos de linguagem podemos elicitar quatro principais estágios de desenvolvimento de pesquisa na área:

- Modelos de Linguagem Estatísticos;
- Modelo de Linguagem Neural (MLN);
- Modelo de Linguagem Pré-Treinado (MLP);
- Modelos de Linguagem de Larga Escala (MLLE).

Os modelos estatísticos foram bastantes utilizados até os anos 2000, a ideia era construir modelo de palavras baseados em previsão de Markov, ou seja, a previsão da próxima palavra é baseada no contexto mais recente, os quais tinham um tamanho limitado de contexto, denominado *n-grama* (ZHAO *et al.*, 2023).

Enquanto os MLN introduziram as redes neurais para calcular a probabilidade de uma sequência de palavras (ZHAO *et al.*, 2023). Uma rede neural é uma máquina projetada para modelar a maneira como o cérebro humano realiza uma tarefa ou função específica de interesse; a rede geralmente é implementada utilizando componentes eletrônicos ou em software (HAYKIN, 2001).

Já as MLP são modelos capazes de capturar representação de palavras conscientes do contexto (ZHAO *et al.*, 2023), pois foram treinados usando as em grandes quantidades de dados, sendo capazes de executar uma gama de tarefas. Utilizam de arquiteturas de redes neurais, como a *Transformer*, um exemplo é o *GPT-2* (ZHAO *et al.*, 2023). Ao aumentar o escopo de dados de treinamento de um MLP, sugiram os Modelos de Linguagem de Larga Escala (MLLE).

2.2 MODELOS DE LINGUAGEM DE LARGA ESCALA

Modelos de linguagem de larga escala (MLLE) são uma categoria de modelos de linguagem que utilizam redes neurais contendo bilhões de parâmetros, treinados com enormes quantidades de dados textuais não rotulados usando a abordagem de aprendizado auto-supervisionado. Ou seja são modelos capazes de aprender padrões complicados, sutilezas da linguagem e ligações semânticas os quais existem com diferentes tamanhos (RAIAAN *et al.*, 2024).

Esses modelos já provaram suas funcionalidades com diversas tarefas linguísticas, incluindo: analise sintática de texto, tradução, sumarização, analise de sentimentos, perguntas e respostas. Alavancando as técnicas de aprendizado profundo e grandes conjuntos de dados (RAIAAN *et al.*, 2024).

Um exemplo é o ChatGPT, que é uma aplicação que usa um dos modelos mais populares existentes no dia de hoje: o GPT-4 é um modelo de grande porte capaz de lidar com tarefas complexas com um nível de desempenho e precisão superior ao de modelos menores (ACHIAM *et al.*, 2023).

Para se ter uma ideia, o GPT-3, sua versão anterior, já contava com 175 bilhões de parâmetros, treinado em clusters de GPUs V100 durante várias semanas (BROWN *et al.*, 2020), um processo que provavelmente custou milhões de dólares. O GPT-4, por sua vez, requer ainda mais recursos e contém significativamente mais parâmetros

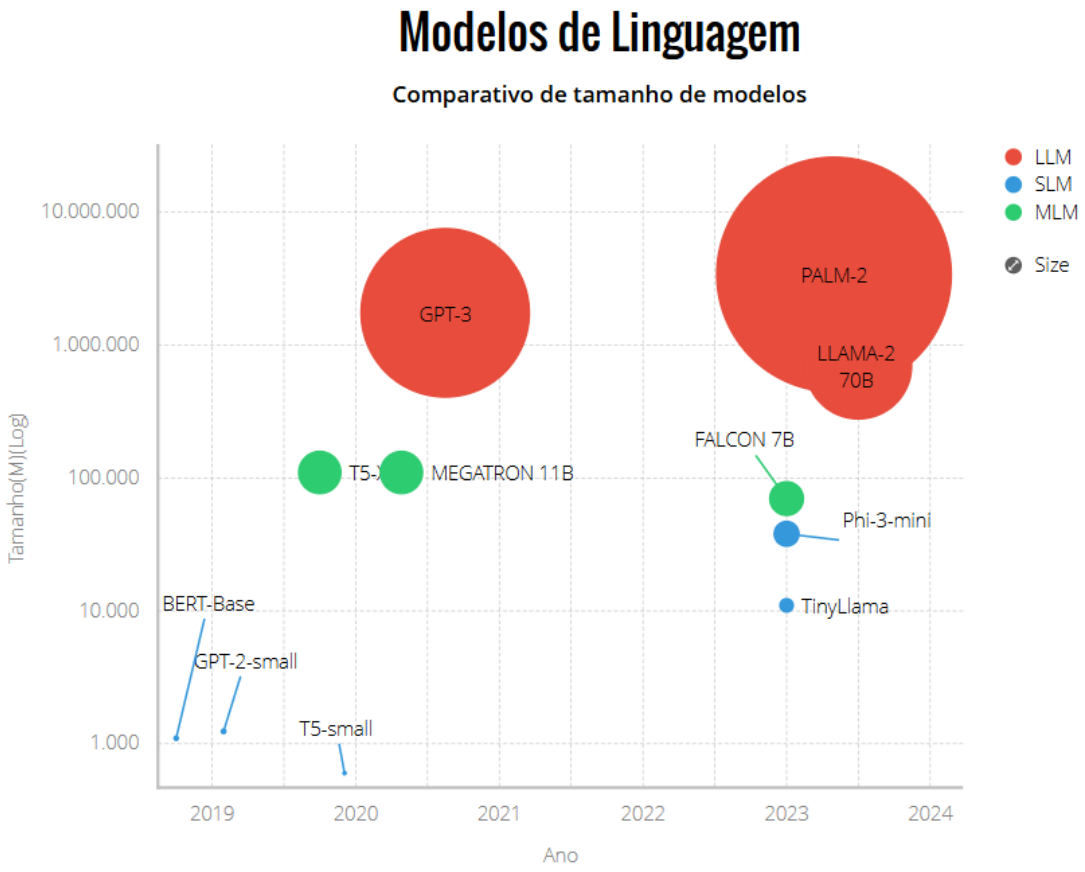

Figura 1: Modelos de Linguagem e suas magnitudes

que o GPT-3 (ACHIAM *et al.*, 2023).

A Figura 1 ilustra a magnitude de diferentes modelos de linguagem, destacando o tamanho do GPT-3 e do PaLM-2 em comparação com outros modelos, como o T5 e o Megatron. Essa representação ajuda a compreender a diferença de escala entre os modelos, proporcionando uma noção visual do impacto do aumento no número de parâmetros.

A ascensão dos MLLE foi devido a altas capacidades de processamento de linguagem natural, todavia o desenvolvimento de tal modelo requer muitos recursos, o que faz impossível a execução em hardware comuns (XU *et al.*, 2023). Isso fez com que os olhares se voltassem para os modelos de linguagem de pequena escala.

Os modelos de linguagem de pequena escala são modelos com menos parâmetros, na casa dos milhões de parâmetros. Esses modelos especializados em uma tarefa de domínio especifico são uma alternativa para as MLLE pois requerem menos poder computacional (HSIEH *et al.*, 2023).

Enquanto os modelos de média escala são uma solução intermediária que

pode oferecer um bom equilíbrio entre precisão e eficiência computacional. No geral os modelos de pequena e média escala podem ser eficientes e terem custo computacional reduzido quando há qualidade nos dados de treinamento desses modelos (HABIB *et al.*, 2024).

2.3 PROCESSAMENTO DE LINGUAGEM NATURAL

Token é uma unidade elementar de texto obtida por meio do processo de tokenização, no qual uma sequência de caracteres (geralmente um texto ou sentença) é segmentada em partes menores, como palavras, pontuações ou subpalavras, dependendo da técnica utilizada.

Em tarefas de PLN, tokens são as unidades sobre as quais os modelos de linguagem operam — por exemplo, na análise gramatical, vetorização, classificação ou tradução (JURAFSKY; MARTIN, 2014). Por exemplo se fosse considerar a seguinte frase *"O sol está bonito hoje!"*, esta frase possui 6 tokens, 6 unidades de texto, incluindo a pontuação, sendo respectivamente : *'O','sol','está','bonito','hoje','!'*

Em diversas aplicações de PLN é interessante desconsiderar algumas palavras que pouco acrescentam ao conteúdo do texto, como preposições, determinantes, conjunções etc. Essas palavras são conhecidas como *stopwords*. É importante notar que, além das palavras clássicas mencionadas, as *stopwords* podem variar de acordo com o contexto específico com o qual estamos trabalhando (CASELI; NUNES, 2024).

2.4 TAREFAS DE PROCESSAMENTO DE LINGUAGEM NATURAL

O conceito de “tarefa” no contexto de Processamento de Linguagem Natural (PLN) refere-se a uma atividade específica que um sistema de PLN pode realizar para processar, entender ou gerar linguagem humana (JURAFSKY; MARTIN, 2014). Nesta seção vamos apresentar algumas tarefas: a extração de informação, resposta automática a perguntas e geração de texto, que são comuns em modelos de linguagem.

2.4.1 Extração de Informação

A extração de informação é desenvolvida com objetivo de se obter uma informação a partir de dados não-estruturados (JURAFSKY; MARTIN, 2014). A tarefa pode ser divida em outras tarefas de interesse, as mais comuns citadas na literatura são o Reconhecimento de Entidades Nomeadas, Extração de Relações e Extração de

Eventos (CASELI; NUNES, 2024).

O Reconhecimento de Entidades Nomeadas tem como propósito identificar e categorizar informações que são cruciais a um estudo, como por exemplo nome de pessoas, locais, organizações, expressões numéricas, por exemplo (MELLO *et al.*, 2024). Enquanto a Extração de Relação consiste em identificar as relações entre as entidades nomeadas no texto, por exemplo a extração de um endereço no texto é uma tarefa de Reconhecimento de Entidades Nomeadas, mas para identificar se esse endereço é um dado sensível é necessário que ele tenha alguma relação com uma entidade pessoa, nesse caso usa-se a Extração de Relação (LUCENA *et al.*, 2024).

A Extração de Eventos por outro lado consiste na tarefa de identificação de uma menção a um evento em uma sentença, se existirem, há também a extração de informação sobre esse evento. Um evento é entendido como uma ocorrência especifica envolvendo participantes, algo que acontece e que pode ser descrito como mudança de estado da qual participam entidades como agentes (CASELI; NUNES, 2024). Por exemplo, a extração de eventos diversos relacionados a vacina em dados não-estruturados provindos de redes sociais (LI *et al.*, 2025).

2.4.2 Resposta Automática a Perguntas

Resposta Automática a Perguntas, em inglês *Question Answering (QA)*, estuda como criar sistemas capazes de responder de forma automática a perguntas em linguagem natural. Esses sistemas buscam a capacidade de compreender a pergunta, recuperar informações relevantes e fornecer respostas precisas e úteis (CASELI; NUNES, 2024). Dessa forma, os modelos são capazes de responder perguntas do tipo: *"Quem é o presidente do STF?"*, *"Considere a Portaria nº462/2017, quais os impactos da implementação do Sistema de Frequência online?"* ou *"Como a implementação do processo eletrônico impactou a eficiência e a transparência do Tribunal de Contas do Estado de Goiás, considerando os desafios e as medidas adotadas para garantir a acessibilidade e a segurança das informações processuais?"* (PARANHOS *et al.*, 2024)

Essas perguntas se diferem pela forma como a resposta é exigida. A primeira é uma pergunta simples de resposta direta. A segunda exige uma resposta mais detalhada, que pode envolver a extração de informações de um documento. Enquanto a última é uma pergunta ampla, permitindo uma resposta exploratória e detalhada, sem necessariamente ter uma única resposta correta (PARANHOS *et al.*, 2024). Esse tipo de consulta incentiva sistemas de geração aumentada por recuperação

(PARANHOS *et al.*, 2024).

2.4.3 Geração de Texto

A geração de texto, também conhecida como geração de linguagem natural tem sido umas das tarefas mais importantes das sub áreas em processamento de linguagem natural. É responsável por produz texto plausível e legível em linguagem humana a partir de diversos formatos de entrada, incluindo texto, imagem, tabelas e conhecimento de base. Tem sido muito utilizado nas últimas décadas para gerar respostas para usuários em conversações, tradução de texto, sumarização a partir de uma fonte textual (LI *et al.*, 2024).

Os *Transformers* que estão por trás de muitos modelos generativos do estado da arte como GPT-3 (CAO *et al.*, 2025). Sua estrutura baseia-se no mecanismo de autoatenção (composta por um *encoder*, que processa a entrada gerando representações ocultas, e um *decoder*, que utiliza essas representações para produzir a sequência de saída. Cada camada contém um módulo de *multi-head attention* - que aprende a ponderar a relevância entre diferentes tokens - seguido por uma rede neural *feed-forward* (VASWANI *et al.*, 2017).

A grande vantagem do Transformer reside em sua capacidade de capturar dependências de longa distância com maior eficiência que os modelos sequenciais tradicionais, além de ser altamente paralelizável, característica que possibilitou seu treinamento em larga escala. Essa combinação de atributos - especialmente a capacidade de priorizar dados em detrimento de vieses indutivos - tornou o Transformer a base para os modernos sistemas de PLN, permitindo sua adaptação a diversas tarefas através do paradigma de pré-treinamento em massa (CAO *et al.*, 2025).

Todavia possuem complexidade quadrática de atenção, o que eleva o custo computacional e restringe a escalabilidade em contextos longos (TAY *et al.*, 2022). Além da falta de memória persistente, onde cada entrada precisa trazer o contexto todo novamente (BERRYMAN; ZIEGLER, 2024). O que afeta diretamente tarefas de raciocínio os quais precisam de atenção distribuídas e comparação sistemática entre os itens (BERRYMAN; ZIEGLER, 2024).

Nos modelos de linguagem de grande escala (MLLE), como o GPT-3, GPT-4 e similares, aspectos como o tamanho da janela de contexto e o controle da temperatura de geração exercem papel fundamental na qualidade e no comportamento da saída textual. A janela de contexto define o número máximo de tokens que o modelo

consegue processar de uma vez, limitando sua capacidade de raciocínio com base em informações anteriores — fator crítico para tarefas com múltiplos documentos ou instruções complexas. Já a temperatura atua como um parâmetro de amostragem que regula o grau de aleatoriedade na escolha de palavras subsequentes: valores baixos promovem respostas mais determinísticas e coerentes, enquanto valores mais altos incentivam saídas criativas, porém menos previsíveis. Compreender e ajustar esses elementos é essencial para adaptar o comportamento dos MLLE a diferentes tarefas, desde geração factual até exploração criativa (BERRYMAN; ZIEGLER, 2024).

2.5 AJUSTE-FINO

O ajuste-fino é uma técnica para ajustar um MLLE pré-treinado com dados específicos de um domínio, para melhorar a sua performance em tarefas diversas como analise de sentimentos, resumos de notícia e resposta a pergunta sobre um tema. A forma padrão de um ajuste inclui treinamento com dados brutos (contexto + pergunta + resposta), onde a resposta é mascarada e o modelo aprende a gerá-la, requerendo conjunto de dados com pares de entrada e saída (LI *et al.*, 2023).

Além do ajuste fino tradicional, existem outras abordagem como ajuste-fino de instrução onde o conjunto de dados carregam instruções que guiam o modelo, e técnicas de eficiência como o LoRa (LI *et al.*, 2023). LoRa é uma forma parcial de se realizar o ajuste-fino, atualizando as camadas densas da rede neural com matrizes de baixo rank conectáveis. Matrizes as quias são independentes da MLLE, podendo ser armazenada e reutilizada em outras tarefas relacionadas. Embora o LoRa seja uma forma mais barata de se fazer o ajuste-fino, a medida que os módulos se acumulam, o custo computacional de gerenciamento aumenta (MAO *et al.*, 2025), o que ainda é uma forma cara de se realizar o ajuste fino e ainda é um processo trabalhoso.

2.6 RECUPERAÇÃO AUMENTADA POR GERAÇÃO

A recuperação aumentada por geração (RAG) é uma técnica que pode oferecer conhecimento externo confiável e atualizado, proporcionando enorme conveniências para inúmeras tarefas. A técnica funciona incorporando informações de fontes de dados externos que servem como suplemento para a consulta de entrada ou saída gerada. RAG é uma técnica viável e eficiente para aplicar em várias tarefas de geração com uma simples adaptação do componente de recuperação, exigindo o mínimo de treinamento e ou nenhum, demonstrando potencial em diversas tarefas (FAN *et al.*,

2024).

Todavia mesmo com a implementação da técnica outros problemas se acarretam como sobrecarga de armazenamento e custo computacional, latência de inferência e aumento indiscriminado, no caso passar informações irrelevantes para a MLLE, as quais já tem conhecimento ou atualizar com informações incorretas (FAN *et al.*, 2024).

3 METODOLOGIA

A metodologia adotada neste trabalho consiste em uma análise qualitativa e comparativa da performance de um modelo de linguagem de larga escala ao lidar com diferentes versões de um contexto textual. O objetivo é avaliar a capacidade do modelo em responder adequadamente as perguntas sobre mensagens reais e avaliar o tempo de respostas e verificar o impacto no tempo de resposta e a qualidade ao reduzir o tamanho desse contexto utilizando diferentes técnicas como *stopwords* e auto gerado.

3.1 DEFINIÇÃO DOS CRITÉRIOS

Nesta seção são apresentados os critérios utilizados para orientar análise e tomada de decisões ao longo do processo. Os critérios foram organizados em categorias, cada uma refletindo um conjunto de aspectos relevantes para avaliação de forma estruturada e coerente.

3.1.1 Seleção de Modelo

Para a escolha de um modelo para analise consideramos neste trabalho o número de tokens que pode ser feito por requisição, o número de requisições disponíveis por dia e por minuto e principalmente o custo, priorizando serviços de modelos gratuitos para a pesquisa.

O foco deste trabalho é em modelos de larga escala devido a grande compreensão de linguagem natural avançada e da versatilidade em múltiplas tarefas (BROWN *et al.*, 2020) e alinhamento com o estado da arte como GPT, Gemini, Deepseek e Claude.

3.1.2 Seleção de Contexto

O contexto para esse trabalho foi escolhido com os critérios de disponibilidade de acesso e processamento de dados. Por esse motivo escolhemos mensagens provinda de WhatApp, pois todo o contexto está em um único arquivo que pode ser obtido, contendo: data, hora, autor e mensagem.

Além de que hoje em dia muitas pessoas utilizam grupos de WhatsApp no trabalho e esses grupos acabam armazenando muitas informações úteis ao longo do tempo e que precisam ser constantemente consultadas. A ideia é utilizar todo o texto de um grupo do WhatsApp como contexto e fazer perguntas sobre esse texto.

As perguntas podem ser genéricas, onde se quer simplesmente obter uma informação compartilhada no grupo, por exemplo *“Como resolver tal problema?”* Sabendo que esse problema já foi respondido por alguém do grupo no passado. Outro tipo de pergunta é a analítica, que requer um computo maior de informações para que possam ser respondidas. Um exemplo de consultas analítica é *“Qual palavra foi mais escrita por fulano?”*. Para responder essa pergunta, o Modelo precisa analisar todo o texto enviado por fulano o grupo.

3.1.3 Processamento de Contexto

O contexto vai ser analisado de três formas distintas:

1. Contexto Completo: exatamente como foi extraído do WhatsApp com o máximo de texto que é capaz de ser carregado pela modelo de linguagem;
2. Contexto Resumido: amostragem do contexto completo obtido através de autogeração a partir de uma requisição ao modelo de linguagem.
3. Contexto Filtrado: contexto completo a partir da remoção das *stopswords*.

O contexto completo refere-se à maior quantidade de texto que um modelo de linguagem de larga escala (MLLE) é capaz de processar em uma única requisição, respeitando o limite máximo de tokens do modelo. Para aproveitar ao máximo esse espaço com informações relevantes, realiza-se uma limpeza inicial no texto, removendo o carácter *Ž00e*, comum em mensagens que indicam anexos (imagens, vídeos ou áudios), os quais não podem ser processados como texto.

Em seguida, o conteúdo é estruturado com base no padrão do WhatsApp, segmentando cada mensagem em data, hora, autor e conteúdo. Essa estruturação é essencial para a posterior análise e validação das respostas geradas pelo MLL. O contexto textual é então reconstruído mantendo os colchetes que delimitam a data e hora, e utilizando quebras de linha para separar as diferentes mensagens.

Caso o texto ainda ultrapasse o limite de tokens do modelo, aplicam-se etapas adicionais de filtragem. Primeiramente, são removidas mensagens automáticas, como *"image omitted"* ou*"Messages and calls are end-to-end encrypted. Only people in this chat can read, listen to, or share them"*. Se necessário, remove-se também o início da conversa (mensagens mais antigas), até que o conteúdo esteja dentro do limite processável pelo modelo.

Enquanto o contexto resumido é o contexto que foi gerado pela MLLE a partir da solicitação: *Com base no contexto fornecido, gere uma amostra grande*.

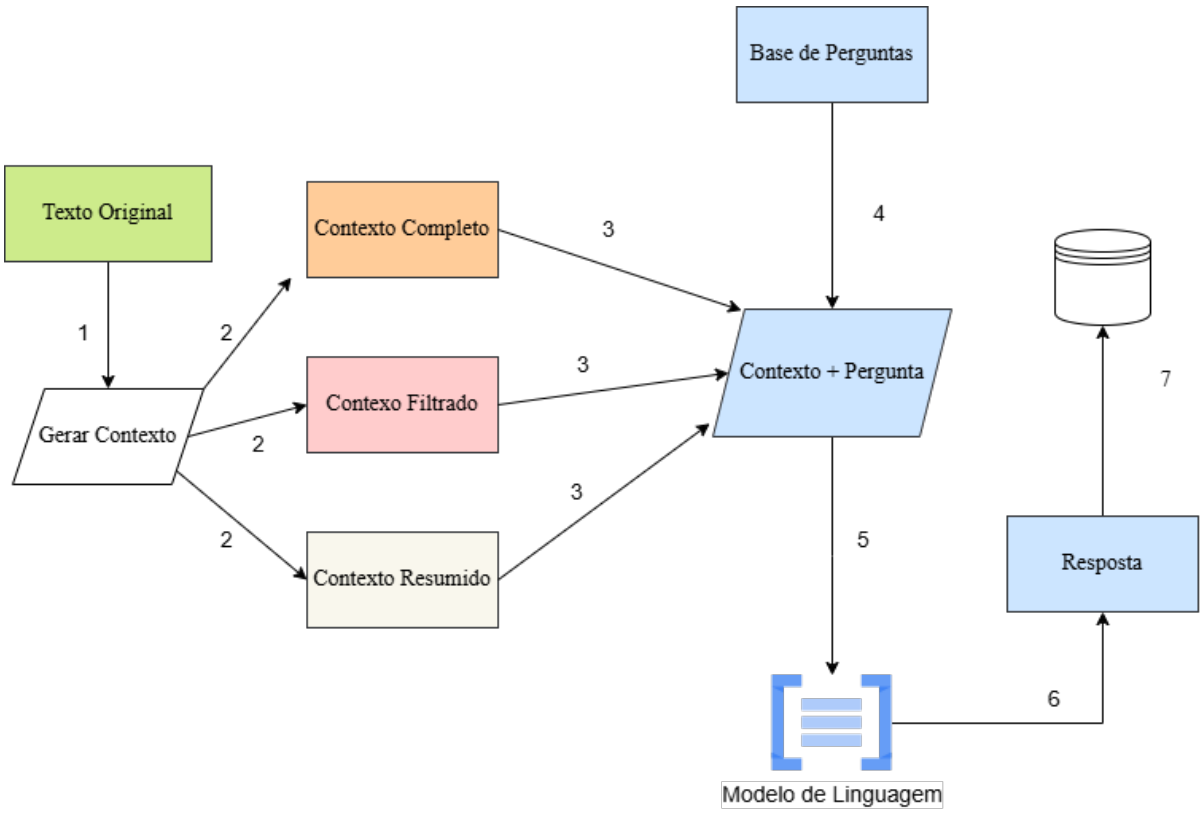

Figura 2: Experimento

O contexto filtrado é obtido a partir da remoção das *stopswords* é o contexto completo com o tamanho máximo de texto que a MLLE consegue processar em uma única requisição que passou por um processo de filtragem de remoção das *stopswords*, Seção 2.3, que por sua vez são palavras muito frequentes e geralmente pouco informativas.

3.1.4 Seleção de Perguntas

As perguntas são escolhidas para que as respostas dadas pelo modelo possam ser contabilizada por técnicas de processamento de linguagem natural e que de certa forma, possam ser julgadas qualitativamente. De forma que seja possível verificar a habilidade de extração de um contexto geral, contabilização e ordenação.

3.2 EXPERIMENTO

O experimento, representado na Figura 2, consiste em gerar 3 contextos a partir de um texto original. Cada um desses contextos é submetido para um modelo de larga escala juntamente com uma base de perguntas pré-definidas. A resposta é registrada

juntamente com o tempo de processamento, que é o tempo desde o envio a chega da requisição.

3.3 AVALIAÇÃO

Na primeira etapa o contexto é reduzido para que possa ser carregado completamente. A partir do contexto reduzido, dois resumos são criados, um utilizado *stopswords* e outro utilizando a própria MLLE para gerar um resumo detalhado.

Esses textos (contextos) juntamento com as perguntas são submetidos a uma MLLE e computamos o tempo de resposta e a resposta fornecida. A reposta dos modelos resumido é comparada com a resposta do modelo completo (resumido para ser enviado), analisando qual contexto consegue se aproximar mais da resposta obtida com o contexto completo.

A seguir avaliamos a qualidade das respostas fornecida, por exemplo para a pergunta: *"Quais falhas do Sagres foram discutidas no texto?"*, espera-se que o sistemas reconheça problemas como: dificuldade nos processos de matricula, lentidão, "bugs"e outros problemas comuns e conhecidos pela comunidade acadêmica e que seja comum aos textos.

Caso o modelo não consiga extrair devido a falta de informação no contexto, é esperado como resposta correta a afirmação do modelo de que ele não pode realizar a tarefa, ao invés de gerar uma resposta sem fundamento.

4 DESENVOLVIMENTO

Este capítulo descreve o processo de implementação da metodologia proposta. A Seção 4.1 descreve a Seleção de Modelo e o que foi levado em consideração para a escolha do modelo, enquanto a Seção 4.2 descreve a seleção do contexto e os processos de geração das diferentes versões do mesmo contexto. A Seção 4.3 descreve como as perguntas foram feitas, a Seção 4.4 como os resultados foram avaliados. Todos os resultados e o código fonte do projeto se encontra no Github [1].

4.1 SELEÇÃO DE MODELO

Para a seleção de modelo foi feita uma avaliação com alguns dos principais modelos no mercado e levamos em consideração o custo e os benefícios de cada versão.

Modelos como GPT[2], Deepseek[3] e Gemini[4] possuem preço para cada 1 milhão de *tokens* sendo por entrada, cache e saída. Para os valores de tokens de entrada podemos observar a Tabela 1, o GPT e o Deepseek não possuem versões gratuitas de serviço como o Gemini.

Modelo	Nível Gratuito	Preço por 1M de Tokens de Entrada
GPT-4.1 nano	Não	$0,100
GPT-4.1 mini	Não	$0,400
GPT-4.1	Não	$2,00
deepseek-chat	Não	$0,07 (Cache Hit)
deepseek-chat	Não	$0,27 (Cache Miss)
Gemini Flash 2.0	Sim	$0,100
Gemini Flash-Lite 2.0	Sim	$0,075

Tabela 1: Modelos GPT e DeepSeek disponíveis

Os modelos do Google[5] possuem diferentes níveis de cobrança que variam do nível gratuito em países qualificados pela empresa que incluem o Brasil, França, Argentina, por exemplo. Os limites são estabelecidos baseado em requisições por dia (RPD), por minuto (RPM) w número de *tokens* por minuto (TPM). Temos por exemplo a Tabela 2, que contém os limites de cada modelo.

Para esse trabalho optamos pelo Gemini-Flash 2.0, que possui versões de uso

[1] https://github.com/laraesquivel/avaliacao_de_llm
[2] https://openai.com/api/pricing/
[3] https://api-docs.deepseek.com/quick_start/pricing
[4] https://ai.google.dev/gemini-api/docs/pricing?hl=pt-br
[5] https://ai.google.dev/gemini-api/docs/rate-limits?hl=pt-br#free-tier

Modelo	Nível Gratuito	RPM	TPM	RPD
Gemini Flash 2.0	Sim	15	1.000.000	1.500
Gemini Flash-Lite 2.0	Sim	30	1.000.000	1.500

Tabela 2: Modelos Gemini disponíveis no nível gratuito e seus limites de *tokens* por requisição

gratuito e é um modelo maior que o Gemini-Flash-Lite. O Gemini-Flash-Lite possui o limite de entrada de 1.048.575 tokens.

4.2 SELEÇÃO DE CONTEXTO

O contexto selecionado são as mensagens provinda do grupo de Whatsapp da Câmera de Graduação da UEFS, é um grupo com temas diversos e relevantes para o funcionamento da Universidade que incluem temas como medidas sanitária ou curricularização de extensão, por exemplo. O grupo conta com 78 autores e contempla período de 24 de Fevereiro de 2022 a 29 de Maio 2025.

Usando as mensagens definimos os contextos. O arquivo de texto inicialmente possui 3,4Mb e 1.295.527 tokens, a cima do limite do Gemini.

Para isso, foi removido os caracteres especiais que indicam anexos, o que não impactou significativamente na redução do tamanho do texto. A seguir esse texto foi estruturado por meio de uma expressão regular em tabelas com os campos: data, hora, autor e mensagem. Os registros que tiveram menções as mensagens automáticas foram removidos, o texto foi restruturado novamente seguindo o padrão: $[data\ hora]\ autor : mensagem$

As remoção das mensagens automáticas reduziu o tamanho do texto para 3,03Mb, mas não foi suficiente para ter o número de tokens processado pelo MLL, já que o texto resultou 1.180.142 tokens. A partir da tabela estruturada, removemos os primeiros registros até que o texto completo tivesse um número menor de tokens do que o limite permitido, o valor processado pelo MLL. O contexto final possui 1.037.996 tokens e 2,67Mb.

O contexto resumido, é uma amostra do contexto completo, obtido através do seguinte *prompt*: *Com base no contexto fornecido, gere uma amostra*, e assim foi retornado uma amostra que possui 8.196 tokens e 0,02Mb.

Por fim, o último contexto foi gerado a partir do contexto completo com a remoção das *stopswords* usando o biblioteca *spacy*[6], o que nos retornou um contexto com 747.987 tokens e 1,87Mb.

[6]https://spacy.io/universe

A influencia das *stopswords* no texto podem ser observadas ao fazer um comparativo entre as nuvens de palavras nas figuras 3 e 4.

Na Figura 3, as nuvens de palavras é formada a partir do contexto completo, todavia os temas do grupo não são claros, e isso poderia facilmente representar qualquer grupo de Whatsapp. Enquanto na Figura 4 os temas são visíveis, e palavras como: colegiado, aluno, curso, estudante, ...etc que são esperada para discussões de um grupo de uma câmera de graduação podem ser detectadas.

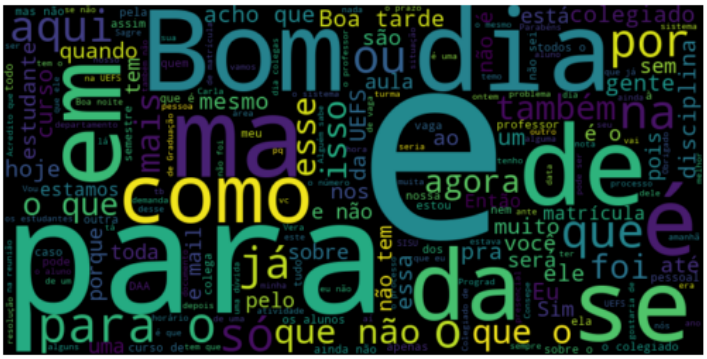

Figura 3: Nuvem de Palavras do Contexto Completo

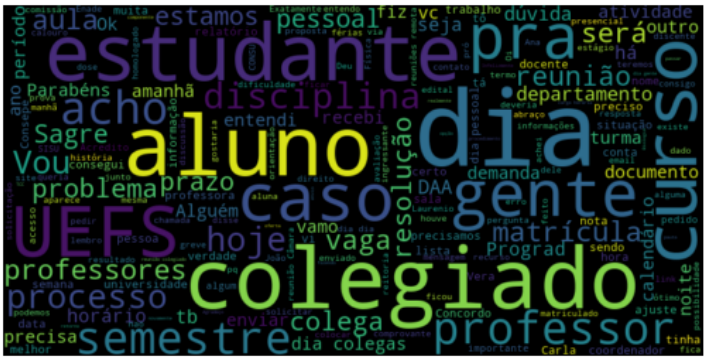

Figura 4: Nuvem de Palavras do Contexto sem *stopwords*

4.3 SELEÇÃO DE PERGUNTAS

Para os 3 contextos armazenamos as respostas e o tempo que o modelo demora a responder para cada um dos contextos distintos em arquivos de extensão *csv*. As perguntas foram feitas com a temperatura, Subseção 2.4.3, de 0.7 para todos os casos.

As perguntas incluem:

- Quais as falhas do Sagres foram discutidas no texto?

- Quais resoluções foram discutidas no texto?
- Listar dicas de uso do Sagres
- Quais as palavras mais mencionadas por Rosa?
- Quantas resoluções CONSEPE foram citadas?
- Quais professores mais participaram da conversa, listando o número de contribuições?

Para cada pergunta é avaliado se o modelo conseguiu responder adequadamente. Estruturando o texto em uma tabela com colunas: data, hora, mensagem e autor para assim verificar se as respostas fornecidas pelo Gemini estão de acordo.

5 RESULTADOS

Esta seção descreve os resultados dos experimentos realizados.

5.1 COMPUTAÇÃO DE CONTEXTO

5.1.1 Computação do Resumo

O resumo foi processado em 81,7 segundos, com 21.248 carácteres e 0,02Mb. O resumo é uma amostra de algumas das principais discussões que envolvem a Câmera de Graduação.

A amostra é uma fatia de mensagens de diversos integrantes, onde discutem temas discutidos incluem: processos administrativos, formatos de defesa do trabalho de conclusão de curso e feriado, Covid 19 e plágio e questões pedagógicas.

5.1.2 Computação de Contexto Filtrado

O contexto filtrado foi gerado a partir da remoção das *stopwords*, o contexto foi processado em 125 segundos, todavia diferente do contexto resumido que foi processado pelo próprio Gemini, o contexto filtrado foi processado pelo modelo *spacy pt_core_news_sm* provindo da biblioteca *spacy*[1].

5.2 AVALIAÇÃO DAS RESPOSTAS

5.2.1 Pergunta 1: Quais as Falhas do Sagres Foram Discutidas no Texto?

Para o contexto completo, o modelo elencou uma lista principal com 12 itens discutindo as falhas que deveriam ser do Sagres e outra lista com outros 95 problemas. As listas incluem, pontos reais como:

- Problemas de Lançamento de notas no Sagres;
- Erros de cálculo de Score;
- Dificuldade de Acesso;
- Sistema mal configurado.

Todavia em ambas listas inclui pontos que não são do Sagres e que não são problemas, e sim a outros sistemas como:

- Dificuldade de realizar Teste de Covid;

[1] https://spacy.io/models

- Problema com o acesso ao UNES;
- Falta de autonomia dos Colegiados.

Além disso, em alguns pontos, o modelo não resumiu e apenas elencou falas dos participantes, como:

- *"Então a resolução atual válida é a anterior a 'pendêmia', que não permitia remoto, confere?"*;
- *"Eu penso do mesmo jeito"*;
- *"Eu vou substituir Irlana, porém não poderei ficar até o final pois tenho aula às 16:30"*;
- *"Os estudantes dos cursos de vocês tem apresentado sintomas gripais frequentes ou estão com COVID"*.

O modelo em suma elencou problemas gerais e listou falas de professores sobre diversas questões, algumas vezes repetidas.

Para o resumo o modelo elencou que não havia menções do Sagres no texto. O que demonstrou que a amostra autogerada não foi representativa.

No contexto filtrado, o modelo identificou 16 itens. Assim como no contexto completo, os problemas mencionados envolvem dificuldades de acesso e instabilidade do sistema, erros e inconsistências nos dados, falhas na digitação de horários e ausência de suporte adequado. A diferença principal é que, no contexto filtrado, os itens foram apresentados de forma mais objetiva e concisa

Todavia cometeu os mesmos erro ao confundir os problemas do Sagres com problemas de outros sistemas, no caso o próprio site da UEFS. O ponto forte foi não capturar as falas dos participantes, como ocorreu no contexto geral.

5.2.2 Pergunta 2: Quais Resoluções Foram Discutidas no Texto?

Para o contexto completo, o modelo foi incapaz de responder a pergunta corretamente, ele elencou resoluções de problemas da UEFS ao invés das resoluções acadêmicas. Ele listou reuniões e eventos, pessoas e seus cargos, mas não elencou nenhuma resolução acadêmica.

Para o contexto resumido, o modelo elencou 4 resoluções:

- Resolução CONSEPE 129;
- Resolução 030/2021;
- Resolução CONSEPE 148/2013;

- Resolução CONSEPE (sugestão de implementação baseada na 106/2021).

O modelo só conseguiu detectar inteiramente (com conselho, código e ano) uma resolução: a Resolução CONSEPE 148/2013. A resolução CONSEPE 106/2021 é mencionada e a CONSEPE 030/2021 via link. Todas essas realmente existem [2]. Não é possível saber se a CONSEPE 129 existe, pois a resolução é mencionada no texto sem o ano da resolução, apenas o que ela abrange: estágio supervisionados.

Para o contexto filtrado, ele retornou as resoluções separadas por conselho, instrução normativa e decreto estadual. Todavia sem seu respectivos códigos, devido a filtragem que os removeu, apenas temas que estão em torno dessas resoluções

5.2.3 Pergunta 3: Listar Dicas de Uso do Sagres

Para o contexto completo o modelo não seguiu a instrução e gerou uma resposta sem sentindo, fazendo um resumo de todas as discussões do grupo, como se o Sagres fosse o próprio grupo da Câmara de Graduação.

Para o resumo, o modelo não gerou dicas pois não havia menções.

Para o contexto filtrado o modelo gerou dicas diversas envolvendo: acesso, navegação, relatórios, documentos e matriculas. Além das dicas, gerou conselhos e avisos sobre o Sistema Sagres, elencado 4 problemas comuns ao sistema. Todavia nem todas as dicas elencadas são referentes ao Sagres mas impactam diretamente sobre processos do sistema, como o comprovante de vacinação para a matricula.

5.2.4 Pergunta 4: Quais as Palavras mais Mencionadas Por Rosa?

Para o contexto completo o modelo menciona temas do grupo e ao final responde a pergunta solicitada: *"Em relação à sua pergunta sobre as palavras mais mencionadas por Rosa Eugênia, o nome dela aparece com frequência, pois ela é quem envia as convocações para as reuniões. Outras palavras que se destacam em suas mensagens são "queridos", "amores", e emojis positivos, demonstrando seu carinho e otimismo."*

Para o resumo, o modelo não conseguiu elencar nenhuma palavra para Rosa, pois não há menções de Rosa no texto entre os 25 participantes que estavam na amostra.

O contexto filtrado primeiro lista palavras mencionadas por Rosa as contabilizando, e em seguida elenca 18 palavras mais mencionadas por Rosa com

[2] https://www.uefs.br/modules/conteudo/conteudo.php?conteudo=146

a sua contagem. Abaixo está as 5 palavras que o mais mencionadas por Rosa para o contexto filtrado:

- dia - 7;
- Amores - 2;
- Acho - 2;
- Pessoal - 2;
- Colegiado - 4;

Todas as palavras elencadas pelos modelos, realmente são menções de Rosa. Todavia na Figura 5 podemos encontrar as palavras mais mencionadas por Rosa por meio de contagem, observe que embora as palavras elicitadas existam e o modelo consegue elencar palavras frequentes como *querido* e *amores*, há outras palavras mais frequentes que foram desconsideradas, como *reunião* e *UEFS* em prol de palavras com baixa frequência como *gente*.

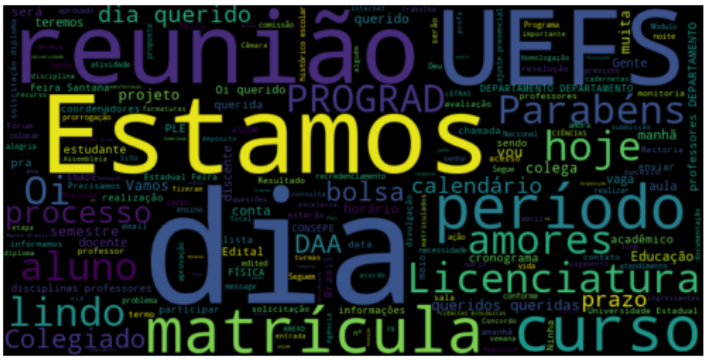

Figura 5: Palavras Mencionadas por Rosa

Além disso, a contagem de palavras resultante do contexto filtrado está errada, o que consequentemente gerou um rank incorreto. O verdadeiro, top3 de Rosa na verdade é:

- dia - 71;
- Oi - 24;
- UEFS - 24.

5.2.5 Pergunta 5: Quantas Resoluções CONSEPE Foram Citadas?

Para o contexto completo, ele não retornou a contagem de resoluções CONSEPE, e sim trechos e parte das mensagens que o modelo identificou referentes

as resoluções, algumas repetidas mais de dez vezes e que por vezes não se referiam as resoluções.

O contexto resumido contabilizou apenas 2 resoluções CONSEPE, embora o modelo tenha identificado 3 na Seção 5.2.2 por ele como CONSEPE. No contexto resumido há 4 resoluções CONSEPE.

O texto filtrado retornou 18 resoluções. Todavia as resoluções CONSEPE não possuem código, existe 48 menções a expressão “resolução consepe” no contexto filtrado. O modelo não explicou como ele contabilizou, como não há código. Caso tenha tema, a palavra resolução aparece 607 vezes, 18 é um número baixo para os diferentes temas e aparições.

5.2.6 Pergunta 6: Quais Professores mais Participaram da Conversa, Listando o Número de Contribuições?

Para o contexto completo ele não elegeu um rank, ele apontou temas e listou professores que ele considerou importantes:

1. *Patricia Guimarães Costa: Responsável pela secretaria das câmaras, enviando lembretes de reuniões e listas de processos*;
2. *Vera Martin (Prograd): Membro da PROGRAD, fornecendo informações sobre prazos, procedimentos, e respondendo a dúvidas*;
3. *Laurenio Sombra, Hélio Ponce, João B. Rocha-Junior, Cláudia: Participantes frequentes nas discussões, compartilhando informações e opiniões*;
4. *Carla Borges: Envolvida na organização de eventos e fornecendo informações sobre a curricularização da extensão*;
5. *Ady Sá Teles: Participante ativa, contribuindo com informações e opiniões*;
6. *Prof Dra Irlana Jane: Compartilhando informações sobre a área de Pedagogia*;
7. *Myria Felix: Expressando preocupações sobre diversos temas e buscando informações*;
8. *Fredson Braz: Apresentando questões sobre o curso de Física e compartilhando informações relevantes*;

O resumo elicitou os nomes de 9 profissionais que apareciam em destaque e o número de aparições no caso:

1. Laurenio Sombra - 13;
2. Hélio Ponce - 13;
3. Marília Lordêlo - 12;

4. Adriana Queiroz - Agronomia: 8;
5. Cláudia - 5;
6. João B. Rocha-Junior - 5;
7. Vera Martin (Prograd) - 5;
8. Fredson Braz - 5;
9. Ady Sá Teles - 5;

Enquanto para o contexto filtrado, o modelo elencou o número de participação de 98 participantes, o grupo tem 78. O seu top5 inclui:

1. Laurenio Sombra - 98;
2. Adriana Queiroz - 88;
3. Marília Lordêlo - 68;
4. Hélio Ponce - 52;
5. Cláudia - 43.

Todavia, ao contabilizar as mensagens reais, observamos a discrepancia dos valores de contagem:

1. Adriana Queiroz - Agronomia - 2480;
2. João B. Rocha-Junior - 1317;
3. Fredson Braz - 1295;
4. Myria Felix - 1273;
5. Luciana Bagdeve - 1128.

5.3 AVALIAÇÃO DO TEMPO DE RESPOSTA

Para a avaliação do tempo de respostas considere as perguntas com os seguintes identificadores:

1. Quais falhas do Sagres foram discutidas no texto?
2. Quais resoluções foram discutidas no texto?
3. Listar Dicas de Uso do Sagres.
4. Quais as palavras mais mencionadas por Rosa?
5. Quantas Resoluções CONSEPE foram citadas?
6. Quais os professores mais participaram da conversa, listando o número de contribuições?

Enquanto os contextos:

1. Contexto Completo;
2. Resumo;
3. Contexto Filtrado (sem as *stopwords)*.

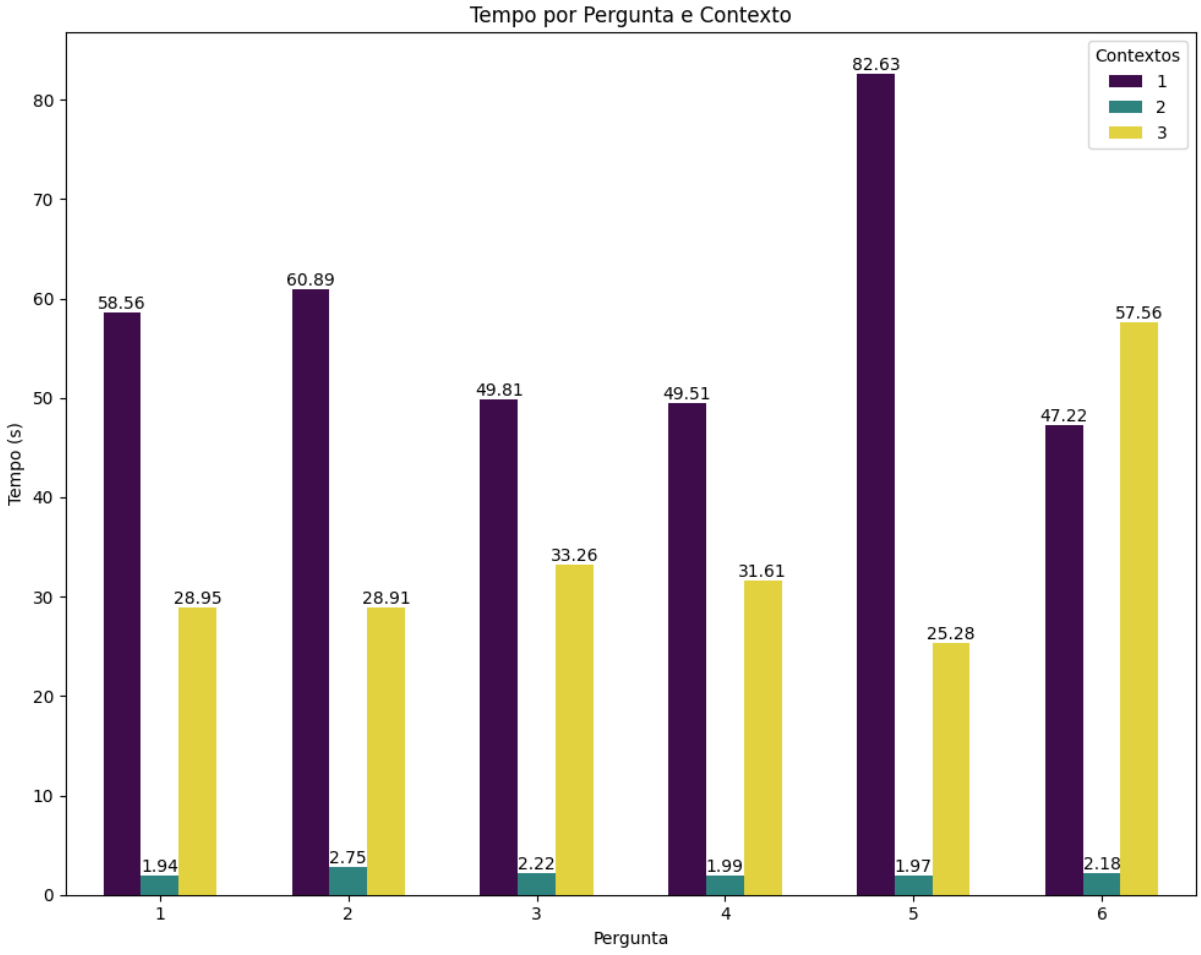

Figura 6: Desempenho de resposta para cada contexto e pergunta

Dado isso o tempo de resposta retornado está representado na Figura 6. Conforme esperado as respostas apresentaram melhor desempenho no contexto resumido, enquanto as outros dois contextos apresentaram altos tempos de resposta devido ao tamanho. O Contexto 3 é no geral mais rápido que o Contexto 1, com exceção na última pergunta na qual o Contexto 1 possuiu um desempenho melhor. Todavia o modelo não executou a tarefa como foi solicitado para a pergunta 6.

A Pergunta 6 envolve sumarização, contagem e ranqueamento de autores, para o Contexto 1, o modelo apenas sumarizou. Embora ele tenha performado melhor, não executou a tarefa corretamente.

6 CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo investigar a capacidade de modelos de linguagem de grande escala (MLLE), com foco no Gemini Flash 2.0, em fornecer respostas completas e precisas a partir de diferentes versões de contexto — completo, resumido e tratado — oriundos de mensagens reais do grupo de WhatsApp da Câmara de Graduação da UEFS.

Os resultados indicam que, embora os MLLE demonstrem boa performance em tarefas de resposta a perguntas diretas, sua completude e confiabilidade ainda são limitadas em contextos onde há necessidade de contabilização, agrupamento e combinação de múltiplas informações. Observou-se, por exemplo, dificuldades na contagem correta de participações, identificação de resoluções CONSEPE e distinção entre informações ruidosas e relevantes em contextos longos.

Além disso, os experimentos mostraram que o desempenho do modelo varia conforme a forma de apresentação do contexto. Enquanto resumo acelera o tempo de resposta, não oferecem informações suficientes para respostas satisfatórias. Já o contexto completo proporciona maior riqueza informacional, mas pode induzir o modelo a erros de interpretação ou alucinação, especialmente em tarefas que exigem precisão quantitativa, além de ter um alto tempo de processamento.

O contexto filtrado, apresentou melhoria significativa em algumas tarefas, e forneceu respostas mais condizentes com as perguntas com menores tempos de resposta. Todavia para contabilizar as resoluções a filtragem gerou uma perda de precisão. Não superou problemas que exigem precisão quantitativa. Ou seja o Gemini Flash 2.0 ainda não superou dificuldades características da própria arquitetura do modelo.

Em termos práticos, os resultados reforçam a importância de curadoria do contexto antes da entrada ao modelo, bem como o papel indispensável de validação humana em cenários críticos. Modelos menores ajustados para tarefas específicas ou estratégias como RAG (Recuperação Aumentada por Geração) continuam sendo alternativas relevantes a serem exploradas para aplicações que exigem maior precisão e menor custo computacional.

Como trabalhos futuros, sugere-se:

- Investigar outras formas de obtenção de amostras;
- Avaliar o desempenho de outras MLLE generalistas;
- Avaliar o desempenho de MLLE especializados e/ou ajustados para domínios acadêmicos ou administrativos;

- Investigar técnicas híbridas de RAG com curadoria automática de contexto;
- Explorar métricas objetivas e automáticas de completude de resposta.

Por fim, este estudo contribui para a reflexão sobre os limites e possibilidades da aplicação de MLLE em ambientes institucionais reais, oferecendo subsídios técnicos e metodológicos para futuras pesquisas e implementações mais eficientes e responsáveis.

REFERÊNCIAS

ACHIAM, J. *et al.* Gpt-4 technical report. **arXiv preprint arXiv:2303.08774**, 2023.

BERRYMAN, J.; ZIEGLER, A. **Prompt Engineering for LLMs: The Art and Science of Building Large Language Model–Based Applications**. [S.l.]: "O'Reilly Media, Inc.", 2024.

BROWN, T. B. *et al.* Language models are few-shot learners. In: **Proceedings of the 34th International Conference on Neural Information Processing Systems**. Red Hook, NY, USA: Curran Associates Inc., 2020. (NIPS '20). ISBN 9781713829546.

CAO, Y. *et al.* A survey of ai-generated content (aigc). **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 57, n. 5, jan. 2025. ISSN 0360-0300.

CASELI, H. d. M.; NUNES, M. d. G. V. Processamento de linguagem natural: conceitos, técnicas e aplicações em português. 2024.

CHEN, J. *et al.* Benchmarking large language models in retrieval-augmented generation. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2024. v. 38, n. 16, p. 17754–17762.

FAN, W. *et al.* A survey on rag meeting llms: Towards retrieval-augmented large language models. In: **Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining**. [S.l.: s.n.], 2024. p. 6491–6501.

FERREIRA, A. B. de H. *et al.* **Mini Aurélio: o dicionário da língua portuguesa**. [S.l.]: Positivo, 2010.

HABIB, F. *et al.* Navigating pathways to automated personality prediction: a comparative study of small and medium language models. **Frontiers in Big Data**, Frontiers Media SA, v. 7, p. 1387325, 2024.

HAN, T. *et al.* Medalpaca–an open-source collection of medical conversational ai models and training data. **arXiv preprint arXiv:2304.08247**, 2023.

HAYKIN, S. **Redes neurais: princípios e prática**. [S.l.]: Bookman Editora, 2001.

HSIEH, C.-Y. *et al.* Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. **arXiv preprint arXiv:2305.02301**, 2023.

JURAFSKY, D.; MARTIN, J. H. **Speech and language processing. Vol. 3**. [S.l.]: Pearson London London, 2014.

LI, J. *et al.* Pre-trained language models for text generation: A survey. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 56, n. 9, abr. 2024. ISSN 0360-0300.

LI, Y. *et al.* Improving entity recognition using ensembles of deep learning and fine-tuned large language models: A case study on adverse event extraction from vaers and social media. **Journal of Biomedical Informatics**, Elsevier, p. 104789, 2025.

LI, Y. *et al.* Large language models in finance: A survey. In: **Proceedings of the Fourth ACM International Conference on AI in Finance**. New York, NY, USA: Association for Computing Machinery, 2023. (ICAIF '23), p. 374–382. ISBN 9798400702402. Disponível em: <https://doi.org/10.1145/3604237.3626869>.

LUCENA, A. d. L. *et al.* Utilizando extração de relação entre entidades para detecção de informações pessoais sensíveis em português. Universidade Federal de Campina Grande, 2024.

MALLADI, S. *et al.* Fine-tuning language models with just forward passes. **Advances in Neural Information Processing Systems**, v. 36, p. 53038–53075, 2023.

MAO, Y. *et al.* A survey on lora of large language models. **Frontiers of Computer Science**, Springer, v. 19, n. 7, p. 197605, 2025.

MELLO, C. E. R. *et al.* Avaliação de grandes modelos de linguagem na extração de informações clínica. **Journal of Health Informatics**, v. 16, n. Especial, 2024.

PARANHOS, S. L. *et al.* Avaliação do impacto de diferentes padrões arquiteturais rag em domínios jurídicos. In: SBC. **Escola Regional de Informática de Goiás (ERI-GO)**. [S.l.], 2024. p. 99–108.

RAIAAN, M. A. K. *et al.* A review on large language models: Architectures, applications, taxonomies, open issues and challenges. **IEEE Access**, IEEE, 2024.

TAY, Y. *et al.* Efficient transformers: A survey. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 55, n. 6, dez. 2022. ISSN 0360-0300.

VASWANI, A. *et al.* Attention is all you need. **Advances in neural information processing systems**, v. 30, 2017.

XU, C. *et al.* Small models are valuable plug-ins for large language models. **arXiv preprint arXiv:2305.08848**, 2023.

ZHAO, W. X. *et al.* A survey of large language models. **arXiv preprint arXiv:2303.18223**, 2023.