UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Eduardo Santos Pires

Avaliação de Modelos de Processamento de Linguagem Natural para Análise de Sentimentos em Avaliações de Restaurantes

Uberlândia, Brasil

2026

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Eduardo Santos Pires

Avaliação de Modelos de Processamento de Linguagem Natural para Análise de Sentimentos em Avaliações de Restaurantes

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Orientador: Profa. Dra. Fernanda Maria da Cunha Santos

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Sistemas de Informação

Uberlândia, Brasil

2026

Eduardo Santos Pires

Avaliação de Modelos de Processamento de Linguagem Natural para Análise de Sentimentos em Avaliações de Restaurantes

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Trabalho aprovado. Uberlândia, Brasil, 26 de maio de 2026:

Profa. Dra. Fernanda Maria da Cunha Santos
Orientador

Profa. Dra. Ana Cláudia Martinez

Prof. Dr. Ronaldo Castro de Oliveira

Uberlândia, Brasil
2026

Dedico este trabalho à minha mãe, que sempre apoiou minha jornada acadêmica, aos meus amigos que me acompanharam em momentos de desafio e de alegria e a cada uma das pessoas que tornaram este trabalho possível.

Agradecimentos

Agradeço especialmente à minha orientadora, **Profa. Dra. Fernanda Maria da Cunha Santos**, cujo acompanhamento, orientação e incentivo me permitiram continuar com este trabalho, mesmo nos momentos mais desafiadores.

Aos membros da banca avaliadora, à **Profa. Dra. Ana Cláudia Martinez** e ao **Prof. Dr. Ronaldo Castro de Oliveira**, cuja contribuição trouxe um maior refinamento do texto, necessário para sua publicação.

À minha mãe **Silvéria**, cujo suporte permitiu que eu me dedicasse à minha vida acadêmica e que me deu apoio em cada momento, sendo a pessoa que me ensinou o significado de força, mostrou-me o valor dos estudos e em quem me inspiro como força motriz.

À minha tia **Vanuse**, cuja atenção, carinho e presença em toda a minha jornada foram de uma preciosidade sem igual.

Aos meus amigos, **Victor Ricarte Silva** e **José Lucas Ferreira de Lima**, que me apoiaram por anos, ao lado dos quais vivi batalhas e dei risadas, e com quem, desde o começo do curso, pude contar.

À equipe do Centro de Tecnologia da Informação e Comunicação da UFU, cujas dicas e apoio foram de grande valia na etapa final de criação desta monografia, em especial ao Coordenador da Divisão de Redes, **Eduardo Nascimento de Souza Rolim**, e ao técnico administrativo **Augusto Carvalho Soares**.

À coordenação, aos técnicos, aos professores, aos alunos e a cada pessoa que se dedica diariamente para que a Universidade Federal de Uberlândia, a Faculdade de Computação e, por fim, o curso de Sistemas de Informação existam e continuem funcionando.

"Não é porque as coisas são difíceis que não ousamos; é porque não ousamos que elas são difíceis."

(Sêneca)

Resumo

O setor de alimentação possui um impacto significativo na economia brasileira e, com a digitalização da forma como a sociedade procura saber sobre estabelecimentos, plataformas de avaliação como o Google Maps tornaram-se determinantes na percepção e decisão dos consumidores. No entanto, a simples visualização das notas gerais em estrelas mascara nuances importantes do sentimento do cliente e dificulta a extração de métricas detalhadas em grandes volumes de texto. Diante dessa lacuna, este trabalho avalia o desempenho de diferentes modelos de Processamento de Linguagem Natural na tarefa de análise de sentimentos em comentários de restaurantes no Google Maps. Para isso, a metodologia consistiu na coleta automatizada de dados (*Web Scraping*) em páginas de estabelecimentos no Google Maps, seguida pelo processamento da base de avaliações textuais utilizando os modelos BERT, XLM-RoBERTa e LeIA. Os resultados demonstraram que o modelo BERT apresentou a melhor adequação para comparações diretas com as notas dos usuários. O XLM-RoBERTa, por sua vez, demonstrou alta confiabilidade, embora sua classificação categórica tenha limitado a precisão em comparações numéricas. Já o LeIA, apesar de utilizar uma abordagem léxica mais clássica, produziu resultados consistentes e comparáveis aos demais. Além disso, a análise revelou uma disparidade comportamental: a polaridade do sentimento extraído exclusivamente dos textos tendeu a ser inferior à nota geral do estabelecimento. Conclui-se que a aplicação de modelos de Processamento de Linguagem Natural fornece métricas complementares que podem vir a ser úteis para a gestão de negócios gastronômicos, superando as limitações analíticas das classificações baseadas apenas em estrelas.

Palavras-chave: Processamento de Linguagem Natural, Análise de Sentimentos, Avaliações Online, Restaurantes, Modelos de Linguagem.

Lista de ilustrações

Lista de abreviaturas e siglas

ABNT	Associação Brasileira de Normas Técnicas
API	*Application Programming Interface*
BERT	*Bidirectional Encoder Representations from Transformers*
BSI	Bacharelado em Sistemas de Informação
C^3E	*Consensus between Classification and Clustering Ensembles*
CAPTCHA	*Completely Automated Public Turing test to tell Computers and Humans Apart*
CDP	*Chrome DevTools Protocol*
CPU	*Central Processing Unit*
CRISP-DM	*Cross-Industry Standard Process for Data Mining*
CSS	*Cascading Style Sheets*
DOM	Modelo de Objeto de Documentos (*Document Object Model*)
GPT-3	*Generative Pre-trained Transformer 3*
HTML	*HyperText Markup Language*
HTTP	*HyperText Transfer Protocol*
IA	Inteligência Artificial
IE	Extração de Informações (*Information Extraction*)
IP	*Internet Protocol*
JS	*JavaScript*
JSON	*JavaScript Object Notation*
LDA	*Latent Dirichlet Allocation*
LeIA	Léxico para Inferência Adaptada
PIB	Produto Interno Bruto
PLN	Processamento de Linguagem Natural

Regex	Expressões Regulares (*Regular Expressions*)
RU	Restaurante Universitário
SVM	*Support Vector Machine*
TF-IDF	*Term Frequency-Inverse Document Frequency*
UFU	Universidade Federal de Uberlândia
URL	*Uniform Resource Locator*
UTF-8	*8-bit Unicode Transformation Format*
VADER	*Valence Aware Dictionary and sEntiment Reasoner*
WebRTC	*Web Real-Time Communication*
XLM-RoBERTa	*Cross-lingual Language Model - Robustly Optimized BERT Pre-training Approach*
XPath	*XML Path Language*

Sumário

1 Introdução

O setor de alimentação tem um impacto significativo na economia brasileira, representando cerca de 10,8% do PIB do país em 2024, o que justifica o aumento de 10,4% do mercado de alimentação fora do lar (ABIA, 2025). Outro setor que está em constante crescimento no mundo todo é o digital e, para o setor gastronômico também seguir nesta tendência, os restaurantes que trabalham tanto por delivery quanto pelo atendimento local próprio precisam se manter atentos às inovações e às atualizações do mercado.

Uma das mudanças que a tecnologia trouxe para negócios locais foi a popularização dos comentários deixados em sites de *review*. Em 2022, 87% dos consumidores utilizaram o Google para avaliar estabelecimentos locais, em pesquisa levantada pela (BRIGHTLOCAL, 2023), demonstrando a influência dessas plataformas na percepção e decisão dos clientes. No setor alimentício, onde a experiência do consumidor é um fator determinante, essas avaliações desempenham um papel ainda mais relevante. No entanto, a simples visualização dos comentários pode dificultar a obtenção de *insights* úteis para melhorias, tornando necessária a categorização e a análise dos feedbacks positivos e negativos.

O trabalho (MORENO, 2015) explora a análise de sentimentos para classificar comentários online, utilizando como base a plataforma Yelp. Para isso, foram extraídos 14.000 comentários sobre produtos turísticos, que passaram por um processo de organização em tópicos e identificação dos termos mais frequentes. Seguindo a metodologia CRISP-DM, aplicou-se a análise de sentimentos para distinguir emoções positivas, negativas e neutras, além da avaliação da utilidade dos comentários. Os resultados obtidos demonstram a influência das avaliações online na percepção dos consumidores e ressaltam a importância da mineração de texto para interpretar grandes volumes de dados de forma eficiente.

Já o trabalho (FARIAS; ANGELUCI; PASSARELLI, 2021) explora a aplicação da ciência de dados e do *Web Scraping* (uma técnica computacional de coleta automatizada que utiliza *scripts* para varrer e extrair grandes volumes de dados não estruturados diretamente de páginas da internet, convertendo-os em formatos estruturados para análise) em resenhas do Google Maps. Este estudo faz a constatação de que usuários da plataforma percebem impactos éticos e de privacidade, tornando-se mais conscientes sobre suas avaliações. Além disso, a pesquisa demonstra que as avaliações *online* influenciam o comportamento do consumidor, criando padrões de percepção sobre os locais e evidenciando que essa fonte de dados possui boa confiabilidade.

1.1 Justificativa

Sistemas de avaliação em plataformas online utilizam classificações em estrelas para mensurar a satisfação do consumidor. Essa métrica numérica, contudo, omite a polaridade expressa nos comentários. Textos associados a notas de valor máximo podem registrar insatisfação, assim como notas de valor mínimo podem abrigar aprovação. Essa divergência demonstra a insuficiência do indicador quantitativo para representar o sentimento do usuário de forma exata. Adicionalmente, o volume de dados gerado impede a categorização manual das opiniões.

Assim, o estudo avalia e compara arquiteturas computacionais na classificação de polaridade de textos informais extraídos da Web, que indicam a satisfação ou não. A aplicação desses modelos automatiza a identificação de opiniões positivas, neutras e negativas. Esse processamento metodológico gera indicadores de sentimento que complementam a limitação das estrelas, entregando bases de dados estruturadas que fundamentam o processo de tomada de decisão.

1.2 Objetivo

O principal objetivo deste trabalho é avaliar o desempenho de diferentes modelos de Processamento de Linguagem Natural na tarefa de análise de sentimentos, utilizando como base os comentários de restaurantes presentes no Google Maps. De forma semelhante ao estudo de Moreno (2015), que demonstrou a influência das avaliações online na percepção dos consumidores, busca-se empregar as mesmas abordagens algorítmicas para interpretar as opiniões dos usuários no contexto gastronômico, comparando a eficácia e as particularidades de modelos distintos (como BERT, XLM-RoBERTa e LeIA). Para isso, serão desenvolvidos scripts para a pré-processamento, processamento e visualização de dados, permitindo a identificação de padrões e demonstrando o valor prático dessa análise para a gestão dos restaurantes.

1.2.1 Objetivos Específicos

Para alcançar o objetivo geral proposto, estabelecem-se os seguintes objetivos específicos:

- Desenvolver um script automatizado de *Web Scraping* para extrair informações textuais e notas de restaurantes no Google Maps;
- Processar os dados extraídos, aplicando diferentes modelos de Processamento de Linguagem Natural para a tarefa de análise de sentimentos;

- Comparar o desempenho e as particularidades dos modelos aplicados, avaliando suas métricas, limitações e sua adaptabilidade à base tratada;
- Elaborar visualizações gráficas dos dados processados para facilitar a identificação de padrões e evidenciar a aplicabilidade gerencial das métricas obtidas.

1.3 Organização do Trabalho

- **O Capítulo 2** apresenta a fundamentação teórica que embasa esta pesquisa, abrangendo as principais áreas de estudo utilizadas em sua execução, bem como a revisão dos principais trabalhos correlatos.
- **O Capítulo 3** descreve as etapas de desenvolvimento da metodologia, detalhando as abordagens, técnicas e tecnologias empregadas na construção do projeto.
- **O Capítulo 4** por sua vez, apresenta os resultados obtidos, ilustrando desde a estrutura dos dados pré-processados até os gráficos resultantes acompanhados de suas respectivas análises.
- **O Capítulo 5** conclui o trabalho, avaliando o êxito no alcance dos objetivos gerais e específicos propostos, além de apresentar sugestões de abordagens e técnicas que podem ser exploradas em trabalhos futuros.

1.4 Uso de Inteligência Artificial Generativa

Os Modelos de linguagem foram utilizados como assistentes de programação. O uso da IA concentrou-se no levantamento de definições técnicas e no suporte à codificação, principalmente na identificação de erros, nos ajustes de sintaxe e na completação automática de código. Todo o código gerado com o auxílio de IA foi submetido a testes e adaptações manuais, visando ao alinhamento com os objetivos analíticos da pesquisa.

2 Fundamentação Teórica e Revisão Bibliográfica

Os temas principais envolvidos neste trabalho baseia-se nos conceitos fundamentais: Processamento de Linguagem Natural (PLN), *Web Scraping*, Análise de Sentimentos e Modelos de Linguagem. Esses elementos permitem a coleta, o processamento e a interpretação de avaliações de usuários, possibilitando uma compreensão mais aprofundada das percepções sobre os estabelecimentos. Neste sentido, este capítulo trará conceitos básicos e fundamentais, além de uma breve descrição dos principais trabalhos científicos que fundamentaram este estudo.

2.1 Fundamentação Teórica

A presente seção estabelece a fundamentação teórica necessária para a compreensão deste estudo, estruturando-se em quatro tópicos principais. Inicialmente, a subseção de Processamento de Linguagem Natural (PLN) apresenta os conceitos basilares para a interpretação computacional de textos. Na sequência, a subseção de *Web Scraping* detalha a técnica de coleta automatizada de dados orgânicos. A subseção de Análise de Sentimentos explora as metodologias aplicadas para classificar a polaridade de opiniões. Por fim, uma breve descrição dos Modelos de Linguagens implementados.

2.1.1 Processamento de Linguagem Natural (PLN)

(CASELI; NUNES; PAGANO, 2023) definem Processamento de Linguagem Natural como um campo de pesquisa que tem como objetivo investigar e desenvolver métodos e sistemas para o processamento computacional da linguagem humana. Essa área busca criar soluções para a análise e interpretação de textos, permitindo que máquinas lidem com a linguagem de forma automatizada.

A história do Processamento de Linguagem Natural (PLN) remonta aos anos 1950, quando Alan Turing propôs o Teste de Turing (TURING, 1950) para avaliar a inteligência artificial com base na capacidade de uma máquina imitar a comunicação humana. Para isso, a máquina precisaria não apenas compreender a linguagem, o que é tornado possível por meio da PLN, mas também gerar respostas coerentes e relevantes, a partir do que hoje conhecemos como Geração de Linguagem Natural.

Nos anos 1980 e 1990, um avanço significativo no campo do PLN ocorreu com o desenvolvimento de técnicas estatísticas que passaram a ser amplamente utilizadas para

extrair dados de grandes volumes textuais. A consolidação dessas bases e o uso de modelos probabilísticos, como o Modelo Oculto de Markov (cuja estrutura representativa é ilustrada na Figura 1) (RABINER, 1989), tornaram possível identificar padrões e relações semânticas em grandes conjuntos de dados. Essas abordagens não apenas aprimoraram a precisão na análise e interpretação da linguagem, mas também possibilitaram a transformação de textos não estruturados em informações estruturadas, facilitando a aplicação de técnicas de mineração de dados e extração de informações (IE).

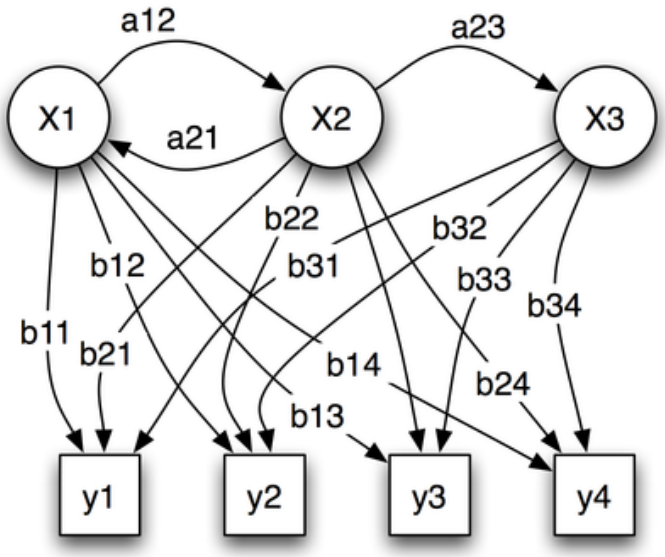

Figura 1 – Representação estrutural dos estados e transições de um Modelo Oculto de Markov.

Fonte: (Wikimedia Commons, 2024)

Atualmente, tecnologias baseadas em Processamento de Linguagem Natural estão presentes em diversos serviços e dispositivos, desde assistentes virtuais como Amazon Alexa[1] até algoritmos de recomendação em plataformas como YouTube[2], motores de busca como Google[3] e aplicações avançadas de IA como o Stable Diffusion[4]. Sua utilidade vai muito além do contexto acadêmico ou de uso exclusivo em tecnologias específicas, adaptando-se de maneira notável a diversos nichos e proporcionando soluções práticas em áreas como saúde, finanças, negócios, educação e marketing.

[1] (AMAZON, 2025)
[2] (GOOGLE, 2025b)
[3] (GOOGLE, 2025a)
[4] (STABILITYAI, 2025)

2.1.2 WebScraping

O (Cambridge Dictionary, 2025) define *Web Scraping* como "a atividade de pegar informações de um site ou tela de computador e colocá-las em um documento ordenado em um computador". Esta técnica permite que pesquisadores e desenvolvedores possam coletar dados do mundo real com facilidade de fontes orgânicas de informação, permitindo assim um amplo leque de pesquisa.

Apesar de sua ampla utilidade, a técnica suscita debates operacionais e legais consideráveis, dado o seu potencial para causar instabilidade técnica nos sistemas que atuam como fontes de dados. Essa dinâmica de sobrecarga é ilustrada pelo litígio relatado em (Jeffrey Neuburger, 2014), referente ao caso judicial entre a rede de varejo norte-americana QVC e o aplicativo de compras Resultly. Na ocasião, a companhia processou o aplicativo sob a alegação de que seus algoritmos de *Web Scraping* operaram de forma excessivamente agressiva para extrair dados de preços em tempo real. O envio automatizado de dezenas de milhares de requisições por minuto esgotou a capacidade dos servidores da QVC, resultando na queda do site por dois dias e em perdas financeiras significativas. Esse caso tornou-se um marco nas discussões sobre os limites éticos e legais da mineração na internet, evidenciando que a extração automatizada, quando desprovida de limitadores de velocidade (conhecidos como *crawl delays*) e executada sem o consentimento da infraestrutura alvo, ultrapassa a barreira da pesquisa e adentra o escopo de abuso computacional.

Atualmente, a *web scraping* é mais utilizada do que nunca, se destacando, por exemplo, em aplicações de Inteligência Artificial. Podendo-se citar o Perplexity AI (PERPLEXITY AI, 2026), um motor de busca conversacional que acessa a internet e realiza a extração de conteúdo de páginas da web em tempo real para coletar informações atualizadas de diversas fontes e fundamentar suas respostas.

Esse aumento no uso do *Web Scraping* foi impulsionado pelos avanços em inteligência artificial e processamento de linguagem natural, tornando a extração de dados mais eficiente e acessível. Ferramentas como Scrapy[5], BeautifulSoup[6] e Selenium[7] permitem a automação do processo, enquanto técnicas avançadas de evasão, como o uso de proxies rotativos (MITCHELL, 2018) e a resolução automática de CAPTCHAs (SIVAKORN; POLAKIS; KEROMYTIS, 2016), ajudam a contornar as restrições de segurança impostas pelos servidores.

[5] (Scrapy Developers, 2025)
[6] (Leonard Richardson, 2025)
[7] (Selenium Developers, 2025)

2.1.3 Análise de Sentimentos

A análise de sentimentos, ou mineração de opinião, é definida por Silva (2016, p. 3) como um campo emergente e multidisciplinar que integra conceitos de mineração de dados, aprendizado de máquina, linguística e Processamento de Linguagem Natural (PLN). O objetivo central dessa disciplina é analisar fragmentos textuais para determinar a atitude, emoção ou opinião do escritor em relação a um tópico específico. Na prática, essa investigação visa extrair sentimentos de textos sobre entidades como produtos e serviços, identificando a polaridade das opiniões para revelar se o posicionamento do usuário é positivo, negativo ou neutro, conforme ilustrado na Figura 2.

Figura 2 – Visualização de Análise de Sentimento.

Fonte: (BINDS.CO, 2023)

Desde o início dos anos 2000, a análise de sentimentos continua sendo uma das principais áreas em PLN. Em 2006, a análise de sentimentos ganhou destaque devido ao crescimento das redes sociais, que facilitaram o acesso às opiniões dos usuários. Plataformas como Facebook e Twitter se tornaram espaços de expressão pública, gerando grandes volumes de dados textuais. Isso impulsionou a necessidade de técnicas automáticas para processar e extrair informações a partir das opiniões subjetivas compartilhadas online (SAMPAIO, 2021; LIU, 2010).

A análise de sentimentos tem evoluído com o avanço de modelos de aprendizado profundo e bibliotecas avançadas de PLN, como a Transformers[8], que oferecem *pipelines* eficientes para a aplicação de redes neurais na extração de sentimentos. Um fator determinante para esse progresso foi o desenvolvimento de *embeddings*, que são representações vetoriais densas onde palavras ou sentenças são mapeadas em um espaço multidimensional. Diferente de técnicas estatísticas simples, os *embeddings* gerados por arquiteturas como BERT[9] e GPT-3[10] são contextuais, o que significa que o vetor numérico de uma

[8] (WOLF et al., 2020)
[9] (DEVLIN et al., 2019)
[10] (BROWN et al., 2020)

palavra se altera conforme o seu entorno na frase. Essa capacidade de codificar relações semânticas complexas aprimorou a detecção de nuances emocionais, permitindo identificar expressões subjetivas, sarcasmo e ironia com maior precisão do que os métodos baseados em correspondência exata de termos.

2.2 Modelos de Linguagem para Análise de Sentimentos

Após o levantamento das metodologias disponíveis na literatura, selecionaram-se três modelos computacionais para a execução deste estudo: BERT, XLM-RoBERTa e LeIA. A seleção fundamenta-se na necessidade metodológica de contrapor paradigmas distintos. O objetivo estrutural dessa escolha é avaliar o desempenho de redes neurais profundas, de arquitetura generalista e especializada, em comparação a métodos determinísticos baseados em dicionários, validando a relação entre custo computacional e precisão na extração de polaridade em textos informais.

2.2.1 BERT Multilingual

O modelo BERT Multilingual (NLP Town, 2020) foi selecionado para atuar como a principal referência (*baseline* neural) de aprendizado profundo deste estudo, visto que representa o padrão estabelecido na literatura para compreensão de texto. A justificativa para sua adoção reside na sua arquitetura *Transformer*, que substitui o processamento sequencial clássico por mecanismos de autoatenção. Essa estrutura viabiliza a atenção bidirecional, avaliando simultaneamente o contexto à esquerda e à direita de um termo para extrair o seu significado (DEVLIN et al., 2018). Em avaliações gastronômicas, onde a polaridade de um adjetivo pode ser invertida por uma palavra distante na mesma frase, a geração desses *embeddings* contextuais é necessária para evitar falsos positivos.

Adicionalmente, a versão multilíngue escolhida justifica-se pelo uso da tokenização por subpalavras (*WordPiece*). Essa técnica fragmenta palavras desconhecidas em unidades morfológicas menores já mapeadas pelo modelo, o que mitiga falhas de processamento causadas por erros ortográficos comuns na internet.

A métrica de confiança (*score*) nesta arquitetura é extraída da última camada da rede neural. Os valores brutos numéricos (*logits*) são submetidos à função de ativação *Softmax*, que os converte em uma distribuição de probabilidade para as classes disponíveis (escala de 1 a 5 estrelas). O cálculo para a classe i é definido pela Equação 2.1:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{2.1}$$

Onde z_i representa o *logit* da classe i, e K corresponde ao número total de categorias. O modelo assume como predição final a classe com a maior probabilidade calculada.

2.2.2 XLM-RoBERTa

O XLM-RoBERTa (*Cross-lingual Language Model - Robustly Optimized BERT Pretraining Approach*) (CardiffNLP, 2021) foi incorporado à metodologia para avaliar o impacto da especialização de domínio e da otimização de treinamento no desempenho da Inteligência Artificial. Tecnicamente, o modelo consiste em uma evolução da arquitetura BERT, submetida a um processo de *Robustly Optimized*. Esse termo refere-se a uma modificação estratégica na receita de treinamento do modelo, que envolve a execução por períodos mais prolongados, a utilização de lotes de dados (*batch sizes*) maiores e a ampliação considerável do volume total de dados de treinamento em comparação à abordagem original do BERT.

A seleção deste modelo justifica-se pela sua robustez linguística, especialmente na versão adaptada para mídias sociais que utiliza bases de dados provenientes do Twitter (CAMACHO-COLLADOS; REZAEE; AL., 2022). O treinamento prévio com esse acervo textual adequou os parâmetros da rede para o processamento de gírias, abreviações e ambiguidades estruturais características de comentários onde existe menor rigor com a linguagem formal. O cálculo do nível de confiança (*score*) preditivo segue a mesma base matemática da função *Softmax* demonstrada anteriormente, sendo dimensionado para as três categorias de saída desta implementação: negativa, neutra e positiva.

2.2.3 LeIA

Para contrastar com as arquiteturas de redes neurais (que exigem elevado poder de processamento de máquina e maior tempo de latência), o LeIA (*Léxico para Inferência Adaptada*) (ANDRADE, 2018) foi selecionado para atuar como uma linha de base baseada em regras gramaticais. A justificativa para sua aplicação é validar se métodos de baixo custo computacional são suficientes para suprir a demanda analítica de comentários de restaurantes.

O modelo opera sob o paradigma léxico, fundamentado no sistema VADER (*Valence Aware Dictionary and sEntiment Reasoner*), adaptado para o português. O funcionamento depende de um dicionário onde cada palavra catalogada possui uma valência emocional associada. No contexto do Processamento de Linguagem Natural, a valência refere-se ao peso numérico intrínseco que quantifica a carga de positividade ou negatividade de um termo isolado (HUTTO; GILBERT, 2014). A essas pontuações lexicais, aplicam-se regras algorítmicas estáticas, como a inversão matemática de polaridade diante de termos de negação.

A métrica central adotada é o *compound*, um índice normalizado que representa a soma das valências ajustadas dos termos na sentença. A normalização é calculada pela Equação 2.2:

$$x = \frac{\sum v}{\sqrt{(\sum v)^2 + \alpha}} \tag{2.2}$$

Onde $\sum v$ corresponde à somatória das valências avaliadas no texto e α é uma constante matemática de suavização estabelecida como 15. O resultado x consiste em uma métrica contínua delimitada no intervalo de -1 (totalmente negativo) a +1 (totalmente positivo). Por fundamentar-se em somatórias determinísticas baseadas em léxico em vez de probabilidades estatísticas, o LeIA não fornece um *score* de confiança preditiva, entregando diretamente a intensidade estrutural calculada do sentimento.

2.3 Trabalhos Relacionados

O trabalho de Shin et al. (2022) tem como objetivo analisar avaliações de restaurantes no Google Maps por meio de técnicas de mineração de texto e processamento de linguagem natural (PLN), com foco na análise de sentimentos. Inicialmente, os autores coletaram 5.427 avaliações em coreano utilizando *Data Crawling* via Selenium, uma ferramenta de automação computacional que varre e extrai os textos diretamente da página web. Em seguida, procedeu-se à etapa de pré-processamento (*Preprocessing*) para a limpeza do conjunto de dados, removendo ruídos linguísticos, como pontuações e *stopwords*. Para viabilizar a análise computacional, o texto limpo foi convertido em valores numéricos por meio da vetorização TF-IDF (*Term Frequency - Inverse Document Frequency*), uma técnica que atribui um peso matemático a cada palavra ao balancear a sua frequência em uma avaliação específica contra a sua raridade em todo o acervo de textos. Por fim, a classificação dos sentimentos foi realizada aplicando-se um algoritmo de *Random Forest* (Floresta Aleatória), um método de aprendizado de máquina que constrói múltiplas árvores de decisão para inferir a polaridade do texto por meio de votação majoritária. Como principais resultados, destacam-se a criação de um dicionário de termos categorizados em quatro critérios (comida, preço, serviço e atmosfera) e a identificação de padrões linguísticos associados a avaliações positivas e negativas.

Esse trabalho relaciona-se diretamente ao presente projeto, uma vez que ambos investigam avaliações de estabelecimentos gastronômicos no Google Maps, empregando métodos análogos de coleta automatizada de dados (*Web Scraping*). No entanto, enquanto os autores utilizaram técnicas clássicas de aprendizado de máquina estatístico, a presente pesquisa avança metodologicamente ao aplicar arquiteturas de redes neurais profundas pré-treinadas (BERT e XLM-RoBERTa) e abordagens léxicas contemporâneas (LeIA), permitindo capturar nuances semânticas mais complexas e adaptadas para o idioma português.

O trabalho de Silva (2019) tem como objetivo principal propor um modelo capaz

de identificar assuntos e analisar sentimentos em comentários em português, utilizando técnicas de mineração de texto e processamento de linguagem natural (PLN). O autor aplicou o modelo em avaliações de restaurantes extraídas do TripAdvisor, empregando o algoritmo LDA (*Latent Dirichlet Allocation*) para a extração de tópicos. O LDA é um modelo estatístico probabilístico que identifica automaticamente temas latentes (ocultos) em grandes volumes textuais, agrupando palavras que coocorrem com alta frequência. Para a classificação dos sentimentos, foram utilizados algoritmos como o *Naive Bayes* e o SVM (*Support Vector Machine*), um método supervisionado de aprendizado de máquina que mapeia os dados em um espaço multidimensional com o intuito de encontrar o hiperplano ideal, ou seja, a fronteira de decisão matemática que melhor separa as diferentes classes (como positivo e negativo). Além disso, o trabalho comparou esses métodos com uma abordagem baseada em dicionários e léxicos de sentimentos estruturados para o idioma português, como o SentiLex (SILVA; CARVALHO; SARMENTO, 2012) e o OpLexicon (SOUZA; VIEIRA, 2012). Os resultados demonstraram que a abordagem com aprendizado de máquina obteve desempenho superior à abordagem puramente léxica, embora tenha enfrentado desafios na identificação de textos negativos, problema que poderia ser mitigado com ajustes de hiperparâmetros e ampliação da base de treinamento.

Esse estudo relaciona-se intimamente com esta monografia, que também visa extrair *insights* de comentários em plataformas *online*. Assim como no trabalho de Silva, este projeto contrapõe o desempenho de técnicas baseadas em léxicos a abordagens de aprendizado de máquina. A grande diferença, contudo, reside na evolução tecnológica adotada: em vez de classificadores tradicionais (SVM) e dicionários mais antigos, esta pesquisa utiliza o léxico otimizado LeIA e modelos avançados de transformadores. Além disso, a dificuldade relatada pelo autor na detecção de textos negativos reforça a relevância do presente trabalho, justificando o uso de ferramentas de Inteligência Artificial mais robustas para lidar com avaliações insatisfatórias disfarçadas de neutralidade ou pontuadas com notas distorcidas.

O trabalho desenvolvido por Silva (2016) tem como objetivo investigar métodos para análise de sentimentos em *tweets*, explorando abordagens supervisionadas e semissupervisionadas. No contexto do aprendizado de máquina, a abordagem semissupervisionada destaca-se por utilizar uma pequena amostra de dados previamente rotulados em conjunto com um grande volume de dados não rotulados, otimizando o treinamento ao reduzir a necessidade de anotação manual exaustiva. O estudo utiliza técnicas como *ensembles* (comitês de modelos), que consistem na combinação estratégica de múltiplos algoritmos de classificação ou agrupamento para gerar predições conjuntas mais robustas e com menor margem de erro do que se atuassem individualmente. Além disso, a pesquisa aplica o algoritmo C^3E (*Consensus between Classification and Clustering Ensembles*) para refinar a inferência de sentimentos. O C^3E opera estabelecendo um consenso: ele ajusta as predições do comitê de classificadores com base nas informações de similaridade extraídas

por modelos não supervisionados (*clusters*), partindo da premissa de que textos estruturalmente semelhantes tendem a compartilhar a mesma polaridade de sentimento. Os resultados mostram que a combinação de classificadores e agrupadores pode melhorar significativamente a robustez e a acurácia da análise, especialmente em cenários com escassez de dados rotulados.

A fundamentação teórica deste trabalho se conecta de forma direta à presente avaliação de modelos, pois ambos lidam com a extração de sentimentos em textos curtos, não estruturados e com alta carga de informalidade. Embora o estudo de Silva (2016) tenha foco no Twitter, a linguagem adotada nessa rede social aproxima-se de forma expressiva da escrita utilizada em avaliações do Google Maps. Inclusive, essa equivalência semântica e estrutural corrobora com a escolha do modelo XLM-RoBERTa nesta monografia, cuja arquitetura foi especificamente treinada sobre bases de dados de *tweets*. Ademais, a necessidade de contornar a ausência de dados previamente rotulados, solucionada pelo autor via *ensembles*, evidencia a importância de se adotar modelos de arquitetura *transformer* pré-treinados, que são capazes de inferir a polaridade sem a exigência de anotações manuais prévias da base de avaliações extraída.

3 Desenvolvimento

Neste capítulo, apresenta-se a metodologia de desenvolvimento do modelo computacional. A Figura 3 ilustra a sequência de passos necessários para que o objetivo do modelo proposto seja atingido. As etapas que merecem destaque são: Coleta de Dados via *Web Scraping*; Análise de Sentimentos com Modelos de IA; e, por fim, Visualização e análise de dados. Todas essas etapas foram executadas utilizando *scripts* em Python, consumindo dados vindos de arquivos JSON.

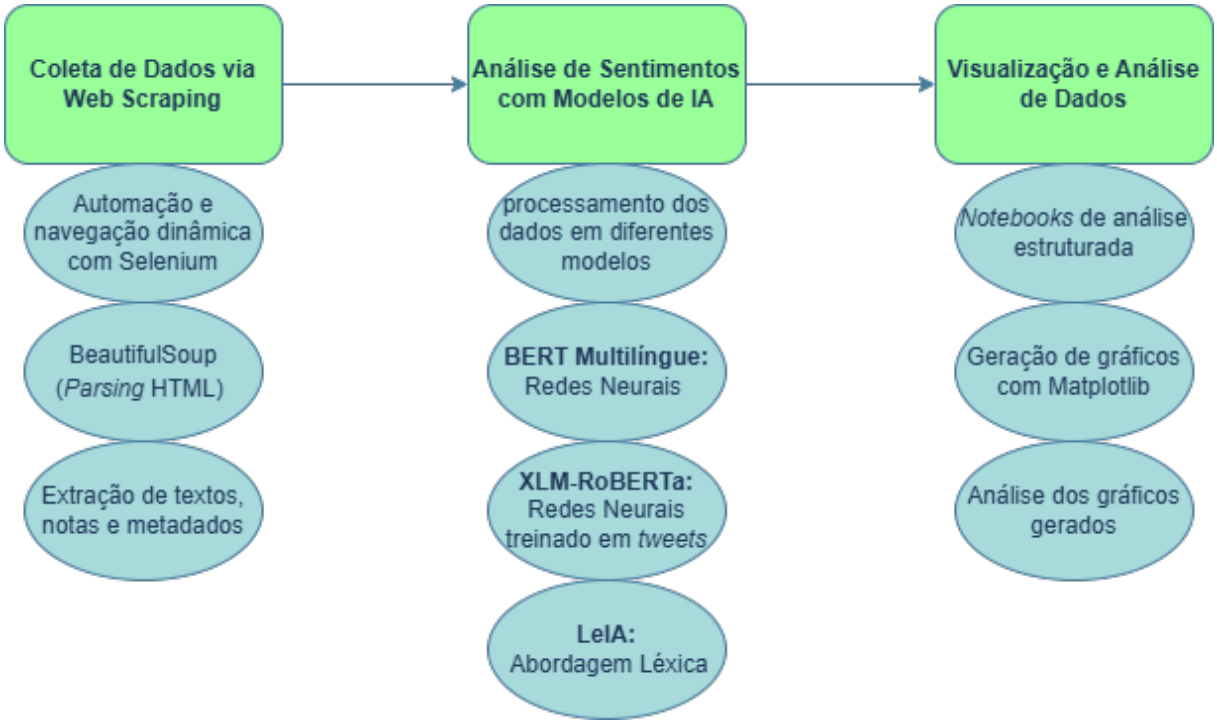

Figura 3 – Organograma das etapas do modelo computacional proposto.

Fonte: Elaborado pelo autor

A tecnologia principal empregada para a execução do projeto foi a linguagem de programação Python[1], adotada devido à disponibilidade de bibliotecas. Na extração de dados, utilizou-se o Selenium[2] para automatizar a navegação e processar o carregamento dinâmico do Google Maps, em conjunto com a biblioteca BeautifulSoup[3], responsável pelo *parsing* do HTML (processo computacional que converte o código-fonte da página web em uma árvore de elementos navegável) e pela extração dos textos. Para organizar os dados

[1] (Python Software Foundation, 2024)
[2] (Selenium Developers, 2026)
[3] (RICHARDSON, 2024)

e viabilizá-los para as próximas fases, os dados foram estruturados no formato JSON[4]. Esses arquivos serão utilizados como dados de entrada para os modelos que irão fazer a análise de sentimentos. Já na etapa de visualização, aplicou-se a biblioteca Matplotlib[5], com destaque para o módulo `pyplot` como a base para a geração dos gráficos comparativos e a análise das métricas, construindo de forma iterativa os eixos, barras e legendas.

3.1 Coleta de Dados via *Web Scraping*

A primeira etapa da metodologia consistiu na coleta automatizada de dados por meio de *Web Scraping* (MITCHELL, 2018), técnica que viabiliza a extração programática de informações de páginas *web*. Para operar em páginas de renderização dinâmica, onde os elementos são carregados assincronamente pelo navegador de forma contínua, o sistema foi desenvolvido na linguagem Python, selecionada devido ao seu suporte nativo a bibliotecas de automação e manipulação de estruturas de dados. A arquitetura do código foi modularizada para segregar responsabilidades, abrangendo a configuração do ambiente, a interação com a interface, a extração dos nós da página, o processamento textual e a persistência final, conforme detalhado na Tabela 1.

Tabela 1 – Módulos da arquitetura do sistema de coleta de dados

Componente Lógico	**Responsabilidade Principal**
Controlador Principal	Orquestração do fluxo de execução e chamada dos submódulos.
Gerenciador de Configurações	Centralização de constantes e seletores do sistema.
Controle de Exclusões	Armazenamento de listas de usuários para evitar redirecionamentos indesejados.
Gerenciador de Sessão	Inicialização do motor de automação com técnicas de evasão de detecção.
Controlador de Navegação	Navegação por URL, cliques em abas e ordenação de avaliações.
Manipulador de Interface	Rolagem de página e expansão de textos truncados via injeção de *scripts*.
Extrator Analítico	*Parsing* do HTML via BeautifulSoup e extração de atributos.
Processador de Texto	Limpeza de metadados e aplicação de atrasos de execução.
Gerenciador de Persistência	Exportação e serialização dos dados coletados no formato JSON.

Fonte: Elaborado pelo autor

[4] (ECMA International, 2017)
[5] (Matplotlib Development Team, 2024)

3.1.1 Configuração e Evasão de Detecção

Devido às políticas anti-automação da plataforma, o módulo gerenciador de sessão do Selenium, *framework* de controle de navegadores, implementou rotinas de evasão para mascarar a origem robótica do tráfego. A primeira estratégia consistiu na rotação do *User-Agent* (FIELDING; NOTTINGHAM; RESCHKE, 2022), um cabeçalho HTTP de identificação do cliente, com o objetivo de simular requisições originadas de múltiplos dispositivos e sistemas operacionais a cada execução.

Simultaneamente, as propriedades padrão que expõem o controle automatizado do *WebDriver* (a interface de comunicação com o navegador) foram desabilitadas via *Chrome DevTools Protocol* (CDP) (Google, 2026). As APIs internas de inspeção do CDP permitiram a sobrescrita das variáveis de ambiente do navegador antes do carregamento da página alvo. Adicionalmente, aplicaram-se restrições ao protocolo de comunicação em tempo real WebRTC (JENNINGS et al., 2021), a fim de bloquear a exposição dos endereços IP reais da máquina executora durante as conexões *peer-to-peer*. Para mitigar o rastreamento comportamental e evitar bloqueios por volume excessivo de tráfego, adotou-se o fracionamento temporal, distribuindo a execução do algoritmo em janelas de tempo em dias distintos.

3.1.2 Navegação e Busca de Informações nos Sites

O sistema iniciou a navegação acessando a URL do estabelecimento para interagir com o Modelo de Objeto de Documentos (DOM) (W3C, 1998), estrutura em formato de árvore que mapeia os nós da página. Inicialmente, o algoritmo localizou a interface de avaliações e acionou o filtro de ordenação cronológica. Para a varredura contínua dos registros, o controlador de navegação executou duas ações iterativas:

- **Rolagem de página (*Scroll*):** O sistema injetou comandos em JavaScript[6] diretamente no console do navegador para manipular o eixo vertical da página, forçando a requisição de novos elementos assíncronos ao servidor. Este laço de repetição operou até que a verificação de altura do contêiner do DOM permanecesse estática após múltiplas tentativas consecutivas, indicando o esgotamento dos registros disponíveis.
- **Expansão de comentários:** Os nós correspondentes aos botões de expansão de texto foram mapeados para receber eventos de clique automatizados, revelando o conteúdo truncado. Durante esta etapa, observou-se que a interação com determinados perfis de usuários acionava redirecionamentos não intencionais da sessão. Para assegurar a continuidade do fluxo, implementou-se uma rotina de verificação baseada em uma lista de exclusão (*blacklist*), instruindo a aplicação a ignorar a interação com identificadores previamente mapeados.

[6] (ECMA International, 2023)

Todas as interações computacionais com o DOM foram espaçadas por intervalos de tempo aleatórios, visando emular atrasos motores humanos e reduzir os padrões determinísticos de acesso.

3.1.3 Extração das Informações e Estruturação dos Dados

Após a renderização completa dos elementos presentes no site, o código-fonte em HTML[7] foi transferido para a biblioteca BeautifulSoup, responsável por realizar o *parsing* (análise sintática) do documento. O extrator iterou sobre os elementos `<div>` correspondentes a cada avaliação para capturar os atributos mapeados.

Para garantir a extração correta dos dados diante de mudanças estruturais do *layout*, o algoritmo implementou mecanismos de contingência (*fallbacks*) na busca pelos nós. A localização primária ocorreu via seletores CSS (W3C, 2023), realizando a busca através das classes de estilização da plataforma; em caso de falha (*timeout* ou elemento não encontrado), o sistema recorreu a consultas via XPath (W3C, 1999), mapeando o caminho absoluto do elemento na hierarquia da árvore do DOM.

Em seguida, o conteúdo extraído foi submetido a filtros baseados em Expressões Regulares (Regex) (FRIEDL, 2006) para a limpeza do texto, removendo padrões literais que indicavam edição da postagem ou métricas do autor inseridas no comentário. O sistema também estabeleceu um critério de parada, interrompendo a coleta ao registrar um limite consecutivo de avaliações sem corpo de texto (contendo apenas a nota quantitativa).

3.1.4 Persistência dos Resultados

Na etapa final, o gerenciador de persistência armazenou as variáveis em um dicionário de dados em memória e procedeu com a serialização para o formato JSON (*JavaScript Object Notation*), um padrão estruturado e hierárquico de intercâmbio de informações. A escrita dos arquivos adotou a codificação UTF-8 para preservar a integridade de caracteres especiais e acentuações inerentes à linguagem natural coletada.

O conteúdo do arquivo gerado foi composto por um nó raiz contendo os metadados da extração (`establishment_name`, `url`, `scrape_date` e `total_reviews`) e um vetor (*array*) denominado `reviews` que englobou os objetos independentes de cada comentário, representado pelas chaves `review_id`, `author`, `rating`, `publish_date`, `text` e `author_review_count`.

[7] (W3C, 2014)

3.2 Análise de Sentimentos com Modelos de Inteligência Artificial

A etapa de processamento analítico dos sentimentos foi estruturada em três fases de execução: a formatação dos dados de entrada, a classificação de sentimentos propriamente dita e o pós-processamento estatístico dos resultados. Para viabilizar esta arquitetura, desenvolveram-se algoritmos na linguagem Python voltados à automação da leitura dos arquivos contendo as avaliações e à submissão sequencial desses textos aos módulos de classificação.

3.2.1 Métodos para Análise de Sentimentos

A fase inicial de integração consistiu na implementação de um módulo gerenciador de arquivos em lote. Este componente utilizou a biblioteca `argparse` para estabelecer uma interface de linha de comando, permitindo a parametrização do caminho do arquivo JSON de entrada. Visando a resiliência da execução, o algoritmo implementou rotinas de tratamento de exceções para gerenciar falhas de formatação (`json.JSONDecodeError`) e a eventual ausência de diretórios no sistema de arquivos. O carregamento do documento em memória ocorreu sob a codificação UTF-8 para assegurar a integridade dos caracteres especiais.

A lógica estrutural responsável por integrar a leitura do arquivo aos modelos preditivos e, posteriormente, o cálculo das métricas de validação, está ilustrada na Figura 4.

Na execução da classificação linguística, o sistema empregou a biblioteca spaCy (HONNIBAL; MONTANI, 2017). Para os propósitos desta arquitetura, a ferramenta atuou primariamente como uma camada de encapsulamento (*wrapper*) para gerenciar e padronizar as chamadas de função no *pipeline* de processamento. Por meio do uso de decoradores (`@Language.component`), os classificadores de sentimento foram anexados como a etapa final do fluxo de execução, registrando os resultados de forma unificada em extensões de atributos customizadas na estrutura do documento (como `doc._.sentiment_raw` e `doc._.compound`).

3.2.2 Configuração dos Modelos de Análise

Para contrastar diferentes paradigmas computacionais de Processamento de Linguagem Natural (PLN), o conjunto de dados foi submetido a três modelos de classificação distintos, instanciados de forma modularizada.

O primeiro algoritmo implementado foi o modelo de atenção bidirecional BERT (NLP Town, 2020; DEVLIN et al., 2019). A integração ocorreu via biblioteca Transformers (WOLF et al., 2020), instanciando a versão `bert-base-multilingual-uncased-sentiment`. Durante a execução, o texto sofreu o processo de tokenização em subpalavras (*subword to-*

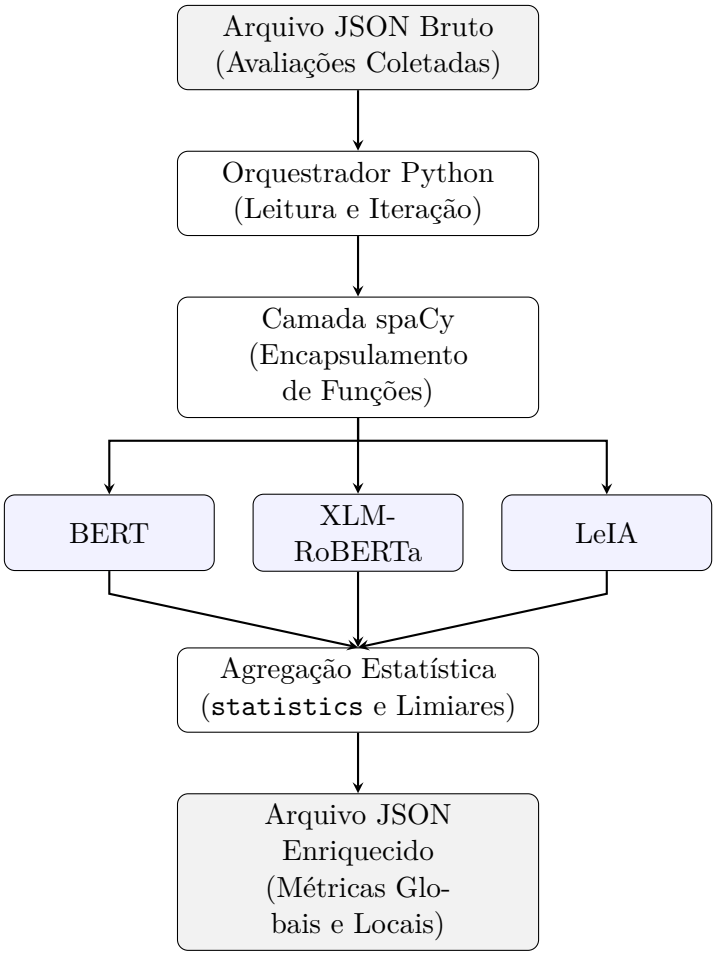

Figura 4 – Fluxograma dos métodos para análise de sentimentos.

Fonte: Elaborado pelo autor.

kenization), técnica que fragmenta termos complexos em subunidades menores para lidar com variações morfológicas e erros ortográficos típicos de comentários em redes sociais. O módulo retornou uma classificação discreta em uma escala de 1 a 5 estrelas, acompanhada de um escore de probabilidade preditiva.

O segundo modelo adotado foi o XLM-RoBERTa (CardiffNLP, 2021; CAMACHO-COLLADOS; REZAEE; AL., 2022), cuja arquitetura também opera sob o paradigma *Transformer*, porém com pesos ajustados para o processamento de microblogs (*tweets*). A configuração da versão `twitter-xlm-roberta-base-sentiment` utilizou um diretório local de armazenamento temporário (*models_cache*) para alocar os pesos neurais. Esta estratégia otimizou o carregamento processual via PyTorch (PASZKE et al., 2019) e eliminou a necessidade de conexões de rede externas durante a execução. A saída do modelo categorizou as avaliações em três rótulos nominais: positivo, neutro ou negativo.

O terceiro algoritmo consistiu na implementação do LeIA (ANDRADE, 2018), uma adaptação lexical do sistema VADER (HUTTO; GILBERT, 2014) para o português.

Diferente das redes neurais profundas, esta abordagem baseou-se em uma arquitetura léxica local, suportada pela leitura de quatro dicionários físicos (*lexicons*) obrigatórios, que mapeiam: a polaridade estática de palavras, intensificadores (*boosters*), marcadores de negação e *emojis*. O classificador aplicou heurísticas textuais estruturadas para identificar advérbios de grau e inverter o valor da polaridade diante de negações. O resultado consolidado consistiu em um valor normalizado denominado *compound*, variando no intervalo contínuo de -1,0 a +1,0.

3.2.3 Pós-processamento e Validação dos Resultados

Após a classificação individual de cada avaliação, o sistema executou rotinas de pós-processamento para extrair métricas globais de cada estabelecimento. Para os modelos baseados em redes neurais, utilizou-se o módulo `statistics` do Python para computar a média aritmética, a mediana e o desvio padrão dos escores de confiança. Estes valores foram estruturados no nó de dados `confidence_metrics`, permitindo avaliar a estabilidade das predições do algoritmo perante o conjunto total de avaliações.

No processamento do modelo LeIA, a agregação focou na distribuição das polaridades. Implementou-se uma lógica de discretização por limiares (*thresholds*): valores de *compound* inferiores a -0,05 foram atribuídos à categoria negativa; pontuações entre -0,05 e +0,05 foram classificadas como neutras; e valores superiores a +0,05 compuseram a categoria positiva. Ao término das operações matemáticas, os dados foram serializados em arquivos JSON, utilizando-se o parâmetro `ensure_ascii=False` para garantir a preservação da acentuação característica da língua portuguesa.

3.3 Visualização e Comparação de Dados

A última etapa do desenvolvimento consistiu na análise das métricas após a classificação e na geração de gráficos. Esta fase consumiu os arquivos JSON resultantes da camada de classificação, processando os dados por meio de *scripts* utilitários e de um ambiente interativo de *notebooks* construído em Python.

Algumas métricas globais (como médias, medianas e os níveis de confiança) foram extraídas e agrupadas a partir dos documentos gerados na etapa anterior. Para cruzar informações e consolidar os resultados de eficácia, foi necessário codificar *scripts* específicos de comparação. A lógica analítica dessas rotinas seguiu o formato de saída inerente a cada classificador:

- **Módulo de Comparação (BERT):** O algoritmo realizou o *parsing* do texto gerado pela rede neural (como a *string* "4 stars") para um valor numérico inteiro. Em seguida, calculou a diferença exata entre a nota original fornecida pelo usuá-

rio e a previsão do modelo, contabilizando a quantidade de acertos e os níveis de divergência.

- **Módulo de Comparação (LeIA):** Como a saída principal deste classificador é o valor contínuo *compound* (que varia no intervalo de -1,0 a +1,0), o código aplicou uma conversão matemática para projetar esse número na escala discreta de 1 a 5 estrelas. Assim, a precisão dos resultados obtidos por esse modelo foi analisada de forma metodologicamente análoga à do BERT.

- **Módulo de Consolidação (RoBERTa):** Diferentemente dos *scripts* anteriores, este algoritmo não calculou a distância numérica em relação às avaliações em estrela. O código iterou sobre os arquivos e contabilizou a distribuição local e global dos sentimentos, somando o total absoluto de avaliações alocadas nas categorias positiva, negativa e neutra.

O fluxo computacional incluiu também *scripts* auxiliares de consolidação voltados a contabilizar a distribuição total das polaridades, convertendo os escores numéricos do *compound* e das estrelas em categorias nominais fixas, a partir de limiares predefinidos. Ao final da execução, os resultados consolidados de todos os modelos foram serializados em novos arquivos JSON, com o propósito de estruturar os dados para a injeção em bibliotecas de plotagem gráfica.

A construção dos gráficos foi realizada em um ambiente interativo em nuvem na plataforma Google Colab[8], utilizando o formato de *notebooks*. Para carregar e estruturar os dados comparativos em formato de tabelas (*DataFrames*), o código fez uso das bibliotecas `pandas`[9] e `numpy`[10]. Com as informações organizadas em memória, a renderização gráfica de barras foi executada pelo módulo `pyplot`, pertencente à biblioteca `matplotlib`. Por fim, por meio dessas visualizações, foi possível interpretar os resultados da fase de análise de sentimentos.

3.4 Fluxo de Dados

Esta seção visa resumir o fluxo de dados do desenvolvimento, documentando o caminho da informação desde a extração inicial até a geração das visualizações gráficas. O processo é estruturado em três etapas principais, nas quais cada fase gera arquivos que serão utilizados na etapa seguinte. O fluxo computacional completo é ilustrado na Figura 5.

[8] (GOOGLE, 2026)
[9] (MCKINNEY, 2010)
[10] (HARRIS et al., 2020)

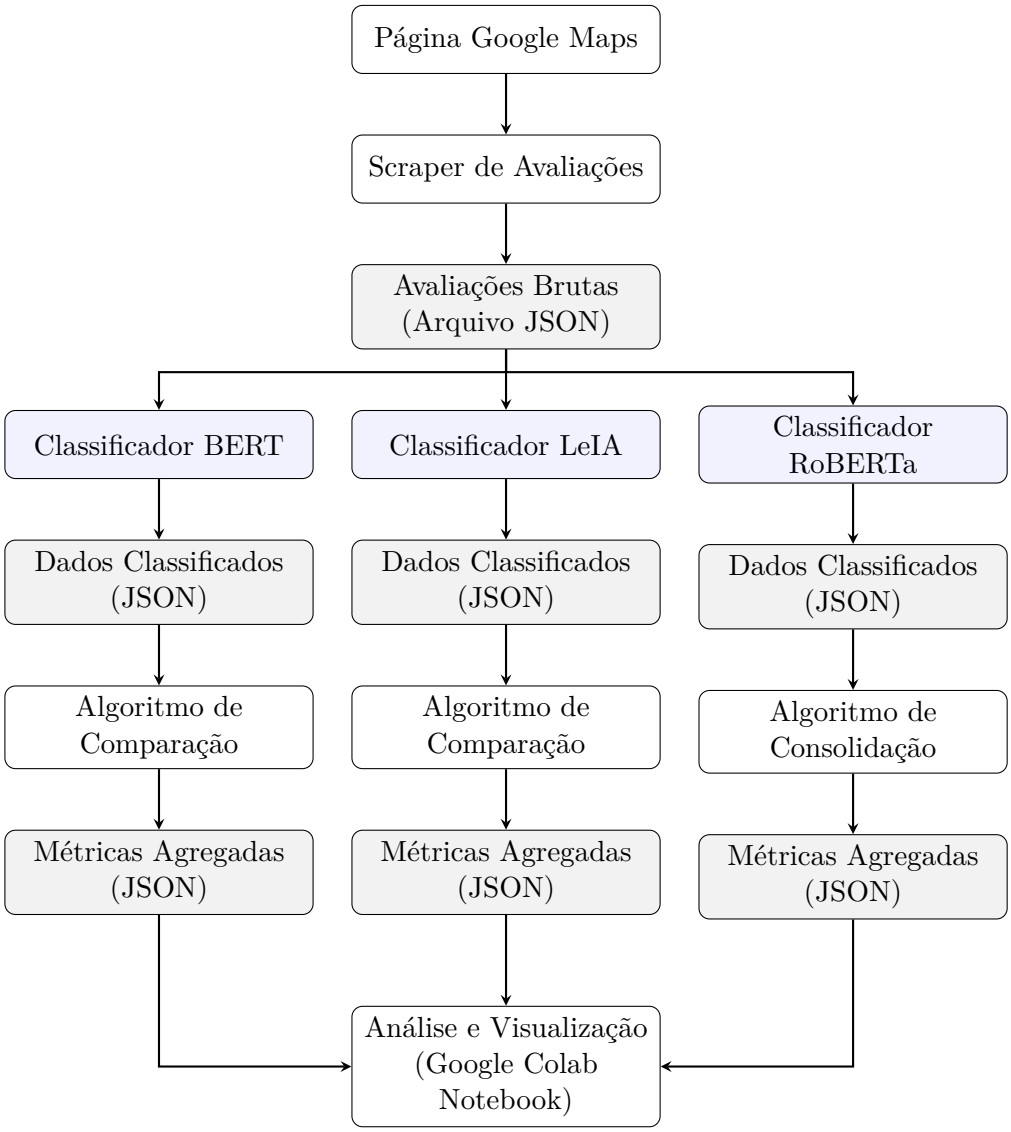

Figura 5 – Fluxograma detalhado do tráfego de informações no *pipeline* computacional.

Fonte: Elaborado pelo autor.

O *pipeline* computacional opera nos seguintes estágios principais:

3.4.1 Coleta de Dados via *Web Scraping*

O fluxo é iniciado pelo módulo de extração automatizada. O algoritmo acessa a página alvo do estabelecimento na plataforma, extrai o conteúdo dinâmico e salva o texto das avaliações e as respectivas notas dos consumidores em um arquivo JSON. Este documento inicial atua como a base de dados em formato bruto do projeto, isolando o momento da coleta orgânica das fases subsequentes de processamento analítico.

3.4.2 Análise de Sentimentos com Modelos de Inteligência Artificial

O arquivo de avaliações em formato bruto é consumido pelos três modelos de classificação selecionados: BERT, LeIA e XLM-RoBERTa. A execução ocorre de forma independente para cada arquitetura computacional. Os modelos processam o texto e geram novos arquivos JSON, os quais contêm as predições de polaridade e os escores estatísticos associados a cada comentário.

3.4.3 Visualização e Análise de Dados

Após a etapa de classificação, os dados produzidos pelos três modelos são consumidos por rotinas utilitárias de consolidação. De forma simplificada, estes algoritmos calculam divergências matemáticas (para o BERT e LeIA) ou agrupam as avaliações em categorias textuais (para o RoBERTa). O objetivo comum desta fase é contabilizar a distribuição global dos sentimentos e exportar as métricas agregadas em novos arquivos JSON.

Para o fechamento do ciclo computacional, essas métricas são inseridas manualmente em um ambiente interativo na plataforma Google Colab. Nesse local, utilizando estruturas tabulares e o módulo de renderização visual da biblioteca `matplotlib`, os dados consolidados são transformados em representações gráficas, viabilizando a análise comparativa que fundamenta a conclusão da pesquisa.

4 Resultados

4.1 Características da Base de Dados

Para compor a base de dados deste estudo e viabilizar as análises propostas, procedeu-se à coleta de comentários de seis estabelecimentos gastronômicos distintos da cidade de Uberlândia. Como critério de seleção, considerou-se a existência de uma grande quantidade de avaliações, não sendo feitas distinções quanto às demais características dos estabelecimentos, como: quantidade de clientes atendidos, faixa etária dos clientes, tamanho do estabelecimento, localização, horário de atendimento ao público, tempo de existência e outros. A quantidade total das amostras foram de 3.566 avaliações. Os dados foram extraídos dos seguintes locais: Restaurante Comedére[1], contabilizando 348 comentários; Restaurante Fogão de Minas[2], com 620 avaliações; Churrascaria Pim Grill[3], com 696 registros; Coco Bambu[4], com 748 avaliações; Minas Tchê[5], contendo 808 comentários; e o Restaurante Universitário (RU Santa Mônica)[6], com 346 avaliações validadas.


```
{
  "establishment_name": "Restaurante Comedére",
  "url": "https://www.google.it/maps/place/Restaurante+Comed%C3%A9re/@-18.9104447,-48.3197987,
  17z/
  data=!3m1!5s0x94a4440aef2b4947:0x2dee024cb546462f!4m8!3m7!1s0x94a4440ae86576ff:0x82e5927c393102
  f6!8m2!3d-18.9104447!4d-48.3172184!9m1!1b1!16s%2Fg%2F11b6p7cwws?entry=ttu&
  g_ep=EgoyMDI1MTAyNi4wIKXMDSoASAFQAw%3D%3D",
  "scrape_date": "2025-12-01T21:35:05.039363",
  "total_reviews": 348,
  "reviews": [
    {
      "reviewer_name": "[redacted]",
      "review_text": "Comida maravilhosa e atendimento impecável. Fazem um peixe muito bom, fora
      que servem um cafézinho sem açúcar perfeito e tiras de rapadura.",
      "relative_date": "um mês atrás",
      "stars_given": 5,
      "owner_response_text": "Agradecemos o carinho e a preferência! Seja sempre bem vindo!"
    },
```


Figura 6 – Estrutura do arquivo JSON da saída da etapa do *web scraping*.

Conforme ilustrado na Figura 6, diversos metadados foram extraídos da página contendo as avaliações dos restaurantes. Para a estruturação da coleta de dados, utilizou-se como referência o modelo de saída da plataforma Apify (Apify Technologies s.r.o.,

[1] Página do Comedére no Google Maps
[2] Página do Fogão de Minas no Google Maps
[3] Página do Pim Grill no Google Maps
[4] Página do Coco Bambu no Google Maps
[5] Página do Minas Tchê no Google Maps
[6] Página do RU Santa Mônica no Google Maps

2026), especificamente o seu extrator de avaliações (*Google Maps Reviews Scraper*). A análise prévia do *schema* gerado por essa ferramenta serviu para mapear os campos de informação necessários para a etapa de processamento de texto.

A partir dessa referência, definiu-se o formato de extração, abrangendo atributos como o nome do revisor, a nota atribuída, a data relativa, eventuais respostas do proprietário e o conteúdo textual. Embora a implementação do *script* de *Web Scraping* tenha sido desenvolvida em Python, a estrutura dos dados armazenados no arquivo JSON acompanhou o modelo relacional da plataforma citada. O objetivo dessa escolha foi organizar a base de dados segundo um padrão de estruturação já estabelecido para esse tipo de extração.

Cabe ressaltar que o foco está no atributo `review_text`. Este campo armazena o conteúdo textual bruto redigido pelos clientes e constituiu a matéria-prima essencial para a aplicação dos modelos de análise de sentimentos. Uma parcela significativa dos demais metadados coletados não foi amplamente explorada nas etapas analíticas subsequentes, servindo primordialmente como identificadores auxiliares para distinguir e organizar os registros, não influenciando a inferência sentimental da Inteligência Artificial.

Após a etapa de classificação fez-se necessário adaptar a estrutura final da base de dados, , devido às particularidades das saídas geradas por cada modelo. A seguir, detalham-se as alterações estruturais e as novas métricas adicionadas ao conjunto de dados por cada abordagem avaliada.

```
{
  "url": "https://www.google.it/maps/place/Restaurante+Comed%C3%A9re/@-18.9104447,-48.3197987,
  17z/
  data=!3m1!5s0x94a4440aef2b4947:0x2dee024cb546462f!4m8!3m7!1s0x94a4440ae86576ff:0x82e5927c393102
  f6!8m2!3d-18.9104447!4d-48.3172184!9m1!1b1!16s%2Fg%2F11b6p7cwws?entry=ttu&
  g_ep=EgoyMDI1MTAyNi4wIKXMDSoASAFQAw%3D%3D",
  "scrape_date": "2025-12-01T21:35:05.039363",
  "total_reviews": 348,
  "average_compound": 0.44253793103448297,
  "median_compound": 0.4404,
  "sentiment_distribution": {
    "negative": 15,
    "neutral": 57,
    "positive": 276
  },
  "reviews": [
    {
      "reviewer_name": "[redacted]",
      "review_text": "Comida maravilhosa e atendimento impecável. Fazem um peixe muito bom, fora
      que servem um cafézinho sem açúcar perfeito e tiras de rapadura.",
      "relative_date": "um mês atrás",
      "stars_given": 5,
      "owner_response_text": "Agradecemos o carinho e a preferência! Seja sempre bem vindo!",
      "sentiment_compound": 0.8402,
      "sentiment_details": {
        "neg": 0.0,
        "neu": 0.654,
        "pos": 0.346,
        "compound": 0.8402
      }
    },
```


Figura 7 – Estrutura do arquivo JSON contendo os atributos gerados pela classificação do modelo LeIA.

A Figura 7 ilustra o formato do arquivo de saída após o processamento do modelo LeIA. Neste arquivo foram acrescentados os campos `sentiment_compound` (a nota agregada final de polaridade, que varia numa escala contínua de -1 a +1) e `sentiment_details`, que detalha matematicamente a proporção de negatividade, neutralidade e positividade identificada pelo léxico na frase. Além disso, para cada estabelecimento foi adicionado os atributos `average_compound` (média dos sentimentos do local), o `median_compound` (mediana) e a `sentiment_distribution`. Essa última métrica contabiliza o volume de avaliações classificadas globalmente como negativas, neutras ou positivas.

```
{
  "url": "https://www.google.it/maps/place/Restaurante+Comed%C3%A9re/@-18.9104447,-48.3197987,
  17z/
  data=!3m1!5s0x94a4440aef2b4947:0x2dee024cb546462f!4m8!3m7!1s0x94a4440ae86576ff:0x82e5927c393102
  f6!8m2!3d-18.9104447!4d-48.3172184!9m1!1b1!16s%2Fg%2F11b6p7cwws?entry=ttu&
  g_ep=EgoyMDI1MTAyNi4wIKXMDSoASAFQAw%3D%3D",
  "scrape_date": "2025-12-01T21:35:05.039363",
  "total_reviews": 348,
  "confidence_metrics": {
    "average_score": 0.5830523077955191,
    "median_score": 0.5646730363368988,
    "stdev_score": 0.16290107484807598,
    "min_score": 0.25383302569389343,
    "max_score": 0.9593322277069092
  },
  "reviews": [
    {
      "reviewer_name": "[redacted]",
      "review_text": "Comida maravilhosa e atendimento impecável. Fazem um peixe muito bom, fora
      que servem um cafézinho sem açúcar perfeito e tiras de rapadura.",
      "relative_date": "um mês atrás",
      "stars_given": 5,
      "owner_response_text": "Agradecemos o carinho e a preferência! Seja sempre bem vindo!",
      "sentiment_label": "5 stars",
      "sentiment_score": 0.8674017786979675
    },
```


Figura 8 – Estrutura do arquivo JSON contendo os atributos gerados pela classificação do modelo BERT.

Em contraste com a abordagem léxica, a Figura 8 apresenta os resultados do modelo BERT, que é baseado em redes neurais profundas. Neste cenário, a avaliação processada recebeu um `sentiment_label` em formato de estrelas (de 1 a 5) e um `sentiment_score`. Cabe ressaltar que esse *score* não representa uma soma de polaridades, mas sim a métrica de confiança (probabilidade *softmax*) da rede neural na classificação escolhida. Além disso, para cada estabelecimento, na raiz do arquivo JSON, é adotado o bloco `confidence_metrics`, englobando métricas estatísticas exclusivas do modelo, como a confiança média (`average_score`), mediana (`median_score`), desvio padrão (`stdev_score`), além das confianças mínima e máxima detectadas no lote de avaliações.

```
{
  "url": "https://www.google.it/maps/place/Restaurante+Comed%C3%A9re/@-18.9104447,-48.3197987,
  17z/
  data=!3m1!5s0x94a4440aef2b4947:0x2dee024cb546462f!4m8!3m7!1s0x94a4440ae86576ff:0x82e5927c393102
  f6!8m2!3d-18.9104447!4d-48.3172184!9m1!1b1!16s%2Fg%2F11b6p7cwws?entry=ttu&
  g_ep=EgoyMDI1MTAyNi4wIKXMDSoASAFQAw%3D%3D",
  "scrape_date": "2025-12-01T21:35:05.039363",
  "total_reviews": 348,
  "confidence_metrics": {
    "average_score": 0.8327116518356334,
    "median_score": 0.8851564228534698,
    "stdev_score": 0.12191317628765219,
    "min_score": 0.37754595279693604,
    "max_score": 0.9419174194335938
  },
  "reviews": [
    {
      "reviewer_name": "[redacted]",
      "review_text": "Comida maravilhosa e atendimento impecável. Fazem um peixe muito bom, fora
      que servem um cafézinho sem açúcar perfeito e tiras de rapadura.",
      "relative_date": "um mês atrás",
      "stars_given": 5,
      "owner_response_text": "Agradecemos o carinho e a preferência! Seja sempre bem vindo!",
      "sentiment_label": "positive",
      "sentiment_score": 0.8888190984725952
    },
```


Figura 9 – Estrutura do arquivo JSON contendo os atributos gerados pela classificação do modelo XLM-RoBERTa

Por fim, a Figura 9 exibe os dados gerados pelo modelo XLM-RoBERTa. Por compartilhar a mesma arquitetura do BERT, o arquivo JSON mantém intacto o bloco raiz de `confidence_metrics` para mensurar a dispersão e a precisão da confiança estatística do modelo em todo o conjunto de dados. A alteração, no entanto, ocorre na classificação individual (`sentiment_label`): em vez de retornar uma nota de 1 a 5 estrelas, o XLM-RoBERTa categoriza o texto diretamente como "positivo", "neutro"ou "negativo", acompanhado do seu respectivo `sentiment_score` de confiança.

4.2 Visualização de Dados

4.2.1 Medianas e Médias dos Modelos BERT e XLM-RoBERTA

Nesta etapa, comparam-se as médias e medianas do BERT e do XLM-RoBERTa, pois estes fornecem métricas de confiança (*score*) em suas predições. Para contextualizar a análise, o *score* é um valor numérico que varia de 0 a 1, representando a probabilidade matemática calculada pela rede neural de que a sua classificação esteja correta. Quanto mais próximo de 1, maior é a certeza do algoritmo sobre o sentimento extraído.

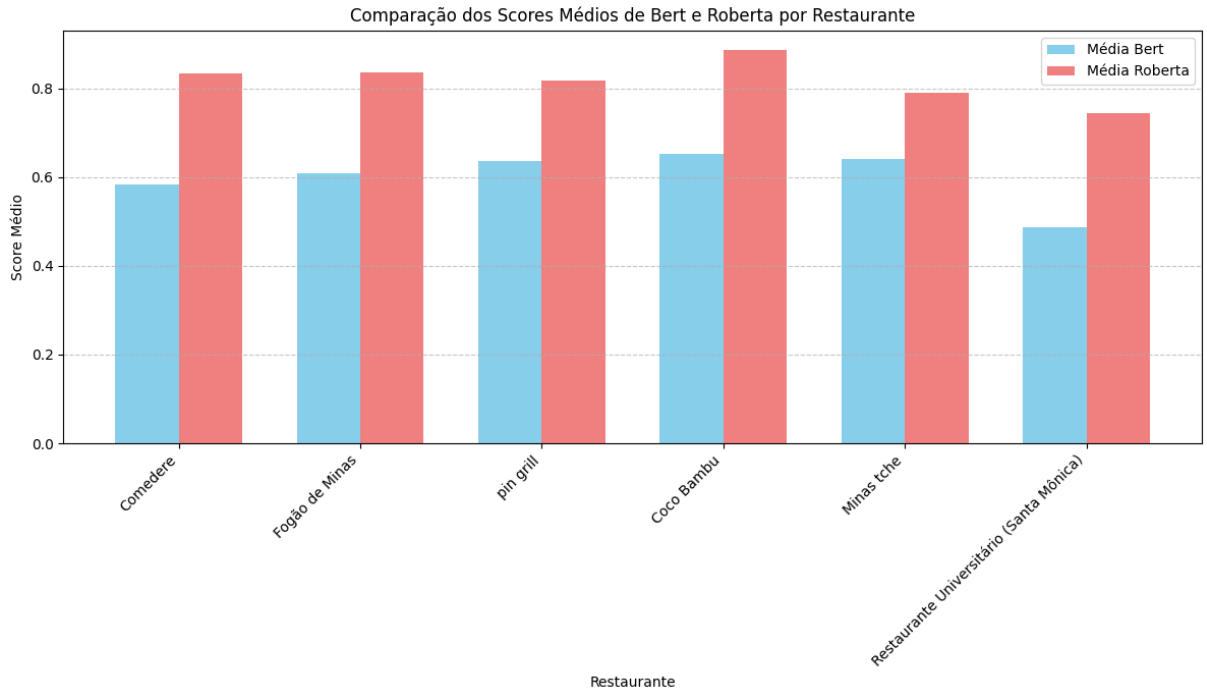

Figura 10 – Comparação da média de confiança entre: XLM-RoBERTa vs. BERT

O gráfico de médias da Figura 10, nota que o XLM-RoBERTa (representado pelas barras vermelhas) apresentou índices de confiança significativamente superiores aos do BERT (barras azuis) em todos os estabelecimentos. Enquanto o BERT registra médias flutuando na faixa de 0.50 a 0.65 (indicando uma certeza apenas moderada em suas classificações), o XLM-RoBERTa mantém um patamar elevado, oscilando entre 0.75 e 0.90. Esse comportamento superior já era esperado, visto que o conjunto de dados de treinamento do XLM-RoBERTa possui uma especialização linguística muito mais próxima da informalidade presente nas avaliações tratadas.

No entanto, os *scores* de confiança frequentemente apresentam assimetria. Valores extremos (*outliers*) podem deslocar a média aritmética de forma a não representar o comportamento típico do modelo. Para confirmar, escolheu-se a mediana como medida de tendência central complementar, devido sua robustez. Segundo Bruce, Bruce e Gedeck (2020), a mediana é considerada um "estimador resistente", pois não é influenciada por valores discrepantes nas caudas da distribuição, oferecendo uma visão mais fiel do centro da massa de dados em cenários de alta variância.

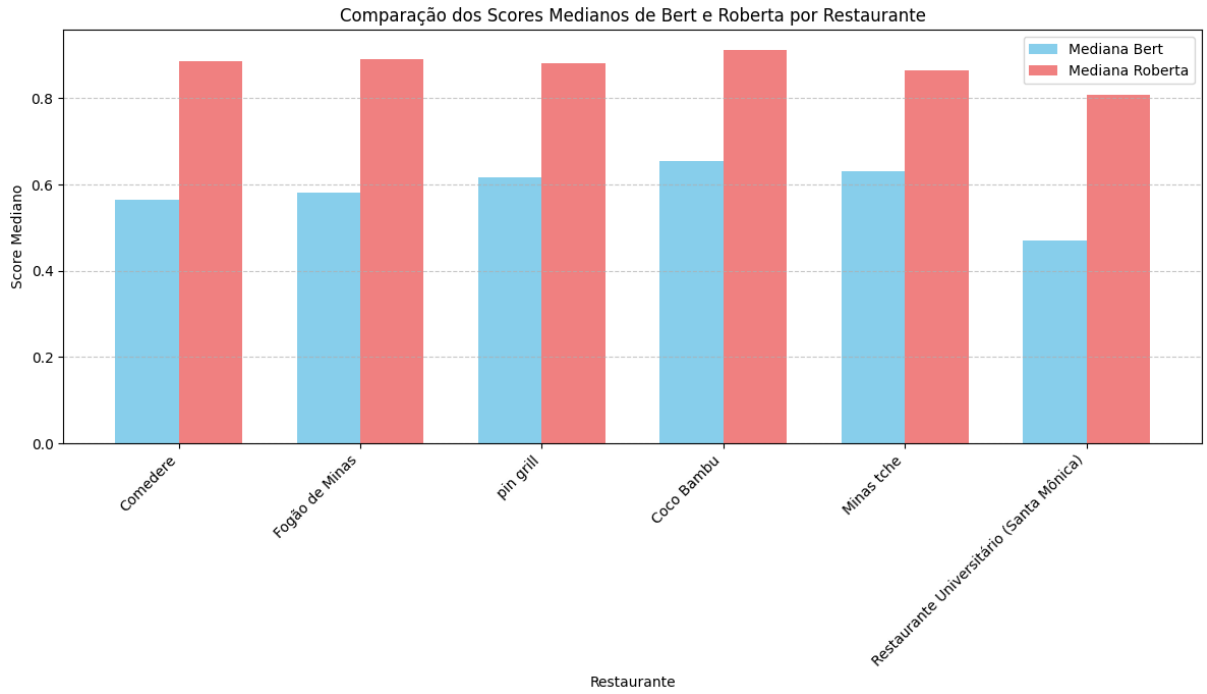

Figura 11 – Comparação da mediana de confiança entre: XLM-RoBERTa vs. BERT

Dessa forma, analisando a Figura 11, constata-se que a mediana do XLM-RoBERTa também se mantém amplamente superior, ou seja, ultrapassa a marca de 0.85 na maioria dos restaurantes, enquanto o BERT cai para valores próximos a 0.50 no Restaurante Universitário. O modelo XLM-RoBERTa não atinge um único pico isolado que poderia elevar sua média artificialmente, a maioria de suas predições se concentram em patamares de confiança muito mais elevados que os do BERT.

4.2.2 Diferença de Estrelas dos Modelos BERT e LeIA em Relação às Notas dos Usuários

Os modelos BERT e LeIA geram resultados numéricos que permitem uma comparação direta com a quantidade de estrelas atribuídas pelos clientes em suas avaliações. O BERT classifica o texto nativamente em uma escala discreta de 1 a 5 estrelas. Já o modelo LeIA exige uma etapa prévia de conversão: o valor do seu atributo *compound*, representado por c, funciona como um índice consolidado que calcula a soma das polaridades das palavras do texto, variando em um intervalo do extremo negativo $[-1]$ ao extremo positivo $[1]$. O processo matemático para extrair a equivalência em estrelas a partir do atributo *compound* é definido pela Equação 4.1:

$$\text{estrelas} = \text{arredonda}\left(\frac{c+1}{2} \cdot 4 + 1\right) \tag{4.1}$$

Nesta formulação, o termo $\frac{c+1}{2}$ normaliza a pontuação original para o intervalo $[0, 1]$. Em seguida, a multiplicação por 4 e a soma de 1 projetam esse valor para o intervalo de $[1, 5]$. Por fim, a função de arredondamento garante que a saída seja um valor inteiro, limitando naturalmente o resultado entre 1 e 5 estrelas.

A partir da obtenção desses valores discretos de predição, estabeleceu-se uma métrica de desempenho baseada na diferença absoluta entre a nota estimada pelo modelo e a avaliação real fornecida pelo consumidor (fornecida em estrelas, em conjunto com a avaliação textual que foi posteriormente analisada pelos modelos). Essa lógica de validação permite a criação de uma forma de mensurar a precisão dos algoritmos, conforme ilustrado na Figura 12. No eixo horizontal do gráfico, o índice "0" representa a coincidência exata entre a predição e o dado real (acerto pleno), enquanto os demais marcadores (de "1" a "4") quantificam o distanciamento, em estrelas, entre a classificação computacional e o julgamento humano, evidenciando a distribuição das divergências para cada abordagem.

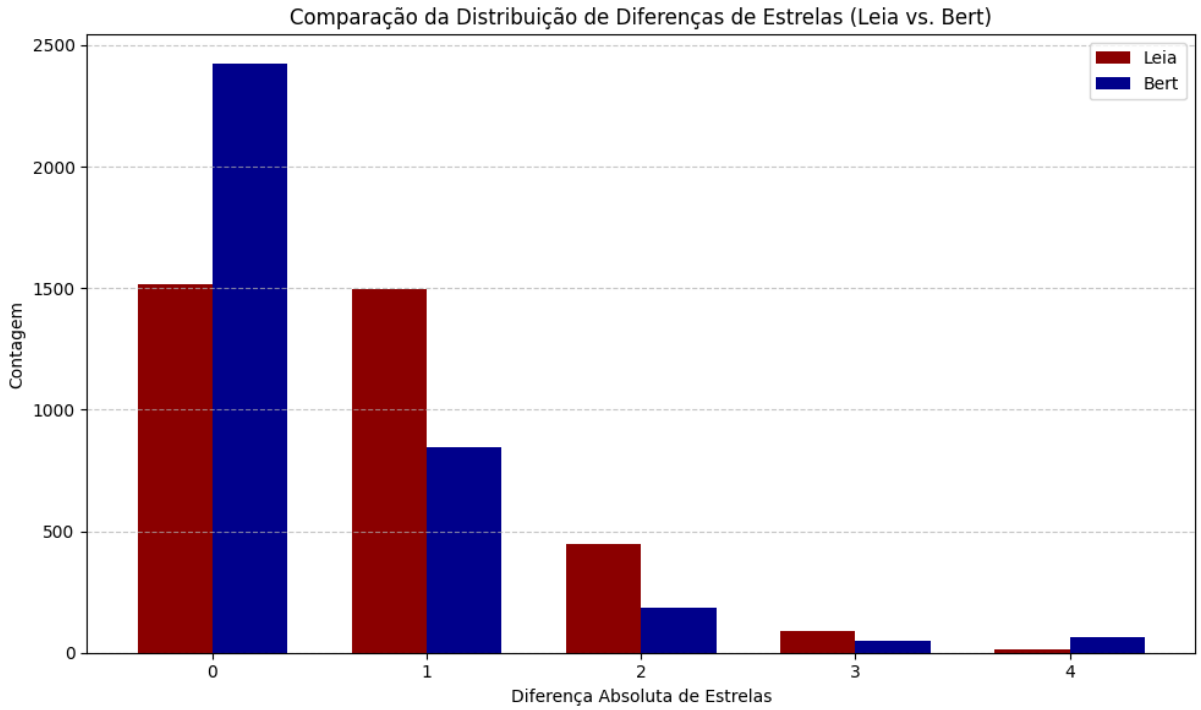

Figura 12 – Distribuição da diferença absoluta entre as predições dos modelos e à nota real (estrelas) dos usuários.

Fonte: Elaborado pelo autor.

É importante ressaltar que, embora o gráfico demonstre que o BERT possui uma taxa de acertos exatos (índice 0) superior à do LeIA, essa métrica isolada não define uma superioridade inquestionável. Existe a possibilidade de o texto escrito pelo usuário expressar um tom emocional ou semântico diferente da nota quantitativa que ele atribuiu

(como, por exemplo, um comentário com tom neutro acompanhado de uma avaliação de 5 estrelas). Essas nuances humanas podem gerar divergências entre a análise literal do texto e a nota final, o que justifica parte da margem de erro dos modelos.

4.2.3 Comparando as Notas Gerais dos Restaurantes e as métricas do LeIA

Conforme notado nas etapas anteriores, a fase de classificação dos sentimentos gerou métricas numéricas contínuas para cada restaurante, como a média e a mediana do *compound* calculadas pelo modelo LeIA. A partir desses dados, elaborou-se uma comparação dos valores obtidos pelo modelo com as notas públicas do estabelecimento, exibidas no Google Maps.

Neste ponto, é fundamental destacar um viés presente na base de dados. Como critério inicial de seleção, priorizaram-se estabelecimentos com alto volume de avaliações, o que resultou na escolha de locais consolidados. Essa característica cria um viés positivo intrínseco aos dados, concentrando as notas no topo da tabela e exigindo adaptações visuais para que as variações entre os restaurantes fiquem evidentes nas plotagens.

Para viabilizar a comparação direta entre a nota em estrelas do Google e a métrica gerada pelo modelo LeIA, a nota geral do restaurante precisou ser normalizada. Na primeira visualização (Figura 13), o valor original de 1 a 5 estrelas foi simplesmente dividido por 5, convertendo-o linearmente para uma escala percentual de 0 a 1.

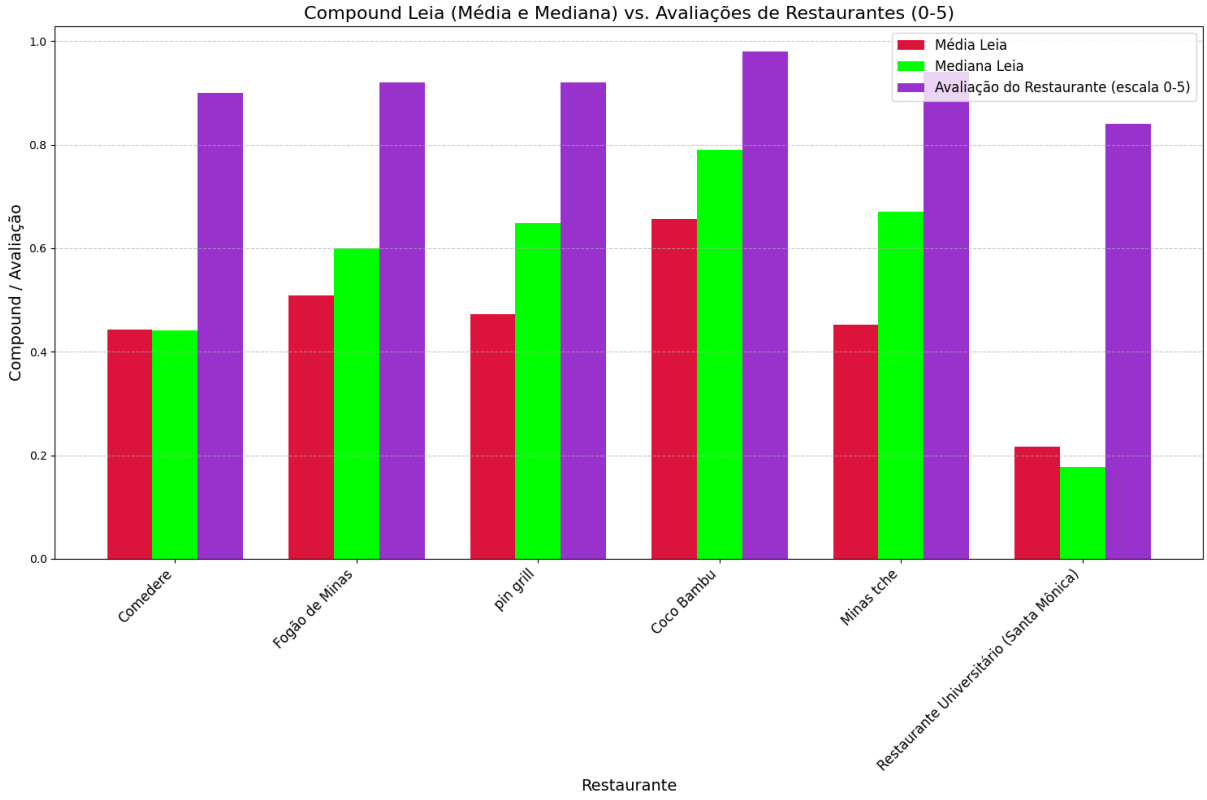

Figura 13 – Comparação entre a nota geral do restaurante pelo Google e a média e mediana do modelo LeIA.

Uma interpretação sobre a Figura 13 revela uma disparidade evidente. A polaridade do sentimento extraído apenas dos textos escritos (barras vermelhas e verdes) é consideravelmente inferior à nota geral do restaurante (barras roxas). Esse fenômeno sugere uma tendência comportamental do consumidor online: usuários parecem ser mais propensos a redigir comentários em texto quando vivenciam experiências negativas ou frustrantes. Em contrapartida, experiências positivas muitas vezes resultam apenas em avaliações silenciosas (a inserção da nota alta sem a redação de um texto associado), o que mantém a média geral do estabelecimento elevada no painel principal da plataforma.

Nesse contexto de disparidade, a visualização também evidencia que a mediana (barras verdes) se ajustou de forma visivelmente mais aderente à tendência das notas gerais do que a média aritmética (barras vermelhas). Como fundamentado anteriormente neste trabalho, esse comportamento prático reitera a robustez da mediana em lidar com distribuições assimétricas. Ao mitigar o impacto de avaliações textuais excessivamente negativas, a métrica conseguiu capturar o sentimento central da maioria dos clientes com maior precisão.

Para observar as nuances entre os estabelecimentos com maior clareza, gerou-se uma segunda visualização gráfica (Figura 14). Considerando o viés positivo da base

relatado anteriormente (todas as notas gerais concentradas acima de 4), aplicou-se um recorte focado apenas nessa faixa de notas altas. Matematicamente, a escala foi ajustada de forma que a nota 4.0 representasse o valor mínimo (0) e a nota 5.0 representasse o valor máximo (1), funcionando como uma ampliação visual das diferenças estatísticas.

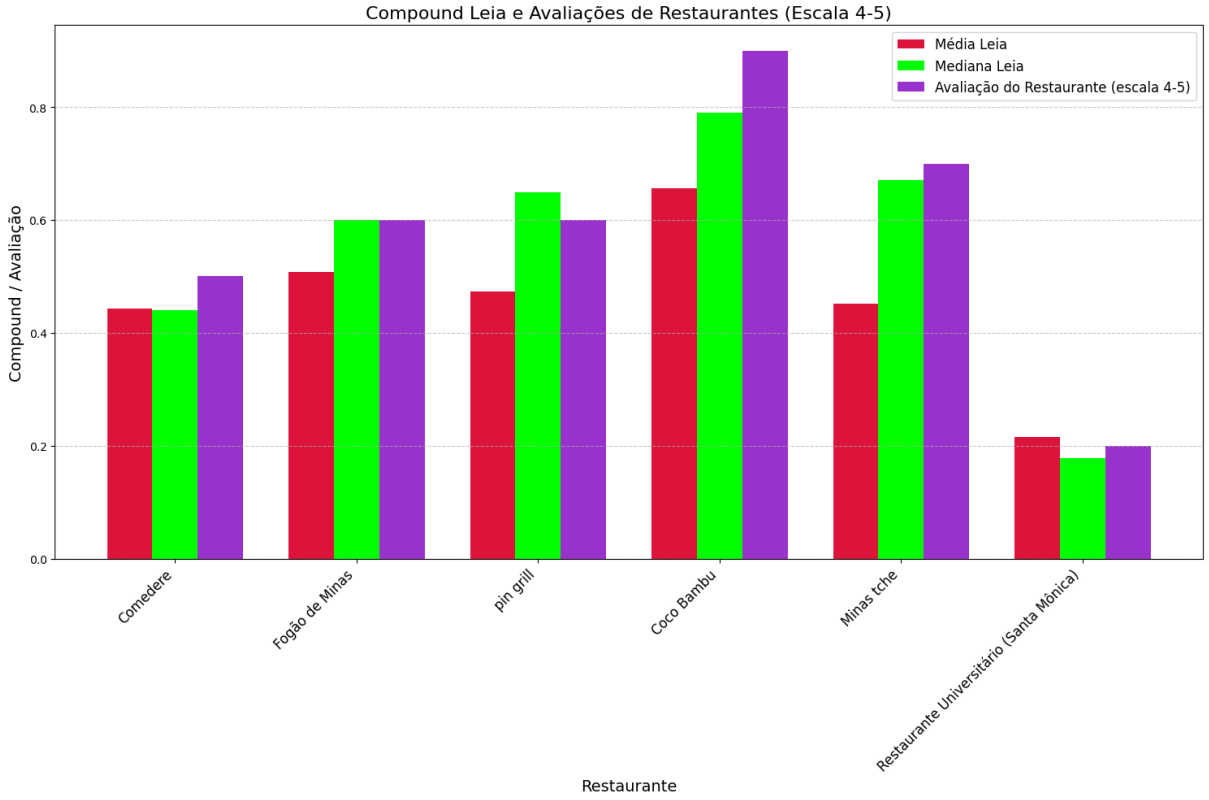

Figura 14 – Comparação com recorte de escala (4 a 5 estrelas) evidenciando a correlação das métricas

Por fim, apesar da inclinação geral para a negatividade nos textos, a Figura 14 demonstra que os comentários escritos ainda exercem uma influência proporcional na nota do local. Ao restringir a escala visual de comparação, fica claro que o sentimento textual acompanha a tendência da avaliação geral. O restaurante "Coco Bambu", por exemplo, que possui a maior nota geral da amostra (4.9), também apresenta os maiores picos de positividade textual. Em contrapartida, o Restaurante Universitário (RU Santa Mônica), que detém a menor nota geral do grupo (4.2), reflete o menor índice de sentimento positivo em seus comentários, confirmando a forte correlação entre o texto e as métricas quantitativas.

4.2.4 Comparação Categórica entre os Modelos

Como observado, o modelo XLM-RoBERTa teve uma participação mais restrita nas comparações numéricas diretas. O motivo para isso é a natureza de sua saída, que é

estritamente categórica, classificando os textos apenas como positivos, neutros ou negativos. Por não gerar um valor numérico de intensidade, a sua conversão direta para a escala de estrelas, de forma idêntica à realizada com os outros algoritmos, tornou-se imprecisa.

Portanto, para viabilizar uma comparação direta e pareada entre os três modelos, utilizando todos os dados coletados da base de avaliações, realizou-se o processo: a conversão das métricas numéricas do BERT e do LeIA para o formato categórico. O BERT, que retorna avaliações na escala de 1 a 5, foi categorizado da seguinte maneira: classificações de 1 e 2 estrelas foram convertidas para o sentimento negativo, 3 estrelas para o sentimento neutro, e 4 e 5 estrelas para o sentimento positivo. Para o LeIA, adotou-se uma regra baseada em seu *compound*: pontuações menores que -0.05 foram classificadas como negativas, pontuações entre -0.05 e 0.05 foram consideradas neutras, e valores superiores a 0.05 representaram o sentimento positivo.

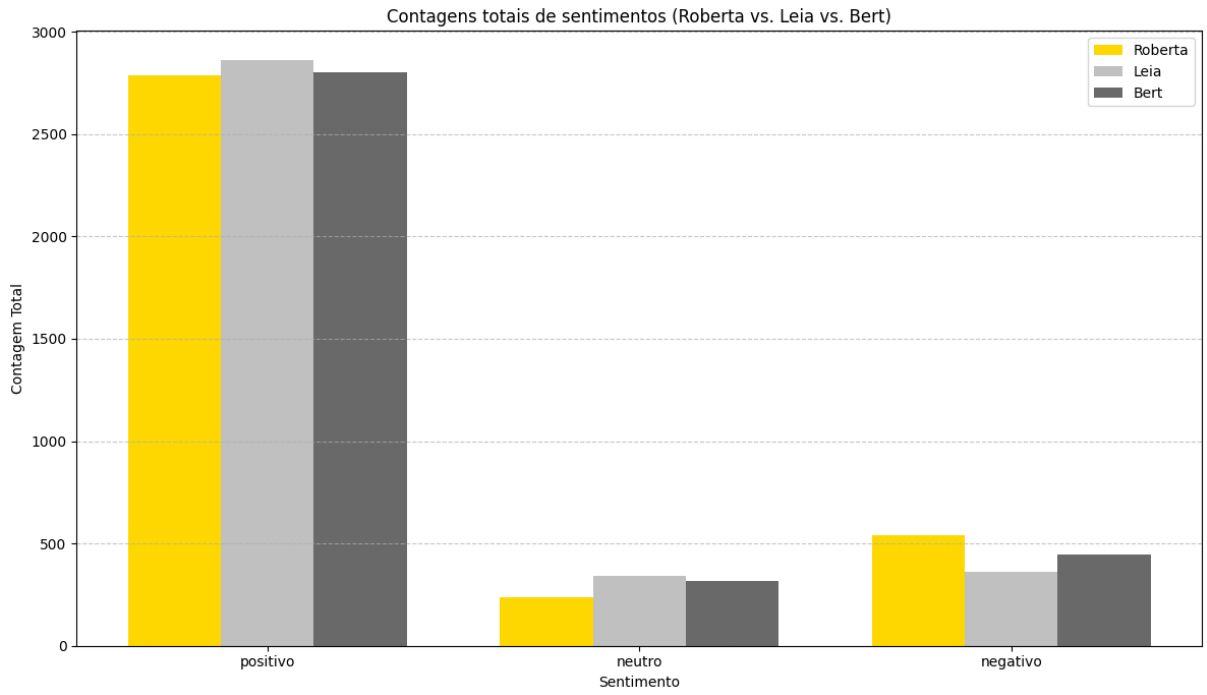

Figura 15 – Classificação dos sentimentos pelos modelos XLM-RoBERTa vs. LeIA vs. BERT.

O resultado dessa padronização categórica pode ser observado na Figura 15. Fica evidente que os três modelos concordam na proporção majoritária de avaliações positivas em toda a base de dados. Cabe ressaltar, contudo, que essa agregação em categorias amplas suavizou consideravelmente as diferenças visuais e numéricas entre os algoritmos. Ao condensar as nuances das métricas do LeIA e da escala de estrelas do BERT em apenas três classes, ocorreu uma perda natural de precisão analítica. Consequentemente, o comportamento dos modelos tornou-se mais homogêneo, ocultando variações de intensidade

que eram visíveis nas comparações anteriores.

5 Conclusão

A proposta deste trabalho foi avaliar o desempenho de técnicas de Processamento de Linguagem Natural juntamente com modelo pré-treinados de IA na tarefa de análise de sentimentos em comentários do Google Maps. Esse objetivo foi cumprido, permitindo comparar o comportamento dos modelos e estruturar a visualização das opiniões dos clientes. Como resultado, o estudo prova que a análise textual dos comentários dos clientes juntos com as notas em estrelas, são dados importantes para que o proprietário faça considerações desejáveis na gestão dos restaurantes.

A primeira etapa da metodologia foi a coleta de dados via *Web Scraping*, que se mostrou a fase mais desafiadora do projeto. A captura dos textos dos comentários e das notas atribuídas pelos usuários dependeu bastante da adaptação das técnicas ilustradas por (GASPARINI, 2020). O trabalho nessa parte consistiu em modernizar o script ensinado pelo autor e adicionar novas técnicas para evitar que a extração fosse bloqueada pela segurança do Google.

Em suma, o processo completo de levantamento e tratamento dos dados consolidou-se como fundamental para os resultados alcançados por esta pesquisa. A extração automatizada de milhares de avaliações e a estruturação dos arquivos JSON, garantiu dados satisfatórios para a classificação dos modelos de IA. Apesar dos desafios técnicos para contornar bloqueios de segurança e padronizar escalas numéricas distintas, o *pipeline* desenvolvido preservou a integridade das informações.

Na fase de análise de sentimentos, foram testados três modelos diferentes: BERT Multilingual, XLM-RoBERTa e LeIA. Em relação às interpretações geradas pelos dados, o BERT foi o modelo que apresentou o melhor desempenho. Ele se destacou por ter uma métrica nativa em estrelas, permitindo uma comparação direta com os metadados extraídos e gerou resultados significativamente mais próximos das impressões reais dos clientes.

O modelo LeIA, que adota uma abordagem léxica, apresenta o atributo *compound* responsável em gerar os resultados deste modelo. Embora o LeIA tenha sido um pouco menos preciso que o BERT, seu uso trouxe perspectivas interessantes e válidas para o estudo.

Já o modelo XLM-RoBERTa entregou uma boa faixa de acertos e confiança. No entanto, por retornar apenas classificações categóricas (positivo, neutro ou negativo) em vez de valores numéricos de intensidade, limitou a precisão na comparação direta com as estrelas.

Um dos achados mais importantes desta análise surgiu ao comparar a nota geral do restaurante presente no site do Google Maps (que inclui avaliações apenas com estrelas, sem texto) com a média do *compound* gerada pelo LeIA. Observou-se que as notas gerais tendem a ser muito maiores do que o sentimento extraído dos textos. Isso permite afirmar que existe uma tendência maior das pessoas deixarem comentários escritos quando a experiência é negativa ou quando há algo a reclamar, embora as notas dos comentários ainda sigam uma proporção em relação à nota geral do local.

A relevância prática da análise de sentimentos ficou evidente ao notar que a métrica padrão de estrelas pode transmitir a falsa impressão de que "está tudo bem". Todos os restaurantes analisados possuíam notas gerais acima de quatro estrelas, mas houve discrepâncias de até 60% quando essas médias foram comparadas com o sentimento real avaliado pelos modelos nos textos. Além disso, as estrelas dadas pelo usuário nem sempre são totalmente fiéis ao que ele escreve: um cliente pode dar 5 estrelas apenas porque "nada deu errado", mas escrever um texto neutro, sem grandes elogios. Sendo assim, o uso dos modelos de linguagem provou ser uma métrica complementar indispensável para a gestão do negócio.

5.1 Trabalhos Futuros

Pensando em trabalhos futuros, seria muito interessante testar e comparar modelos de linguagem que entreguem resultados totalmente compatíveis entre si. Isso facilitaria bastante a criação de visualizações e tornaria as análises mais diretas e precisas.

Continuando nos modelos, uma oportunidade promissora para desenvolvimentos futuros seria a aplicação de técnicas de *fine-tuning* (ajuste fino) em arquiteturas pré-treinadas, especializando-as estritamente no domínio do problema. Embora o presente estudo tenha obtido êxito ao adotar um modelo cuja base de dados de treinamento apresentou características linguísticas muito próximas ao escopo desejado, lidando bem com a informalidade dos textos, o treinamento com um conjunto de dados composto exclusivamente por avaliações de nicho gastronômico tem o potencial de elevar significativamente a precisão e a confiabilidade das classificações.

Além disso, existe a possibilidade de criar uma base de dados supervisionada onde outras pessoas avaliam os comentários. Como citado anteriormente, a métrica de estrelas dada pelos usuários serve como uma boa referência para o modelo, mas pode acabar gerando falsos positivos e negativos eventualmente.

Outra abordagem promissora para projetos futuros seria juntar a análise de sentimentos com a identificação de temas específicos abordados nos textos, como "tempo de espera", "qualidade da comida"e "atendimento". Essa ideia acabou ficando fora do escopo deste projeto, mas seria muito pertinente para que se saiba não só a intensidade do senti-

mento do cliente, mas exatamente qual ponto da operação do restaurante motivou aquele comentário.

Referências

ABIA. **Balanço Econômico da Indústria de Alimentos e Bebidas - 2024**. 2025. Acesso em: 6 mar. 2025. Disponível em: <https://intranet.abia.org.br/vsn/temp/z2025221OnePage25web.pdf>. Citado na página 12.

AMAZON. **Amazon Alexa: Assistente Virtual Inteligente**. 2025. Disponível em <https://www.amazon.com/alexa>. Acesso em março de 2025. Citado na página 16.

ANDRADE, R. J. de. **LeIA - Léxico para Inferência Adaptada**. 2018. <https://github.com/rafjaa/LeIA>. GitHub repository. Accessed: 2026-03-10. Citado 2 vezes nas páginas 20 e 29.

Apify Technologies s.r.o. **Google Maps Scraper: Data Output Schema**. 2026. Acesso em: 12 mar. 2026. Inspirado na estrutura de dados para automação de coleta de reviews. Disponível em: <https://apify.com/apify/google-maps-scraper>. Citado na página 35.

BINDS.CO. **Visualização de Análise de Sentimento**. 2023. Acesso em: 23 mar. 2025. Disponível em: <https://binds.co/img/section/faces-sentiment.png>. Citado na página 18.

BRIGHTLOCAL. **Local Consumer Review Survey 2023**. 2023. Acessado em: 15 mar. 2025. Disponível em: <https://www.brightlocal.com/research/local-consumer-review-survey-2023/>. Citado na página 12.

BROWN, T.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A. et al. Language models are few-shot learners. **Advances in Neural Information Processing Systems**, v. 33, p. 1877–1901, 2020. Disponível em: <https://arxiv.org/abs/2005.14165>. Citado na página 18.

BRUCE, P.; BRUCE, A.; GEDECK, P. **Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python**. 2. ed. Sebastopol, CA: O'Reilly Media, Inc., 2020. ISBN 978-1492072942. Citado na página 39.

CAMACHO-COLLADOS, J.; REZAEE, K.; AL. et. Tweetnlp: Cutting-edge natural language processing for social media. In: **Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**. [s.n.], 2022. Disponível em: <https://aclanthology.org/2022.emnlp-demos.5/>. Citado 2 vezes nas páginas 20 e 29.

Cambridge Dictionary. **Web Scraping - Definition**. 2025. Accessed: 2025-03-13. Disponível em: <https://dictionary.cambridge.org/us/dictionary/english/web-scraping>. Citado na página 17.

CardiffNLP. **Twitter XLM-RoBERTa base sentiment**. 2021. <https://huggingface.co/cardiffnlp/twitter-xlm-roberta-base-sentiment>. Hugging Face model card. Accessed: 2026-03-10. Citado 2 vezes nas páginas 20 e 29.

CASELI, H. de M.; NUNES, M. das G. V.; PAGANO, A. O que é pln? In: **Livro de Processamento de Linguagem Natural**. GLOBE, 2023. cap. 1. Uso permitido sob licença Creative Commons CC BY-NC-ND 3.0 BR. Disponível em: <https://brasileiraspln.com/livro-pln/1a-edicao/>. Citado na página 15.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018. Citado na página 19.

_____. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2019. Disponível em: <https://arxiv.org/abs/1810.04805>. Citado 2 vezes nas páginas 18 e 28.

ECMA International. **ECMA-404: The JSON data interchange format**. Genebra, Suíça, 2017. Disponível em: <https://www.ecma-international.org/publications-and-standards/standards/ecma-404/>. Acesso em: 29 mar. 2026. Citado na página 25.

_____. **ECMAScript 2023 Language Specification**. [S.l.], 2023. Standard ECMA-262. Disponível em: <https://262.ecma-international.org/14.0/>. Citado na página 26.

FARIAS, M. T.; ANGELUCI, A.; PASSARELLI, B. Web scraping e ciência de dados na pesquisa aplicada em comunicação: um estudo sobre avaliações online. **Revista observatório**, 2021. Citado na página 12.

FIELDING, R. T.; NOTTINGHAM, M.; RESCHKE, J. **HTTP Semantics**. [S.l.], 2022. Acesso em: 31 mar. 2026. Disponível em: <https://www.rfc-editor.org/rfc/rfc9110.html>. Citado na página 26.

FRIEDL, J. E. **Mastering Regular Expressions**. [S.l.]: O'Reilly Media, Inc., 2006. Citado na página 27.

GASPARINI, M. Scraping google maps reviews in python. **Towards Data Science**, apr 2020. Acesso em: 24 fev. 2026. Disponível em: <https://pt.scribd.com/document/790641073/Scraping-Google-Maps-reviews-in-Python-by-Mattia-Gasparini-Towards-Data-Science>. Citado na página 47.

GOOGLE. **Google: Motor de Busca**. 2025. Disponível em <https://www.google.com>. Acesso em março de 2025. Citado na página 16.

_____. **YouTube: Algoritmos de Recomendação**. 2025. Disponível em <https://www.youtube.com>. Acesso em março de 2025. Citado na página 16.

Google. **Chrome DevTools Protocol**. 2026. Acesso em: 31 mar. 2026. Disponível em: <https://chromedevtools.github.io/devtools-protocol/>. Citado na página 26.

GOOGLE. **Google Colaboratory**. 2026. <https://colab.research.google.com/>. Acesso em: 2 abr. 2026. Citado na página 31.

HARRIS, C. R.; MILLMAN, K. J.; WALT, S. J. van der; GOMMERS, R.; VIRTANEN, P.; COURNAPEAU, D.; WIESER, E.; TAYLOR, J.; BERG, S.; SMITH, N. J.; KERN, R.; PICUS, M.; HOYER, S.; KERKWIJK, M. H. van; BRETT, M.; HALDANE, A.; RÍO, J. F. del; WIEBE, M.; PETERSON, P.; GÉRARD-MARCHANT, P.; SHEPPARD, K.; REDDY, T.; WECKESSER, W.; ABBASI, H.; GOHLKE, C.; OLIPHANT, T. E. Array programming with NumPy. **Nature**, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357–362, 2020. Citado na página 31.

HONNIBAL, M.; MONTANI, I. spacy: Industrial-strength natural language processing in python. Zenodo, 2017. Citado na página 28.

HUTTO, C. J.; GILBERT, E. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: **Proceedings of the International AAAI Conference on Web and Social Media**. [s.n.], 2014. Disponível em: <https://ojs.aaai.org/index.php/ICWSM/article/view/14550>. Citado 2 vezes nas páginas 20 e 29.

Jeffrey Neuburger. **QVC Sues Shopping App for Web Scraping That Allegedly Triggered Site Outage**. 2014. Accessed: 2025-03-13. Disponível em: <https://newmedialaw.proskauer.com/2014/12/05/qvc-sues-shopping-app-for-web-scraping-that-allegedly-triggered-site-outage/>. Citado na página 17.

JENNINGS, C.; BOSSON, H.; KAUFMANN, M.; HAZELL, D. **WebRTC 1.0: Real-Time Communication Between Browsers**. [S.l.], 2021. Acesso em: 31 mar. 2026. Disponível em: <https://www.w3.org/TR/webrtc/>. Citado na página 26.

Leonard Richardson. **Beautiful Soup Documentation**. 2025. Accessed: 2025-03-17. Disponível em: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>. Citado na página 17.

LIU, B. Sentiment analysis and subjectivity. In: **Proceedings of the 2010 conference on Empirical Methods in Natural Language Processing**. [S.l.]: Association for Computational Linguistics, 2010. p. 410–421. Citado na página 18.

Matplotlib Development Team. **Matplotlib: Visualization with Python**. 2024. Disponível em: <https://matplotlib.org/>. Citado na página 25.

MCKINNEY Wes. Data Structures for Statistical Computing in Python. In: WALT Stéfan van der; MILLMAN Jarrod (Ed.). **Proceedings of the 9th Python in Science Conference**. [S.l.: s.n.], 2010. p. 56–61. Citado na página 31.

MITCHELL, R. **Web Scraping with Python: Collecting More Data from the Modern Web**. 2. ed. Sebastopol, CA: O'Reilly Media, Inc., 2018. Citado 2 vezes nas páginas 17 e 25.

MORENO Águeda C. **Análise de Sentimentos na Classificação de Comentários Online Aplicando Técnicas de Text Mining**. Dissertação (Dissertação de Mestrado) — ISCTE-IUL Business School, Lisboa, Portugal, outubro 2015. Citado 2 vezes nas páginas 12 e 13.

NLP Town. **BERT base multilingual uncased sentiment**. 2020. <https://huggingface.co/nlptown/bert-base-multilingual-uncased-sentiment>. Hugging Face model card. Accessed: 2026-03-10. Citado 2 vezes nas páginas 19 e 28.

PASZKE, A.; GROSS, S.; MASSA, F.; LERER, A.; BRADBURY, J.; CHANAN, G.; KILLEEN, T.; LIN, Z.; GIMELSHEIN, N.; ANTIGA, L. et al. Pytorch: An imperative style, high-performance deep learning library. In: **Advances in Neural Information Processing Systems 32**. [S.l.]: Curran Associates, Inc., 2019. p. 8024–8035. Citado na página 29.

PERPLEXITY AI. **Perplexity: AI Search Engine**. 2026. Disponível em: <https://www.perplexity.ai/>. Acesso em: 28 mar. 2026. Citado na página 17.

Python Software Foundation. **Python: A Dynamic, Open Source Programming Language**. [S.l.], 2024. Disponível em: <https://www.python.org/>. Citado na página 24.

RABINER, L. R. **A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition**. [S.l.]: IEEE, 1989. v. 77. 257–286 p. Citado na página 16.

RICHARDSON, L. **Beautiful Soup Documentation**. 2024. Disponível em: <https://www.crummy.com/software/BeautifulSoup/>. Citado na página 24.

SAMPAIO, A. **ANÁLISE DE SENTIMENTOS**. Monografia (type) — Pontifícia Universidade Católica de Goiás, 2021. Citado na página 18.

Scrapy Developers. **Scrapy Documentation**. 2025. Accessed: 2025-03-17. Disponível em: <https://docs.scrapy.org/en/latest/>. Citado na página 17.

Selenium Developers. **Selenium Documentation**. 2025. Accessed: 2025-03-17. Disponível em: <https://www.selenium.dev/documentation/>. Citado na página 17.

_____. **Selenium Documentation**. 2026. Acesso em: 11 mar. 2026. Disponível em: <https://www.selenium.dev/documentation/>. Citado na página 24.

SHIN, B.; RYU, S.; KIM, Y.; KIM, D. Analysis on review data of restaurants in google maps through text mining: Focusing on sentiment analysis. **Journal of Multimedia Information System**, Korea Science, v. 9, n. 1, p. 61–68, 2022. Citado na página 21.

SILVA, D. d. S. **OTTO: Descobrindo Assuntos e Sentimentos em Comentários**. Tese (Doutorado) — Universidade do Vale do Rio dos Sinos (UNISINOS), São Leopoldo, Brasil, 2019. Citado na página 21.

SILVA, M. J.; CARVALHO, P.; SARMENTO, L. Sentilex-pt 02: Léxico para análise de sentimentos e extração de opinião em português. In: **Encontro Nacional de Inteligência Artificial**. [S.l.: s.n.], 2012. Citado na página 22.

SILVA, N. F. F. d. **Análise de sentimentos em textos curtos provenientes de redes sociais**. Dissertação (Mestrado) — Universidade de São Paulo, São Carlos, 2016. Citado 3 vezes nas páginas 18, 22 e 23.

SIVAKORN, S.; POLAKIS, J.; KEROMYTIS, A. D. I'm not a human: Breaking the google recaptcha. In: **Proceedings of the Black Hat**. Las Vegas, NV: [s.n.], 2016. p. 1–15. Citado na página 17.

SOUZA, M.; VIEIRA, R. Construction of a portuguese opinion lexicon from multiple resources. In: SPRINGER. **International Conference on Computational Processing of the Portuguese Language**. [S.l.], 2012. p. 59–66. Citado na página 22.

STABILITYAI. **Stable Diffusion: Geração de Imagens via IA**. 2025. Disponível em <https://stablediffusionweb.com>. Acesso em março de 2025. Citado na página 16.

TURING, A. M. Computing machinery and intelligence. **Mind**, Oxford University Press, v. 59, n. 236, p. 433–460, 1950. Citado na página 15.

W3C. **Document Object Model (DOM) Level 1 Specification**. 1998. Acesso em: 31 mar. 2026. Disponível em: <https://www.w3.org/TR/REC-DOM-Level-1/>. Citado na página 26.

_____. **XML Path Language (XPath) Version 1.0**. 1999. Acesso em: 31 mar. 2026. Disponível em: <https://www.w3.org/TR/xpath/>. Citado na página 27.

_____. **HTML5: A vocabulary and associated APIs for HTML and XHTML**. 2014. Acesso em: 31 mar. 2026. Disponível em: <https://www.w3.org/TR/html5/>. Citado na página 27.

_____. **Cascading Style Sheets (CSS) Snapshot 2023**. 2023. Acesso em: 31 mar. 2026. Disponível em: <https://www.w3.org/TR/css-2023/>. Citado na página 27.

Wikimedia Commons. **Hidden Markov Model Diagram**. 2024. [Acessado em 23 março 2025]. Disponível em: <https://commons.wikimedia.org/wiki/File:HiddenMarkovModel.png>. Citado na página 16.

WOLF, T.; DEBUT, L.; SANH, V.; CHAUMOND, J.; DELANGUE, C.; MOI, A.; CISTAC, P.; RAULT, T.; LOUF, R.; FUNTOWICZ, M.; DAVISON, J.; SHLEIFER, S.; PLATEN, P. von; MA, C.; JERNITE, Y.; PLU, J.; XU, C.; SCAO, T. L.; GUGGER, S.; DRAME, M.; LHOEST, Q.; RUSH, A. Transformers: State-of-the-art natural language processing. In: **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**. [S.l.: s.n.], 2020. p. 38–45. Citado 2 vezes nas páginas 18 e 28.