Os Segredos da Análise de Sentimentos: Um Estudo Aplicado às Lojas Americanas

Érica Ribeiro[1], Júnior Fernandes Marques[2], Luís Vogel[3], Marlon José Martins[4], Raphael Franco Chaves[5], Thiago Ambiel[6]
ICMC-USP

1 Introdução

A análise de sentimentos tem se tornado uma ferramenta essencial para compreender as opiniões e percepções dos consumidores em relação a produtos e serviços, especialmente no contexto do e-commerce. Neste trabalho, será realizada uma investigação aprofundada utilizando o Corpus "B2W-Reviews01.csv", disponibilizado pelas Lojas Americanas. Através da análise das avaliações contidas neste Corpus, buscamos não apenas classificar os sentimentos expressos pelos consumidores, mas também explorar as nuances dessas opiniões, a fim de identificar padrões e tendências que podem informar estratégias de marketing e desenvolvimento de produtos. Este estudo pretende contribuir para uma compreensão mais profunda das dinâmicas de consumo na era digital, revelando insights que podem beneficiar tanto as empresas quanto os consumidores.

2 Objetivo Geral

O objetivo geral do nosso projeto corresponde ao desenvolvimento de uma aplicação automatizada de análise de sentimentos, permitindo uma melhor compreensão das opiniões dos consumidores. A seguir, apresentamos os principais elementos da aplicação, incluindo os tipos de usuários, os dados de entrada requeridos, os resultados fornecidos, bem como a finalidade geral da aplicação.

[1] ericaribeiro@usp.br

[2] junior.marques@usp.br

[3] luisvlopes@usp.br

[4] mjmartins@alumni.usp.br

[5] raphaelchaves@usp.br

[6] thiago.ambiel@usp.br

2.1 Usuários da Aplicação

Os usuários da nossa aplicação são gestores e profissionais de marketing de lojas de varejo, como plataformas de e-commerce ou marketplaces, que desejam compreender as opiniões dos clientes sobre seus produtos. A ferramenta foi projetada para ser simples e acessível, não exigindo conhecimento técnico em computação, permitindo que qualquer profissional utilize os resultados para tomar decisões estratégicas.

2.2 Dados de Entrada

Os dados de entrada que devem ser fornecidos incluem única e exclusivamente as avaliações em formato textual providas pelos consumidores, que serão processadas para identificar os sentimentos expressos.

2.3 Resultados Fornecidos

A aplicação se baseia num modelo simbólico o qual classifica cada avaliação como **positiva**, **negativa** ou **neutra**, com base na opinião expressa. Além disso, fornece um índice de sentimento (de -1 a +1) que indica a intensidade da opinião, onde valores próximos a +1 são muito positivos, próximos a -1 são muito negativos, e próximos a 0 são neutros. Os resultados são apresentados em uma tabela simples, fácil de entender.

2.4 Finalidade

Entendemos que tal solução beneficiará tanto as empresas, ao ajustar suas estratégias de marketing e desenvolvimento de produtos, quanto os consumidores, ao melhorar a qualidade dos produtos e serviços oferecidos. Adicionalmente, tal solução será de código aberto, resultando em algumas vantagens significativas, por exemplo: transparência dos códigos, custo-efetividade e personalização.

3 Informações sobre o Corpus Escolhido

O Corpus escolhido para o desenvolvimento do presente trabalho corresponde à base de dados e informações “B2W-Reviews01.csv” disponibilizada pelas Lojas Americanas. Tal base de dados e informações encontra-se disponível no Github oficial das Americanas-tech[7], e corresponde a um Corpus aberto de avaliações de produtos ofertados pelas Lojas Americanas na sua plataforma de e-commerce. Este Corpus contempla mais de 130 mil avaliações de clientes, as quais foram coletadas do site Americanas.com entre os meses de janeiro e maio de 2018. Adicionalmente, este Corpus oferece informações importantes para a análise exploratória de dados contemplando o perfil do

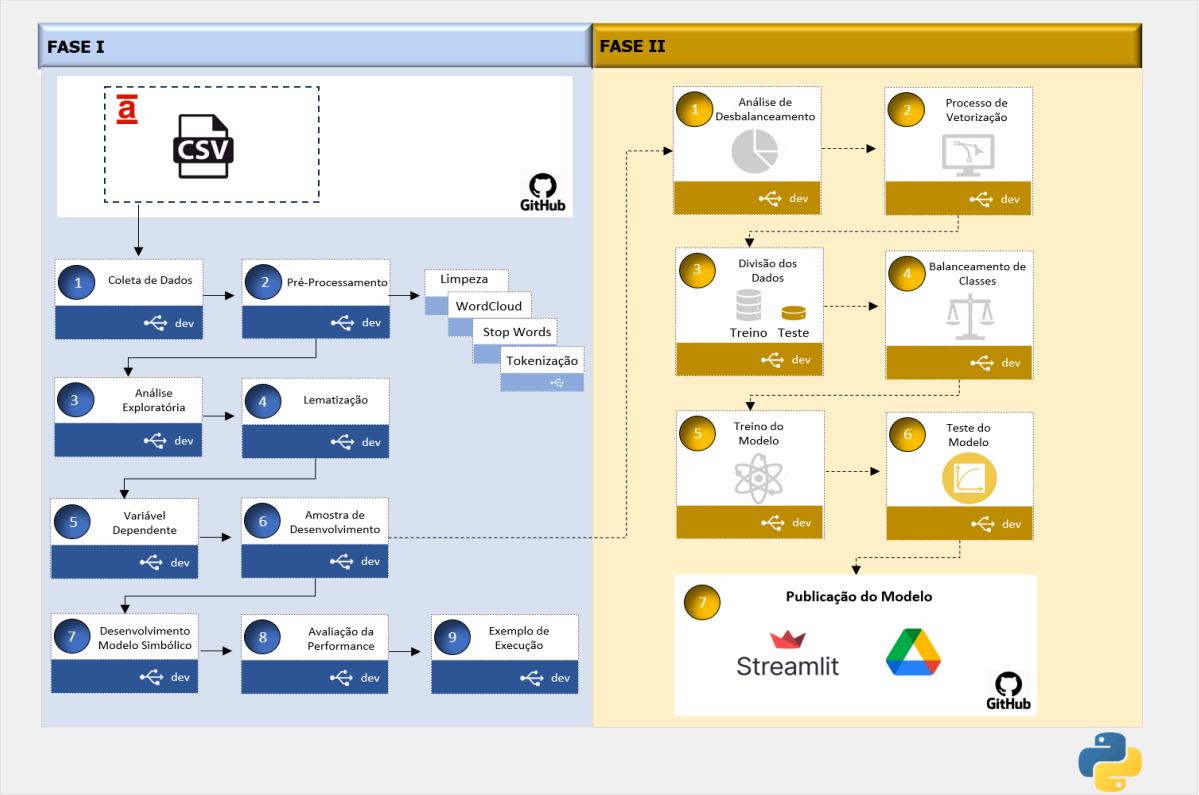
[7] Dados disponíveis em: https://github.com/americanas-tech/b2w-reviews01/blob/main/B2W-Reviews01.csv

avaliador, como gênero, idade e localização geográfica. O Corpus também apresenta duas diferentes taxas de avaliação providas por clientes, dentre as quais se destacam:

1) Uma escala de avaliação entre 1 e 5 pontos, em que: 1 = Ruim, 2 = Regular, 3 = Bom, 4= Ótimo e 5 = Excelente; e
2) Uma pergunta com as alternativas "Sim“ ou “Não" que representa a disposição do cliente em recomendar o produto a outra pessoa.

4 Arquitetura da Solução Proposta

O diagrama ilustrado abaixo, contempla a arquitetura da solução proposta pelo grupo para a Fase 1. Adicionalmente, apresentamos um resumo da arquitetura planejada referente a Fase 2.



Fonte - Elaborado pelos autores.

5 Coleta de Dados

A coleta de dados corresponde a um processo de extrema importância para o desenvolvimento de qualquer modelo simbólico eficaz. Neste capítulo, detalharemos o processo através do qual os dados foram obtidos, processados e preparados para análise. Com base no que foi discutido no capítulo anterior, utilizamos o repositório oficial da Americanas-tech no Github como a principal fonte de dados. Para garantir a precisão e a eficiência na extração dos dados, foram desenvolvidos códigos específicos na linguagem Python, proporcionando uma coleta automatizada e sistemática. Esses

códigos, juntamente com a documentação detalhada, estão disponíveis no material anexo, permitindo a reprodução e a verificação dos métodos utilizados.

6 Pré-processamento dos Dados

O pré-processamento dos dados pode ser considerada uma das fases mais importantes no desenvolvimento de modelos preditivos, a qual envolve várias atividades essenciais para garantir a qualidade e a eficácia do modelo. Os capítulos a seguir, detalham as principais etapas efetuadas.

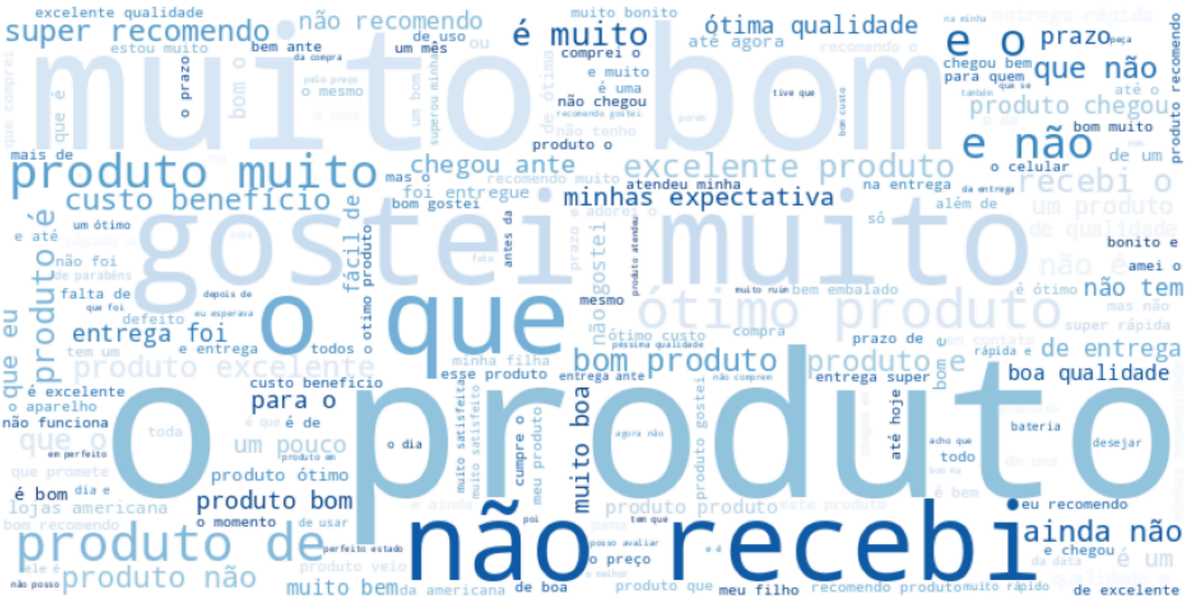
6.1 Limpeza dos Dados

No processo de limpeza dos dados, visamos garantir que as informações utilizadas no desenvolvimento do nosso modelo simbólico estivessem em um formato adequado para análise, sem ruídos que pudessem comprometer os resultados finais da nossa análise. Dentre os processos de limpeza de dados efetuados, destacam-se:

1) Valores Nulos: Primeiramente, removemos todas as observações que continham valores nulos em qualquer uma das colunas. A função dropna() foi utilizada para este propósito, garantindo que apenas registros completos fossem considerados na análise.
2) Dados Duplicados: Em seguida, valores duplicados foram removidos com o objetivo de evitar a redundância de informações. Isso foi feito utilizando a função drop_duplicates(), mantendo apenas a primeira ocorrência de cada registro duplicado.
3) Padronização: Para padronizar os textos das avaliações dos clientes, todos os caracteres foram convertidos para minúsculas. Isso ajuda a evitar a distinção entre palavras que deveriam ser consideradas iguais, independentemente de estarem em maiúsculas ou minúsculas.
4) Pontuações: Os caracteres de pontuação dos textos das avaliações foram substituídos por espaços. Isso foi feito para garantir que apenas palavras e espaços fossem mantidos, facilitando a análise subsequente.
5) Números: Por fim, todos os números presentes nos textos das avaliações foram removidos. Números geralmente não contribuem para a análise de sentimentos e podem introduzir ruídos nos dados.

6.2 Nuvem de Palavras

A Nuvem de Palavras corresponde a uma representação visual que destaca as palavras mais frequentes em um conjunto de textos, onde o tamanho de cada palavra é proporcional à sua frequência. A imagem a seguir, contribuiu para que o nosso grupo pudesse identificar rapidamente as palavras mais comuns nos comentários dos clientes, proporcionando insights visuais sobre os temas e sentimentos predominantes:



Fonte - Elaborado pelos autores.

Para o desenvolvimento da nuvem de palavras ilustrada acima utilizamos a biblioteca WordCloud do Python.

6.3 Stop Words

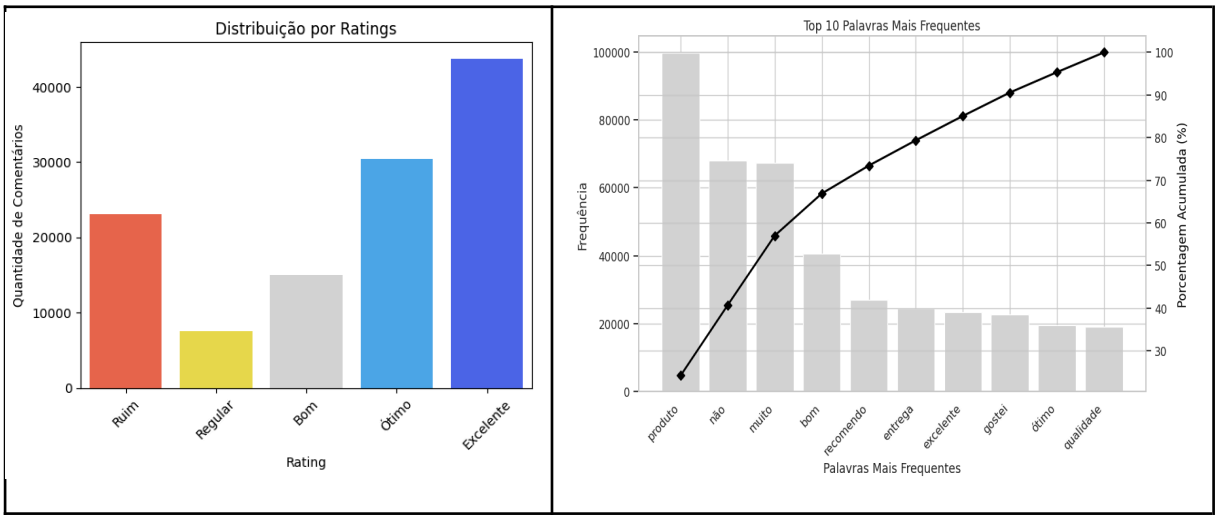
Em projetos de análise de sentimentos, a etapa de remoção de stop words é uma etapa fundamental do pré-processamento de texto. Via de regra, as stop words correspondem a palavras que, em geral, não agregam significado relevante ao contexto do texto quando analisado em um modelo de análise de sentimentos. No Corpus utilizado para o desenvolvimento deste trabalho foram identificadas 177 diferentes stop words. Ao efetuarmos uma análise detalhada de cada stop word, entendemos que a única stop word que deveria ser mantida correspondia a palavra “Não”, dado que tal stop word foi comumente utilizada pelos consumidores para expressar um sentimento negativo referente a um determinado produto. No processo de remoção de stop words foram consideradas as funcionalidades providas pela biblioteca NLTK.

6.4 Tokenização

A Tokenização corresponde ao processo de dividir um texto em unidades menores, chamadas de "tokens". Esses tokens podem ser palavras, frases ou até mesmo caracteres, dependendo do nível de tokenização que se deseja aplicar. No nosso trabalho utilizamos a função word_tokenize da biblioteca NLTK para dividir os comentários providos pelos clientes em palavras individuais. Tal processo permitiu ao nosso grupo manipular e analisar os comentários dos clientes de maneira mais granular, o que é fundamental para extrair informações significativas e relevantes.

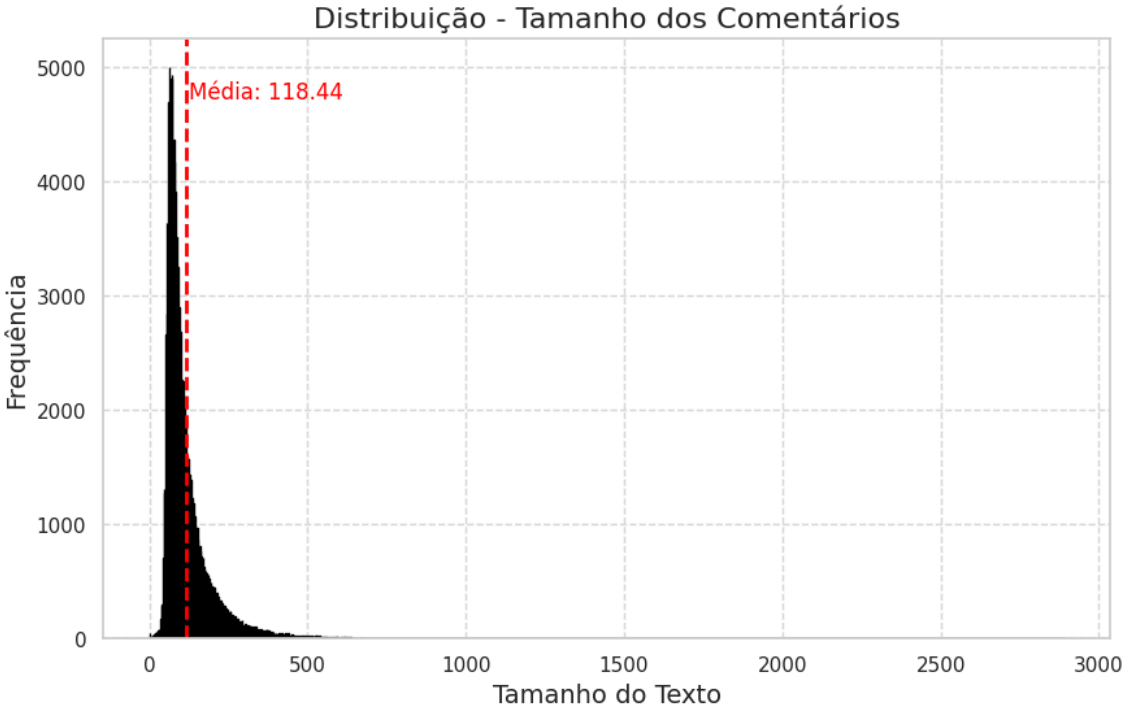
7 Análise Exploratória dos Dados

O intuito dessa etapa é permitir uma compreensão inicial do conjunto de dados disponível. Primeiramente, duas visualizações de dados foram desenvolvidas, as quais nos auxiliaram a entender a distribuição dos ratings atribuídos pelos clientes, bem como as dez palavras utilizadas com maior frequência.



Fonte - Elaborado pelos autores.

Adicionalmente, foi desenvolvido um histograma contendo a distribuição do comprimento dos comentários após o pré-processamento. Essa visualização nos permitiu identificar como os tamanhos dos comentários variam, fornecendo uma visão clara da densidade e dispersão dos dados. Ao visualizarmos a distribuição dos tamanhos dos comentários, pudemos identificar padrões, como a prevalência de comentários curtos ou longos. Isso pode indicar a natureza das interações dos clientes, como comentários mais curtos talvez indicando feedback rápido e direto, enquanto os mais longos podem conter opiniões mais detalhadas.



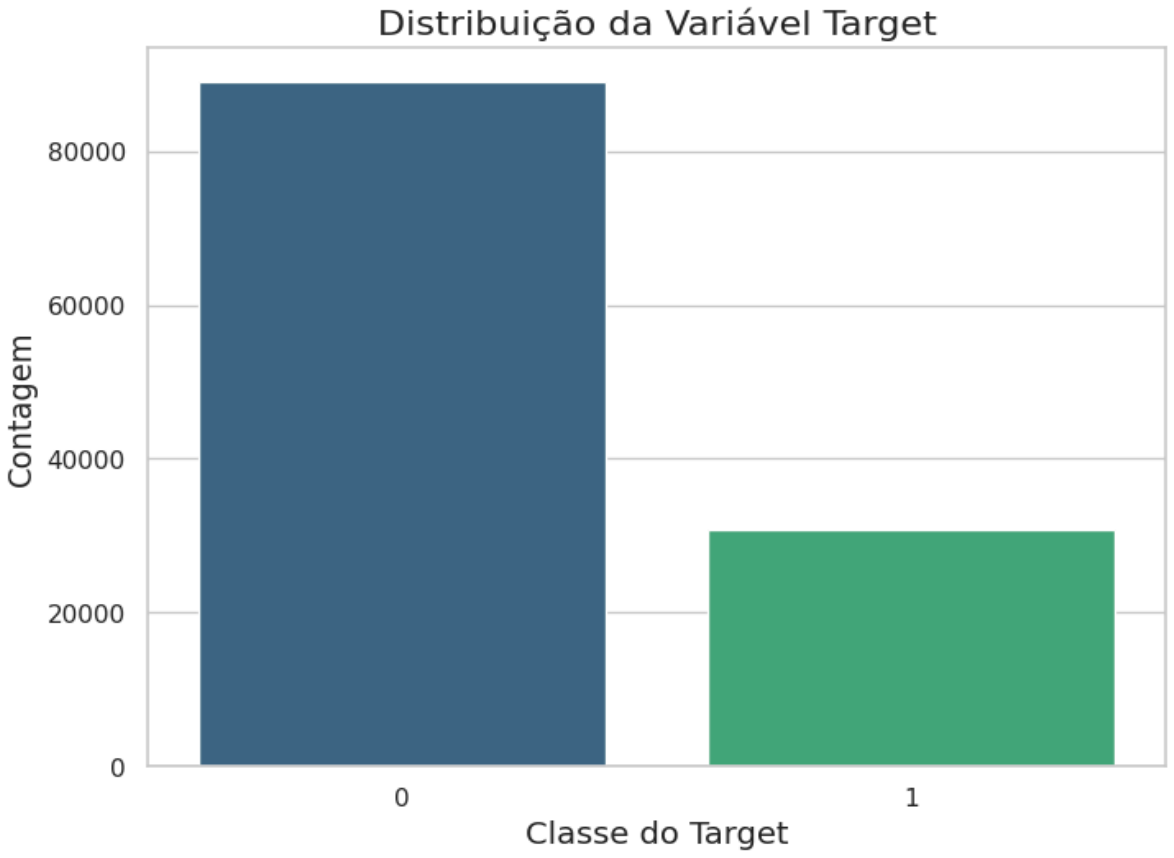
Fonte - Elaborado pelos autores.

8 Processo de Lematização

A lematização tem como principal objetivo reduzir as palavras em um texto à sua forma base ou raiz, conhecida como *lemma*, enquanto preserva o contexto e a integridade semântica das palavras. Esse processo consiste em identificar termos que efetivamente pertencem à língua, levando em conta sua respectiva classe gramatical. Entendemos que este aspecto é de extrema importância para o nosso projeto de análise de sentimentos, onde a precisão na compreensão dos significados das palavras é crucial para a interpretação correta dos sentimentos expressos nos textos. Para o desenvolvimento desta etapa utilizamos a biblioteca spaCy do Python,

9 Definição da Variável Dependente

No contexto da análise de sentimentos, a variável dependente binária adotada neste modelo é o atributo "recommend_to_a_friend", que classifica as respostas em duas categorias: "Yes" e "No". Para a construção do modelo, essas categorias foram mapeadas para valores numéricos binários, onde "Yes" foi codificado como 0 e "No" como 1. Essa abordagem permite que o modelo identifique e analise a polaridade das opiniões dos usuários, facilitando a interpretação dos resultados. A nova variável 'target' foi então incorporada ao conjunto de dados, possibilitando a realização de análises mais aprofundadas sobre a disposição dos consumidores em recomendar um produto ou serviço a um amigo. A seguir, apresentamos a distribuição desta variável:



Fonte - Elaborado pelos autores.

10 Amostra de Desenvolvimento

Essa é uma etapa crucial no processo de construção de modelos de análise de sentimentos, pois define a base sobre a qual o modelo será treinado e testado. Ao término de todas as etapas mencionadas nos capítulos anteriores, dispomos agora de uma base de dados e informações processada e pronta para ser consumida, a qual representa um passo significativo na construção do nosso modelo de análise de sentimentos. Esta base de dados e informações foi utilizada para a construção do modelo simbólico, bem como para a construção do modelo de aprendizado de máquina supervisionado referente a fase II.

11 Modelo Simbólico

O modelo simbólico elaborado pelo nosso grupo combinou regras léxicas e o processamento linguístico para análise de sentimentos. Tal modelo foi desenvolvido com adaptações do VADER para português, inspirando-se no código disponibilizado no repositório do NLTK (Hutto, 2021, https://github.com/nltk/nltk/blob/develop/nltk/sentiment/vader.py) e utilizando léxicos especializados obtidos do projeto LeIA (Araujo, 2021, https://github.com/rafjaa/LeIA). Destacam-se os seguintes léxicos utilizados:

- **vader_lexicon_ptbr.txt**: 7.458 termos com polaridades validadas (ex: "abandono" = -1.9, "amoroso" = +2.3);
- **booster.txt**: Intensificadores como "muito" (+0.293) e "pouco" (-0.293), ajustando a força dos sentimentos;
- **negate.txt**: 60 termos de negação (ex: "nunca", "nem") que invertem polaridades dos termos seguintes;
- **emoji_utf8_lexicon_ptbr.txt**: 3.570 emojis traduzidos (ex: 😊 = "rosto sorridente").

O processamento de um determinado texto pelo modelo simbólico ocorre em uma sequência estruturada de etapas interligadas, conforme ilustrado abaixo:

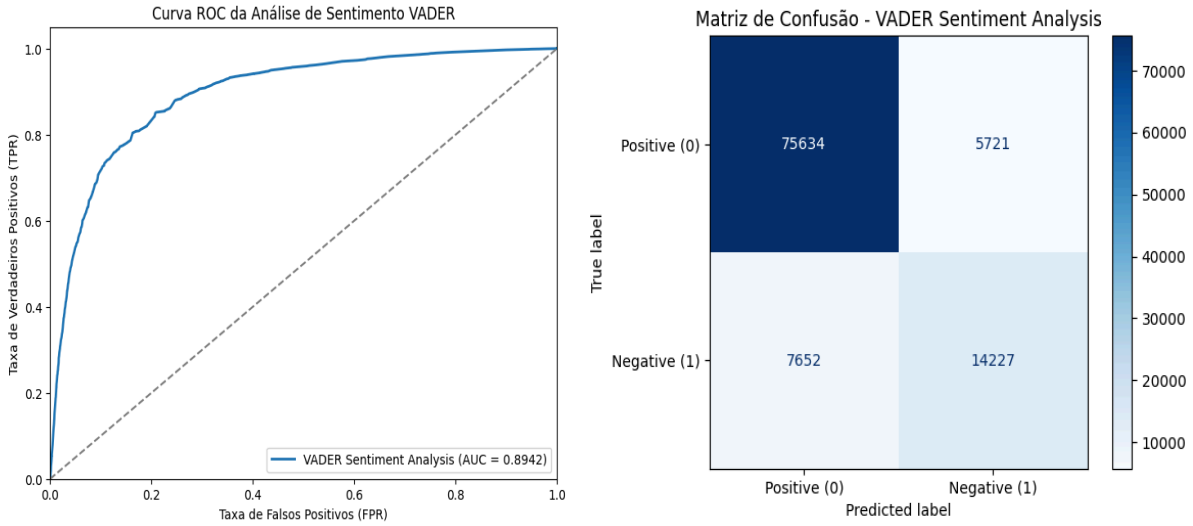
Primeiramente, o texto é normalizado através da remoção de acentos e pontuação, padronizando a entrada para análise léxica. Em seguida, o algoritmo verifica bigramas para identificar expressões idiomáticas pré-mapeadas (ex: "mó ruim") e termos compostos no léxico, garantindo que combinações específicas não sejam fragmentadas. Paralelamente, intensificadores ("muito", "pouco") são detectados e acumulam valores de *boosting* (B_INCR / B_DECR) que ajustam a intensidade do sentimento subsequente. Quando uma negação é identificada (ex: "não", "jamais"), um sinalizador é ativado para inverter a polaridade do próximo termo relevante. Conjunções adversativas, como "mas", disparam um *reset* imediato do contexto, zerando o *score* acumulado para evitar contradições.

Finalmente, a valência total é calculada integrando os efeitos cumulativos de *boosters*, negações e *resets*, com ajustes empíricos para casos onde múltiplas regras interagem (ex: "nunca foi

completamente ruim" → negação dupla + *booster*). Essa *pipeline* garante que interações complexas entre palavras sejam modeladas sem dependência de treinamento estatístico.

12 Performance do Modelo Simbólico

O modelo simbólico baseado no VADER, desenvolvido pelo nosso grupo, alcançou uma acurácia global de 87% em 103.234 avaliações, com F1-score macro de 80%, demonstrando capacidade robusta para classificação de sentimentos em larga escala. Para a classe majoritária, positiva, com 81.355 casos, obteve precisão de 91% e recall de 93%, refletindo eficiência na identificação de padrões claros, como adjetivos positivos, por exemplo, "excelente" e "fantástico", além de expressões idiomáticas mapeadas. No entanto, para a classe minoritária, negativa, com 21.879 casos, as métricas foram menores, com precisão de 71% e recall de 65%, indicando desafios na captura de nuances contextuais. A matriz de confusão revela 75.634 verdadeiros positivos, 5.721 falsos negativos, 7.652 falsos positivos e 14.227 verdadeiros negativos, confirmando o desempenho sólido, mas com erros mais frequentes na classe negativa. A AUC de 0.8942 demonstra boa capacidade de discriminação entre as classes. As principais limitações incluem dificuldades em interpretar textos com linguagem informal, erros gramaticais e vocabulário não padrão, como em avaliações que descrevem "propaganda enganosa", "produto de qualidade inferior" ou "lixo", frequentemente classificadas incorretamente. Além disso, o modelo enfrenta desafios com sarcasmo, ironia e o viés causado pelo desbalanceamento das classes, com proporção aproximada de 3.7:1. Apesar disso, o modelo mostrou-se robusto para aplicações práticas em varejo, apontando caminhos para melhorias futuras, como expansão do léxico para lidar com linguagem coloquial e tratamento de contextos semânticos ambíguos.



Fonte - Elaborado pelos autores.

13 Exemplo de Execução

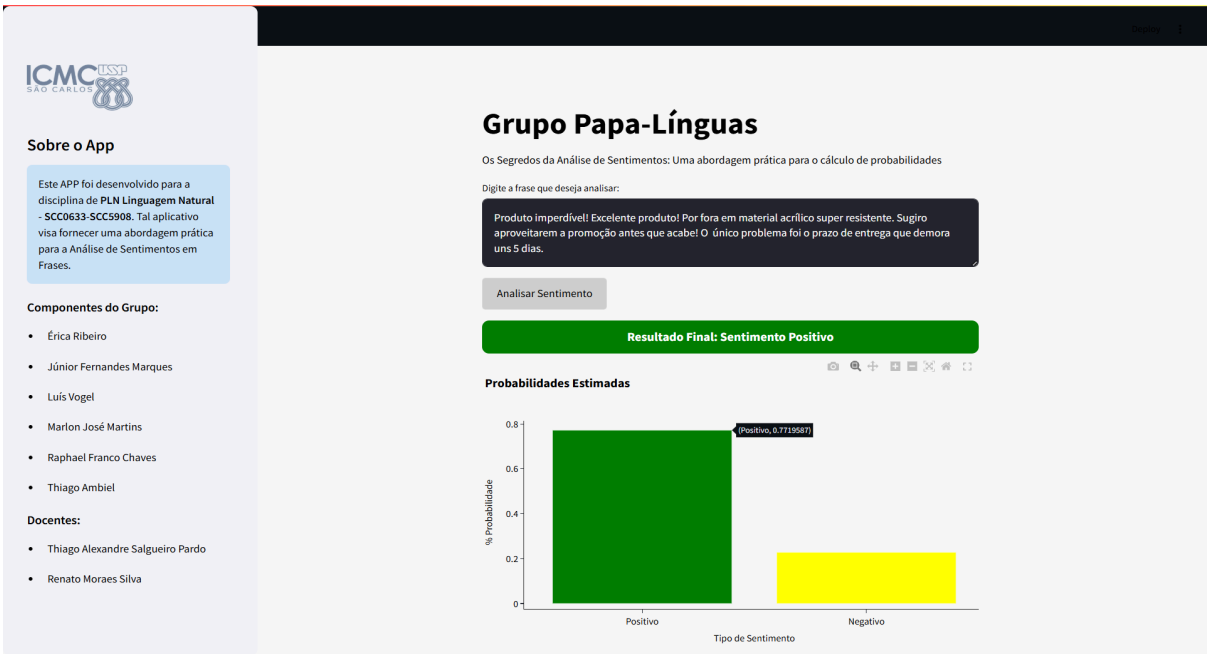
A aplicação do modelo ocorre por meio da classe *SentimentIntensityAnalyzer*, que analisa textos para identificar sentimentos expressos em avaliações de consumidores. A função *analise_sentimento* processa textos pré-processados, classificando-os como positivos, negativos ou neutros com base no índice de sentimento do VADER. No dataset, a função é aplicada à coluna de avaliações lematizadas, gerando uma classificação automática para milhares de reviews, o que permite aos gestores de varejo entender rapidamente as opiniões dos clientes. Por exemplo, uma avaliação como "preço imbatível e qualidade excelente" é classificada como positiva, enquanto "produto de qualidade inferior, material frágil" é identificada como negativa. Além disso, a função interativa *analisar_comentario* possibilita que usuários insiram comentários em tempo real, recebendo a classificação instantânea, ideal para análises pontuais. Contudo, o modelo enfrenta dificuldades com textos em linguagem informal ou com erros gramaticais, como em avaliações que mencionam "propaganda enganosa e produto de péssima qualidade", que podem ser incorretamente classificadas como neutras devido a vocabulário não padrão ou estruturas confusas. Essas limitações destacam a necessidade de expandir o léxico para lidar com a linguagem coloquial. Ainda assim, a aplicação é eficaz para identificar padrões claros de sentimento, atendendo ao objetivo de fornecer insights valiosos para estratégias de marketing e melhoria de produtos no varejo.

```
Comentário: E achei a entrega muito demorada
Score: 0.29
Sentimento simbólico: neutro

Comentário: Eu gostei do produto, mas faltou atenção na entrega
Score: -1.40
Sentimento simbólico: negativo

Comentário: Eu não gostei da devolutiva após minha reclamação
Score: -1.80
Sentimento simbólico: negativo

Comentário: Eu amei a entrega. Chegou certinho
Score: 0.00
Sentimento simbólico: neutro

Comentário:
Score: 0.00
Sentimento simbólico: neutro
Análise encerrada.
```

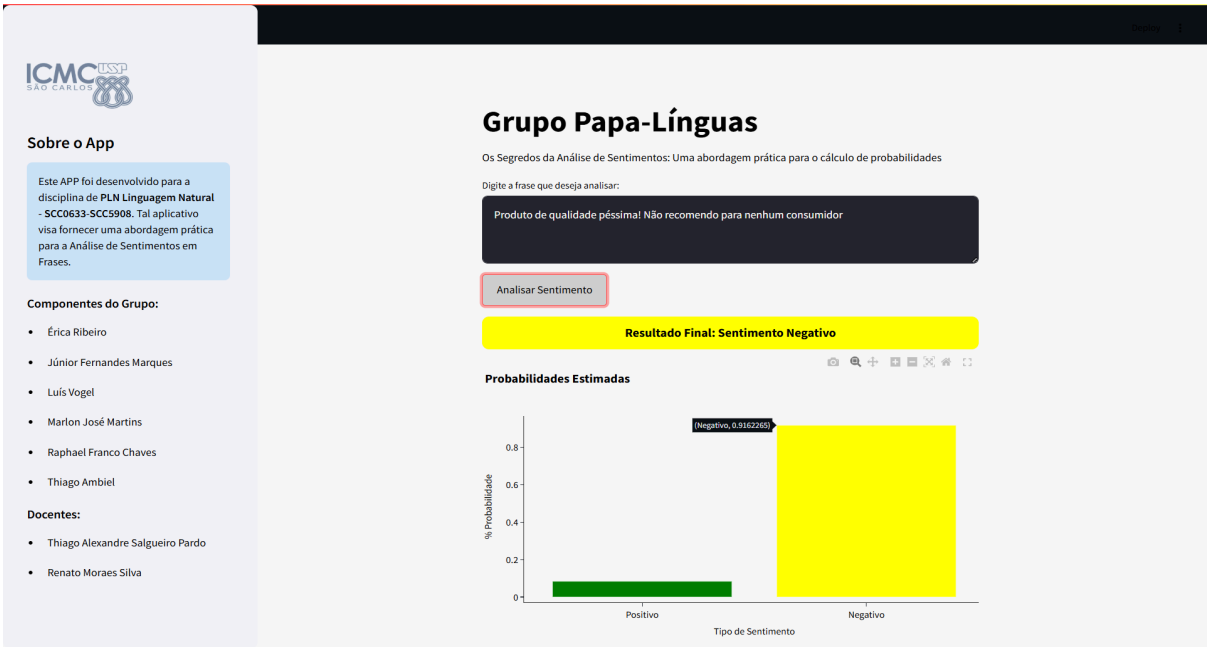
Fonte - Elaborado pelos autores.

14 Planejamento da Fase II

Conforme previamente ilustrado no capítulo 4, o nosso grupo já dispõe de uma arquitetura planejada para a Fase II. Ressaltamos que alguns testes já foram realizados com diferentes algoritmos de aprendizado de máquina supervisionado, com o objetivo principal de aumentar a acurácia do nosso modelo de análise de sentimentos. Nossa expectativa é que o nosso modelo seja capaz de classificar se o conteúdo de uma frase fornecida por um(a) determinado(a) usuário(a) apresenta um sentimento positivo ou negativo, além de fornecer a probabilidade associada a tal classificação. As imagens a seguir, demonstram o nosso planejamento para a publicação em produção utilizando o pacote Streamlit do Python, na qual a primeira imagem corresponde a uma frase que obteve uma classificação final referente a um sentimento positivo, e a segunda imagem a um sentimento negativo:



Fonte - Elaborado pelos autores.



Fonte - Elaborado pelos autores.