

INSTITUTO DE TECNOLOGIA E LIDERANÇA

**APLICAÇÃO UTILIZANDO PROCESSAMENTO DE
LINGUAGEM NATURAL**

Banco BTG Pactual

Autores:

Gustavo Monteiro;

Izabella Almeida de Faria;

Luiz Augusto Pompeo de Camargo Franco Ferreira;

Patrick Victorino Miranda;

Raduan Muarrek;

Ueliton Moreira Rocha;

Vitória Rodrigues de Oliveira.

Controle do documento

Histórico de revisões

| Data | Autor | Versão | Resumo da atividade |
|------------|---|--------|---|
| 13/06/2023 | Izabella Faria | 1.0 | Criação do documento |
| 14/06/2023 | Izabella Faria | 1.1 | <ul style="list-style-type: none">• Sumarização do documento• adição do texto referente à metodologia |
| 19/06/2023 | Izabella Faria | 1.2 | <ul style="list-style-type: none">• Personas• Matriz de risco• Canva da proposta de valor• Matriz oceano azul• Análise financeira do projeto• User stories• Arquitetura da solução• Processamento dos dados• vetorização e word embedding• Modelagem |
| 20/06/2023 | Vitória Rodrigues de Oliveira | 1.3 | <ul style="list-style-type: none">• Diagrama de Implantação UML |
| 20/06/2023 | Patrick Victorino Miranda e Luiz Ferreira | 1.4 | <ul style="list-style-type: none">• Deploy e visualização do modelo |
| 21/06/2023 | Gustavo Monteiro | 1.5 | <ul style="list-style-type: none">• Incremento do Diagrama de Implantação |
| 22/06/2023 | Gustavo Monteiro | 1.6 | <ul style="list-style-type: none">• Incremento dos textos de resultado |
| 22/06/2023 | Patrick Victorino Miranda, Gustavo Monteiro, Ueliton Rocha, Raduan Oliveira | 1.7 | <ul style="list-style-type: none">• Número de figuras adicionado• Introdução, Solução, Objetivos e Problema• Ferramentas e Biblioteca |
| 22/06/2023 | Raduan Oliveira | 1.8 | <ul style="list-style-type: none">• Revisão geral |
| | | | |
| | | | |

Sumário

| | |
|---|-----------|
| 1. Introdução..... | 6 |
| 2. Problema..... | 6 |
| 3. Objetivo..... | 6 |
| 4. Solução..... | 6 |
| 5. Entendimento do Negócio..... | 7 |
| 5.1.1. Matriz oceano azul..... | 7 |
| 5.1.2. Matriz de risco..... | 11 |
| 5.1.3. Canvas da proposta de valor..... | 13 |
| 5.1.4. Análise financeira do projeto..... | 15 |
| 6. Entendimento da experiência do usuário..... | 17 |
| 6.1. Personas..... | 17 |
| 6.2. User stories..... | 19 |
| 7. Metodologia..... | 27 |
| 7.1. Metodologia CRISP-DM..... | 27 |
| 7.1.1. Entendimento dos Dados..... | 28 |
| Análise descritiva dos dados..... | 28 |
| 7.1.2. Preparação dos Dados..... | 31 |
| 7.1.2.1. Pré processamento dos dados..... | 31 |
| 7.1.2.2. Pipeline de pré-processamento..... | 32 |
| 7.1.3. Vetorização e word embedding..... | 33 |
| 7.1.3.1. Bag of words..... | 33 |
| 7.1.3.1.1. Nuvem de palavras..... | 35 |
| 7.1.3.2. Word2vec continuous bag of word..... | 36 |
| 7.1.3.3. Word2vec skip-gram..... | 37 |
| 7.1.3.4. Sentence transformers..... | 37 |
| 7.1.4. Modelagem..... | 37 |
| 7.1.4.1. Naive bayes com bag of word..... | 37 |
| 7.1.4.2. Naive bayes com Word2Vec..... | 38 |
| 7.1.4.3. Support vector machine com Word2Vec..... | 39 |
| 7.1.4.4. Rede neural com transformers..... | 40 |
| 7.1.5. Métricas para Avaliação..... | 42 |
| 7.1.6. Implantação..... | 43 |
| 7.1.6.1. Arquitetura da solução..... | 43 |
| 8. Resultados..... | 47 |
| 9. Discussão e Conclusão..... | 52 |
| 9.1. Comparação entre os modelos..... | 52 |
| 9.2. Modelo escolhido..... | 53 |
| 9.2.1. Critérios e justificativas para escolha do modelo..... | 55 |
| 10. Deploy e visualização do modelo..... | 57 |
| 11. Ferramentas..... | 62 |
| 12. Bibliotecas..... | 62 |
| 13. Referências bibliográficas..... | 63 |

1. Introdução

O documento a seguir contextualiza e apresenta a solução desenvolvida por alunos do 3º semestre de Sistemas de Informação do Inteli para atender a uma demanda apresentada pelo BTG Pactual, utilizar Processamento de Linguagem Natural (PLN) para facilitar a análise das redes sociais e da opinião pública sobre a empresa.

Será apresentado o contexto de experiência do usuário (UX), a visão de negócios para a solução e o processo de criação do modelo, assim como o racional utilizado pelo grupo, para eleger o modelo principal que seria usado, além de detalhes sobre as especificidades do nosso código.

2. Problema

Muitas empresas utilizam redes sociais para manter uma proximidade com seus clientes e os usuários da rede, promovendo a marca e obtendo feedbacks constantes de seus produtos e serviços.

O marketing representa uma parte importante do orçamento total de uma empresa, 13,6% aproximadamente, de acordo com a Pesquisa Anual de CMO da Deloitte. Isso está provocando uma mudança na priorização do marketing, focando mais em rede social, já que o impacto de usuários com comentários negativos sobre a reputação da marca pode ser significativo.

Mas a gestão de redes sociais pode demandar muito tempo para a equipe de marketing, necessitando de uma forma de análise mais apropriada para identificar os comentários negativos de usuários, prevenindo crises e cancelamentos na imagem e reputação da marca.

3. Objetivo

Utilizar Processamento de Linguagem Natural para identificar rapidamente comentários negativos de usuários na rede social do Instagram do parceiro de projeto, identificando palavras-chave e classificando-as utilizando um modelo preditivo que utilize a técnica de aprendizado de máquina em análise de sentimento.

4. Solução

A solução desenvolvida é um programa no qual os colaboradores do BTG podem analisar os sentimentos expressos pelos seguidores em uma determinada rede social. O programa recebe um arquivo CSV que deve conter, no mínimo, os comentários ou textos a serem analisados, juntamente com a data em que foram feitos. Com base nessas informações, são gerados gráficos e tabelas que auxiliam na análise dos sentimentos e na frequência de termos e palavras.

5. Entendimento do Negócio

5.1.1. Matriz oceano azul

Desenvolvida por W. Chan Kim e Renée Mauborgne, no livro "Blue Ocean Strategy" (Estratégia do Oceano Azul), a matriz oceano azul é uma ferramenta que auxilia na identificação de oportunidades pouco exploradas pelo mercado, o que atenua a entrada ou atuação em ambientes que já possuem muitos produtos semelhantes que competem diretamente entre si. Ao utilizar a matriz, é possível elaborar o modelo de quatro ações, que visa identificar diferentes oportunidades para diferenciação de produtos já existentes:

- **Eliminar:** essa ação remove atributos que a indústria vê como essenciais, mas que não interferem tanto na visão de valor do cliente.
- **Reduzir:** os atributos são reduzidos quando comparados ao padrão do setor, tendo em vista que isso não impacta negativamente o usuário final.
- **Aumentar:** nessa ação é possível incrementar atributos a fim de deixá-los mais em evidência com relação às empresas concorrentes analisadas.
- **Criar:** essa ação permite a criação de um novo atributo que a indústria não explorou ou ofereceu aos seus clientes.

| Atributos | Feel Good | Hootsuite | Sprout Social | Brandwatch |
|---------------------------|-----------|-----------|---------------|------------|
| Relação qualidade/preço | 10 | 7 | 6 | 5 |
| Qualidade | 6 | 9 | 9 | 7 |
| Suporte ao cliente | 0 | 8 | 9 | 10 |
| Disponibilidade nacional | 10 | 9 | 7 | 0 |
| Personalização do produto | 10 | 7 | 7 | 7 |
| Praticidade | 9 | 9 | 10 | 8 |
| Escalabilidade | 6 | 8 | 9 | 8 |
| Método de coleta de dados | 10 | 8 | 8 | 8 |
| Transparência | 10 | 7 | 7 | 7 |
| Velocidade de entrega | 7 | 10 | 10 | 10 |
| Modulação dos resultados | 10 | 0 | 0 | 0 |

Tabela 1: tabela referente à matriz oceano azul, que compara a nossa solução com as demais existentes no mercado.

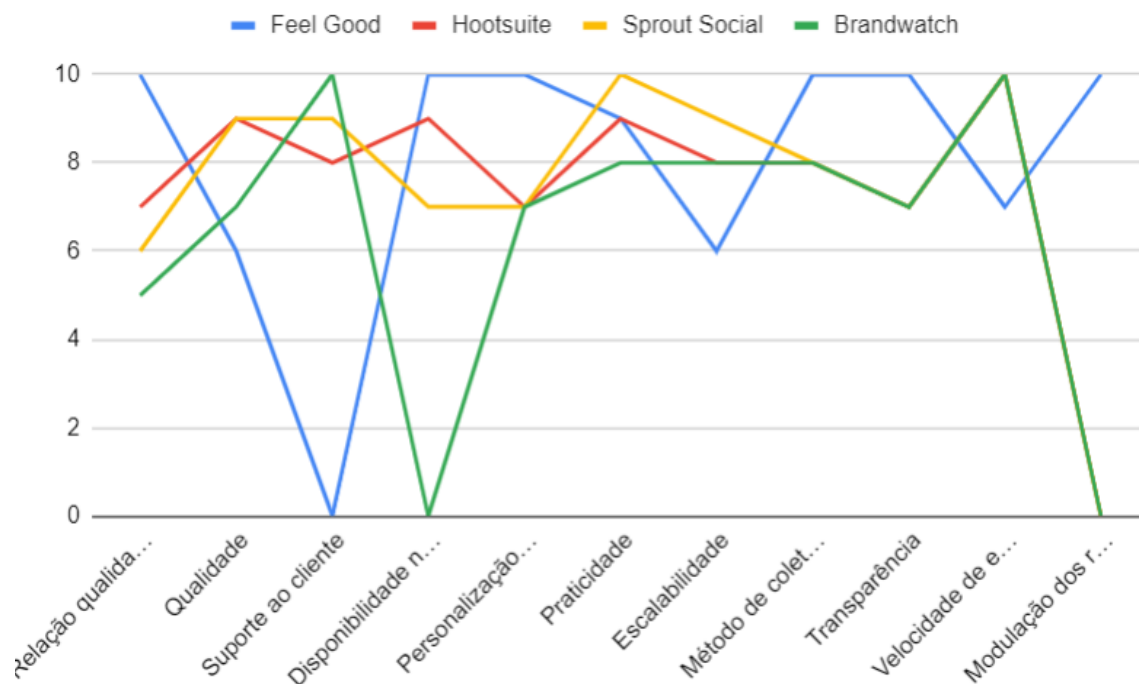


Gráfico 1: Imagem referente à representação gráfica dos dados dispostos na tabela 1.

- 1. Relação qualidade/preço:** Esse parâmetro avalia a qualidade do produto em relação ao preço cobrado pela empresa. Ele deve descrever se o preço do serviço está de acordo com a qualidade do produto oferecido e como ele se compara com outras empresas do mercado.
- 2. Qualidade:** A qualidade do produto é um fator crucial para a satisfação do cliente. Esse parâmetro deve avaliar a qualidade dos dados coletados e das análises realizadas pela empresa, bem como a precisão das informações fornecidas.
- 3. Suporte ao cliente:** Esse parâmetro avalia a qualidade do suporte oferecido pela empresa aos seus clientes. Ele deve descrever a disponibilidade e qualidade do atendimento ao cliente, além da capacidade da empresa em resolver problemas e tirar dúvidas dos clientes.
- 4. Disponibilidade:** A disponibilidade do produto é um fator importante para muitos clientes. Esse parâmetro deve avaliar a capacidade da empresa em fornecer o serviço em tempo hábil e com disponibilidade adequada, levando em conta a demanda e as necessidades dos clientes.
- 5. Personalização do produto:** Esse parâmetro avalia a capacidade da empresa em personalizar o produto de acordo com as necessidades dos clientes. Ele deve descrever como a empresa pode adaptar suas soluções para atender às demandas específicas dos clientes e como isso pode afetar o preço e a qualidade do serviço.

6. Praticidade: Esse parâmetro avalia a facilidade de uso e a praticidade do produto oferecido pela empresa. Ele deve descrever a qualidade da interface do usuário, a facilidade de acesso aos dados e a eficácia das ferramentas de análise disponibilizadas.

7. Escalabilidade: Esse parâmetro avalia a capacidade da empresa em atender às necessidades de crescimento dos clientes. Ele deve descrever a escalabilidade do serviço oferecido, levando em conta a possibilidade de aumento da demanda e de expansão das atividades dos clientes.

8. Método de coleta de dados: Esse parâmetro avalia a eficácia do método utilizado pela empresa para coletar os dados utilizados na análise de sentimento. Ele deve descrever como a empresa coleta os dados, sua eficácia em coletar informações relevantes e a capacidade de manter a privacidade e segurança das informações coletadas.

9. Transparência: Esse parâmetro avalia a transparência da empresa em relação ao processo de análise de sentimento. Ele deve descrever como a empresa realiza as análises, como apresenta os resultados e como lida com a possibilidade de erros e inconsistências nos resultados obtidos. A transparência é importante para a confiança do cliente na empresa.

10. Tempo: Tempo necessário para realizar todas as atividades relacionadas à análise de sentimento, desde a coleta até a apresentação dos resultados, incluindo a eficiência dos processos e a capacidade da empresa de cumprir prazos estipulados pelo cliente.

11. Modulação dos resultados: Possibilidade de utilizar partes das soluções e os algoritmos de forma isolada. Com isso os usuários podem extrair insights úteis para suas respectivas equipes e setores.

Modelo de quatro ações:

Eliminou: diante dos parâmetros listados, o suporte ao cliente foi eliminado (devido a isso, a nota desse atributo na matriz é igual a 0), uma vez que equipe de desenvolvimento, por se tratar de um time acadêmico que segue um cronograma rígido, não disponibilizará suporte ou manutenção para modificações futuras.

Reduziu: Por se tratar de uma solução desenvolvida por estudantes, que não gera nenhum tipo de custo para a empresa, a qualidade do produto foi reduzida. Diante das circunstâncias, a prioridade do time é desenvolver uma solução cabível e bem estruturada, sempre prezando pelo aprendizado do time. Devido a esse fator, o atributo qualidade recebeu uma nota menor que as outras empresas do mercado.

A escalabilidade foi reduzida para tornar a solução mais voltada para um tipo específico de rede social, aumentando a eficiência e confiança nos resultados

obtidos, visto que a especificidade favorece a construção de um algoritmo que pode perceber melhor os padrões do tipo de rede social escolhido. Tendo isso em vista, a nota foi reduzida perante os concorrentes de mercado que possuem tanto web scraping como também várias redes sociais compatíveis para a solução.

Aumentou: A velocidade gasto para a realização do projeto por parte do time será de dez semanas, enquanto que as empresas que prestam esse serviço no mercado levam de algumas horas até alguns dias para fornecer os resultados finais. Diante dessa comparação, quando colocada lado a lado com outras soluções, o projeto em questão aumentou o tempo gasto na realização da análise e, por isso, obteve a menor nota nessa linha da tabela.

A disponibilidade da solução, de forma nacional e gratuita, é importante para divulgar e possibilitar o teste do produto, que será entregue com código aberto. Tais características tornam o produto mais competitivo em relação aos seus concorrentes, aumentando, também, a relação qualidade/preço, uma vez que os gastos com a utilização desse projeto serão praticamente nulos. Quando comparado a outras soluções, esse atributo é mais visível na solução que está sendo construída nesse projeto.

Tendo em vista que o produto é totalmente voltado para as necessidades do parceiro desse projeto, a característica de personalização é muito relevante e robusta. Isso também envolve o método de coleta de dados, que será realizado pela própria empresa, não tendo necessidade de permitir o acesso de terceiros a dados sensíveis e permitindo que o cliente participe de todos os processos de construção do produto. Com relação às de outras empresas, essa solução apresenta um aumento da característica de personalização do produto, o que justifica a nota mais alta dessa linha da tabela.

É de extrema importância que o método de coleta de dados de uma empresa seja seguro e transparente. Em nosso projeto, visando a segurança e a privacidade dos dados, o método de coleta foi aprimorado, uma vez que a solução está focada na proteção da privacidade e na construção de uma relação de confiança com o cliente. Todas as ações a serem realizadas com os dados fornecidos só serão realizadas, de fato, a partir da clara permissão do parceiro e todas elas ficarão visíveis ao final do projeto, dado que se trata de uma solução de código aberto. Esses fatores aumentam a transparência, que evita violações e práticas inadequadas com os dados fornecidos.

Criou: Nas empresas tradicionais, que disponibilizam serviços de inteligência e gerenciamento de mídia social, somente é obtido o resultado final dos algoritmos que usam, não contemplando resultados referentes às etapas anteriores desse processo. No caso desse projeto, além de contemplar esse resultado através da análise de sentimento, também é possível extrair informações de diferentes etapas, como a de pré-processamento, possibilitando extrair insights dos dados em

diferentes estados. Assim, essa modularização da solução possibilita a visualização e obtenção de resultados diferentes, a depender do modo como as etapas são observadas, o que traz uma novidade ao mercado, demonstrando a criação de um valor antes inexplorado.

5.1.2. Matriz de risco

A Matriz de Risco é uma ferramenta que proporciona uma análise abrangente das ameaças e oportunidades do projeto. Ela permite mensurar a importância de cada risco com base na probabilidade de ocorrência e no nível de impacto no projeto. Com ela, podemos identificar as ameaças com maiores probabilidades e impactos, assim como as oportunidades existentes no desenvolvimento, permitindo a criação de um plano de ação efetivo.

| Matriz de risco | | | | | | | | | | |
|-----------------|--------|-------------|-------|-------|------|--------------|------------|------|-------|-------|
| Probab | Riscos | | | | | Oportunidade | | | | |
| Muito Alta | 1 | | | | | 11 | 12 | | | |
| Alta | 2 | | 1 | | | 13 | | 14 | | |
| Médio | 3 | | 2 | 3 | 4 | | | | | |
| Baixa | 4 | | | 5 | 6 | 7 | | 15 | | |
| Muito Baixa | 5 | | | 8 | 9 | 10 | | 16 | | |
| | | 1 | 2 | 3 | 4 | 5 | 5 | 4 | 3 | 2 |
| | | Muito Baixo | Baixo | Médio | Alto | Muito alto | Muito alto | Alto | Médio | Baixo |
| | | Impacto | | | | | | | | |

Tabela 2: tabela referente à matriz de risco do grupo.

Abaixo, é possível visualizar a legenda, referente à matriz de risco acima:

1. Dados fornecidos de forma inconsistente na tabela fornecida para treinamento do modelo. Esse risco apresenta-se com uma alta probabilidade de ocorrência, levando em consideração projetos anteriores, e um médio impacto, visto que o principal objetivo é o aprendizado do grupo.
2. Má seleção de palavras chave. Essa opção possui uma probabilidade baixa de ocorrência e um baixo impacto, principalmente devido ao fato de que todos os grupos da sala precisam fazer isso, então, caso haja erros, será fácil comparar e consertar esse processo.

3. Baixo conhecimento técnico das bibliotecas a serem utilizadas. Esse risco possui médio impacto e média probabilidade, uma vez que, caso venha a acontecer, seremos todo o aparato de professores disponíveis para ajudar o grupo.
4. Falta de dados na tabela fornecida para treinamento do modelo. Esse risco possui média probabilidade e impacto alto, tendo em vista que acreditamos que os parceiros fornecerão uma tabela completa, mas, caso isso não ocorra, teremos muita dificuldade para seguir com o projeto.
5. Não disponibilidade dos instrutores. Esse risco possui baixa probabilidade e médio impacto, uma vez que, mesmo que os nossos instrutores regulares estejam ocupados, outros professores do inteli se dispõem a ajudar, caso precisemos.
6. Vazamento de dados sensíveis. A probabilidade desse risco ocorrer é baixa, visto que todos tomaremos o máximo de cuidado possível, contudo, caso isso ocorra, terá um impacto muito alto, dado que esses dados são particulares do banco e de seus respectivos clientes.
7. Parte do grupo fica doente ou não consegue comparecer. A probabilidade desse risco acontecer é baixa, mas, se isso ocorrer, teremos sérios problemas de organização e realização de tarefas, o que confere a ele um impacto muito alto.
8. A solução não ser capaz de fornecer resultados conclusivos sobre o impacto emocional das propagandas. Diante de tantas bibliotecas e ferramentas possíveis, esse risco apresenta uma probabilidade muito baixa de acontecer, bem como um impacto médio, caso ocorra.
9. Desativação das bibliotecas utilizadas. Essas bibliotecas estão no mercado há um bom tempo, então é praticamente improvável que elas sejam desativadas no decorrer da realização desse projeto, por isso, a baixa probabilidade. Porém, caso ocorra, esse fator terá um alto impacto no projeto, visto que teremos que identificar novas bibliotecas para trabalhar no restante do desenvolvimento.
10. Baixo nível técnico dos integrantes em linguagem natural.
11. Grande volume de dados gerados por redes sociais que serão utilizados para treinamento eficaz do modelo.
12. Redução do tempo gasto em análises de problemas para possíveis campanhas de marketing.

Plano de ação :

Risco 1: Comunicar com o Orientador para que ela possa falar com o cliente sobre a tabela fornecida. Responsável Vitória.

Risco 2: Reunir outros grupos para avaliar a seleção de palavras chave e comparar os resultados obtidos. Responsável Izabella.

Risco 3: Comunicar com o professor de programação para atuar no entendimento das bibliotecas. Responsável Raduan.

Risco 4: Fazer o tratamento dos dados e usá-los da melhor forma após debatermos sobre esse problema com o Orientador. Responsável Gustavo.

Risco 5: Iremos recorrer ao Orientador para conversarmos sobre esse caso. Responsável Vitória.

Risco 6: Recorrer imediatamente ao professor de programação para remediar a situação e tratar o caso. Responsável Patrick.

Risco 7: Comunicar com os membros que tiverem problemas pessoais e traçar um cronograma de reuniões remotas e entregas online. Responsável Izabella.

Risco 8: Investir mais tempo no treinamento do modelo preditivo e na seleção e tratamento dos dados, para direcionar melhor os resultados, a fim de suportar as análises feitas. Responsável Uelinton.

Risco 9: O grupo, como um todo, irá conversar com o professor de programação e buscar alternativas, como outras ferramentas.

Risco 10: Um integrante do grupo irá se reunir com outros da turma para trocar informações e conhecimentos relacionados a alternativas para as bibliotecas desativadas. Responsável: Luiz Augusto.

A partir do nono risco listado, estão dispostas as oportunidades que o projeto possui. Para essas oportunidades, não foram listados responsáveis ou planos de ação a serem realizados, visto que, em sua maioria, são compostos por fatores que não compete à equipe desenvolvedora da solução tomar alguma iniciativa.

5.1.3. Canvas da proposta de valor

O Canvas de Proposta de Valor tem como objetivo apresentar, de forma visual, uma compreensão empática da persona, descrevendo suas principais atividades diárias e os desafios que enfrenta, bem como os ganhos que obtém. Em seguida, a solução proposta é apresentada, com uma descrição geral de sua proposta de valor e principais características que geram benefícios para a persona. Além disso, as funcionalidades ou recursos da solução são destacados para demonstrar como podem aliviar as dores da persona.

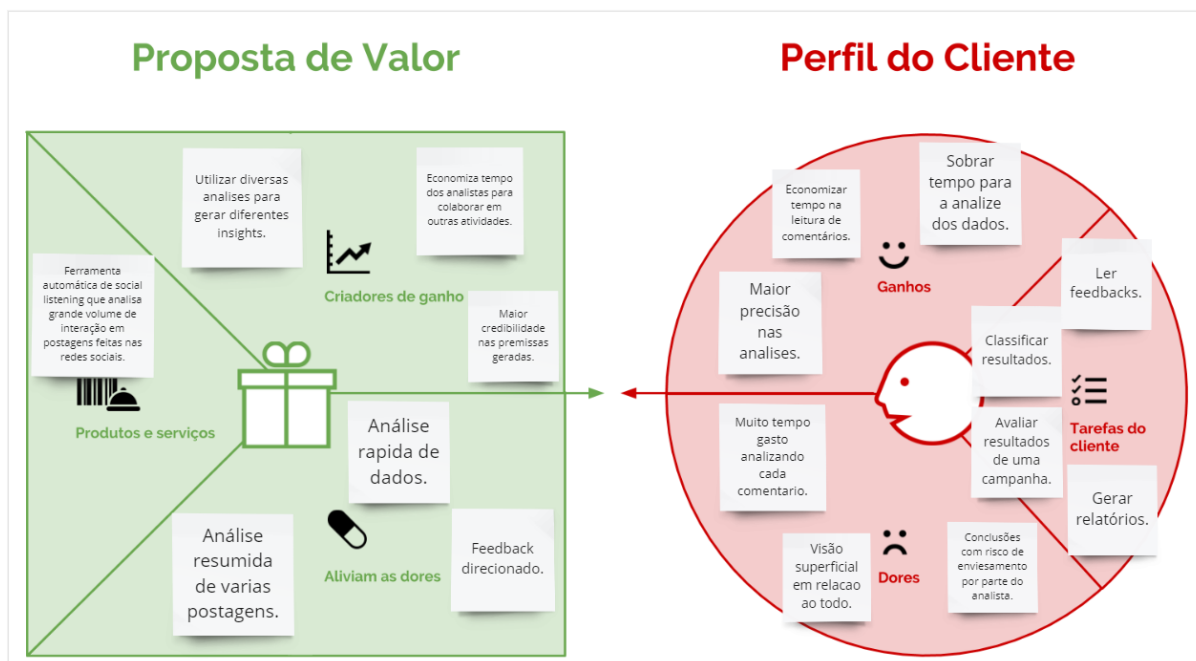


Figura 1: imagem referente ao canvas da proposta de valor elaborado.

Perfil do cliente:

Tarefa do cliente:

Nossa persona tem tarefas bem mecânicas em seu dia a dia, como ler comentários/feedback dos clientes nas redes, classificar aquilo que faz sentido, avaliar, gerar relatórios etc....

Dores:

Essa monotonicidade causa grandes dores em seu dia a dia. Se não precisasse fazer esse trabalho manual, de ler comentário por comentário, recortar o que faz sentido ou não, sobraria mais tempo para analisar e tirar insights dos principais materiais de cada campanha. Além disso, com tantas tarefas manuais em pouco tempo, a análise final sofre alguns riscos negativos como ter uma análise superficial, pois faltou tempo para a parte mais importante, a análise em si. Ou também o enviesamento, dito que o analista BTG passou tanto tempo analisando os dados, acabou criando opinião própria antes mesmo de fazer a análise final, ancorando assim o resultado na opinião do colaborador BTG.

Ganhos:

Para o analista de marketing/automation terminar seu expediente satisfeito, ele deve conseguir fazer boas análises e tirar bons insights. Para isso, precisa direcionar o tempo com as análises e não deve ter o trabalho de ler comentários diretamente nas redes ou trabalhar na base de dados.

Proposta de valor:

Produtos e serviços:

Definida nossa proposta de valor: “Ferramenta automática de social listening que analisa grande volume de interação em postagens feitas nas redes sociais.” que possui suas principais funções, reconhecer palavras chaves que estão sendo repetidas e classificar campanha entre positivo, negativo e neutro.

Alivia as dores:

Temos uma ferramenta que diferente do tempo de um humano, faz os recortes das principais informações e insights de cada campanha de forma rápida onde é possível replicar isso à várias postagens. Assim como recortar de onde exatamente vem os insights: Data, campanha x, em qual período de tempo? Tendo assim um feedback direcionado.

Criadores de ganhos:

E como ganho, a solução não terá erro humano, tendo então premissas mais confiáveis, para o analista BTG chegar em conclusões mais confiáveis. Assim como economizar tempo para fazer o mais importante, analisar e extrair insights e conseguir gerar diferentes outputs para gerar diferentes resultados.

5.1.4. Análise financeira do projeto

Inicialmente, o parceiro - banco BTG Pactual - pretende investir R\$250.000,00. Para o desenvolvimento desse projeto são esperados os custos:

- R\$100.000,00 -> Gastos com produtos AWS.
- R\$150.000,00 -> Custos de 4 desenvolvedores por 6 meses de desenvolvimento.
- O projeto não possui entradas pois é um projeto interno da empresa.

Não foi indicado uma intenção de retorno econômico da solução por parte do parceiro. Apresentaremos, então, uma possibilidade em que o BTG conseguiria retorno financeiro com a solução desenvolvida:

Incorporação de um modelo de negócios.

A primeira ação seria dar à solução um caráter empresarial e, visando a sua atuação no mercado com um modelo de negócios baseado na cobrança de mensalidades. Para chegar na projeção, partimos de algumas premissas que devem ser revistas pelo BTG Pactual. São elas:

- Ticket médio da mensalidade -> R\$ 1.250,00
- M0 = 2 cliente
- Churn = 0%

- Crescimento de 15% no número de clientes por mês. Sendo assim, obtivemos a seguinte projeção:

| Mês | Crescimento | Número de clientes | Entradas | | | |
|-----|-------------|--------------------|---------------|------------|--------------|----------------|
| 1 | 2,000 | 2 | R\$ 2.500,00 | | ticket medio | R\$ 1.250,00 |
| 2 | 2,300 | 2 | R\$ 2.500,00 | | Soma total | R\$ 258.750,00 |
| 3 | 2,530 | 3 | R\$ 3.750,00 | | | |
| 4 | 2,783 | 3 | R\$ 3.750,00 | | | |
| 5 | 3,061 | 3 | R\$ 3.750,00 | | | |
| 6 | 3,367 | 3 | R\$ 3.750,00 | | | |
| 7 | 3,704 | 4 | R\$ 5.000,00 | | | |
| 8 | 4,075 | 4 | R\$ 5.000,00 | | | |
| 9 | 4,482 | 4 | R\$ 5.000,00 | | | |
| 10 | 4,930 | 5 | R\$ 6.250,00 | | | |
| 11 | 5,423 | 5 | R\$ 6.250,00 | | | |
| 12 | 5,966 | 6 | R\$ 7.500,00 | | | |
| 13 | 6,562 | 7 | R\$ 8.750,00 | | | |
| 14 | 7,218 | 7 | R\$ 8.750,00 | | | |
| 15 | 7,940 | 8 | R\$ 10.000,00 | | | |
| 16 | 8,734 | 9 | R\$ 11.250,00 | | | |
| 17 | 9,608 | 10 | R\$ 12.500,00 | | | |
| 18 | 10,568 | 11 | R\$ 13.750,00 | | | |
| 19 | 11,625 | 12 | R\$ 15.000,00 | | | |
| 20 | 12,788 | 13 | R\$ 16.250,00 | | | |
| 21 | 14,067 | 14 | R\$ 17.500,00 | | | |
| 22 | 15,473 | 15 | R\$ 18.750,00 | | | |
| 23 | 17,021 | 17 | R\$ 21.250,00 | | | |
| 24 | 18,723 | 19 | R\$ 23.750,00 | | | |
| 25 | 20,595 | 21 | R\$ 26.250,00 | Break even | | |

Tabela 3: *tabela referente à projeção de ganhos no decorrer do tempo de vida do projeto.*

O entendimento de quanto a solução iria economizar com o aumento da efetividade de suas campanhas é algo muito difícil de mensurar, pois precisaríamos do produto pronto para entender o quão efetivo, de fato, seria, além de entender toda a esteira de processos dentro do marketing. Nessa situação, ainda seria necessário assumir uma série de premissas que teriam uma baixa confiabilidade, como: “quanto o BTG investe em marketing por ano”, “a efetividade que nossa solução traria iria converter em qual economia por campanha gasta pelo BTG?”. Pela quantidade de riscos e pela falta de confiança no resultado final, optamos por não apresentar essa projeção, contentando-nos apenas com a análise realizada anteriormente.

6. Entendimento da experiência do usuário

Com o objetivo de criar artefatos de planejamento estratégico que possam orientar a compreensão da experiência do usuário da aplicação, foram desenvolvidas duas personas e cinco histórias dos usuários. Ambos os artefatos estão apresentados abaixo.

6.1. Personas

Nome: Gabriel Rodrigues

Idade: 29 anos

Cargo: Líder de operações automatizadas

Gênero: Masculino

Cidade: São Paulo - SP

Renda média: R\$ 25.000,00 - R\$26.000,00

Personalidade: Gabriel é um homem muito seguro de si, exigente, responsável e carismático. Ele se sente muito confortável ao lidar com pressão e não se intimida facilmente. Gabriel sempre foi afeiçoado por situações desafiadoras e não consegue se manter estático por muito tempo, seja na sua vida pessoal ou profissional. Além disso, é um homem muito dedicado que busca realizar suas tarefas com excelência e responsabilidade.

Interesses: Gabriel se interessa muito por esportes, música e tecnologia. Ele costuma trabalhar muito e, quando está no seu tempo livre, gosta de estar com seus amigos e familiares.

Dores com o problema: Gabriel trabalha na área de Automação do Banco BTG Pactual. Atualmente desenvolve um novo projeto, no qual, se faz necessário que seu trabalho ocorra em parceria com um outro setor do BTG, o setor de marketing. Nesse projeto, ele recebeu a tarefa de desenvolver um algoritmo capaz de realizar análises de sentimentos a partir do processamento de linguagem natural. Contudo, o algoritmo que foi desenvolvido não se mostrou eficiente, apresentando erros que comprometem, de forma direta, a análise do time de marketing.

Objetivos com o problema: Tendo isso em mente, Gabriel gostaria de ter acesso a ferramentas e códigos que consiga suprir as falhas apresentadas por seu desenvolvimento anterior.. Dessa forma, seu principal objetivo nesse momento é potencializar as análises realizadas pelo algoritmo, permitindo que o time de marketing atue de forma assertiva no desenvolvimento de outras campanhas para as redes sociais. De modo geral, essa melhoria permitirá que o banco tenha postagens mais bem estruturadas e direcionadas, o que resultará em um lucro mais

pungente.

Nome: Fernanda Soares

Idade: 26 anos

Cargo: Gerente de Marketing

Gênero: Feminino

Cidade: São Paulo - SP

Renda média: R\$15.000,00 - R\$16.000,00

Personalidade: Fernanda é uma mulher muito criativa, inteligente, ágil e esforçada. Ela busca sempre ser positiva e resiliente, tendo em vista que acredita que as pessoas ao seu redor merecem ter contato com a sua melhor versão. Além disso, Fernanda é muito dedicada, principalmente quando sabe que aquilo que ela desempenha terá um impacto positivo na realidade, ou seja, que está trabalhando em algo que acredita

Interesses: Fernanda se interessa muito por design, música, arte e culinária. Ela ama visitar museus, teatros e galerias de arte no seu tempo livre e sempre fica impactada com o fato de que esse tipo de coisa tem o poder de gerar impressões nas pessoas e influenciar suas ações, mesmo que indiretamente. Além disso, ela se interessa muito por livros voltados para o entendimento da mente humana e para a compreensão da influência das propagandas na vida das pessoas.

Dores com o problema: No último mês, o banco BTG Pactual, empresa na qual Fernanda trabalha, solicitou uma análise de sentimento das campanhas de marketing veiculadas no instagram da empresa. O time de automação, a pedido do time de marketing, desenvolveu um algoritmo capaz de realizar essas análises. Porém, durante a sua utilização, Fernanda percebeu que é muito difícil compreender o funcionamento da solução, tendo em vista que a interface desenvolvida apresenta uma quantia excessiva de informações concentradas no mesmo lugar, o que sobrecarrega a visão dos usuários. Além disso, apesar de eficiente, Fernanda sente que ainda é possível potencializar os resultados obtidos, mediante um algoritmo ainda mais eficaz.

Objetivos com o problema: Portanto, Fernanda gostaria de obter uma interface mais clara e intuitiva para o recebimento dos dados do algoritmo. Ela sente que, se conseguir compreender melhor todas as informações passadas, obterá resultados mais vantajosos. Ademais, com um algoritmo mais eficiente e com menos erros, é possível desenvolver estratégias mais certeiras e bem estruturadas, visando a melhoria da satisfação dos clientes do banco.

6.2. User stories

As histórias do usuário (ou user stories) representam uma técnica de gerenciamento de projetos que ajuda as equipes de desenvolvimento a compreender melhor as necessidades e expectativas do seu público alvo, no qual vão utilizar o sistema. Elas são escritas com base em três elementos: no papel do usuário, em sua necessidade e no valor que será agregado a ele. Para facilitar o entendimento do sistema, utilizamos os critérios de aceitação e exemplos de teste de aceitação. Segue abaixo as user stories:

| Número | User story 1 |
|-------------------------------|--|
| Épico | Pré-processamento dos dados. |
| Persona | Gerente de marketing. |
| História | Eu, como gerente de marketing, quero que os dados sejam filtrados e selecionados de maneira que seja fácil de visualizar as principais palavras chaves e comentários, para garantir a escolha dos comentários tratados para a análise. |
| Critérios de aceitação | <p>- Critério 1: É necessário alimentar o algoritmo com o arquivo CSV com os comentários que serão analisados.</p> <p>Critério 2: É necessário que o algoritmo reconheça os dados tratados e dê início à etapa de modelagem.</p> |
| Testes de aceitação | <p>Teste 1 para o critério 1: O funcionário tenta alimentar o algoritmo com dados no formato correto:</p> <p>- Conseguiu: correto.</p> <p>- Não conseguiu: incorreto, é necessário que essa funcionalidade seja corrigida.</p> |

Teste 2 para o critério 1: O funcionário tenta alimentar o algoritmo com dados no formato incorreto, só deve ser aceito CSV ou Excel:

- Conseguiu: incorreto, é necessário que haja uma condicional que impeça a adição de dados no formato incorreto.
- Não conseguiu: correto.

Teste 1 para o critério 2: O funcionário tenta alimentar o algoritmo com os dados não tratados:

- Conseguiu: incorreto, não deve ser possível rodar o algoritmo com dados não tratados, esse erro precisa ser corrigido.
- Não conseguiu: correto.

Tabela 4: *user story 1.*

| Número | User story 2 |
|----------|--|
| Épico | Pré-processamento dos dados. |
| Persona | Líder da área de automação. |
| História | Eu, como funcionário da área de automação, quero usar os melhores métodos de tratamento para realizar o pré-processamento dos dados. |

| | |
|-------------------------------|---|
| Critérios de aceitação | <p>Critério 1: É necessário que os métodos escolhidos sejam relevantes para a análise desejada.</p> <p>Critério 2: É necessário que seja escolhido um número mínimo de parâmetros, sendo esse maior que 1.{retirada de acentos, retirada de artigos, tratamento emoji}</p> <p>Critério 3: É necessário que haja uma indicação dos melhores métodos a serem utilizados para um resultado específico (isso pode estar estruturado em comentários no código).</p> |
| Testes de aceitação | <p>Teste 1 para o critério 1 e para o critério 2: O funcionário não escolhe nenhum método de tratamento de dados e tenta passar para a etapa de modelagem:</p> <ul style="list-style-type: none"> - Conseguiu: incorreto, é necessário que haja um pré processamento de dados antes de passar para a próxima etapa. Portanto, essa possibilidade deverá ser corrigida. - Não conseguiu: correto. <p>Teste 1 para o critério 3: O funcionário utiliza os métodos indicados e consegue o melhor resultado após o pré-processamento de dados:</p> <ul style="list-style-type: none"> - Conseguiu: correto. <p>Não conseguiu: é preciso revisar os métodos indicados pelo código, a fim de apresentar as melhores opções para o usuário final.</p> |

Tabela 5: *user story 2.*

| Número | User story 3 |
|--------|-------------------------|
| Épico | Modelagem do algoritmo. |

| | |
|-------------------------------|---|
| Persona | Líder da área de automação. |
| História | Eu, como líder da área de automação, desejo separar os dados entre teste e treino para estruturar o modelo e receber os resultados da análise de sentimento. |
| Critérios de aceitação | <p>Critério 1: Para que seja possível realizar a modelagem, é necessário utilizar os dados tratados anteriormente.</p> <p>Critério 2: Os dados de teste e treino devem estar divididos de forma que haja maior proporção em treino do que em teste. Seguindo, por exemplo, a proporção: 70% Treino, 30% teste.</p> <p>Critério 3: Os resultados não podem apresentar métricas perfeitas.</p> |

Testes de aceitação

Teste 1 para o critério 1: O funcionário de automação tenta utilizar os dados não tratados para treinar o modelo.

- Conseguiu: incorreto, não deve ser possível treinar o modelo utilizando dados não tratados.
- Não conseguiu: correto.

Teste 2 para critério 1: O funcionário da área de automação tenta utilizar os dados anteriormente tratados para o treinamento do modelo.

- Conseguiu: correto.
- Não conseguiu: incorreto, não pode haver erros no processo de utilização dos dados tratados, essa funcionalidade precisa ser revisada e corrigida.

Teste 1 para critério 2: O funcionário de automação tenta treinar o modelo utilizando proporções de 50% e 50% ou com uma proporção de dados maior para teste do que para treino.

- Conseguiu: incorreto, é necessário que haja uma maior proporção de dados para treino do que para teste (baseado nas condições de construção de inteligências artificiais), visando um resultado relevante .
- Não conseguiu: correto

Teste 2 para critério 2: O funcionário de automação tenta treinar o modelo utilizando uma maior proporção de dados para treino quando comparada aos dados de teste (respeitando a proporção mínima de 30% e 70%).

- Conseguiu: correto.
- Não conseguiu: incorreto, deve ser possível realizar a modelagem com essa proporção, portanto, essa funcionalidade precisa ser corrigida.

Teste 1 para critério 3: O modelo acerta todas as análises e apresenta métricas perfeitas.

- Incorreto: caso essa situação ocorra, é necessário revisar e corrigir o modelo preditivo, tendo em vista que ele se encontra viciado nos dados fornecidos.

Teste 2 para critério 3: O modelo erra a maioria das análises e apresenta métricas abaixo daquelas oferecidas pelo time de automação.

- Incorreto: diante dessa situação, é necessário que o modelo criado seja revisitado, tendo em vista que apresenta erros que não podem ser tolerados.

Tabela 6: *user story 3.*

| Número | User story 4. |
|------------------------|---|
| Épico | Validação do algoritmo. |
| Persona | Líder da área de automação. |
| História | Eu, como líder da área de automação, quero que o modelo seja capaz de fornecer análises com base em métricas de avaliação para que seja possível gerar gráficos que possibilitem a visualização dos resultados. |
| Critérios de aceitação | <p>Critério 1: A solução deve ser capaz de fornecer análises sobre os resultados dos diferentes modelos.</p> <p>Critério 2: O modelo deve ser capaz de gerar gráficos para analisar a performance de cada modelo.</p> |

Critério 3: As métricas utilizadas devem ser comuns para todos os modelos, visando a comparação dos resultados.

**Testes de
aceitação**

Teste 1 para o critério 1: O Líder da área de automação tenta executar as funções de análise do notebook:

- Conseguiu: correto.
- Não conseguiu: incorreto, essa funcionalidade precisa estar funcionando para que seja possível analisar os resultados.

Teste 2 para o critério 1: O Líder da área de automação executa as funções de análise do notebook, sem executar previamente o modelo:

- Não gera análise: correto.
- Gera análise: Incorreto, uma vez que os dados não existem, não seria possível gerar a análise, sendo necessário revisar de onde estão vindo os dados da análise.

Teste 1 para critério 2: O Líder da área de automação tenta executar as funções de visualização de gráficos:

- Conseguiu: correto.
- Não conseguiu: Incorreto, provavelmente há um erro na geração do gráfico ou uma inconsistência na visualização, sendo necessário revisar os resultados obtidos ou o processo de geração do gráfico.

Teste 1 para o critério 3: O líder da área de automação tenta executar a comparação entre os diferentes modelos utilizando as mesmas métricas:

- Conseguiu: correto.
- Não conseguiu: Incorreto, essa funcionalidade precisa ser corrigida ou implementada.

Teste 2 para o critério 3: O líder da área de automação tenta executar a comparação entre os diferentes modelos utilizando métricas distintas:

- Conseguiu: Incorreto, não deve ser possível realizar comparações utilizando métricas distintas.
- Não conseguiu: correto.

Tabela 7: user story 4.

| Número | User story 5 |
|-------------------------------|--|
| Épico | Visualização dos resultados da análise de sentimento. |
| Persona | Gerente de marketing; líder da área de automação. |
| História | Eu, como gerente de marketing, desejo visualizar a performance e repercussão de cada propaganda, para analisar o impacto das publicações na empresa. |
| Critérios de aceitação | Critério 1: O projeto deve ser capaz de fornecer, em uma interface, gráficos com base nos dados de desempenho de cada propaganda analisada. |
| Testes de aceitação | <p>Teste 1 para o critério 1: O gerente de marketing tenta visualizar as informações na interface, após a execução do modelo.</p> <ul style="list-style-type: none"> - Sucesso na visualização gráfica do desempenho das propagandas: correto. - Os gráficos não foram exibidos na interface: Incorreto, o processo de geração dos gráficos ou de exibição precisa ser revisado. <p>Teste 2 para o critério 1: O gerente de marketing tenta visualizar as informações na interface sem executar o modelo</p> <ul style="list-style-type: none"> - Sucesso na visualização gráfica do desempenho das propagandas: Incorreto, pois não deveriam existir dados para gerar a visualização - Aviso que o modelo não foi executado:Correto |

Tabela 8: user story 5.

7. Metodologia

Nesta seção do documento serão abordados os principais tópicos relacionados ao desenvolvimento e métricas de avaliação dos modelos preditivos, bem como a introdução ao modelo desenvolvido nessa sprint, o Support Vector Machine, e a sua comparação com modelos de sprints anteriores.

Também serão exploradas as otimizações realizadas nos modelos, sejam por hiperparâmetros ou outros tipos de vetorização de palavras para o processamento de linguagem natural.

A equipe de desenvolvimento recebeu o banco de dados no formato xlsx, em um excel, que contém 11 colunas e 12356 linhas. Os dados foram extraídos do perfil do Instagram @btgpactual pelo nosso parceiro de projeto, o banco BTG Pactual. É importante destacar a confiança depositada na equipe de desenvolvimento pelo nosso parceiro, que é fundamental para o desenvolvimento e o sucesso deste projeto. Além disso, a segurança e o sigilo das informações contidas no banco de dados são de extrema importância, por isso, a equipe deve se manter atenta à proteção desses dados e garantir que informações sensíveis não sejam divulgadas.

7.1. Metodologia CRISP-DM

No desenvolvimento do projeto referido neste documento, foi escolhida a metodologia CRISP-DM. De acordo com o IBM SPSS Modeler CRISP-DM Guide, o Cross-Industry Standard Process for Data Mining (CRISP-DM) constitui uma das mais importantes metodologias relacionadas ao processo de mineração de dados. É por meio do CRISP-DM que os dados de uma empresa podem ser transformados em informações capazes de guiar o gerenciamento do negócio. Essa metodologia é composta pelas seguintes etapas:

- **Business understanding (entendimento do negócio):** antes de iniciar o processo de mineração de dados, é necessário refletir sobre o que o cliente espera obter como resultado. Para isso, é fundamental examinar as metas, riscos e recursos disponíveis para o desenvolvimento do produto.
- **Data understanding (entendimento dos dados):** a partir da compreensão dos objetivos do negócio, é necessário explorar a base de dados disponível, a fim de compreender o conjunto de informações que será minerado. Constituindo uma fase da metodologia que apresenta uma significativa demanda por tempo, essa etapa exige que

os atributos presentes e os valores preenchidos nos registros existentes sejam analisados com precisão.

- **Data preparation (preparação dos dados):** constitui a etapa de preparação dos dados disponíveis para que eles possam ser devidamente lidos e interpretados pelos processos de mineração aos quais serão submetidos. É quando ocorre a Feature Engineering, constituída pela seleção de atributos que serão utilizados, bem como os procedimentos de limpeza dos dados, agregação de registros, derivação de novos atributos e separação de conjuntos de dados para treinamento e teste.
- **Modeling (modelagem):** a modelagem de dados é a fase na qual os dados, já preparados, são submetidos a diferentes algoritmos - utilizando, a princípio, os parâmetros padronizados do modelo. Posteriormente aos testes realizados, ocorre, ainda, a aplicação de uma série de técnicas de manipulação de dados responsáveis pelo refinamento dos modelos construídos, a exemplo das ferramentas de ajustes de hiperparâmetros.
- **Evaluation (avaliação):** nessa fase, os modelos testados a partir da base de dados são avaliados por meio de métricas definidas na fase de entendimento do negócio. São os chamados “critérios de sucesso”, os quais serão capazes de indicar se os procedimentos até então realizados estão tecnicamente corretos e efetivos para o objetivo do cliente.
- **Deployment (implantação):** a última etapa da metodologia CRISP-DM se refere à implementação do modelo final definido na fase de avaliação. É nessa fase que podem surgir novos insights para aprimorar o produto final. Além disso, é quando uma revisão do projeto é conduzida a fim de atestar que os objetivos foram alcançados ao final do projeto.

7.1.1. Entendimento dos Dados

Análise descritiva dos dados

Após a recepção dos dados, uma análise foi conduzida pela equipe para identificar as características relevantes necessárias para a criação de um modelo bag of words. Isso incluiu a remoção das colunas irrelevantes antes do início da etapa de pré-processamento dos elementos da tabela.

O objetivo da análise foi compreender o corpus dos dados, identificando as características necessárias, tratamentos importantes e limpezas a fim de reduzir ruídos e outros fatores. Serão apresentados os nomes das colunas da tabela, os conteúdos dos campos e as colunas que foram eliminadas, juntamente com a estratégia utilizada para tomar essas decisões. É importante ressaltar que os nomes das colunas serão escritos exatamente como foram recebidos pela equipe, sendo alterados apenas por algoritmos em Python, e não manualmente na tabela do Excel, uma vez que essa etapa é crucial no tratamento dos dados.

- **Coluna "id":** Essa coluna apresenta o índice para visualização da planilha e pode ser usada como chave primária para os comentários. No entanto, essa coluna não é relevante para a construção do modelo, pois sua utilidade está apenas em garantir a unicidade das linhas. Portanto, essa coluna não será utilizada.
- **Coluna "dataPublicada":** Essa coluna se refere à data de publicação do comentário. Para a construção do modelo bag of words, essa coluna não possui relevância, portanto, não será utilizada. No entanto, sua utilização pode ser necessária para investigar os períodos das campanhas.
- **Coluna "autor":** Essa coluna se refere à conta do Instagram que realizou o comentário na postagem. Embora não seja utilizada no modelo bag of words, ela será importante para o agrupamento de comentários relacionados à empresa BTG em outros modelos.
- **Coluna "texto":** Essa coluna contém o texto presente nos comentários. Para a construção do modelo, essa é a coluna mais relevante, pois são os conteúdos dos comentários que precisam ser analisados pelo modelo bag of words.
- **Coluna "sentimento":** Essa coluna representa o alvo da classificação dos dados. Será utilizada para o treinamento posterior do modelo, uma vez que contém o resultado esperado. Os comentários foram classificados como POSITIVE, NEGATIVE e NEUTRAL. No entanto, após uma análise manual das classificações, foi observado que alguns comentários foram classificados de forma incorreta, pois possuem um teor positivo, mas foram classificados como negativos ou neutros. Considerando o modelo de bag of words, essa coluna não será utilizada.
- **Coluna "tipoInteracao":** Essa coluna indica o tipo de interação à qual o comentário pertence, como resposta ou marcação. Ela não será utilizada na construção do modelo bag of words.

- **Coluna "anomia":** Essa coluna é usada para indicar a presença de links ou intenções maliciosas nos comentários. Os valores variam entre 0 (zero) para comentários que não se enquadram nessa condição e 1 (um) para comentários que se enquadram. Como não está relacionada ao sentimento expressado pelo usuário, essa coluna não será utilizada.
- **Coluna "probabilidade Anomia":** Essa coluna indica a probabilidade de ocorrência de links ou intenções maliciosas nos comentários, com valores variando entre 0 (zero) e 100 (cem) de acordo com a chance do comentário se enquadrar nessa condição. Por não estar relacionada ao sentimento expressado pelo usuário, essa coluna também não será utilizada.
- **Coluna "linkPost":** Essa coluna contém o link da postagem da qual os comentários foram retirados. Para a análise de sentimento, essa coluna não é relevante e, portanto, não será utilizada.
- **Coluna "processado":** Essa coluna indica se o comentário foi analisado pelo algoritmo usado pela empresa parceira para classificar o sentimento do texto do usuário e definir a classificação na coluna "sentimento". Por ser apenas uma verificação adicional e todos os comentários na tabela já terem sua classificação definida, essa coluna não será utilizada por enquanto.
- **Coluna "contemHyperlink":** Não temos informações suficientes para definir o significado dessa coluna, portanto, ainda não é possível determinar se ela será utilizada ao longo do desenvolvimento do projeto. Por enquanto, essa coluna não será utilizada.

A equipe utilizou o critério de funcionamento do modelo bag of words para decidir quais colunas excluir e quais manter. Nessa abordagem, cada palavra é tratada como uma unidade independente, tornando irrelevante manter informações sobre a origem ou contexto, como as colunas "autor" e "dataPublicada". Se a estratégia fosse selecionar um conjunto específico de datas, seria necessário destacá-las e eliminar os elementos fora desse período selecionado. As palavras-chave relevantes são provenientes dos textos escritos pelos usuários da rede social, e adicionar palavras não inseridas pelos usuários geraria muito ruído. Portanto, todas as colunas foram excluídas antes da criação do modelo, mantendo apenas o texto dos comentários para o pré-processamento.

Para a visualização gráfica dos dados, foram desenvolvidos métodos que possibilitaram a análise exploratória dos dados presentes na tabela fornecida pela empresa parceira. Por meio da visualização de valores nulos e do agrupamento por autores, foi possível compreender melhor o comportamento dos dados e determinar a abordagem a ser adotada pelo grupo. Os critérios para essa parte foram baseados

nos conhecimentos prévios dos membros da equipe, que sugeriram dois métodos de análise exploratória conhecidos.

7.1.2. Preparação dos Dados

7.1.2.1. Pré processamento dos dados

A etapa de pré-processamento é considerada crucial para garantir a qualidade dos resultados obtidos, e sua extensão depende da qualidade dos dados brutos (Rudkowski et al, 2018). Com base na análise descritiva do corpus, foram selecionados alguns pré-processamentos que são considerados essenciais para preparar os dados para o modelo Bag of Words. Essas etapas são fundamentais para normalizar o texto, eliminar ruídos, padronizar e tratar ambiguidades.

A seguir estão listados todos os pré-processamentos selecionados:

- **Remoção de acentos:** A remoção de acentos é uma prática comum no pré-processamento de textos, pois visa padronizar os componentes do texto e evitar interpretações equivocadas. Isso ajuda a identificar palavras idênticas que foram acentuadas de maneiras diferentes, melhorando a qualidade e a eficácia da análise de dados em linguagem natural.
- **Tratamento de letras maiúsculas:** O tratamento de letras maiúsculas é importante para evitar problemas de análise do algoritmo causados pela diferença entre letras maiúsculas e minúsculas. Essa diferença pode levar a resultados imprecisos, especialmente ao contar a frequência de palavras. Portanto, é necessário converter todas as letras para minúsculas ou tratar adequadamente a diferença entre maiúsculas e minúsculas.
- **Tokenização:** A tokenização é o processo de dividir os valores de uma coluna em pedaços menores, como palavras ou frases. Cada pedaço, conhecido como token, recebe um valor específico para identificação, permitindo que cada palavra seja tratada de forma independente. Esse processo torna o texto mais gerenciável e facilita a análise e o processamento posterior dos dados.
- **Remoção de Stopwords:** A remoção de Stopwords é importante para eliminar palavras irrelevantes que podem prejudicar a precisão do modelo final. Isso inclui artigos, preposições, conjunções e outros conectores que não têm um valor semântico significativo. Esse tratamento ajuda na eficácia da classificação de texto e na redução do vocabulário e de ruídos.
- **Tratamento de abreviações:** Dado que o trabalho envolve a manipulação de textos extraídos das redes sociais, em que há uma quantidade significativa de comentários com abreviações, é necessário um tratamento adequado para esses casos. Isso envolve substituir as abreviações por suas versões completas, permitindo que o modelo compreenda e processe melhor essas expressões.

- **Tratamento de emojis:** Emojis são amplamente utilizados por usuários de redes sociais para expressar emoções e sentimentos. Portanto, é importante realizar um tratamento adequado dos emojis para melhor entender o sentido de uma mensagem e fornecer uma análise mais precisa dos sentimentos.

A padronização da variação de palavras é importante para o modelo bag of words, que conta a ocorrência das palavras considerando variações na grafia. Portanto, os tratamentos de remoção de acentos, transformação de letras maiúsculas para minúsculas e substituição de abreviações têm o objetivo de reduzir as diferentes variações nas grafias de palavras, tornando-as consistentes e com o mesmo sentido.

A remoção de Stopwords é fundamental para melhorar a análise de dados em linguagem natural, pois essas palavras não possuem significado por si só e podem causar ruído nos modelos de processamento de linguagem natural. Ao eliminá-las, é possível obter uma análise mais precisa e de melhor qualidade.

A tokenização é um processo básico na criação de um modelos de análise de sentimento, pois permite a separação de frases em palavras isoladas. Isso facilita a identificação de palavras-chave, a análise de tendências e padrões nos textos escritos por diferentes usuários.

O objetivo de todas essas etapas descritas anteriormente é garantir um conjunto padronizado de palavras, reduzindo o ruído e criando modelos com melhor qualidade.

7.1.2.2. Pipeline de pré-processamento

A utilização de uma pipeline é fundamental para organizar o fluxo de trabalho, garantindo coerência e eficiência na aplicação dos algoritmos. Ela permite a visualização clara das etapas do processo e facilita a identificação e correção de erros ou problemas em cada fase, melhorando significativamente a qualidade do resultado final. Além disso, a pipeline promove a comunicação efetiva entre os membros da equipe sobre os processos realizados. No contexto específico deste projeto, voltado para o pré-processamento de texto, a pipeline é essencial para demonstrar a sequência de etapas do tratamento dos dados, que ocorre da seguinte forma:

- Entrada:** Primeira etapa, com os dados em seu estado bruto, exatamente como constam na base de dados.
- Remoção de acentos:** Consiste em eliminar os acentos das palavras, evitando discrepâncias nas formas de escrita.
- Conversão para minúsculas:** Transformação de todas as letras do texto em minúsculas, para padronizar as palavras.
- Tradução de emojis:** Conversão de emojis presentes no texto para sua forma textual.
- Tratamento de abreviações:** Identificação e expansão de abreviações, tornando o texto mais legível para análises posteriores.

- f. **Tokenização:** Divisão do texto em unidades menores, como palavras ou subpalavras, conhecidas como tokens, possibilitando a criação do modelo bag of words.
- g. **Remoção de stopwords:** Eliminação de palavras comuns, como artigos e preposições, que não contêm informações relevantes para as análises.
- h. **Saída:** Resultado final após os dados passarem por todas as etapas.



Imagem 1: Pipeline desenvolvida com as etapas do pré-processamento dos dados, com uma frase utilizada como exemplo para demonstrar o resultado de cada etapa no texto.

7.1.3. Vetorização e word embedding

Word embedding é uma técnica de processamento de linguagem natural que mapeia palavras para vetores contínuos de números reais em um espaço multidimensional. Esses vetores representam o significado semântico e a relação entre as palavras. O objetivo do word embedding é capturar as nuances e complexidades das palavras em um formato numérico que possa ser facilmente compreendido e utilizado por algoritmos de aprendizado de máquina.

7.1.3.1. Bag of words

Dados são muito importantes para enriquecer as tomadas de decisão das empresas. Porém, analisar os dados de forma manual demanda muito tempo, principalmente quando o profissional analisa esses dados diretamente em uma plataforma de rede social, por exemplo. Nesta etapa do desenvolvimento do modelo de machine learning para análise de sentimentos, um modelo de bag of words será criado. A técnica de processamento de linguagem natural transforma textos em vetores de palavras, ignorando a estrutura gramatical e a ordem das palavras. Cada palavra é ranqueada de acordo com a quantidade de ocorrências no banco de dados fornecido pelo parceiro de projeto.

Antes de criar o modelo de bag of words, é essencial analisar e compreender os dados para realizar o pré-processamento adequado. A relevância do tratamento e pré-processamento dos dados é ressaltada, especialmente quando se trata de modelos de linguagens, que contêm informações subjetivas. Isso garante a qualidade dos dados para análise e a obtenção de resultados mais sólidos e robustos. A presença de ruídos e vieses é reduzida como resultado desse processo.

A técnica de "bag of words" identifica e classifica a frequência e ocorrência de "words" (palavras ou elementos) em cada "bag" ou "bolsa", que representa o conjunto do banco de dados utilizado. Cada modelo criado pode possuir diferentes conjuntos de elementos (Lee et al, 2022). A instância de cada palavra no modelo desconsidera a ordem e a gramática, separando adjetivos de substantivos, desconectando palavras compostas, por exemplo, e diferenciando variações de palavras, de acordo com sua escrita, como "água" e "agua", ou "Gato" e "gato". Isso resalta a importância do tratamento dos dados para padronizar as palavras (Qader, Ameen, Ahmed, 2019).

O modelo "bag of words" pode ser utilizado para construir uma nuvem de palavras, facilitando a visualização e análise dos resultados obtidos e tornando os insights mais intuitivos (Powell et al, 2017).

Os resultados obtidos podem ser observados na tabela com a frequência das dez palavras que tiveram mais ocorrências no banco de dados:

| Palavra | Repetição |
|---------------|-----------|
| btg | 570 |
| banco | 514 |
| limite | 370 |
| conta | 309 |
| cartão | 263 |
| investimentos | 233 |
| estamos | 222 |

| Palavra | Repetição |
|---------|-----------|
| dia | 192 |
| ajudar | 187 |
| melhor | 185 |

Tabela 9: Ocorrência das palavras que mais apareceram no banco de dados após criação do modelo bag of words.

7.1.3.1.1. Nuvem de palavras

Uma nuvem de palavras, também conhecida como nuvem de tags ou word cloud em inglês, é uma representação visual das palavras mais frequentes em um conjunto de texto. Nessa representação, as palavras são exibidas em tamanho e formato diferentes, de acordo com a sua frequência de ocorrência no texto. Ou seja, as palavras que aparecem com mais frequência são exibidas em tamanho maior, enquanto as menos frequentes são mostradas em tamanho menor.

A visualização de nuvem de palavras é uma forma de representar de maneira visual os principais resultados da análise de bag of words. Ao analisar os dados, algumas palavras-chave são recorrentes, tais como "btg", "banco", "limite", "conta", "cartão" e "ajudar".

Também foi possível perceber variações de palavras que não foram tratadas, mas que possuem o mesmo sentido, como plurais, e palavras com tempos verbais diferentes, bem como palavras que não representam nenhum insight importante, como "vocês", "dia", "estamos", entre outras.

Outro ponto a destacar é palavras que estão diretamente ligadas ao nosso parceiro de projeto, como "btg", "banco", mas que não trazem nenhum insight, já que são naturalmente esperadas de aparecer em uma rede social de um banco, podendo ser utilizadas também como stop words.

considera múltiplas palavras em um contexto específico para fazer a previsão da palavra de destino.

Geralmente, ele produz representações vetoriais mais densas e, por lidar com as palavras mais frequentes, é capaz de evitar seus ruídos e garantir seus contextos. Além disso, por possuímos um conjunto de dados de treinamento pequeno, o modelo CBOW é mais favorável e eficiente para capturar as informações contextuais mais próximas.

7.1.3.3. Word2vec skip-gram

Foi escolhido testar o modelo Word2Vec com Skip Gram devido à sua capacidade de capturar relacionamentos contextuais entre palavras em um texto. Esse modelo aprende a representar palavras com base em seu contexto, permitindo que palavras semanticamente e sintaticamente relacionadas sejam mapeadas para pontos próximos no espaço vetorial.

A abordagem do Skip Gram consiste em treinar uma rede neural para aprender vetores de palavras, em que cada palavra é representada por um vetor denso de valores reais. A dimensão desses vetores, chamada de dimensão de embedding, é um hiperparâmetro definido pelo usuário. O objetivo é encontrar os melhores vetores de palavras que capturem as relações semânticas e sintáticas presentes no corpus de treinamento.

7.1.3.4. Sentence transformers

Sendo uma vetorização avançada, esse método utiliza o contexto das palavras em relação à frase para gerar classificações mais precisas e significativas.

Os vetores gerados capturam, além do significado das palavras individuais, mas também as nuances e relações semânticas entre as frases, o que permite uma compreensão mais profunda do texto.

Além de ser amplamente utilizado no mercado por empresas como Google e OpenAI, uma vantagem é por ser uma solução de código aberto, não requerendo uma licença específica para sua utilização.

A vetorização Sentence Transformers é frequentemente utilizada como input para redes neurais recorrentes (RNNs), onde essa combinação potencializa a capacidade de análise de texto, apresentando um desempenho aprimorado.

7.1.4. Modelagem

7.1.4.1. Naive bayes com bag of word

O modelo Naive Bayes é um classificador que utiliza o Teorema de Bayes para calcular probabilidades. Ele assume a independência condicional entre os

recursos (palavras) dadas as classes (sentimentos positivo, negativo ou neutro). Nesta sprint, aplicamos o modelo Naive Bayes utilizando a abordagem Bag of Words (BoW) para classificar os comentários.

A escolha do modelo Naive Bayes com BoW é baseada na natureza estatística do Naive Bayes, que se encaixa bem no uso de BoW para capturar a recorrência das palavras nos comentários.

O modelo Naive Bayes utiliza o conjunto de recursos construído para calcular as probabilidades condicionais de um comentário pertencer a cada uma das categorias de sentimento (positivo, negativo ou neutro). Com base nessas probabilidades, o modelo atribui uma classe ao comentário.

A aplicação do Naive Bayes se justifica pela capacidade do modelo em lidar com grandes conjuntos de dados, por ser um modelo mais simples e requerer menos recursos de processamento, comportando tanto a abordagem de Bag of Words, quanto o Word2Vec, facilitando a escalabilidade do algoritmo com inputs diferentes, em comparação a classificadores mais complexos, como as redes neurais.

7.1.4.2. Naive bayes com Word2Vec

O Naive Bayes é um modelo de aprendizado de máquina que se baseia no teorema de Bayes para realizar previsões em problemas de classificação. Ele assume a independência condicional das características para simplificar o cálculo das probabilidades. Apesar dessa suposição simplificadora, o Naive Bayes pode fornecer resultados satisfatórios, especialmente quando há dados de treinamento suficientes, sendo amplamente utilizado em tarefas de processamento de linguagem natural e classificação de documentos.

O raciocínio por trás do modelo Word2Vec baseia-se na ideia de que palavras com significados semelhantes aparecem em contextos semelhantes. O modelo aprende representações vetoriais (embeddings) para cada palavra, de modo que palavras semanticamente relacionadas fiquem próximas umas das outras no espaço vetorial.

Para treinar o modelo, normalmente é utilizado um grande corpus de texto. O algoritmo Word2Vec utiliza duas abordagens principais: o Skip-gram e o Continuous Bag of Words (CBOW).

No método Skip-gram, o objetivo é prever as palavras de contexto (as palavras ao redor de uma palavra-alvo) com base na palavra-alvo. Já no método CBOW, o objetivo é prever a palavra-alvo com base nas palavras de contexto. O modelo ajusta os pesos das conexões entre as palavras, de forma a maximizar a probabilidade de prever corretamente as palavras de contexto.

Uma vez treinado, o modelo Word2Vec produz vetores densos para cada palavra. Esses vetores podem ser usados para várias tarefas de processamento de linguagem natural, como classificação de sentimentos, agrupamento de documentos, identificação de tópicos, entre outros. Além disso, os vetores podem ser utilizados para calcular similaridade entre palavras, permitindo a realização de buscas semânticas e identificação de relações entre termos.

7.1.4.3. Support vector machine com Word2Vec

O SVM é um algoritmo de aprendizado de máquina supervisionado que pode ser aplicado tanto a problemas de classificação quanto para de regressão. Ele busca uma linha de separação (hiperplano) entre duas classes distintas, baseando-se nos pontos mais próximos de cada classe. Este modelo foi escolhido pela sua performance com uma base de dados complexa e de alta dimensionalidade, em modelos de classificação, e pela facilidade em aplicar hiperparâmetros para ajustar o seu desempenho.

A partir de seu foco no treinamento e classificação de um dataset, seu objetivo principal é encontrar um hiperplano que melhor separa os pontos de dados em diferentes classes ou aproxime os pontos de dados com menor erro possível.

Uma das principais vantagens do SVM é sua capacidade de lidar com conjuntos de dados não lineares e possuir propriedades de regularização embutidas, o que ajuda a evitar o overfitting (sobreajuste) do modelo aos dados de treinamento.

Há dois tipos principais de SVM: o modelo rígido (hard margin) e o modelo com margem flexível (soft margin). O modelo rígido visa encontrar um hiperplano que separe os dados em duas classes, assumindo que eles são linearmente separáveis e sem cometer erros de classificação. Já o modelo com margem flexível permite erros de classificação e além de encontrar um hiperplano que separe os dados da melhor maneira e que maximize a margem, ele busca minimizar esses erros de classificação também.

Em nosso modelo, utilizamos o SVM (Support Vector Machine) com margem flexível por permitir que o modelo seja mais flexível na adaptação dos dados e um certo grau de erro nos dados de treinamento. Isso evita o chamado "overfitting", onde o modelo se ajusta muito bem aos dados de treinamento, mas tem baixa capacidade de generalização para novos dados.

Otimização da Modelagem:

Hiperparâmetros:

Para melhor configurar o desempenho da modelagem preditiva para classificação, os hiperparâmetros são ajustes definidos antes do treinamento do modelo, influenciando no seu aprendizado. A otimização desses hiperparâmetros se

dá pela configuração e combinação desses parâmetros, que resultam no melhor desempenho do modelo, ressaltando a importância de evitar o overfitting, melhorando a precisão dos resultados obtidos com a generalização do treinamento do modelo.

Uma técnica que pode ser utilizada para realizar a otimização de hiperparâmetros é a partir do Grid Search, que envolve a definição de um conjunto de valores possíveis para cada hiperparâmetro, que configura vários testes para a escolha dos ajustes ideais para o conjunto de dados utilizado, evitando overfitting e melhorando o desempenho do modelo.

É importante dizer que colocar um limite de configurações que o Grid Search realiza é importante para reduzir o tempo de execução do modelo, pois pode levar um tempo muito elevado e o desempenho não mudar, por algum problema na base de dados ou na pipeline dos dados, por exemplo, reduzindo a eficiência do Grid Search.

O Support Vector Machine (SVM) é um modelo de aprendizado de máquina amplamente utilizado para problemas de classificação e regressão. Ele busca encontrar um hiperplano de separação ótimo entre as classes, maximizando a margem entre os pontos de cada classe. O SVM é capaz de lidar com conjuntos de dados de alta dimensionalidade e separações complexas, graças ao uso de funções de kernel que permitem mapear os dados para espaços de maior dimensionalidade. Apresentando uma interpretação geométrica clara, o SVM é uma opção eficaz para problemas de classificação.

7.1.4.4. Rede neural com transformers

O modelo de Rede Neural Recorrente (RNN) é um tipo de classificador que usa a arquitetura Transformer para entender melhor as sentenças. Essa técnica considera a ordem das palavras nas sentenças ou documentos.

Foi escolhido usar a RNN com o Transformer porque a RNN é boa em lidar com sequências, e o Transformer é ótimo para entender como as palavras nas sentenças se relacionam umas com as outras, o que é muito importante para entender o texto.

Para melhorar a nossa análise, foi utilizada uma outra rede neural para entender o sentimento dos textos. Isso nos ajuda a olhar os dados de uma maneira mais completa, considerando todo o contexto para classificar o sentimento (se é positivo, negativo ou neutro).

Foi usado o modelo RNN porque ele é bom em lidar com a complexidade e a ordem das palavras encontradas em muitos textos. Mesmo sendo um modelo mais complicado e que precisa de mais recursos para processar os dados, a RNN trabalha bem com o Transformer para entender melhor as sentenças. Isso ajuda o

algoritmo a lidar com muitos tipos diferentes de entradas e a entender melhor o contexto e o significado das sentenças, em comparação a classificadores mais simples, como o Naive Bayes. Usamos uma segunda rede neural para classificar o sentimento usando as informações fornecidas pelo Transformer.

A implementação da Rede Neural Recorrente juntamente com a arquitetura Transformer, e a adição de uma segunda rede neural para a classificação de sentimentos, parece ter influenciado positivamente os resultados da análise de sentimentos. Essa observação se apoia tanto em resultados quantitativos obtidos na fase de teste, quanto em alguns exemplos qualitativos notáveis.

A complexidade desses modelos proporcionou uma oportunidade para explorar as nuances do sentimento nos textos de forma mais detalhada, possibilitando um entendimento mais aprofundado do contexto e da semântica das sentenças. Isto parece ter levado a uma classificação de sentimento mais precisa e minuciosa, indicando o potencial que tais métodos avançados de modelagem de linguagem natural podem ter.

A experiência adquirida ao longo deste processo forneceu insights úteis e parece reforçar a utilidade potencial dessas técnicas contemporâneas de processamento de linguagem natural para a análise de sentimentos.

Durante a construção do modelo de rede neural com Transformers, foram aplicados os seguintes métodos:

Arquitetura do Transformer: A arquitetura do Transformer foi implementada como a base do modelo. Foram incluídas camadas de codificação e decodificação, permitindo que o modelo capture relações de dependência entre as palavras em uma sequência. Essa arquitetura avançada contribuiu para uma melhor compreensão do contexto e uma capacidade aprimorada de classificação de texto.

Fine-tuning: O modelo foi ajustado (fine-tuned) para a tarefa de classificação de texto específica do projeto. Os pesos do modelo pré-treinado foram refinados utilizando dados rotulados relacionados ao problema em questão. Esse processo de fine-tuning permitiu que o modelo se adaptasse aos dados específicos da sua tarefa, otimizando seu desempenho e melhorando a capacidade de realizar classificações precisas.

Aumento de dados (data augmentation): Para expandir o conjunto de dados de treinamento e melhorar a capacidade de generalização do modelo, foram aplicadas técnicas de aumento de dados. Transformações simples, como rotações, inversões ou substituições de palavras, foram utilizadas para criar variações nos dados de treinamento. Isso proporcionou ao modelo uma maior exposição a diferentes casos e variações nos dados de entrada, tornando-o mais robusto e capaz de lidar com uma variedade de situações.

A aplicação desses métodos no projeto com a rede neural com Transformers permitiu a construção de um modelo avançado, capaz de aprender representações contextualizadas e capturar relações complexas entre as palavras. Isso resultou em um desempenho aprimorado na tarefa de classificação de texto, proporcionando resultados mais precisos e relevantes.

A Rede Neural com Transformers é um modelo avançado de aprendizado de máquina utilizado em tarefas de processamento de linguagem natural. Baseada na arquitetura de transformers, ela utiliza atenção baseada em mecanismos para capturar relações de dependência entre as palavras em uma sequência, melhorando a representação contextual das palavras. Essa abordagem tem se mostrado altamente eficaz, permitindo que o modelo lide com problemas complexos de linguagem, como tradução automática, resumo de texto e análise de sentimentos, alcançando resultados promissores.

7.1.5. Métricas para Avaliação

As métricas de avaliação são fundamentais para comparar a performance e o desempenho dos modelos utilizados. Nesse contexto, após obter os resultados dos modelos, foi gerada uma matriz de confusão para avaliar o desempenho (Franceschi, 2019).

A matriz de confusão é uma tabela que apresenta o desempenho de um modelo de classificação, dividindo as previsões em quatro categorias, como demonstrado a seguir:

| True label | 0.0 | 1.0 |
|-----------------|---------------------|---------------------|
| | Verdadeiro Positivo | Falso Positivo |
| 1.0 | Falso Negativo | Verdadeiro Negativo |
| Predicted label | | |

Tabela 10: Tabela que apresenta o desempenho de um modelo de classificação.

A acurácia é uma métrica simples que calcula a proporção de acertos do modelo, pela seguinte fórmula:

$$Acurácia = \frac{VerdadeirosPositivos + VerdadeirosNegativos}{VerdadeirosPositivos + VerdadeirosNegativos + FalsosPositivos + FalsosNegativos}$$

Figura 11: Figura com a fórmula da acurácia.

Representa a soma dos verdadeiros positivos e verdadeiros negativos, dividida pelo total de elementos utilizados na predição (verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos). No entanto, é importante destacar que a acurácia sozinha não é suficiente para avaliar completamente o desempenho dos modelos, pois não considera a distribuição das classes ou possíveis desequilíbrios no conjunto de dados (Chen, et al, 2020). Por isso, além da acurácia, também foi aplicada a revocação, com a seguinte fórmula:

$$Revocação = \frac{VerdadeirosPositivos}{VerdadeirosPositivos + FalsosNegativos}$$

Figura 12: Figura com a fórmula da revocação.

A revocação é uma métrica que avalia a capacidade de um modelo em identificar corretamente os exemplos positivos em relação ao total de exemplos positivos presentes nos dados, ela é calculada dividindo o número de Verdadeiros Positivos pela soma dos Verdadeiros Positivos com os Falsos Negativos.

7.1.6. Implantação

7.1.6.1. Arquitetura da solução

A arquitetura macro da solução é uma representação de alto nível que descreve a estrutura geral de uma solução. Ela abrange os principais blocos funcionais e as etapas-chave do processo do sistema e tem como objetivo fornecer uma representação visual desde a origem dos dados até a obtenção do resultado desejado. Nossa arquitetura macro inclui componentes como fontes de dados, processamento de dados, modelos, interfaces de usuário, entre outros.

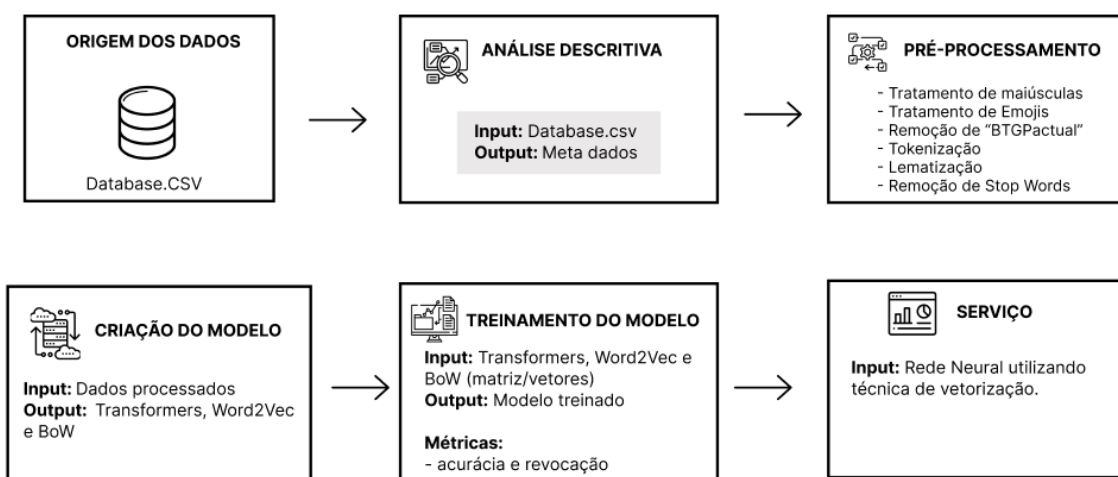


Figura 2: *Arquitetura da Solução, ilustrando as etapas desde a origem dos dados, passando pela análise exploratória dos dados, pré-processamento realizados, modelo desenvolvido, seu treinamento, e o serviço gerado, possibilitando a interação do sistema com pelo usuário.*

A arquitetura se inicia recebendo os dados de entrada proveniente de um arquivo "baseDados.csv" em formato Excel e a partir desses dados, é feita a análise descritiva. Essa análise envolve a extração de metadados relevantes dos dados utilizados e visa explorá-los de forma a identificar padrões, tendências e distribuições descritivas que possam ser relevantes para o desenvolvimento da solução.

Na etapa seguinte, ocorre o pré-processamento dos dados, que consiste na realização de várias tarefas para facilitar a vetorização ou a transformação dos dados em uma matriz adequada para o treinamento do modelo. Essas técnicas incluem a normalização dos dados, o tratamento de valores ausentes, o processo de Tokenização, a remoção de ruídos, entre outras, a fim de garantir a qualidade dos dados.

Após os dados serem pré-processados, já é possível a criação do modelo. Ela envolve a definição de sua arquitetura, a seleção dos algoritmos adequados, a escolha de hiperparâmetros e a divisão dos dados em conjuntos de treinamento, validação e teste. Essa etapa é fundamental para o desenvolvimento de um modelo capaz de aprender padrões e fazer previsões precisas.

Uma vez que o modelo é criado, é necessário treiná-lo com os dados disponíveis. Isso envolve alimentar o modelo com os dados de treinamento e ajustar seus parâmetros para otimizar o desempenho. O treinamento pode ser iterativo e exigir várias rodadas até que o modelo alcance uma precisão satisfatória.

Por fim, o serviço é disponibilizado através de um dashboard intuitivo, que proporciona aos usuários uma visualização clara e compreensível dos resultados. O dashboard é capaz de fornecer métricas, gráficos e insights relevantes, auxiliando o colaborador de marketing do BTG a tomar decisões embasadas e estratégicas com base nas informações geradas pela solução.

7.1.6.2. Diagrama de Implantação UML

O diagrama de Implantação UML é um tipo de diagrama usado para descrever a arquitetura física de um sistema identificando os componentes de hardware e software e as conexões entre eles.

Ele representa a arquitetura física de um sistema a partir da relação entre a disposição dos nós físicos e lógicos, como computadores, servidores, dispositivos de rede, etc.

Além de ser usado como uma ferramenta de comunicação em relação à arquitetura e as outras partes interessadas do projeto, ele pode ajudar a identificar problemas de desempenho e segurança.

A seguir pode ser verificado o diagrama UML, representando o sistema e as conexões entre os componentes necessários para utilizar o sistema.

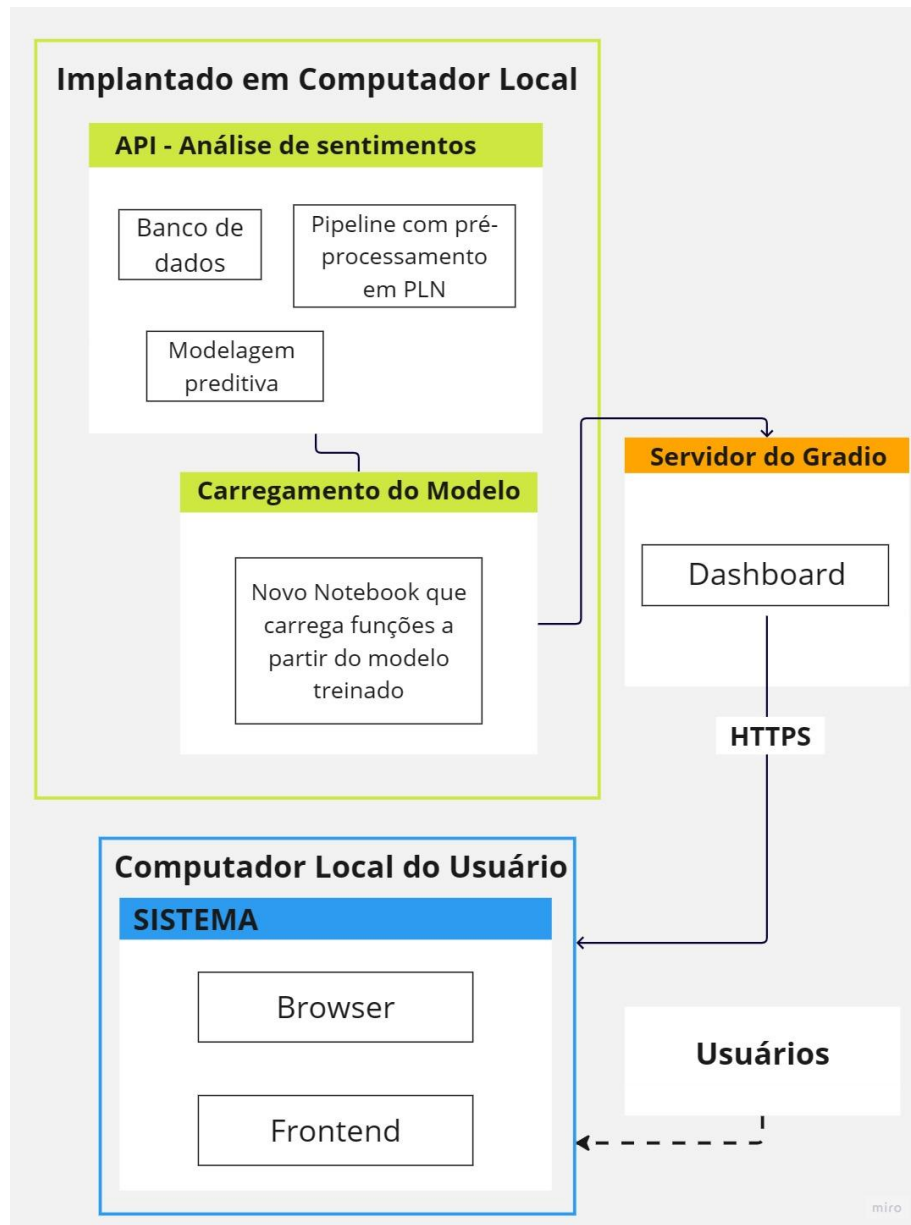


Figura 3: Diagrama de Implantação UML, mostrando a comunicação entre os blocos de hardware, e os componentes dos sistemas utilizados em cada um.

O processo é iniciado no sistema local que contém a implantação da API, com o banco de dados, a pipeline com os pré-processamentos realizados, e o modelo desenvolvido, no caso deste projeto, o de Rede Neural utilizando vetorização com Transformers. Neste sistema também é gerado um novo Notebook de Google Colab, com o modelo preditivo exportado, e o deploy utilizando Gradio,

com a configuração do dashboard que possibilita a interação do usuário com a solução. A partir do deploy com o Gradio, é gerado um servidor com o dashboard, que pode ser acessado por um computador local do usuário, por meio do protocolo HTTP. Com sistema do computador, usando o browser, é possível ter acesso a um frontend, que contém a funcionalidade do dashboard.

8. Resultados

Os resultados obtidos a partir dos modelos de classificação desenvolvidos são de grande relevância e fornecem insights valiosos para o problema proposto. Durante as análises, os modelos foram avaliados por meio de métricas como acurácia e revocação, que demonstram a sua capacidade de realizar previsões corretas e identificar corretamente os targets dos comentários analisados.

8.1. Naive Bayes com Bag of Words:

- Matriz de confusão:

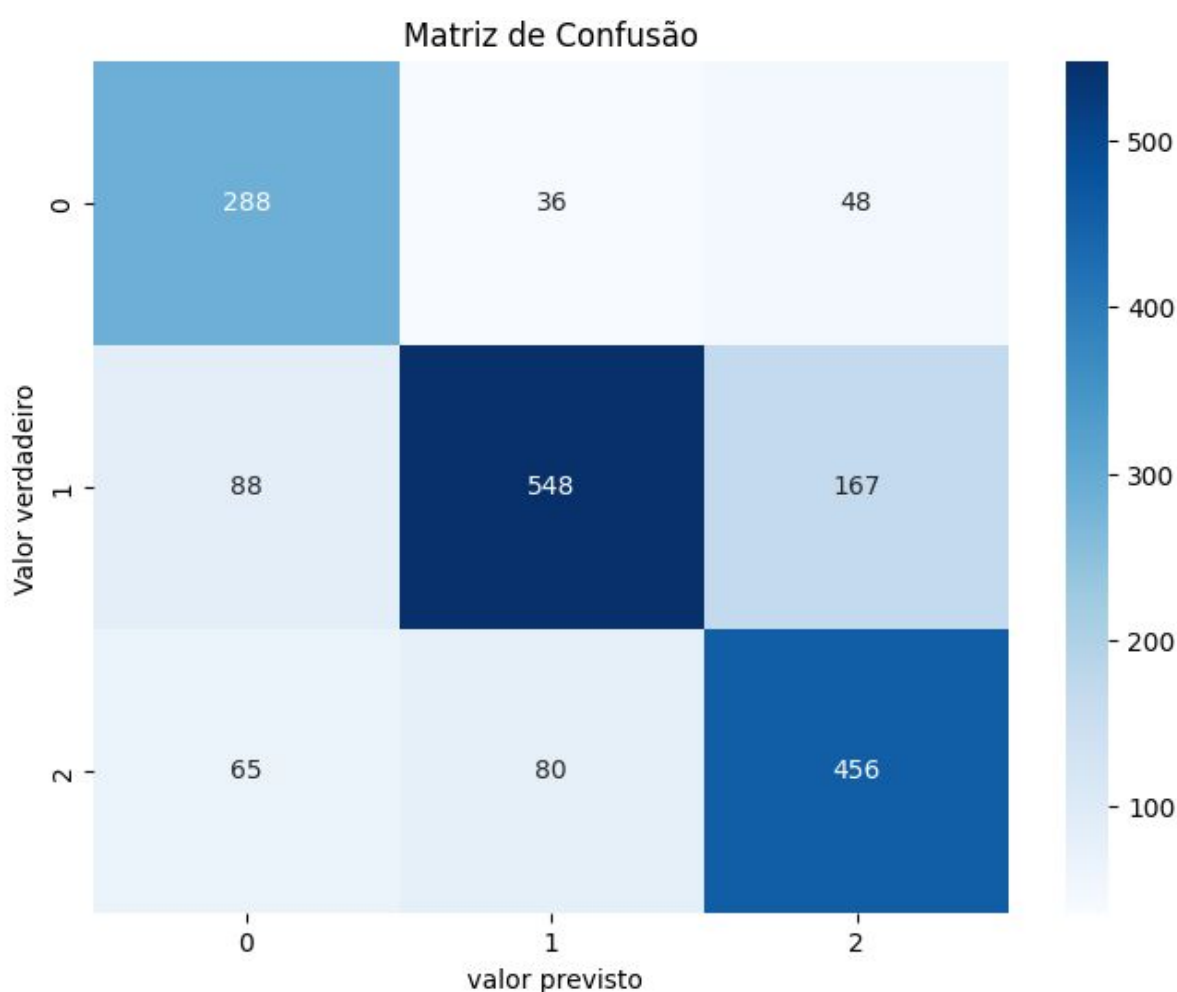


Figura 4. Matriz de confusão do modelo Naive Bayes com vetorização do tipo Bag of Words, onde 0 representa comentários neutros, 1 representa comentários positivos e 2 os comentários negativos..

-Métricas:

| | |
|------------------|-----------------|
| <i>Modelo</i> | <i>NB – BoW</i> |
| <i>Acurácia</i> | 73,9% |
| <i>Revocação</i> | 72,7% |

Tabela 13: Tabela com as métricas acurácia e revocação do modelo Naive Bayes com BoW.

8.2. Naive Bayes com Word2Vec:

- Matriz de confusão:

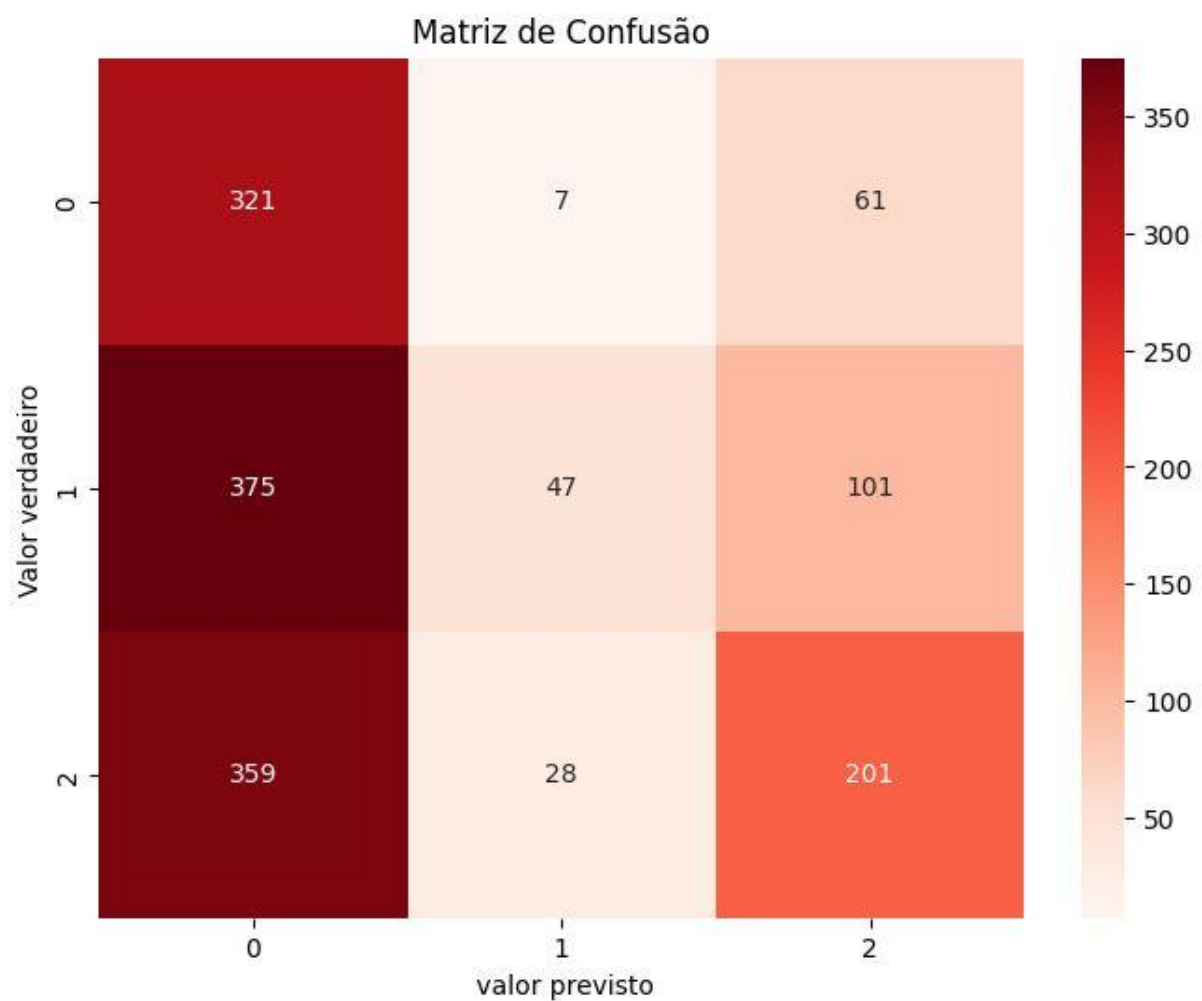


Figura 5. Matriz de confusão do modelo Naive Bayes com vetorização do tipo Word2Vec, onde 0 representa comentários neutros, 1 representa comentários positivos e 2 os comentários negativos..

-Métricas:

| | |
|------------------|-----------------|
| <i>Modelo</i> | <i>NB – W2V</i> |
| <i>Acurácia</i> | 49,5% |
| <i>Revocação</i> | 37,9% |

Tabela 14: Tabela com as métricas acurácia e revocação do modelo Naive Bayes com W2V.

8.3. Support Vector Machine com Word2Vec:

- Matriz de confusão:

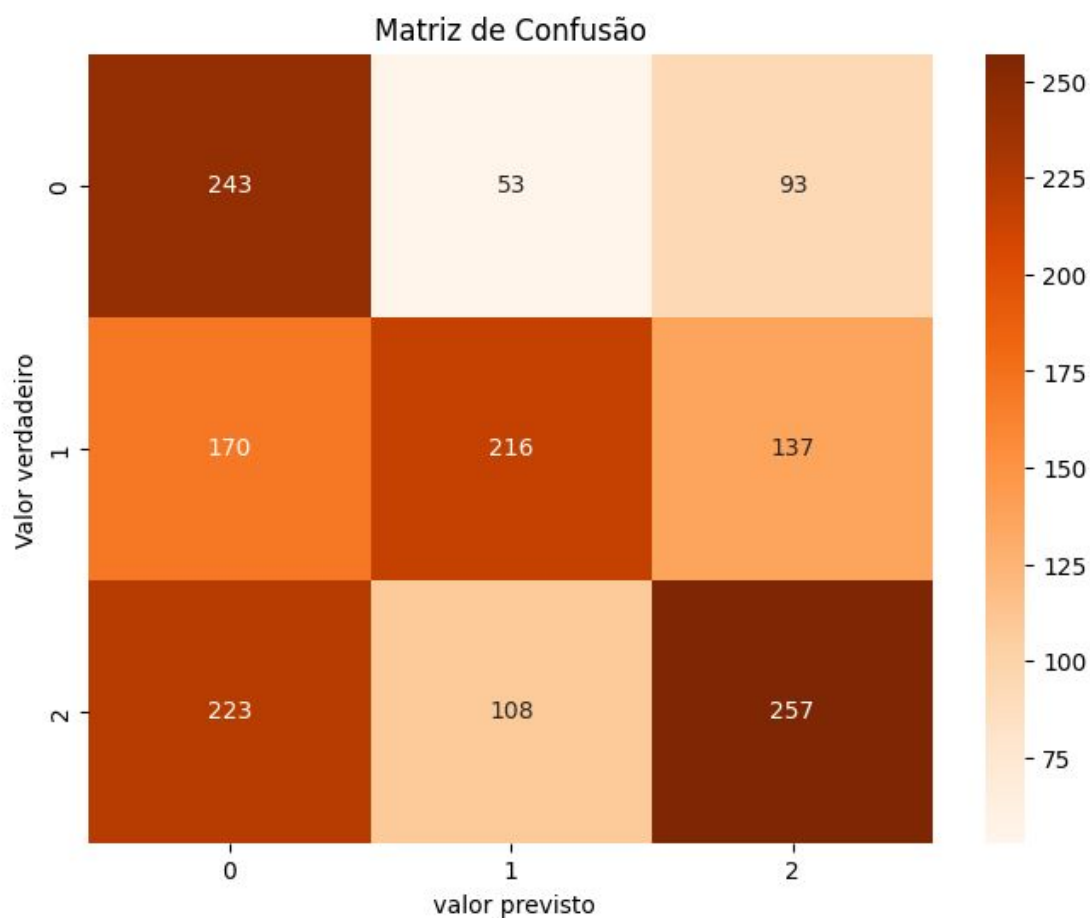


Figura 6. Matriz de confusão do modelo Support Vector Machine com vetorização do tipo Word2Vec, onde 0 representa comentários neutros, 1 representa comentários positivos e 2 os comentários negativos..

-Métricas:

| | |
|------------------|------------------|
| <i>Modelo</i> | <i>SVM – W2V</i> |
| <i>Acurácia</i> | 50,5% |
| <i>Revocação</i> | 47,7% |

Tabela 15: Tabela com as métricas acurácia e revocação do modelo Support Machine com vetorização do tipo Word2Vec.

8.4. Redes Neurais com Transformers:

- Matriz de confusão:

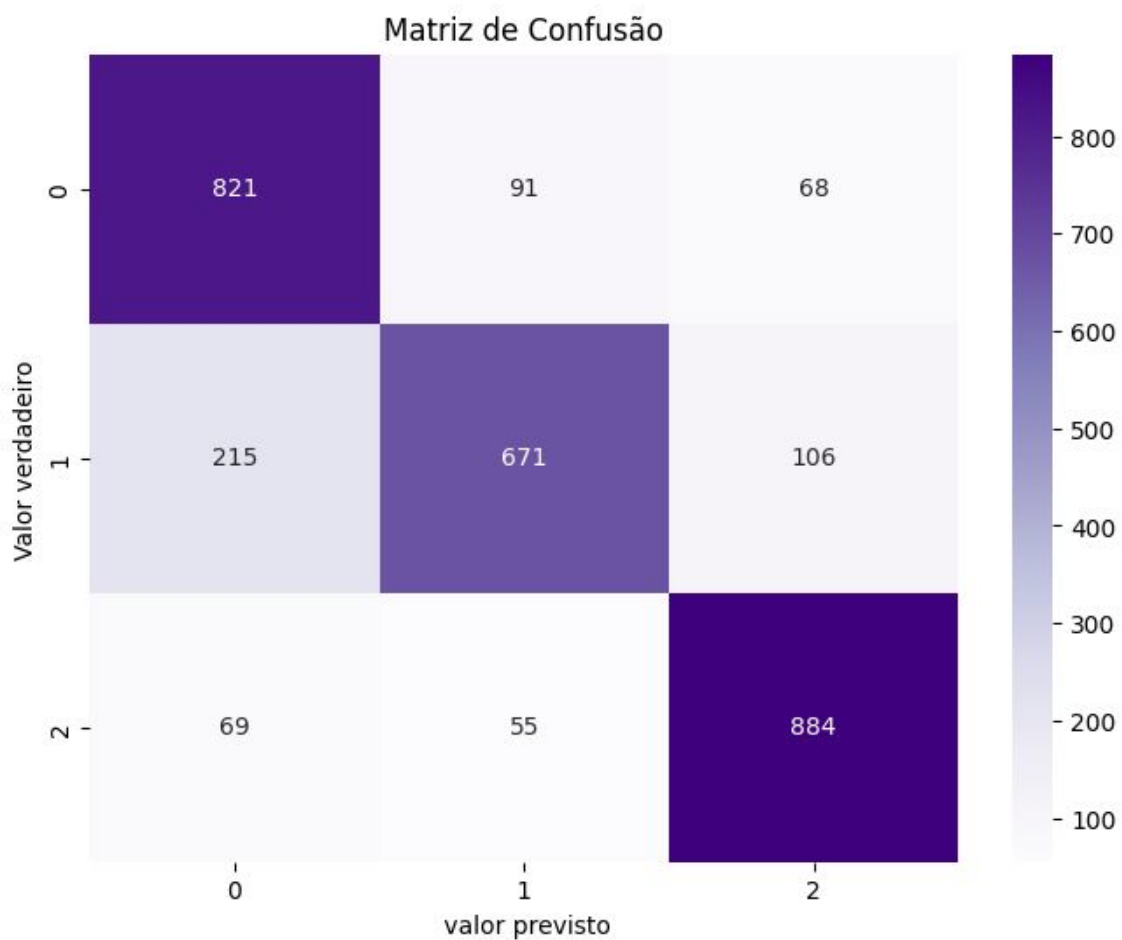


Figura 7. *Matriz de confusão do modelo Rede Neural com vetorização do tipo Transformers, onde 0 representa comentários neutros, 1 representa comentários positivos e 2 os comentários negativos..*

| | | | |
|-----------------|---|------------------|---------------|
| Legenda: | | -Métricas: | |
| <i>Positivo</i> | 0 | <i>Modelo</i> | <i>RN – T</i> |
| <i>Neutro</i> | | <i>Acurácia</i> | 80,0% |
| <i>Negativo</i> | 2 | <i>Revocação</i> | 79,7% |

Tabela 16: *Tabela com as legendas positivo, neutro, negativo.*

Tabela 17: *. Tabela com as métricas de acurácia e revocação do modelo Rede Neural com vetorização do tipo Transformers.*

9. Discussão e Conclusão

Com os resultados obtidos, os modelos serão confrontados a partir dos valores de acurácia e revocação, bem como foi destacado na definição de escolha dos critérios. Também serão levantados possíveis pontos fortes e limitações dos modelos e também do tipo de vetorização utilizada pelos mesmos.

A começar pelo modelo Naive Bayes, que apresentou uma acurácia muito maior utilizando a vetorização de Bag of Words que Word2Vec, 73,9% e 49,5% respectivamente, e uma diferença ainda maior levando em conta a revocação do modelo utilizando Bag of Words em comparação ao que utilizou o Word2Vec, 72,7% e 37,9% respectivamente, levanta algumas hipóteses. O modelo Naive Bayes, é um modelo de aprendizado de máquina relativamente simples, baseado no teorema de Bayes, e na suposição de independência condicional entre os atributos. Isso torna o modelo mais rápido de treinar, em relação a modelos mais complexos, como os de redes neurais, por exemplo. Tendo isso em mente, ao analisar os resultados que utilizaram diferentes tipos de vetorização, pelo Naive Bayes ser baseado na suposição de independência condicional entre os atributos, no caso deste projeto, assumindo que as palavras no texto são independentes umas das outras, essa suposição pode ser mais adequada em cenários onde a relação semântica e contextual não são tão importantes.

A vetorização em Word2Vec captura essas relações semânticas entre as palavras, considerando a ordem e o contexto, diferente da vetorização em Bag of Words, que considera apenas a frequência em que as palavras ocorrem na base de dados do vocabulário criado. Isso pode justificar uma melhor performance da modelagem do Naive Bayes utilizando a vetorização em Bag of Words que o Word2Vec.

O modelo SVM apresentou uma acurácia e revocação de aproximadamente 50.5%, com a vetorização em Word2Vec, e apesar de ter hiperparâmetros ajustados com grid search, esse resultado indica que o modelo não conseguiu traçar um hiperplano eficaz para classificar os elementos do banco de dados, sugerindo que este modelo não conseguiu capturar as complexas relações semânticas da vetorização, tornando este um modelo ineficaz para análise de sentimentos deste banco de dados específico.

9.1. Comparação entre os modelos

Durante as últimas etapas de desenvolvimento, diversos modelos de predição classificatória foram explorados, resultando, de forma geral, em resultados satisfatórios para o problema proposto. No entanto, algumas escolhas não alcançaram as métricas esperadas e, conseqüentemente, foram descartadas. Um exemplo disso é o modelo Support Vector Machine, que apresentou uma acurácia

de 50,5% e uma revocação de 47,7%, indicando uma dificuldade em acertar os targets dos comentários analisados.

Da mesma forma, o modelo Naive Bayes, quando utilizado com a técnica de vetorização word2vec, mostrou dificuldades na predição dos sentimentos, com uma acurácia de 49,5% e uma revocação de 37,9%. Apesar desses resultados desfavoráveis, dois modelos se destacaram com métricas satisfatórias, sendo um deles o modelo Naive Bayes com Bag of Words.

Com uma abordagem de modelagem e metodologia mais simples, o modelo Naive Bayes, empregando a vetorização com Bag of Words, alcançou métricas superiores a 70%. Esses resultados encorajadores evidenciam a efetividade desse modelo na classificação de comentários. No entanto, em termos de desempenho, o modelo de rede neural baseado na arquitetura Transformers se destacou ainda mais.

O modelo de rede neural com Transformers demonstrou uma performance superior em relação aos demais modelos analisados. Sua capacidade de lidar com a complexidade do problema e obter resultados promissores o tornou a escolha para compor a solução final.

Em resumo, embora o Support Vector Machine e o Naive Bayes com word2vec tenham apresentado limitações, o modelo Naive Bayes com Bag of Words mostrou-se sólido e foi considerado como uma opção viável. No entanto, o modelo de rede neural com Transformers se destacou como o mais eficiente entre todos os modelos avaliados, apresentando ótimos resultados e sendo escolhido como a solução final.

| <i>Modelo</i> | <i>NB – BoW</i> | <i>NB – W2V</i> | <i>SVM – W2V</i> | <i>RN – T</i> |
|------------------|-----------------|-----------------|------------------|---------------|
| <i>Acurácia</i> | 73,9% | 49,5% | 50,5% | 80,0% |
| <i>Revocação</i> | 72,7% | 37,% | 47,7% | 79,7% |

Tabela 18: Tabela com as métricas dos modelos utilizados.

9.2. Modelo escolhido

Tendo sido apresentados todos os resultados dos modelos utilizados neste projeto, o modelo de Rede Neural com vetorização por Transformers obteve a maior acurácia e revocação, 80% e 79% respectivamente. A fim de evitar o risco de overfitting, os resultados de treino e teste foram aplicados em um gráfico de curvatura de acurácia (Figura 6), para validar o desempenho desse modelo, para

avaliar se o modelo está conseguindo generalizar a sua predição, e não apenas memorizando os dados de treino.

Este gráfico permite a visualização do desempenho do modelo à medida que o tamanho do conjunto de treinamento aumenta, em épocas (cada unidade de época representa a passagem do modelo em todo o conjunto de treinamento).

Gráfico curvatura de acurácia

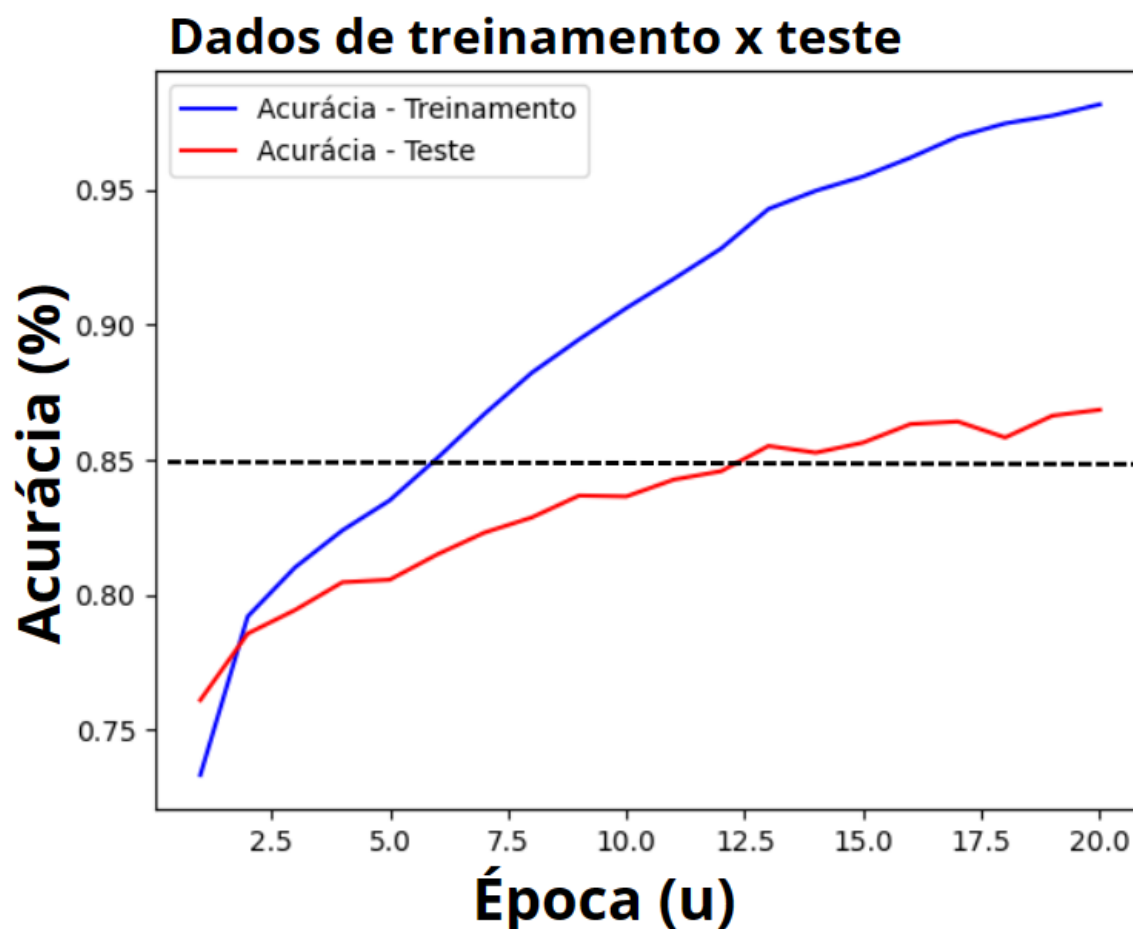


Figura 8. Gráfico de Curvatura do Modelo de Rede Neural com Vetorização do tipo Transformers.

Com base no gráfico, é visível que conforme o modelo passa por todo o conjunto de dados repetidamente, em cada época, a acurácia do treinamento aumenta rapidamente (curva azul), em relação à acurácia nos dados de teste (curva vermelha), que também apresenta um crescimento, porém mais discreto. Ao final de 20 épocas, a linha de treino (azul) chega a 100% de acurácia. Já a linha de teste (vermelha) atinge um valor de 80% de acurácia, aproximadamente, entre 12 e 13 épocas, enquanto a linha de treino (azul) ainda está com 90% de acurácia, aproximadamente.

Por estar com uma média de 10% a mais na acurácia de treinamento, em relação à acurácia do teste, 90% e 80%, respectivamente, entre a época 12 e 13, o gráfico indica que não há um condicionamento por overfitting expressivo no modelo, mantendo um grau de generalização relevante, se as épocas forem limitadas neste valor.

9.2.1. Critérios e justificativas para escolha do modelo

É importante definir os critérios para a escolha do modelo mais adequado ao problema proposto neste projeto, e para isso, é necessário considerar o valor mais relevante das métricas utilizadas, ressaltando que a métrica mais relevante pode variar, de acordo com a classe mais importante a ser classificada de forma correta, levando em conta qual tipo de erro da predição pode ser mais prejudicial ao usuário que irá utilizar a solução desenvolvida neste projeto.

Dentre as três classes de categorias de sentimentos, Positivo, Neutro e Negativo, o sentimento mais importante de ser identificado de forma correta é o Negativo, pelo impacto que elementos dessa classe podem apresentar ao usuário do sistema desenvolvido. Sendo assim, o erro mais crítico é o de Falsos Negativos, que podem passar despercebidos como sendo de outras classes, enquanto é na verdade da classe Negativo.

Analisando este cenário, é desejável utilizar o modelo que apresenta um alto índice de revocação, ou seja, que mais corretamente identificam elementos como sendo da classe Negativo, reduzindo assim o número de Falsos Negativos. Sendo essa a principal métrica de avaliação para o critério de escolha do modelo com base nos resultados obtidos.

Também é importante destacar que, ao confrontar os resultados dos modelos, seja levado em consideração o modelo com acurácia mais alta, além da revocação, e o tipo de vetorização que o modelo melhor se desempenhou. A combinação desses critérios indicará o modelo mais adequado para o projeto.

Após considerar os critérios estabelecidos, o modelo escolhido para a modelagem preditiva foi a Rede Neural com vetorização do tipo Transformers de sentença. Esse modelo se destacou por sua capacidade de capturar o contexto das palavras e por sua complexidade, levando em consideração as relações entre elas.

Ao avaliar a importância de identificar corretamente a classe Negativo, levando em conta o impacto que elementos dessa classe podem ter para o BTG, o modelo escolhido apresentou uma alta revocação, garantindo uma correta identificação dos verdadeiros negativos e reduzindo os falsos negativos.

Os resultados obtidos demonstraram uma acurácia de 80% e uma revocação de 79,7% para o modelo. Essas métricas refletem a capacidade do modelo em

identificar corretamente os sentimentos dos textos, principalmente da classe Negativo.

Portanto, com base nos critérios estabelecidos, a escolha do modelo de Rede Neural com vetorização do tipo Transformers se mostrou adequada para o projeto, pois atende tanto ao objetivo de identificar corretamente a classe Negativo quanto à obtenção de uma alta eficácia e revocação. Isso permitirá uma análise precisa dos sentimentos dos textos, maximizando a eficácia do sistema desenvolvido.

10. Deploy e visualização do modelo

Ferramenta de visualização:



Figura 9: Imagem de exemplo de visualização da solução, após o seu deploy com Gradio.

Como o modelo foi transferido:

Vale mencionar que o modelo utilizado no código de nossa interface de visualização foi importado do notebook de treinamento de modelos do nosso grupo. E posteriormente carregado em uma função de predição, a qual é chamada por nossa interface para análise de sentimento e criação de gráficos.

Plataforma de visualização:

Para criação de nossa interface foi utilizada a biblioteca Gradio, para visualização, em conjunto com um código em python, criação de gráficos e processamento das informações. A escolha do gradio se deve ao fato de ele possibilitar que os inputs fornecidos sejam processados e devolvidos em forma de dashboards de forma rápida e eficiente.

Inputs necessários:

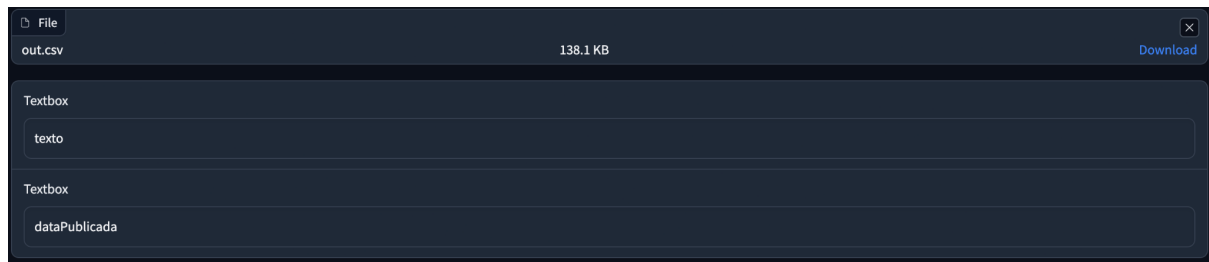


Figura 10: Inputs necessários para o início do processamento.

Arquivo csv:

O arquivo em forma de csv contém as colunas necessárias para fazer o processamento, sendo necessárias uma coluna com os textos a serem analisados e outra com sua data de publicação.

Nome da coluna que contém os textos:

A coluna do arquivo csv, que contém os textos das publicações para serem processados dentro do modelo.

Nome da coluna que contém as datas das publicações:

A coluna do arquivo csv, que contém as datas das publicações, é importante para fazer o gráfico de sentimentos dos comentários em função do tempo.

Outputs esperados:

Número de linhas:

Para que o usuário tenha um entendimento melhor sobre a quantidade de linhas que tem dentro do arquivo, ele pode visualizar esse número em nossa interface.

Textbox

Número de Tuplas: 450

Figura 11: *Output de metadados esperados.*

Palavras mais frequentes:

A nossa interface é capaz de calcular as palavras mais frequentes presentes nos textos do csv fornecido. Gerando através de um gráfico feedback quais são as palavras mais frequentes e quantas vezes cada uma aparece.

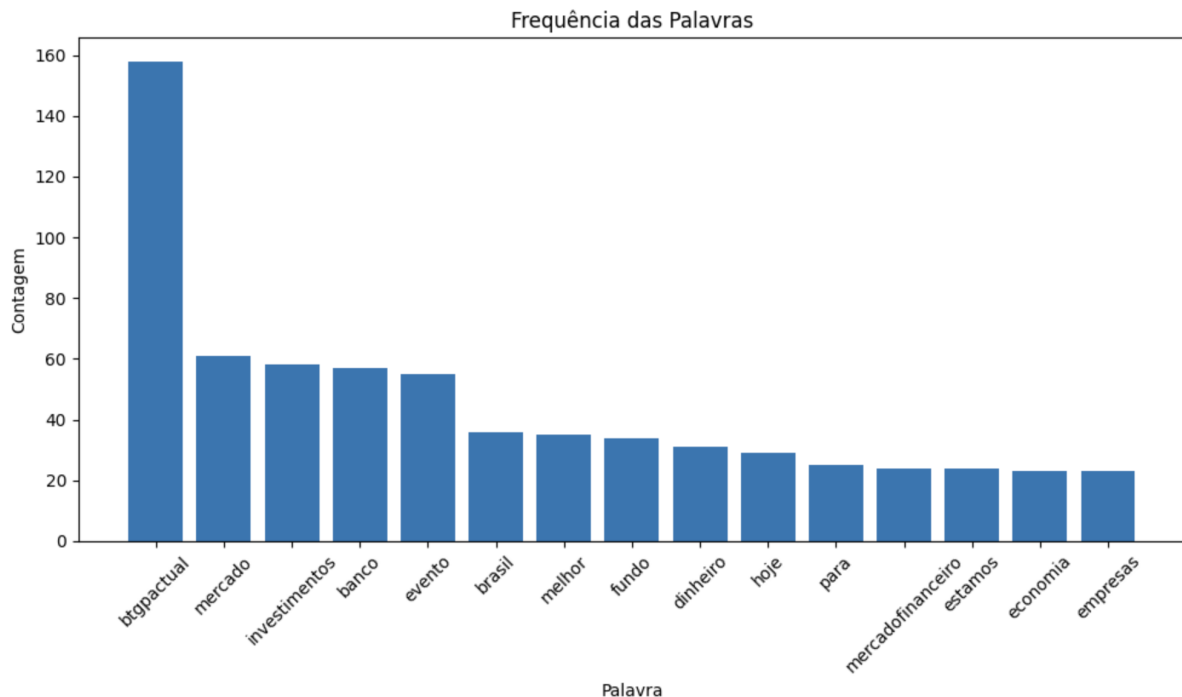


Gráfico 2: *Frequência das palavras mais frequentes presentes.*

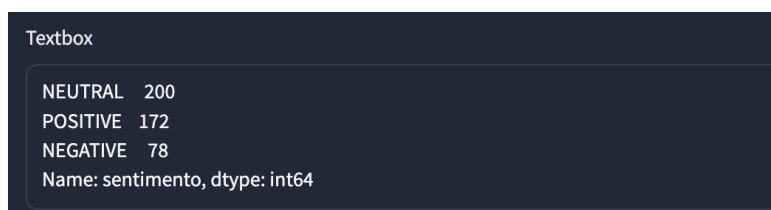
Nuvem de palavras:

A nossa interface é capaz de gerar nuvem de palavras com as palavras mais frequentes que aparecem no csv fornecido para ela. Possibilitando assim insights sobre tendências e assuntos em alta presentes nos comentários.



Sentimento do Corpus:

A nossa interface é capaz de classificar, através de nossa rede neural, todos os textos em positivos e negativos e neutros. E depois, através de nossa interface, fornecer feedback sobre o sentimento dentro do dataframe.



Distribuição de Sentimento:

Através da classificação dos sentimentos presentes nas frases, nossa interface é capaz de gerar um gráfico que fornece o número total de positivos neutros e negativos. Possibilitando a visualização do sentimento mais e menos recorrente.

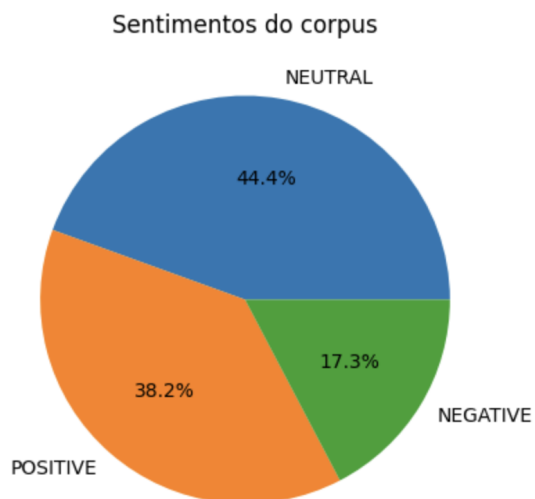


Gráfico 3: Gráfico do número total de positivos neutros e negativos.

Sentimento em Função do Tempo:

Através da classificação dos sentimentos presentes nas frases, em conjunto com a data de publicação dos textos, nossa interface é capaz de gerar um gráfico de sentimento por tempo. Possibilitando a visualização dos sentimentos mais recorrentes em determinado período de tempo.

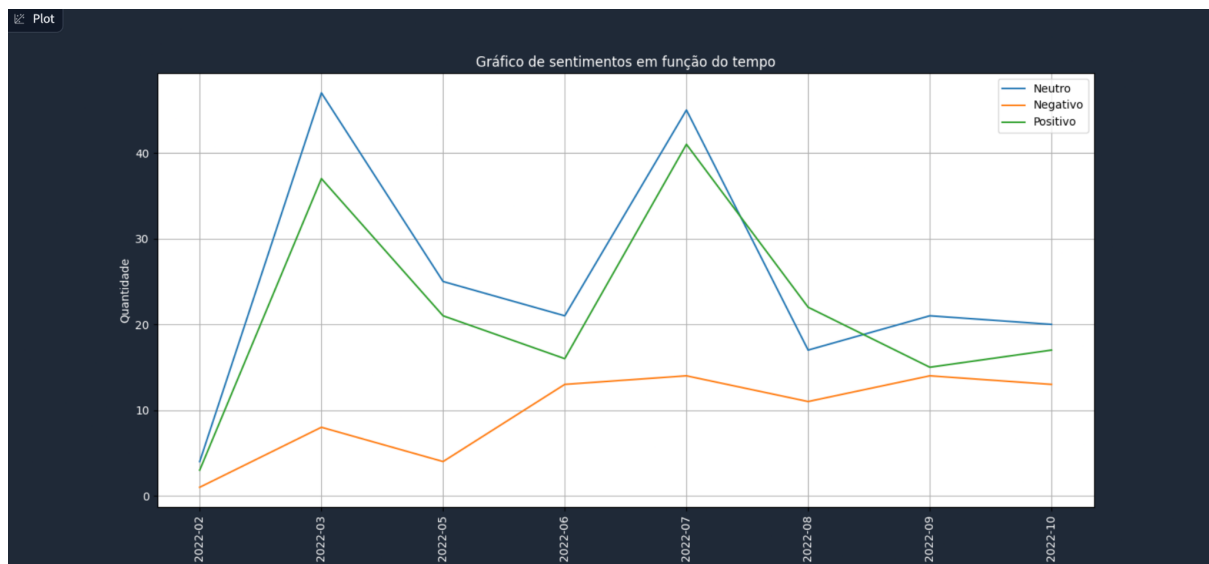


Gráfico 4: Gráfico da classificação dos sentimentos neutro, negativo, positivo presentes nas frases.

11. Ferramentas

- Google Colab: Plataforma baseada em nuvem para escrever e executar códigos Python em um ambiente de notebook colaborativo.
- GitHub: O GitHub é uma plataforma de hospedagem de código-fonte e colaboração, para hospedagem e compartilhamento de projetos, versionamento de código, entre outros.
- Jupyter: Aplicação web de código aberto que permite criar e compartilhar documentos que contêm código interativo, visualizações e texto narrativo.
- Documentos CSV: Formato de arquivo utilizado para armazenar dados em formato tabular, em que cada linha representa uma entrada e os valores são separados por vírgulas. Muito útil para importar e exportar dados de bases de dados.

12. Bibliotecas

- Pandas : biblioteca para visualização de dataframes
- Numpy : biblioteca para operações matemáticas e manipulação de arrays
- NLTK : biblioteca com kit para operações e processamento de linguagem natural
- Spacy : biblioteca com um kit para PLN, utilizada para stopwords no projeto.
- Unidecode: Biblioteca utilizada para remoção de acentos em textos
- Re:Biblioteca para o uso de expressões regulares
- Emoji: Biblioteca para transformar emoji em textos
- Pickle: Biblioteca para exportação de modelos no formato PKL
- Sklearn: Biblioteca para o treinamento de modelos preditivos
- Genshin: Biblioteca utilizada para o treinamento de redes neurais
- Mathplotlib: Bibliotecas para plotagem e criação de gráficos
- Tensorflow: Biblioteca para o treinamento de redes neurais
- Sentence_Transformers: Biblioteca para representação vetorial de textos através de modelos pré-treinados.
- Wordcloud: Biblioteca capaz de gerar nuvem de palavras.
- Gradio: Biblioteca para criação de interface gráfica para visualização dos dados obtidos com modelo.
- Tempfile: Bibliotecas para criação de arquivos temporários para ser plotado no gradio.

13. Referências bibliográficas

Lee, J., Warner, E., Shaikhouni, S. et al. "Unsupervised machine learning for identifying important visual features through bag-of-words using histopathology data from chronic kidney disease", Scientific Reports, 2022.

Qader, W. A., Ameen, M. M. and Ahmed, B. I. "An Overview of Bag of Words;Importance, Implementation, Applications, and Challenges", International Engineering Conference (IEC), 2019.

Rudkowsky, E., Haselmayer, M., Wastian, M., Jenny, M., Emrich, S., Sedlmair, M., "More than Bags of Words: Sentiment Analysis with WordEmbeddings", Communication Methods and Measures, 2018.

Powell, R. T., Olar, A., Narang, S., Rao, G., Sulman, E., Fuller, G. N., Rao, A. "Identification of Histological Correlates of Overall Survival in Lower Grade Gliomas Using a Bag-of-words Paradigm: A Preliminary Analysis Based on Hematoxylin & Eosin Stained Slides from the Lower Grade Glioma Cohort of The Cancer Genome Atlas", Journal of Pathology Informatics, 2017.

<http://nilc.icmc.usp.br/nilc/index.php/repositorio-de-word-embeddings-do-nilc>

CHEN, D., NIGRI, E., OLIVEIRA, G.,SEPULVENE, L., ALVES, T.: Métricas de Avaliação em Machine Learning: Classificação - Kunumi Blog, medium, 2020.

FRANCESCHI, P, R.: Modelagens Preditivas de Churn: O Caso do Banco do Brasil, Universidade do Vale do Rio dos Sinos, 2019.

Qi, Y., Sachan, D. S., Feliz, M., Padmanabhan, S. J., Neubig, G., When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?, Language Technologies Institute, Carnegie Mellon University, 2018.

Feng, B., Thuremella, D., A Tale of Two Encodings: Comparing Bag-of-Words and Word2vec for VQA, Princeton University, 2018.

<https://www.inf.ufpr.br/dagoncalves/IA07.pdf>

<https://medium.com/turing-talks/turing-talks-12-classificação-por-svm-f4598094a3f1>