UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Análise de sentimento para língua portuguesa: *Fine-Tuning* de Modelos de *Word Embedding* Contextuais

Renan Peres Martins

São Carlos – SP

Análise de sentimento para língua portuguesa: *Fine-Tuning* de Modelos de *Word Embedding* Contextuais

Renan Peres Martins

Orientador: **Prof. Dr. Ricardo Marcondes Marcacini**

Monografia final de conclusão de curso apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação.
Aprendizado de máquina e Mineração de opiniões em dados textuais

USP – São Carlos
Junho de 2022

AGRADECIMENTOS

Agradeço, principalmente, e de todo meu coração, a minha mãe, Andréa, e meu pai, Marcos, que sempre me deram suporte e amor. Agradeço a minha namorada Marianna que me apoiou, confiou e sempre viu o melhor em mim. Ao meu orientador, prof. Dr. Ricardo Marcacini, pela oportunidade, diálogos e transmissão de conhecimento ao longo da nossa jornada nesse projeto. A todos os professores que tive ao longo de minha vida, que me ajudaram a me descobrir como cidadão e me formar como um profissional ético e qualificado.

Por fim, mas com igual importância, agradeço aos meus amigos André Prado, Bruno Stefano, Diego Parra, Edson Andrade, Hiago de Franco, Leonardo Dias, Marcelo Coelho, Marcos Arce, Mateus Doimo, Pedro Natali, Raphael Costa, Victor Amaral e todos que estiveram presente e compartilharam comigo noites de estudo no bloco 4 e conversas no bandejão.

Agradecimentos especiais são direcionados ao Laboratório de Inteligência Computacional da Universidade de São Paulo (LABIC/USP) e seus pesquisadores.

“Ninguém ignora tudo. Ninguém sabe tudo.
Todos nós sabemos alguma coisa. Todos nós ignoramos alguma coisa.
Por isso aprendemos sempre.”
(Paulo Freire)

RESUMO

MARTINS, R. P.. **Análise de sentimento para língua portuguesa: *Fine-Tuning* de Modelos de *Word Embedding* Contextuais**. 2022. 53 f. Monografia (Graduação) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

A ampla inserção de tecnologia no cotidiano, seja nas formas de comunicação, vendas ou entretenimento, proporciona um cenário propício ao desenvolvimento de inteligências artificiais voltadas à compreensão das expressões de sentimentos e opiniões humanas. Para que haja cenários de avanço real nas pesquisas e desenvolvimento de modelos e aplicações para análise de sentimentos mais Assertivo e com melhores desempenhos, é fundamental a divulgação de modelos de base funcionais e com fácil transmissão de conhecimento para dada linguagem.

Quando volta-se à língua portuguesa, há um cenário que ainda carece de recursos para treinamento de modelos computacionais para a análise de sentimentos. Dessa forma, esse projeto estabelece técnicas e métodos para coleta de comentários textuais em português com anotação de sentimentos para diferentes domínios de aplicação de modo a apresentar um *dataset* robusto para treinamento. Além disso, investiga-se a qualidade e desempenho após um ajuste fino, *Fine-Tuning*, em dado modelo de redes neurais profundas baseadas em *word embeddings* para classificação de textos - baseado no modelo pré-processado de *codificador bidirecional de Transformers - BERT*.

Assim, apresenta e disponibiliza-se o modelo ajustados e o *dataset* extraído para uso aberto em demais projetos que necessitem da funcionalidade de análise de sentimentos para a língua portuguesa, contribuindo para o desenvolvimento sustentável da comunidade nesse idioma.

Palavras-chave: Análise de sentimento, Mineração de dados para língua portuguesa, Word Embedding, Processamento de linguagem natural, Modelo BERT, *Reviews* de aplicativos.

ABSTRACT

MARTINS, R. P.. **Análise de sentimento para língua portuguesa: *Fine-Tuning* de Modelos de *Word Embedding* Contextuais**. 2022. 53 f. Monografia (Graduação) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

The insertion of technology in everyday life, whether in the social networks, business or communication, offers a great settings for the development of IA systems aimed at predicting human feelings and opinions. For there, it's essential to provide great base models with easy transmission of knowledge, for advancement in researchs and development of models and applications for a given language.

The Portuguese language needs resources for computational evironment models, including Sentiment Analysis. This study apply some techniques and methods for extracting samples for textual's excerpts in Portuguese with annotation of feelings for different applications domains. Furthermore, it investigates the quality and performance of a fine-tuning training for models of deep NPL based on word embeddings for text classification - using the pre-processed model *Bidirectional Encoder Representations from Transformers - BERT*.

Once completed, the model and dataset has been available for open use in other projects that need the sentiment analysis functionality for the Portuguese language.

Key-words: Sentiment analysis, Opinion mining for portuguese language, Bidirectional Encoder Representations from Transformers - BERT Model, App's reviews.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

Capítulo 1

INTRODUÇÃO

1.1 Motivação e Contextualização

Desde a pré-história, a organização social pauta-se no estabelecimento de relações sociais e transmissão de conhecimento, informações e opiniões (BARROS; SOUZA; TEIXEIRA, 2020). O desenvolvimento da comunicação humana organizada e estruturada em símbolos fonéticos e, posteriormente, gráficos caminha ao lado do estabelecimento de sociedades volumosas e organizadas, surgimento de representantes de poder e, até mesmo, leis para defender o **interesse** coletivo dos grupos de pessoas. Ao suprassumo do debate de ideias e influência da opinião do emissor aos ouvintes tem-se o período popularmente conhecido como *democracia ateniense*, onde um conjunto limitado de homens nascidos na cidade expressavam seus descontentamentos a fim de convencer uma maioria e definir leis para a *poli* (ABRAO, 2004).

Com a revolução tecnológica e a tomada dos meios de comunicação pela *internet* e redes sociais, a expressão de sentimentos tornou-se muito mais simples, de fácil acesso e com potencial de influência muito grande. Assim, surgem os recentes estudos acadêmicos e privados focados em estabelecer modelos computacionais que consigam compreender a expressão e sentimento humano em suas expressões na *internet* - por meio de manifestações diretas em textos ou até padrões de comportamentos em aplicativos.

A importância e constância desse tipo de estudo e o uso dessas tecnologias nos dias atuais é visível, ora pelos escândalos envolvendo redes sociais com aquisição de dados e comportamentos sem consentimento dos usuários para impactar seu desempenho financeiro (ALECRIM, 2017) e pelas investigações de influência política durante as recentes eleições pelo mundo, ora por projetos voltados para a área de educação (MISURACA; SCEPI; SPANO, 2021) e prevenção às *Fake News* a respeito da pandemia de *COVID-19* (FREIRE; GOLDSCHMIDT, 2021). Dessa forma, é fundamental que se desenvolvam projetos abertos que fundamentem outros sucessivamente, para um maior entendimento da tecnologia e utilização da mesma de modo a trazer frutos positivos a sociedade e combater o uso danoso.

Dito isso, a língua portuguesa ainda carece de modelos base funcionais para análise de sentimentos. Para estabelecer um desses modelos, é necessário atingir uma base de dados de amostras com um domínio de estudo e objetivos claros e com possibilidade de expansão de horizontes para projetos futuros. Quadro contextual que encaixa-se muito bem para o treinamento orientado a **avaliações** na *internet* - para o projeto utiliza-se **sites de aplicativos** - pois trazem

um excerto textual redigido por pessoas comuns, utilizando de linguagem funcional (CASTRO, 2015). Isso possibilita que se obtenha amostras em demasia para um público geral e com indicação de sentimento já atribuído, por meio das notas relacionadas ao *review*. Para utilizar esse sentimento inerte aos textos, é vantajoso a escolha de um modelo que objetive uma análise para polarização desse, definindo-os em categorias espectralmente opostas, como 'positivo' e 'negativo' ou 'bom' e 'ruim' (ARAUJO; GOLO; MARCACINI, 2022).

Para elaboração desse modelo, é necessário que se realize uma **mineração de opinião** com as amostras de dados do domínio selecionado. Nela extraem-se as entidades e contexto que compõem o mapeamento dos sentimentos, de acordo com uma das possíveis adaptações da **quíntupla de sentimento**. Nesse formato, tem-se as entidades - que representam o aplicativo - dada suas características extraídas de seu contexto - que são as categorias de aplicativos. De modo que a polarização da classificação do sentimento empregado é dado pela relação entre esses elementos (YUGOSHI, 2018).

A maneira que esse modelo é implementado para uma linguagem natural é a tradução das palavras em vetores numéricos, onde suas posições expressam as relações de similaridade entre elas, uma técnica denominada **Word Embedding**. Esses valores associados as posições dos vetores são determinados por métricas matemáticas que descrevem a distância entre as palavras em um plano cartográfico com n dimensões, sendo n o número de elementos associados ao vetor. Assim é possível determinar, por meio do contexto, quais palavras tem maiores proximidades, consequentemente, maior relação com aquele.

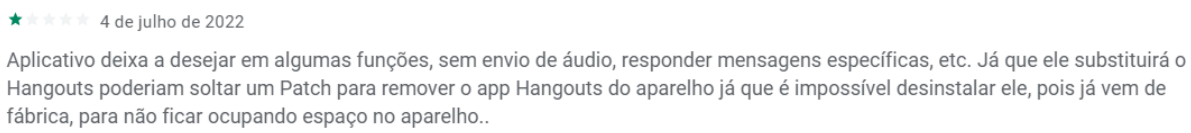
Para que a ideia de continuidade na construção de um ambiente saudável para a língua portuguesa, é necessário que esse modelo seja estruturado sobre um método que permita facilmente a **transmissão de conhecimento** adquirido durante os projetos e seja eficiente. Assim, o projeto *open source* para treinamento bidirecional de *transformers* **BERT** que é pré-treinado para o modo multilinguagem, desenvolvido e utilizado atualmente dentro do motor de pesquisa da *Google*, atende as necessidades ao permitir o refinamento de seu desempenho a uma dada função. Nesse caso, sua capacidade de identificar palavras em regiões e contexto será aplicado para classificar os sentimentos contidos por esses textos (DEVLIN *et al.*, 2018).

1.2 Objetivo

O objetivo principal deste projeto é contribuir com o avanço de modelos para análise de sentimentos na língua portuguesa, por meio de um ajuste fino (*Fine-Tuning*) de um algoritmo de aprendizado profundo pré-treinado e consolidado (*Bidirectional Encoder Representations from Transformers* - BERT *multilingual*). Para o domínio, opta-se pelo uso das avaliações de aplicativos disponíveis na *Google Play Store*, por apresentar uma quantidade colossal em volume de dados e dispor das características linguísticas e semânticas desejadas. Na Figura 1, tem-se um modelo de amostra desejada, nele há um comentário avaliativo realizado sobre um aplicativo

(nesse caso *Google Chat*). A partir dele, tem-se construções de frases com termos associados que revelam um sentimento a respeito desse aplicativo, nesse caso, um sentimento negativo devido a avaliação de uma estrela.

Figura 1 – Exemplo de amostra desejada para o projeto. Extraído de: Google Play - Aplicativo: 'Google Chat'



Além disso, pretende-se avaliar o uso de categorias como forma de contexto, substituindo processos de extração de aspecto, normalmente utilizados, para esse tipo de abordagem e validando a eficiência do mesmo. Para o caso em exemplo na Figura 1, o aplicativo, *Google Chat*, pertence a categoria "Corporativo", portanto seu contexto seria associado a tal.

Com o intuito de concretizar o objetivo principal, define-se metas específicas, sendo elas:

- Coletar e organizar um corpus textual para treinamento de análise de sentimentos sobre diferentes categorias;
- Desenvolver e avaliar épocas de um modelo ajustado, comparando performance gerais e por categoria entre as opções;
- Avaliar o efeito das categorias como contextualização de predições em relação a um modelo de predição livre de contexto;
- Disponibilizar um modelo eficiente de avaliação de opiniões positivas ou negativas dado um excerto.

1.3 Organização

Esta monografia apresenta os conceitos, métodos, tecnologias e processos de desenvolvimento do projeto seguindo a estrutura apresenta a seguir.

No Capítulo 2 é apresentado os conceitos iniciais de mineração de opiniões - contextualizando seu uso, apresentando o nível de profundidade de análise, explicando o conceito da quíntupla de sentimento e esquematizando o fluxo de um projeto para mineração de opiniões; Aprofunda essa ideia especificando-a no uso aplicado a domínios textuais para *word embedding*, apresentando os conceitos e técnicas de *Word2Vec* e, posteriormente, o modelo *BERT* e seus processos de aprendizagem para as modelagens pré e pós treinadas.

O Capítulo 3 apresenta as etapas de desenvolvimento do projeto, mostrando os processos seguidos, os cuidados tomados, decisões de projeto e a forma de avaliação utilizada para obter os

resultados apresentados ao final. Além disso, apresenta as dificuldades e limitações enfrentadas no decorrer do desenvolvimento.

Por fim, o Capítulo 4 apresenta uma reflexão sobre as contribuições para as áreas que esse projeto traz, avalia se os objetivos iniciais foram alcançados e debate possíveis caminhos futuros que podem ser trilhados a partir desse projeto.

Capítulo 2

MÉTODOS, TÉCNICAS E TECNOLOGIAS UTILIZADAS

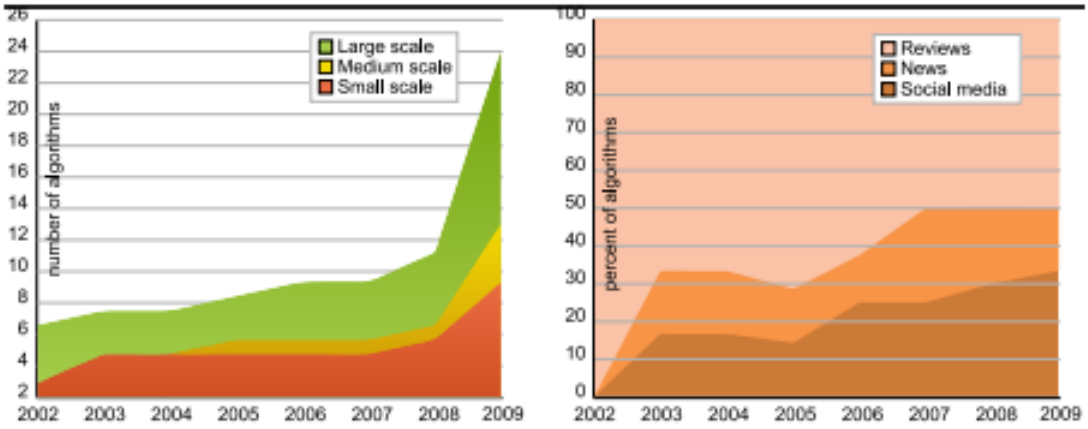
2.1 Mineração de Opiniões

Mineração de opiniões - ou análise de sentimentos - é uma vertente do desenvolvimento tecnológico recente de processamento de linguagem natural, que dialoga com áreas de mineração de dados, aprendizado de máquina e inteligência artificial. Pode ser definida como o estudo computacional sobre sentimentos que certos sujeitos projetam sobre algum objeto alvo ou sobre certas características desse (ZHANG; LIU, 2017), (PRAGER, 2007) e (KARIN; TUMITAN, 2013). Por sua vez, os sentimentos representam opiniões, visões, emoções e conceitos subjetivos ao comportamento humano e esse objeto pode representar inúmeras facetas com uma pessoa, empresa, serviço, produto, objeto físico, ou uma dúvida (YUGOSHI, 2018) e (GANTZ; REINSEL, 2010).

Seja recomendando um produto em uma loja de vendas, uma crítica gastronômica publicada em um site, ou um *tweet* criticando o atendimento de certa empresa, as relações humanas para o contexto da *internet*, pautam-se na expressão de opinião que um indivíduo apresenta sobre uma entidade e no reflexo dessa nos demais usuários ao qual a mensagem chega. Com a proliferação de inúmeras redes sociais, o volume de opiniões e o alcance das mesmas, ainda que proferidas por usuários anônimos e ocultados por um *nickname* fictício, cresce em proporções absurdas e desencadeia três fatores que fundamentam o cenário de mineração de opiniões: **um volume massivo de amostras** de opiniões para os mais diversos tipos de entidades; Uma necessidade de empresas, instituições e pessoas em zelar pela opinião pública sobre elas em ambiente virtual, que leva ao **investimento nessa tecnologia**, bem como estudos de instituições públicas e privadas no **desenvolvimento de algoritmos** capazes de compreender, analisar e prever o comportamento humano sobre certo assunto.

A Figura 2 (TSYTSARAU; PALPANAS, 2012), demonstra esse crescimento na escala dos algoritmos de análise de sentimento e a relação com os domínios de aplicação ao longo dos anos, onde pode ser observado um crescimento no tamanho das escalas e do tema de abordagem das redes sociais próximo ao seu *boom* global, nos arredores de 2010.

Figura 2 – A esquerda, comparativo entre a escala de dados dos algoritmos de análise de sentimento e a direita, percentual dos principais domínios de interesse dos algoritmos ao longo dos anos.



Fonte: Tsytsarau e Palpanas (2012).

2.1.1 Granularidade e Quíntupla de Opinião

Uma sentença com indicação subjetiva, considerada em análise de sentimentos, trabalha com dois elementos chaves que necessitam estar associados: um objeto alvo e um sentimento referente a ele (KARIN; TUMITAN, 2013). Desse modo, associa-os e classifica-se o sentimento polarizando em 'negativo' ou 'positivo'. O objeto alvo pode ser uma entidade (ao exemplo genérico: "Esse **produto** é bom") ou uma parte que compõe essa entidade (ao exemplo: "A **aparência desse produto** é feia), ele é quem vai receber a carga de sentimento, caso haja.

Há níveis de granularidade para cada tipo de análise:

- **Documento**: As análises se referem ao todo de um documento, identificando e polarizando a opinião de um ponto de vista geral (PANG; LEE; VAITHYANATHAN, 2002) e (TURNEY, 2002);
- **Sentença**: Análise de cada sentença ou cláusula que compõe o documento, uma a uma, identificando a polarização do sentimento para cada (WIEBE; BRUCE; O'HARA, 1999);
- **Entidade e atributos**: Realiza associações e análise entre cada conjunto de palavras que compõe as sentenças, que, por sua vez compõe o documento. Ou seja, define os sentimentos individuais empregados sobre um alvo ou dada característica sua, atribuindo polaridade entre as relações (YUGOSHI, 2018).

A **polaridade do sentimento** citada ilustra a intensidade e orientação do mesmo. A forma mais comum é a classificação entre sentimentos **Positivos** ou **Negativos**, por vezes, também considera-se o sentimento **Neutro**. Todavia há outras formas avaliativas como o uso de uma escala numérica e ou de diferentes classes que compõe um espectro emocional, que são considerados dependendo do intuito do modelo. Por exemplo, para se determinar o sucesso de

um filme na opinião popular, pode-se avaliar os comentários do público e da crítica especializada de forma polarizada em 'bom' e 'ruim', definindo um percentual de aceitação, ou definir uma escala mais detalhada, variando de 0 a 10, por exemplo.

A classificação entre as relações de sentimento com as entidades e atributos de um texto, utiliza-se do conceito da **quíntupla de Opinião**. Seja uma opinião O sobre um objeto alvo, ela será descrita por meio de suas entidades (e_i) - os nomes atribuídos a esse objeto - que possui os atributos e características (a_{ij}), na qual um sentimento é expresso por um indivíduos (h_k), em determinados instantes de tempo (t_l). Dessa forma, nossas polarizações de sentimento (s_{ijkl}) são relações dependentes desses elementos e, assim tem-se a quíntupla de elementos, expressa em 2.1, que definem a opinião sobre o alvo. O objetivo de uma análise de sentimentos é mapear todas as relações $(e_i, a_{ij}, s_{ijkl}, h_k, t_l)$ apresentadas (YUGOSHI, 2018).

$$O = (e_i, a_{ij}, s_{ijkl}, h_k, t_l) \tag{2.1}$$

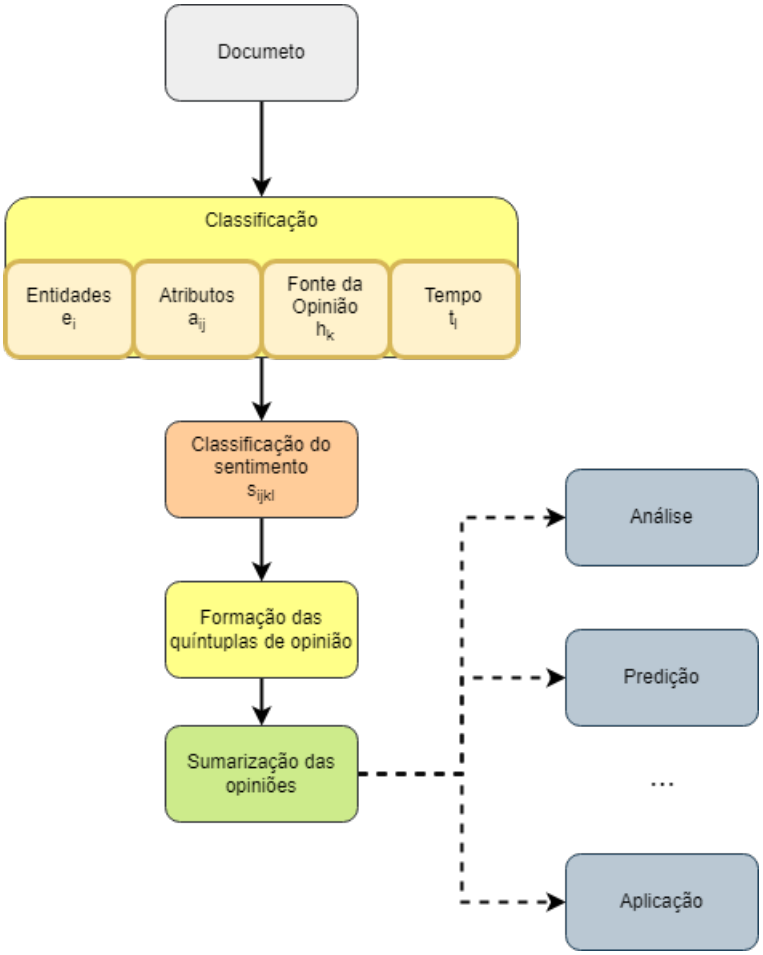
2.1.2 Etapas da Mineração de Opinião

Uma mineração de opinião estrutura-se através de tarefas individuais que compõe o objetivo de mapear as quíntuplas de opiniões de um excerto. Tendo uma amostra, em linguagem natural, em mãos, é necessário realizar a **identificação dos quatro elementos da quíntupla de opinião** inerentes ao texto, ou seja, determinar os nomes associados a entidade e seus atributos (a expressão de sentimento atribuída diretamente a uma entidade é tratado como um atributo geral - a_{i0}), o indivíduo fonte da opinião e o tempo em que essa opinião foi emitida. Em seguida é necessário **classificar a polarização do sentimento** associado ao alvo dessa opinião por alguma forma de abordagem definida pelo método (alguns modelos comuns são abordagens baseadas em dicionários, estatística ou por aprendizado de máquina). Desse modo, tem-se os cinco elementos constituintes da quíntupla, levando a **construção de todas as quíntuplas de opinião** para o documento.

Neste ponto, a extração das opiniões sobre um alvo está completa no que diz respeito ao mapeamento, entretanto não é possível, ainda, atribuir uma função a esses dados. Seja para expor a um usuário uma média de avaliações, para determinar o sentimento predominante de um documento ou aplicar algorítimos para prever tendências. Então é necessário que haja métricas e um tratamento dessa lista de quíntuplas, de modo a sumarizar e quantificar os parâmetros extraídos. Valendo-se de uma **sumarização** de informações quantitativas e qualitativas, a respeito de um sentimento ao longo do tempo, é possível, por exemplo, montar a curva de satisfação sobre um produto e tentar prever uma tendência futura (KARIN; TUMITAN, 2013).

A Figura 3 ilustra esse fluxo de etapas, onde parte-se de um documento com opiniões embebidas; Extrai-se as entidades, seus atributos, o emissor da opinião e o tempo; Classifica-se o sentimento atribuído a esses elementos; Monta-se as quíntuplas de opiniões encontradas e sumariza-se as informações.

Figura 3 – Fluxo das etapas envolvidas no processo de uma análise de sentimentos



Fonte: Elaborada pelo autor.

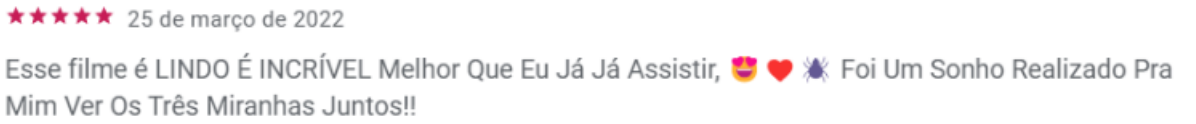
2.2 Análise de *Reviews* de aplicativos e serviços

A definição de um ambiente para ser tomado como domínio de um modelo estudado, deve apresentar as características em quantidade, disposição e especificidades desejadas, por conta disso, destacam-se bastante os modelos de análise textuais que se utilizam de *reviews* de aplicativos, filmes, e vendas para identificar conotações 'positivas' e 'negativas' (ARAUJO; GOLO; MARCACINI, 2022). Esses modelos são amplamente utilizados no dia-a-dia, por exemplo em sistemas de recomendações de *E-commerces*, e, como demonstrado na Figura 2, são expressivas as quantidades de modelos que utilizam-se dessas amostras para aprendizado, principalmente por associar uma crítica textual a uma **nota**. Entretanto, para se obter modelos preditivos com grande acurácia e assertividade, é necessário prever e contornar pontos de complicação nas análises que comentários escritos a esmo e sem devida revisão específica para análise trazem.

2.2.1 Opiniões indiretas e implícitas

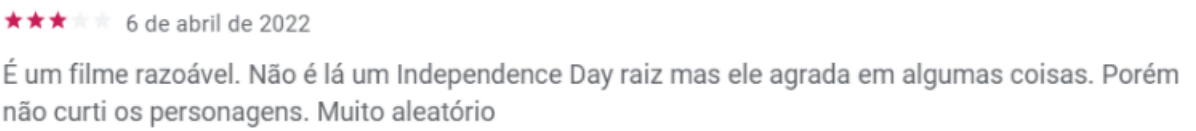
Como uma derivativa do processamento de linguagem natural, as amostras providas de opiniões sobre aplicativos, carregam alguns desafios inerentes à adaptação computacional para as subjetividades das comunicações humanas. Mais do que realizar apenas uma análise sintática, identificando a função de cada palavra no construto de uma frase, esse universo computacional necessita compreender os maneirismos, as figuras de linguagem, a ocultação de infamações e elementos baseado nos contextos que são empregados em uma linguagem natural (KARIN; TUMITAN, 2013). A forma mais fácil de análise é na expressão direta e explicita da mesma, ou seja, associa no discurso substantivos e adjetivos com significados literais que rementem a uma entidade e suas características grafadas no texto. Como no exemplo da Figura 4, pode-se observar que o indivíduo que expressa sua opinião utiliza-se de ideias explicitamente presente no texto para avaliar positivamente o filme por si, associando termos com conotações positivas, o que expressa seu contentamento.

Figura 4 – Avaliação de um filme de modo explícito. Extraído de: Google Play - Movies: 'Homem-Aranha: Sem Volta para Casa'



Entretanto, no panorama geral, uma opinião não se expressa dessa forma exata. Muitas vezes os atributos são qualificados comparativamente a produtos semelhantes, dando um caráter implícito a essas opiniões e dificultando a identificação da entidade. Na Figura 7, tem-se um exemplo de uso de uma comparação entre dois filmes para expressar uma opinião, onde, implicitamente, a qualidade entre o objeto avaliado é expressa por meio da qualidade do outro filme semelhante.

Figura 5 – Avaliação de um filme através de comparações. Extraído de: Google Play - Movies: '*Moonfall*'



2.2.2 Co-referência para uma mesma entidade

É considerado boa prática gramatical o uso de sinônimos e artifícios linguísticos a fim de evitar repetições em demasia de textos. Entretanto, para um modelo de aprendizado, que necessita identificar a entidade alvo de análise, distintos termos para se referir ao mesmo elemento causa certos empecilhos e dificuldade no aprendizado de diversos modelos. O uso de pronomes pessoais como "ele/ela"e suas variações, bem como o uso de substantivos equivalentes ao objeto alvo da

análise, cria uma duplicidade na interpretação, considerando entidades e/ou atributos diferentes para a mesma instância semântica. Dessa forma, é necessário que os modelos busquem tratar e referenciar adequadamente os sinônimos e pronomes de suas amostras textuais.

2.2.3 Uso de ironia e avaliação não condizente com o texto

Análise de sentimentos em linguagem natural, especialmente quando trata-se do meio artístico, depara-se com as barreiras das figuras de linguagens (duas delas já abordadas são as metáforas e comparações). Porém vale comentar as dificuldades associadas a uma outra figura de linguagem comum à literatura, marcada pela expressão de sentimento oposta ao significado direto das palavras. A ironia atrapalha o aprendizado de certos modelos de polarizações de opiniões não adaptados, por criar falsas relações entre as ideias (CARVALHO *et al.*, 2009).

No domínio de *reviews* de aplicativos e serviços, felizmente, não é comum o uso desse tipo de recurso linguístico, pois entende-se que uma avaliação clara tende a apresentar melhor compreensão. Entretanto um evento com desdobramento semelhante é recorrente em amostras de *reviews*, seja por descuido ou falta de afinidade com os recursos do site: avaliações com notas positivas e texto com expressão de sentimentos negativos (e o contrário diametralmente). Felizmente, esse desvio da normal não é comum, e pode ser compensado pelo grande volume de dados de amostra, que durante o treinamento dos modelos tende a suprimir esses equívocos.

Figura 6 – Avaliação de um filme com divergência entre sentimento expresso e avaliação numérica. Extraído de: Google Play - Movies: 'Morbius'

2.3 Análise de Sentimentos usando Word Embeddings

Entendendo o conceito e ideia de se analisar uma opinião através de um excerto, depara-se com a questão: "Como fazer com que uma máquina compreenda o conceito de palavras?". De forma geral, o processo de associar conceitos a uma palavra ou expressão para um algoritmo consiste em 'traduzir' seu significado através de valores numéricos, processo denominado **word embedding**. Uma das formas de assimilar significado entre palavras que apresenta melhor desempenho atualmente, de modo a manter-se em um escopo computacional viável para um universo idiomático com milhões de opções, é o uso de um **vetor numérico de características**. Para isso, o número de cada posição associa um valor dentro de um *range* que identifica o quão próximo essa palavra está dos extremos que representam características semânticas, similaridades e relação com o contexto (FONSECA, 2021) e (BROWNLEE, 2017).

O treinamento de um modelo que realize essa transcrição de significado na linguagem natural para um vetor numérico pode ser processado por modelos computacionais probabilísticos, com matrizes ocorrência ou por modelos de treinamento de máquina de linguagem natural, que é o caso dos algoritmos **Word2Vec**.

2.3.1 Word2Vec

Criado e patenteado em 2013 pela equipe de pesquisa de Tomas Mikolov, em associação a empresa *Google* (MIKOLOV *et al.*, 2013a) e (MIKOLOV *et al.*, 2013b), os algoritmos de aprendizado de máquina pela abordagem *Word2Vec* destacam-se entre as propostas de *word embedding* por sua habilidade de elaborar vetores numéricos com estimativas precisas para identificação da relação entre as palavras. Esse destaque permitiu seu uso em difundidas áreas que implementam recursos de inteligência computacional, como sistemas de recomendações, análises de similaridades, aplicações médicas para mapeamento de genomas e análise de sentimentos.

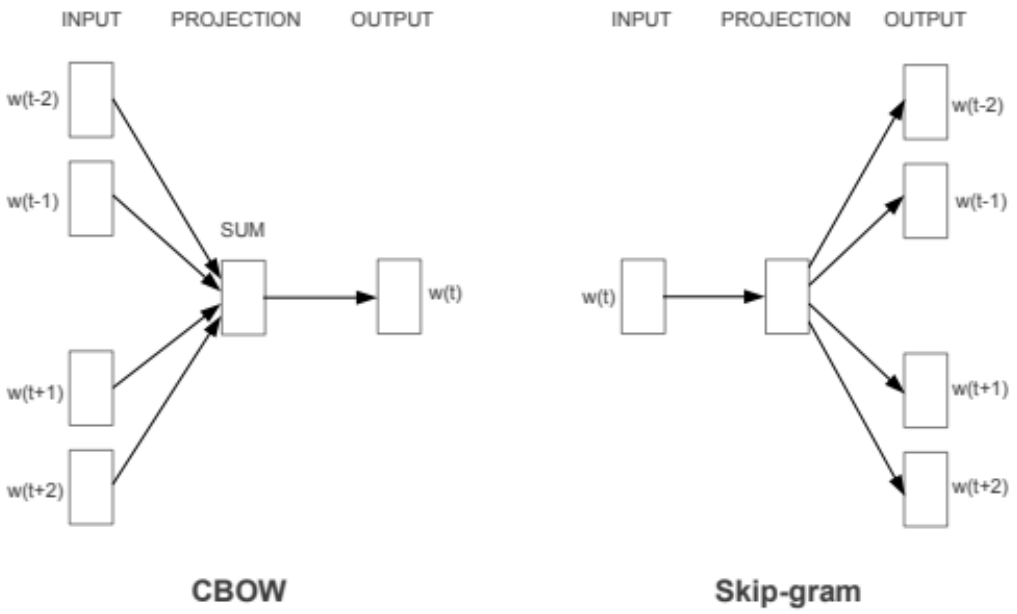
Alguns pontos interessantes para se considerar com os dados de amostra para treinamento de um modelo *Word2Vec* são o tamanho de suas janelas de contexto, a quantidade de palavras diversificadas que estarão disponíveis e o grau de dimensionalidade escolhido (o tamanho do vetor que descreverá cada palavra). Dessa forma, tentar atingir o intervalo ideal entre esses valores para criar fortes ligações dentro de um ambiente de processamento aceitável é a chave para o uso desse recurso.

Existem duas principais arquiteturas para os modelos *Word2Vec*, sendo elas (MIKOLOV *et al.*, 2013a):

- **Continuous Bag of Words (CBOW)**: Utiliza-se de um contexto e valores previamente associados às palavras desse para estimar onde uma nova palavra encaixa-se, ou seja, parte-se de um conjunto de amostras estruturado e define a melhor localização para uma nova palavra dentro desse conjunto. Como o ponto inicial é o conjunto de amostras, os modelos que utilizam essa estratégia necessitam de maiores quantidade de amostras para o treinamento e tem melhor precisão para palavras com grandes frequências de uso;

- **Continuos Skip-Grama**: Realiza-se um processo diametralmente oposto ao modelo anterior, onde utiliza-se a nova palavra para estimar as relações de contexto entre o conjunto de dados de treinamento, ou seja, parte-se de uma nova palavra para, a partir dessa, estruturar as demais localizações do conjunto de amostras. Como não parte de um contexto, funciona bem para conjuntos de treino menores e tem um padrão de acretividade mais homogênea do que o primeiro método, independente da frequência de uso de cada palavra.

Figura 7 – Esquema de precições realizadas pelos modelos *Word2Vec* para os tipos *CBOW* e *Skip-Gram*



Fonte: Mikolov *et al.* (2013a).

2.3.2 Modelo de aprendizado BERT

Bidirectional Encoder Representations from Transformers - BERT ou *Transformer com Encoder Bidirecional*, é um modelo de aprendizado de máquina que adapta os modelos de atenção - chamados de *Transformers* - para uma análise de aprendizado bidirecional, isso é, considera o contexto e associação das palavras em ambas direções de sequência, diferente de outros modelos *Word2Vec* que consideram o fluxo textual da esquerda para a direita normalmente. Desenvolvido por uma equipe de pesquisa dos laboratórios de inteligência artificial da *Google* DEVLIN *et al.*, o projeto publicado em caráter *open source* trouxe grandes avanços e contribuições em diversos outros projetos devido à facilidade com a qual realiza **transferência de aprendizado**, isso é, permite que se faça um modelo base **pré-treinado** (com parâmetros iniciais definidos através de processos não supervisionados), e modelos com **ajuste fino**, onde aplica-se treinamentos supervisionados para refinar os parâmetros iniciais dado um certo domínio e função específicos. Exemplos de sua capacidade são os resultados obtidos e apresentados em **SQuAD** - *Stanford Question Answering Dataset* - uma aplicação para respostas de uma base de perguntas, e na avaliação das informações na *benchmarch* **GLUE** - *General Language Understanding Evaluation*, dispostos em (DEVLIN *et al.*, 2018), onde observa-se resultados cerca de 7 pontos percentuais melhores que antigos modelos publicados.

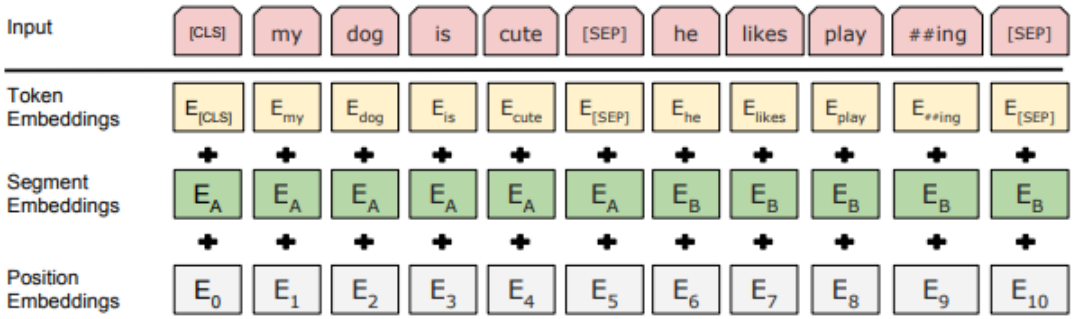
Para uma transferência de conhecimento, é necessário que o modelo esteja pré-treinado com uma sólida base de amostras e apresente valores consideráveis de resultados para a construção de seus parâmetros base. Originalmente, o modelo *BERT* implementado em determinada língua dá-se em um *state-of-the-art* modulado sobre um *dataset* que considera centenas de milha-

res de artigos do site *Wikipédia* - totalizando cerca de 2,5 bilhões de palavras. As etapas de um processamento inicial são desempenhadas conjuntamente, de modo a permitir ao modelo supor qual palavra se encaixa em dado contexto e estabelecer ligação entre sentido de duas sequências de palavras, em etapas denominadas **Masked LM - MLM** e **Next Sentence Prediction - NSP**.

A **MLM** consiste em determinar um percentual de palavras das amostras de treino e ocultá-las aleatoriamente, deixando a cargo do aprendizado estimar qual palavra cabe melhor, baseada nas palavras ao seu redor, ou seja, ao seu contexto. Em termos mais práticos, ele cria uma matriz de dimensões igual ao tamanho do vocabulário e estima a probabilidade de cada palavra estar na região ocultada, selecionando a que apresenta um melhor dado numérico.

Por sua vez, o **NSP** consiste em tentar prever se uma frase *B* é sequência direta de outra *A*. Logo, recebe tuplas de frases as quais, para uma certa porcentagem, a segunda frase é imediatamente sequência da primeira, e para outra porcentagem, não é. Os excertos textuais apresentam *tokens* que identificam o início de uma tupla e a separação entre frases - delimitando os segmentos de palavras, assim, o modelo processa todas as frases, cria estimativas simples para cada relação considerando a probabilidade e vai aprimorando o aprendizado ao calcular as chances de ser ou não a próxima frase.

Figura 8 – Modelo de pré-treinamento do *BERT*. A linha de *input* refere-se à informação textual de entrada (onde *[CLS]* e *[SEP]* são marcadores de separação de frases para **NSP** e *##ing* representa a máscara aplicada a uma palavra), essa informação é a soma das demais características de símbolo, frase e posição.



Fonte: Devlin *et al.* (2018).

Um modelo com esses treinamentos finalizado para um conjunto suficiente de amostras, é dito um modelo pré-treinado que consegue elaborar vetores numéricos que representam bem uma palavra baseado em seu contexto. A partir disso, pode-se aprimorar o desempenho com um *Fine-Tuning* para um certo domínio de aplicação. Esse refinamento, quando direcionado para o uso de predições de sentimento contidos em excertos textuais, pode levar o modelo ao uso para a análise de sentimos em uma linguagem natural.

2.3.3 Fine-Tuning e análise de sentimento para a língua portuguesa com o BERT

Em geral, para os suportes pré-treinados do modelo **BERT**, seu *state-of-the-art*, ou seja, o nível de conhecimento prévio adquirido pelo modelo se dá baseado em artigos expositivos retirados de páginas da *Wikipédia*, fazendo com que eles, do ponto de vista de uma abordagem de opiniões nos textos, não se deem de forma suficiente para compreender os indícios subjetivos dos discursos humanos em um contexto de opinião. Assim, para aprimorar o modelo pré-existente para a determinação de um sentimento, é necessário estimular o aprendizado de textos subjetivos para a língua ao qual se pretender e criar uma camada extra no fluxo de processamento do modelo que não apenas vai identificar relações de palavras com o contexto, mas também vai conseguir associar essas identificações com o sentimento que melhor se encaixa.

O intuito ao final do treinamento é alcançar um modelo preparado para textos dissertativos e argumentativos em linguagem natural - sendo essa linguagem o português brasileiro. Vale ressaltar que alguns projetos nacionais já utilizaram a base do modelo *BERT* para o domínio de suas aplicações, em especial, pode-se citar o *BERTimbau* (SOUZA; NOGUEIRA; LOTUFO, 2020), disponibilizado abertamente para uso, que consiste em um modelo capaz de traçar similaridades, e reconhecimento de palavras nos textos, similar ao modelo *BERT* original, porém destinado à língua portuguesa.

Para o *fine-tuning* neste projeto, utiliza-se o modelo *multilingual* do *BERT* desenvolvido inicialmente por (DEVLIN *et al.*, 2018), modelo esse que dá suporte amplo para diferentes línguas diretamente e vem obtendo ótimos resultados em projetos recentes (PIRES; SCHLINGER; GARRETTE, 2019). Neste caso, utilizaremos como foco textos em ambientes digitais provenientes de comentários e avaliações de aplicativos, que utilizem um certo modelo de contexto passado para melhorar o desempenho. Essa ultima parte é importante, pois, dependendo da situação de uso, uma palavra pode apresentar maior ou menor similaridades com o sentimento empregado, sendo um **contexto específico** o mecanismo que pode promover a diferenciação. Supondo que tem-se o seguinte texto: "Ele me ajudou a #### melhor", o melhor complemento a essa frase pode ser dada por diferentes palavras, sendo assim, a inferência de um contexto pode ajudar. Caso fosse um comentário em um contexto sobre alimentação ou saúde, provavelmente há maior proximidades estabelecida com palavras como "comer", "hidratar", "dormir"; Já em um contexto onde se fala sobre trabalho, a palavra "comer"estaria mais distante metricamente do que termos como "trabalhar"ou "focar".

Desa forma, tenta-se otimizar os resultados do *Fine-Tuning*, utilizando a designação de categorias dos excertos textuais apresentados para gerar um contexto e nichar os resultados para um maior grau de acerto ao final dentro de cada uma dessas áreas. Essa aplicação de um contexto pode ser feita por diferentes maneiras, alguns projetos como (XU *et al.*, 2019) realizam uma pré-etapa de preparação onde extraem-se aspectos característicos a dados contextos e utilizam-nos como esses demarcadores que conduzem o aprendizado. Entretanto, para o desenvolvimento

deste, busca-se a utilização de um aspecto inerente às amostras textuais utilizadas, reduzindo assim barreiras quanto ao tempo de consumo no processamento em ambientes computacionais. Esse aspecto baseia-se no uso da categoria do aplicativo ao qual o texto portador de opinião foi selecionado.

Capítulo 3

DESENVOLVIMENTO

3.1 Descrição do Problema

O projeto foca em realizar o treinamento de um ajuste fino (*Fine-Tuning*) para um modelo pré-treinado de aprendizado profundo baseado em *Word Embedding*, para análise de sentimento. Nesse caso, foi selecionado o modelo **BERT**, apresentado nas seções 2.3.2 e 2.3.3, para realizar um aprimoramento com foco na língua portuguesa, utilizando avaliações de aplicativos com o intuito de classificar os textos em opiniões **positivas** ou **negativas**. Para isso, faz-se necessário a composição de um *dataset* suficientemente grande e variado, em quesitos de tipos de amostras e de público, para que seu treinamento seja eficiente e consiga abranger uma ampla gama de características e particularidades de diferentes discursos na língua portuguesa, o que possibilita um amplo leque de possibilidades futuras.

Para a elaboração do *dataset*, é preciso definir as características que serão utilizadas como contexto dos modelos, visando extrair, possivelmente, um melhor resultado das predições. Por fim, é necessário treinar o modelo para um certo número de épocas e validar os resultados obtidos, escolhendo o que melhor se apresenta para os cenários de teste.

Em síntese, determina-se um fluxo tradicional de etapas a serem seguidas durante o desenvolvimento do projeto, como exemplificado em (BISWAS *et al.*, 2021), de modo a seguir a seguinte sequência de de atividades:

- Definição dos tipos de dados textuais e seu contexto a serem utilizados como amostra → Previamente definido o uso de avaliações de aplicativos da *Google Play* com categoria como contexto;
- Definição do modelo pré-treinado que será base do ajuste fino → Previamente definido o uso do modelo *BERT multilingual*;
- Extração dos dados de amostra para treino e teste por categoria de aplicativos;
- Filtragem dos dados extraídos para um nivelamento dos grupos mais sensíveis;
- Treinamento do modelo para um *Fine-Tuning* em certas quantidades de épocas suficientes;
- Uso dos modelos pelas amostras de teste, definindo a qualidade do desempenho;
- Publicação do modelo obtido.

3.2 Atividades Realizadas

Nessa seção são apresentadas as atividades desenvolvidas para concretizar as etapas dispostas na seção 3.1 para extração e filtragem de dados, bem como o *Fine-Tuning* do modelo. As métricas obtidas pelos testes realizados e a publicação estão dispostas na seção subsequente, seção 3.3.

3.2.1 Extração das amostras para treinamento e teste

Para construir um conjunto de dados de amostras optou-se por selecionar *reviews* de aplicativos da loja de aplicativos *Google Play Store*, adquirindo amostras de todas as 33 categorias de *apps* disponíveis - excluindo as categorias de jogos - selecionando de 5 (cinco) a 8 (oito) dos melhores aplicativos para aquisição. A seleção dos melhores *apps* se deu considerando o número de avaliações indicado pelo próprio site, mas ponderando a popularidade do aplicativo no Brasil (uma vez que esse número de avaliações indicado reflete o número global de avaliações e considera notas sem texto associado). A Tabela 1 reflete as categorias utilizadas e a quantidade de *apps* selecionados por categoria.

Após selecionados os aplicativos, utiliza-se uma ferramenta de extração em massa para adquirir as informações referentes a todas as avaliações de cada um desses *apps*, denominada **Crawler**. Um *crawler* é um *script* automatizado que acessa uma determinada URL e extrai algum tipo de informação pré-definida. Os motores de busca do *Google*, por exemplo, utilizam de *crawlers* para avaliar conteúdos de páginas *web* listadas durante uma pesquisa. Nesse caso, o *crawler* utilizado é uma ferramenta implementada para a linguagem *Python* que recebe uma URL de busca e outros filtros opcionais - como nota da avaliação, determinado período de tempo, linguagem e país de origem, gerando metadados com as informações de *reviews* para o aplicativo da URL solicitada. Nessa situação, utilizou-se apenas o filtro de região, localizando os comentários brasileiros, e em língua portuguesa. Os metadados obtidos são convertidos em formato de *DataFrame* da biblioteca **Pandas**, que trata as informações como uma tabela com colunas de informações que compreendem desde o nome do aplicativo e do usuário, à categoria, data de avaliação, texto de opinião emitida, nota e até imagem de usuário e códigos de identificação do site.

Como a maior parte das informações não são relevantes ao treinamento, as tabelas foram realocadas selecionando apenas as colunas de interesse: **aplicativo**, **categoria**, **avaliação textual** e **nota**. Em seguida os dados foram exportados para um arquivo estruturado em formato *CSV*, que possibilita a leitura do mesmo em interfaces mais amigáveis e com ferramentas de filtragem para tabelas, como o *Microsoft Excel*, permitindo maior facilidade na análise das quantidades e qualidade das amostras. Os valores de amostras obtidos para cada categoria encontram-se na Tabela 1.

Ao final da extração de dados, obtém-se um *dataset* bruto com quantidades de *samples*

Tabela 1 – Total de aplicativos, em **unidades** (10^0), e avaliações, em **milhões** (10^6), selecionados e extraídos por categoria. Considerando número de *reviews* gerais e popularidades. Organizados por ordem alfabética.

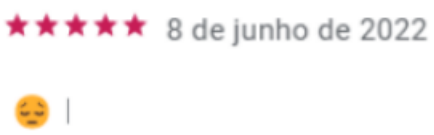
Categoria	Total de Aplicativos	Total de avaliações
Arte e Design	5	0,219
Beleza	5	0,006
Biblioteca e Demos	5	0,012
Casa e Decoração	5	0,044
Clima	5	0,117
Comer e Beber	6	0,636
Compras	7	0,459
Comunicação	8	1,118
Corporativo	6	0,149
Criar os Filhos	5	0,010
Educação	6	0,267
Encontros	5	0,134
Entretenimento	8	0,648
Esportes	5	0,007
Estilo de Vida	6	0,130
Eventos	5	0,002
Ferramentas	8	0,398
Finanças	8	0,897
Fotografia	7	0,422
Humor	6	0,007
Livros e Referências	6	0,170
Mapas e Navegação	6	0,272
Medicina	6	0,176
Mostradores de Relógio	5	0,002
Música e Áudio	8	0,762
Notícias e Revistas	5	0,047
Personalização	6	0,2425
Produtividade	8	1,077
Reproduzir e Editar Videos	8	0,609
Saúde e Fitness	8	0,156
Social	8	0,284
Turismo e Local	8	0,458
Veículos	6	0,100
TOTAL	**208**	**1,0E+7**

na ordem de 10^7, ou seja, dezena de milhões, o que julga-se suficiente para que, mesmo após as filtragens de dados - próxima etapa do projeto explicada na seção 3.2.2 - a base de amostras esteja com um tamanho funcional.

3.2.2 Filtragem das amostras

Para evitar um treinamento com comportamento vicioso e tendencioso, é preciso tomar alguns cuidados com o *dataset* utilizado, fazendo-se necessário algumas filtragens e preparações a fim de apresentar um ambiente balanceado para os sentimentos que deseja-se prever. Como é esperado que esse modelo determine os excertos em sentimentos **positivo** e **negativo**, deve-se converter a anotação de avaliação típica do site, que considera uma nota de 1 (um) a 5 (cinco), no formato esperado. As avaliações são consideradas do tipo *positivo* quando recebem nota 4 (quatro) ou 5 (cinco) e do tipo *negativo* quando recebem nota 1 (um) ou 2 (dois). As avaliações com nota igual a 3 (três), que representaria um sentimento neutro, são desconsideradas, uma vez que essa nota mediana é muito dependente da interpretação de usuários e poderia comprometer o desempenho do modelo. Além disso, também são removidos das tabelas de dados os comentários vazios ou que dispostos apenas por caracteres não convencionais, como um sequência de *emojis*.

Figura 9 – Exemplo de amostra a ser desconsiderado - texto com, apenas, sequência de caracteres não convencionais. Extraídas de: Google Play - Movies: '*Peso do Sucesso*'



Finalizada essa filtragem inicial, o tratamento de dados atenta-se a balancear a quantidade de amostras entre os dois tipos de classificação de sentimento, igualando a quantidade de avaliações **por aplicativo** de acordo com o **dado mais sensível**, isso é, tomando como teto no número de amostras com o menor entre as quantidades de dados *positivas* e *negativas*. Matematicamente, tem-se um total de amostras por aplicativo i escolhido igual as Equações 3.1 e 3.2.

$$N_i = N_{positive_i} + N_{negative_i} = 2 * Min(n_{positive_i}, n_{negative_i}) \tag{3.1}$$

$$N_{positive_i} = N_{negative_i} \tag{3.2}$$

Assim, ao final, tem-se um *dataset* com a **mesma quantidade de amostras para cada sentimento** que pretende-se considerar por categoria de aplicativo. Essa tratativa é de grande importância pois quando um número de amostras que refletem à um tipo de sentimento se sobrepõe em demasia aos demais, pode-se criar um comportamento tendencioso para a classificação desse devido à maior exposição durante a fase de processamento do modelo.

Por fim, ainda balanceando os valores de amostras por categorias, é realizado o *split* das amostras em tabela de treino - utilizados na próxima etapa para o treinamento dos modelos - e teste - utilizados posteriormente para a avaliação da qualidade dos modelos. Para essa divisão utiliza-se a proporção de 70% (setenta por cento) de amostras para treino e 30% (trinta por cento) de amostras para teste. A divisão é feita por categoria e por sentimento para garantir que,

principalmente para os grupos com baixo número de amostras, haja exemplares de cada uma em ambos as tabelas, e, como pretende-se considerar as categorias como contexto nas predições, seria um erro crasso não conter certas quantidades em ambas as tabelas. Dessa forma, tem-se dados para treino e teste representadas, respectivamente, pelas Equações 3.3 e 3.4, sendo *i* cada categoria de aplicativos.

$$DF_{train} = \sum_i 0.7(N_{positive_i} + N_{negative_i}) \tag{3.3}$$

$$DF_{test} = \sum_i 0.3(N_{positive_i} + N_{negative_i}) \tag{3.4}$$

Após a filtragem de dados descrita visando manter o equilíbrio do aprendizado às quantidades, a quantidade obtida de amostras totais, de treino e de teste encontram-se na Tabela 2, ressaltando que, para todas as células, o valor apresentado é a soma de uma igual quantidade de amostras representativas de *Positivo* e *Negativo*. Apesar de uma numerosa perda após o balanceamento dos dados, em especial devido ao nivelamento pelo atributo de sentimento mais sensível, os *datasets* persistem em uma quantidade de amostras na ordem de 10^6, ou seja, algumas unidades de milhões de dados, suficiente para suprir a necessidade do aprendizado profundo supervisionado do *Fine-Tuning*. Essa tabela então é exportada para um formato de objeto serializado - do tipo **pickle** - pronta para ser importada e utilizada nos processos subsequentes do projeto, concluindo-se assim as etapas de extração e tratamento de amostras.

3.2.3 Treinamento dos modelos

O processo de treinamento de recurso computacional para análise de sentimentos é custoso e exige recursos computacionais dedicados (placas gráficas) para que seja realizado em tempo aceitável. Essa etapa só foi possível por conta de uma máquina virtual em ambiente *Microsoft Azure*, com 54*Gb* de memória RAM, 380*Gb* de armazenamento, processamento com 6 núcleos e uma placa gráfica *NVidea Tesla K80*, fornecida pelo Laboratório de Inteligência Computacional da Universidade de São Paulo (LABIC/USP). Dessa forma, foi possível computar todo o processo em cerca de *19 horas* por época treinada.

O treinamento foi realizado, também, em linguagem *Python*, utilizando-se de bibliotecas para aprendizado de máquina específica, **KTrain** (MAIYA, 2020), e de métricas de verificação. Esse repositório importado oferece recursos para treinamento, ajuste, análise do *fit* de cada época, seleção de parâmetros de processamento (como tamanho das *batchs* e das amostras), seleção entre linguagens de modelos pre-processados, entre outros recursos para diversos tipos de métodos de aprendizado suportado, entre eles, o *BERT*. O modelo *BERT* disponível na *KTrain* já apresenta um pré-processamento inicial multilinguagem, citado em 2.3.3, de modo que está pronto para o treinamento de um ajuste fino de acordo com as finalidades desejadas.

As amostras serializadas, preparadas anteriormente, são carregadas no ambiente virtual descrito e separadas em dados de entrada, que consideram a informação textual portadora do

Tabela 2 – Quantidade de amostras totais, treino e teste, em **milhares** (10^3) **de amostra**, por categoria de aplicativo. Organizados por ordem alfabética.

Categoria	Total	Treino	Teste
Arte e Design	43	30	13
Beleza	2	1	1
Biblioteca e Demos	5	3	2
Casa e Decoração	18	13	5
Clima	16	11	5
Comer e Beber	292	204	88
Compras	183	128	55
Comunicação	523	366	157
Corporativo	30	21	9
Criar os Filhos	3	2	1
Educação	92	64	28
Encontros	77	54	23
Entretenimento	392	275	118
Esportes	6	4	2
Estilo de Vida	28	20	8
Eventos	2	1	1
Ferramentas	122	85	37
Finanças	584	409	175
Fotografia	87	61	26
Humor	3	2	1
Livros e Referências	48	34	14
Mapas e Navegação	175	122	52
Medicina	78	54	23
Mostradores de Relógio	1,4	1	0,4
Música e Áudio	268	188	80
Notícias e Revistas	13	9	4
Personalização	24	17	7
Produtividade	157	110	47
Reproduzir e Editar Videos	183	128	55
Saúde e Fitness	9	6	3
Social	181	126	54
Turismo e Local	118	83	35
Veículos	25	18	7
TOTAL	**3787**	**2651**	**1136**

sentimento, o contexto dado pela categoria de aplicativo do qual foi extraído essa avaliação e dados de saída com a anotação exata do sentimento (em *negative* ou *positive*), para métricas do aprendizado como perda e acurácia. Em seguida, os dados textuais são transformados para atender os padrões de *BERT Based Multilingual Cased*, com tamanho máximo de 64 palavras por *review*.

Por fim, inicia-se o treinamento para aprimorar o *learner* avaliando seu desempenho e salvando o modelo de predição por época (considera-se um $fit = 5E-5$ e $batch\_size = 32$). A execução deu-se completa por 3 (três) épocas, na qual, pelas métricas de validação do *learner*, observou-se a convergência dos valores de execução, encerrando, com isso, a etapa de treinamento e selecionando o referente à terceira e última época como o modelo que utilizado para as validações.

3.3 Resultados

Para análise dos resultados, considera-se que a saída de um modelo de classificação de sentimentos que tem como base a polarização entre *positive* e *negative* apresenta quatro possibilidades quanto a sua saída: apontar um 'negativo' corretamente - *True Negative*, apontar um 'negativo' erroneamente - *False Negative*, apontar 'positivo' corretamente - *True Positive* - ou apontar 'positivo' erroneamente - *False Positive*.

Os valores referentes a essas 4 classificações obtidas definem a chamada **Matriz de Confusão**, que ilustra os resultados obtidos quanto a assertividade do modelo. Por meio da matriz é possível calcular outros parâmetros que refletem sobre a qualidade do desempenho para os testes. Os testes de validação do modelo realizados apresentam quantidades dadas pela seguinte matriz de confusão:

$$\textit{Matriz de Confusão} = \begin{bmatrix} \text{True Positive} & \text{False Positive} \\ \text{False Negative} & \text{True Negative} \end{bmatrix} = \begin{bmatrix} 507641 & 44165 \\ 44143 & 507895 \end{bmatrix}$$

Dessa forma, a soma das colunas da matriz apresentam a quantidade total de amostras contendo cada sentimento, sendo a coluna a esquerda o total de amostras *positive* e a da direita *negative*, o que é um valor importante para o cálculo do *sensitivity/recall* ou *Specificity* do modelo. Analogamente, a soma das linhas da matriz apresenta outro fator importante, que é utilizado para calcular a precisão, *precision*, do modelo, esse fator fornece, para a linha superior, a quantidade de amostras *positive* e, para a linha inferior, a quantidade de amostras *negative* que o modelo apontou. Por fim, a soma dada na diagonal principal apresenta o valor que reflete a quantidade total de amostras corretamente classificadas pelo modelo.

3.3.1 Recall

A medida de *Recall*, também sendo denominada como a sensibilidade do modelo - *sensitivity*, é a relação entre o valor das predições positivas encontradas em relação ao total de amostras positivas de entradas. Dessa forma, sendo o total de amostras positivas no *dataset* de teste iguais a *True Positive* + *False Negative*, o valor da sensibilidade do modelo é dada pela Equação 3.5.

$$R = Sensitivity = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} = \frac{507641}{551784} \approx 0.92 \tag{3.5}$$

Esse parâmetro demonstra o quanto de uma quantidade de texto com anotação de sentimento positivo é corretamente classificados, ou seja, cerca de 92% das amostras *positive* foram classificadas como esperado.

3.3.2 Specificity

A medida de especificidade do modelo, *Specificity*, é a relação entre o valor das predições negativas encontradas em relação ao total de amostras negativas de entradas. Dessa forma, sendo o total de amostras negativas no *dataset* de teste iguais a *True Negative* + *False Positive*, o valor da especificidade do modelo é dada pela Equação 3.6.

$$Specificity = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}} = \frac{507895}{552060} \approx 0.92 \tag{3.6}$$

Esse parâmetro demonstra o quanto de uma quantidade de texto com anotação de sentimento negativa é corretamente classificados, ou seja, cerca de 92% das amostras *negative* foram classificadas como esperado.

3.3.3 Precision

A medida de precisão do modelo, *precision*, é a relação entre o valor das predições de um dado sentimento classificadas corretamente em relação ao total apontado pelo modelo com esse sentimento, sendo essa classificação correta ou não. Dessa forma, o total de amostras de sentimentos positivos apontados iguais a *True Positive* + *False Positive*, o valor da precisão para o sentimento positivo é:

$$P = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} = \frac{507641}{551806} \approx 0.92 \tag{3.7}$$

Esse parâmetro demonstra a precisão do sistema ao classificar certo tipo de sentimento, no caso, o sentimento positivo. Para o modelo obtido, tem-se precisão de, aproximadamente, 92% .

3.3.4 Accuracy

A medida de acurácia do modelo, *accuracy*, é medida que aponta a quantidade de amostras corretamente classificadas em relação ao total de amostras do *dataset* de teste utilizado. Sendo o valor das amostras corretamente classificadas dada pela diagonal principal da matriz de confusão, *True Positive* + *True Negative*, o valor da acurácia do modelo é dada por:

$$Accuracy = \frac{\text{True Positive + True Negative}}{\text{Total de amostras de teste}} = \frac{1015536}{1103844} \approx 0.92 \tag{3.8}$$

A acurácia do modelo dá a taxa de acerto geral, então, para o treinado nesse experimentos, tem-se 92% de acerto do *dataset* de amostras de teste utilizado.

3.3.5 F1 - Score

Em alguns casos, a precisão e a sensibilidade podem apresentar valores diametralmente extremos, por exemplo, o modelo apresentar valores de precisão extremamente alto (praticamente não apresentar falsos positivos), mas não refletir bem o conjunto de dados completo (muitas amostras positivas classificadas como falso negativo). Dessa forma, considerar apenas as métricas apresentadas anteriormente pode não refletir o balanceamento dos dados de teste, levando a necessidade de uma métrica que considere ambas *precision* e *recall* para calcular seu valor dado. Para isso, o *F1-Score* considera uma média harmônica entre esses outros dois parâmetros e, com isso, adiciona um comportamento de peso aos valores apresentados por ambos, refletindo melhor o balanço dos resultados apresentados.

$$F1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} = 0.92 \tag{3.9}$$

Dessa forma, como o modelo treinado durante o experimento apresentou valores semelhantes para ambas as medidas, e como a quantidade de amostras de teste estavam balanceadas em quantidade, a relação harmônica entre elas apresenta um valor também semelhante, sendo o *F1-Score* cerca de 92%.

3.3.6 Comparativo com um Baseline

Além de apenas calcular as valores dos parâmetros apresentados acima, para uma avaliação quanto à funcionalidade do modelo no cenário atual, é preciso uma comparação com outro modelo que atue sobre um mesmo domínio, de modo a analisar se o projeto apresenta ganho para a comunidade científica e se será interessante seu uso futuro. Para isso, escolhe-se o modelo **LeIA** (ALMEIDA, 2018), uma adaptação da ferramenta de análise de sentimentos **VADER** (GILBERT, 2014) para atuar com textos em português. Como o domínio de alocação do *LeIA* é focado para o ambiente digital, treinado com textos de mídias sociais, compreende o mesmo aspecto e apresenta-se como um bom *baseline* de comparação de desempenho para esse projeto.

VADER é configurado para dispor resultados em classificação de sentimentos positivos, negativos ou neutros, de acordo com um valor atribuído a ele que varia de -1 a $+1$. Para refletir o comparativo em apenas duas classificações, as definições de sentimentos consideraram valores no intervalo de -1 a 0 como negativo e 0 a $+1$ como positivo, nenhuma amostra apresentou os parâmetros completamente zerados - portanto não houve caso de indecisão.

Quanto aos resultados, os valores medidos são apresentados separados por categoria, uma vez que o intuito é utilizar as categorias como demarcadores de contexto. O número utilizado foi de até 20 mil amostras por categoria, limitado ao número máximo de amostras daquela categoria quando inferior, conforme valores da Tabela 2. As mesmas configurações de quantidade, amostras selecionadas e ordem de execução foram utilizadas para os dois modelos, e os valores para *precision - P*, *Recall - R* e *F1-Score* apresentam-se na Tabela 3. Uma vez que, para algumas categorias a quantidade de amostras de teste positivas e negativas não se dispões exatamente iguais, opta-se por utilizar os valores de média de classificações considerando o sistema de pesos para cada um dos dois sentimentos representados.

Como pode-se notar, para as classificações nos parâmetros de análise explicados, em todas as categorias, o modelo do experimento apresentou valores consideravelmente superiores, ou seja, uma maior capacidade de classificar os sentimentos corretos. Em termos numéricos, o modelo de *fine-tuning* derivado do *BERT* apresentou um desempenho superior cerca de, ao menos, 10% dos resultados do modelo de *baseline*. Além disso, para a maioria das categorias, o desempenho em todas as métricas avaliativas deu-se superior à taxa de 90% de acerto, um valor consideravelmente expressivo.

3.3.7 Curva ROC

Outra forma de avaliar e comparar o desempenho dos modelos, é considerando a curva *Receiver Operating Characteristic - ROC* e sua *Area Under the Curve - AUC* associada.

A **curva ROC** é uma relação entre as taxas de *True Positive* e *False Positive* que representa a capacidade de um modelo de prever corretamente os sentimentos de um texto de entrada. Seu eixo x representa a **taxa de positivos falsos**, ou seja, é o comportamento conforme aumenta-se o número de valores erroneamente dado como positivo pelo modelo que também pode ser representado pelo oposto à taxa de negativos verdadeiros, *Specificity*, como mostra a Equação 3.10. Quanto maior a taxa de positivos falsos, menor o número de predições corretas do valor *negative*, nos extremos, quando a taxa se da como 0, não há positivos falsos e, consequentemente, todos os valores negativos foram classificados corretamente. Já quando a taxa é 1, o numerador e denominador são iguais, ou seja, o valor de negativos verdadeiros é zero e, portanto, todas as classificações de valores negativos são incorretas.

$$\text{False Positive Rate} = 1 - \text{Specificity} = \frac{\text{False Positive}}{\text{True Negative} + \text{False Positive}} \tag{3.10}$$

Tabela 3 – Comparativo entre o modelo obtido pelo experimento e um modelo *baseline* VADER PT-BR

	Proposta			*LeIA*		
Categoria	*P*	*R*	F_1	*P*	*R*	F_1
Geral	0.92	0.92	0.92	0.82	0.67	0.73
Arte e Design	0.92	0.92	0.92	0.83	0.66	0.73
Beleza	0.95	0.95	0.95	0.87	0.68	0.76
Biblioteca	0.88	0.88	0.88	0.83	0.66	0.73
Casa e Decoração	0.95	0.95	0.95	0.87	0.71	0.78
Clima	0.93	0.93	0.93	0.84	0.64	0.71
Comer e Beber	0.96	0.96	0.96	0.85	0.71	0.78
Compras	0.96	0.96	0.96	0.86	0.74	0.79
Comunicação	0.88	0.88	0.88	0.81	0.64	0.71
Corporativo	0.92	0.92	0.92	0.86	0.66	0.74
Criar os Filhos	0.91	0.91	0.91	0.83	0.68	0.75
Educação	0.92	0.92	0.92	0.82	0.62	0.69
Encontro	0.92	0.92	0.92	0.82	0.63	0.71
Entretenimento	0.92	0.92	0.92	0.82	0.63	0.71
Esportes	0.94	0.94	0.94	0.85	0.63	0.72
Estilo de Vida	0.93	0.93	0.93	0.83	0.66	0.73
Evento	0.95	0.94	0.94	0.90	0.76	0.82
Ferramenta	0.90	0.90	0.90	0.84	0.64	0.72
Finanças	0.95	0.95	0.95	0.83	0.75	0.78
Fotografia	0.92	0.92	0.92	0.81	0.66	0.72
Humor	0.85	0.85	0.85	0.74	0.50	0.60
Livros e Referências	0.90	0.90	0.90	0.81	0.58	0.67
Mapas e Navegação	0.92	0.92	0.92	0.79	0.68	0.73
Medicina	0.97	0.97	0.97	0.90	0.76	0.83
Mostradores de Relógio	0.91	0.91	0.91	0.90	0.58	0.70
Música e Áudio	0.92	0.92	0.92	0.81	0.60	0.68
Noticias e Revista	0.90	0.90	0.90	0.85	0.60	0.68
Personalização	0.92	0.92	0.92	0.87	0.65	0.67
Produtividade	0.92	0.92	0.92	0.87	0.65	0.73
Reproduzir e Editar Vídeos	0.88	0.88	0.88	0.77	0.64	0.70
Saúde	0.86	0.85	0.85	0.79	0.51	0.59
Social	0.90	0.90	0.90	0.79	0.66	0.72
Turismo e Local	0.91	0.92	0.92	0.84	0.61	0.71
Veículos	0.94	0.94	0.94	0.86	0.69	0.76

Por sua vez, o eixo *y* representa a **taxa de positivos verdadeiros**, ou seja, é o comportamento conforme aumenta-se a sensibilidade do modelo, como mostra a Equação 3.11. Quanto maior a taxa de positivos verdadeiros, maior o número de predições corretas do valor *postive*, nos extremos, quando a taxa se da como 0, não há positivos verdadeiros e, consequentemente, todos os valores positivos foram classificados erroneamente. Já quando a taxa é 1, o valor de positivos falsos é zero e, portanto, todas as classificações de valores negativos estão corretaS.

$$\text{True Positive Rate} = \text{Sensitivity} = \frac{\text{True Positive}}{\text{True Negative} + \text{False Positive}} \tag{3.11}$$

Idealmente, busca-se um modelo que acerte todos os valores tanto para classificação negativa quanto para positiva. Assim, busca-se uma curva ROC que passe pela coordenada $(x,y) = (0,1)$, mantendo-se constante em $x = 0$ para toda a excursão de y e mantendo-se constante em $y = 1$ para toda a excursão de x - isso representa um modelo com 100% de *Sensitivity* e *Specificity*, e quanto mais próximo da coordenada buscada, melhor o desempenho do modelo (ZWEIG; CAMPBELL, 1993).

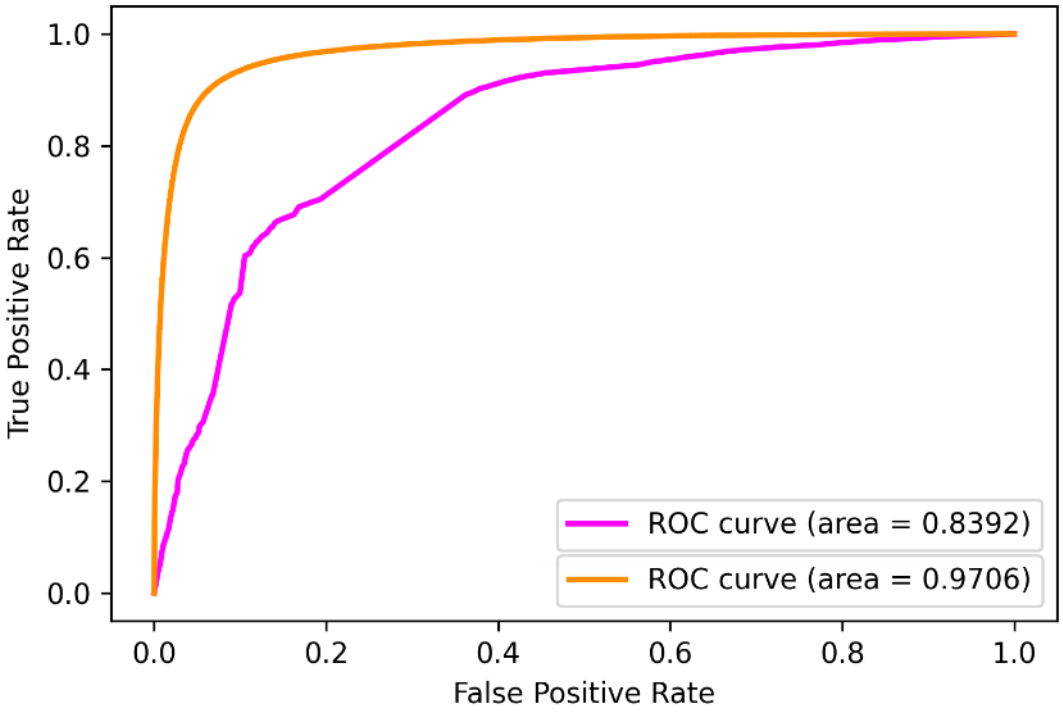
A área delimitada abaixo da curva ROC, que é capaz de traduzir numericamente a eficácia do modelo demonstrada pela curva, chama-se **AUC**. O valor calculado na área AUC busca-se apresentar como 1, sendo a expressão numérica onde a condição de um modelo ideal pela curva ROC é atingida. Quanto maior o valor da área, maior a possibilidade de acerto no sentimento apresentado e maior a capacidade de distinguir e classificar os excertos textuais. Podemos tratar os significados dos valores de AUC de acordo com os intervalos a seguir:

- $AUC = 1$: Modelo de predição ideal, todas as classificações estão corretas tanto para sentimentos positivos quanto para sentimentos negativos;
- $0.5 < AUC < 1$: Modelo de predição com altas chances de acerto, a maioria das classificações estão corretas tanto para sentimentos positivos quanto para sentimentos negativos;
- $AUC \leq 0.5$: A partir de uma área de 0.5, tem-se modelos onde a quantidade de classificações corretas é, no máximo, igual a quantidade de erros. Em termos práticos, isso representa que o modelo não consegue classificar os excertos e está distribuindo aleatoriamente classificações ou utilizando apenas um sentimento para classificá-los;

Para os casos gerais de teste com o modelo proposto e com o modelo de *baseline* - *LeIA* foram geradas as curvas ROC e calculada sua área abaixo para comparação de resultados, disposta na Figura 10.

O comparativo entre os dois modelos sobreleva a conclusão obtida com as classificações dos parâmetros anteriormente. O modelo desenvolvido no projeto apresenta uma curva bem constante nos valores desejados, aproximando consideravelmente da situação ideal, desempenho evidenciado na comparação com a curva gerada para o modelo de comparação. Os valores

Figura 10 – Curva ROC e AUC das amostras de teste consideradas para o modelo desenvolvido no projeto, em laranja, e para o modelo de *baseline* - *LeIA*, em rosa



Fonte: Dados da pesquisa.

calculados para AUC confirmam essa característica, onde, além de apresentar um valor de área cerca de 16% maior em relação ao modelo de comparação, o valor obtido encontra-se nas vizinhanças do almejado $AUC = 1$. Dessa forma, por meio de diferentes métricas de validação, pode-se afirmar a grande capacidade de classificar sentimentos do fruto deste projeto.

3.3.8 Importação e uso do modelo

O modelo obtido foi disponibilizado na comunidade *Hugging Face*, que tem como objetivo a divulgação e implementação prática de repositórios *Open Source* de modelos *Machine learning* pré-treinados e prontos para uso. O repositório contendo o modelo final, após 3 épocas de ajuste, encontra-se em *BERPT*.

Para uso do modelo, em ambiente *Python*, é necessário, apenas, a instalação da biblioteca **KTrain** e do repositório contendo o modelo. Esses requisitos podem ser realizados por meio da execução em terminal de comando do ambiente computacional como dispostos no Código-Fonte 1.

Código-fonte 1: Instalação das dependências para uso do modelo do projeto.

```
1 !git lfs install
2 !pip install ktrain
3 !git clone https://huggingface.co/renanperes/BERPT
```

Com as dependências devidamente instaladas, a averiguação de sentimento atribuído a um excerto pode ser realizada de maneira análoga ao exemplo expresso no Código-Fonte 2. Nele, uma *string* contendo o texto a ser analisado pode ser associado a um contexto (por meio de uma *tupla*) e, então, solicita-se ao modelo importado a conotação expressa em **Positive** ou **Negative**.

Código-fonte 2: Exemplo de código que determina, através do modelo do projeto, o sentimento atribuído a um excerto.

```
1 import ktrain
2
3 model = ktrain.load_predictor('./BERPT')          # Importa o modelo
4
5 text = "Gostei muito da dinâmica do aplicativo"  # Excerto
6 context = "Corporativo"                           # Contexto
7 test_input = (text, context)                      # Mapeamento da tupla
8
9 result = model.predict(test_input)                # Predição do sentimento
10
11 print("Classificação: ", result)
```

3.4 Dificuldades e Limitações

As principais dificuldades enfrentadas no desenvolvimento do projeto foram limitações quanto aos ambientes computacionais disponíveis, em especial para as etapas de treinamento e análise de resultados. A princípio, os ambientes computacionais em nuvem disponíveis deparam-se com limitadores quanto ao uso de memórias, GPU, instâncias simultâneas e, principalmente, tempo de execução. Dessa forma, como comentado na Seção 3.2.3, houve certa necessidade em encontrar um ambiente computacional disponível que oferecesse ao, menos, a disponibilidade de computação paralela por GPU e tempo de execução contínua suficiente para o processo de treinamento do modelo. O ambiente disponibilizado pelo Laboratório de Inteligência Computacional da Universidade de São Paulo (LABIC/USP) para o processo de treinamento, hospedado pela *Microsoft Azure*, supriu as necessidades ao oferecer um processador, memória RAM e, principalmente, GPU com altíssimo desempenho e pelo tempo necessário, permitindo assim o treinamento de modelo para as 3 épocas, como já especificado em 3.2.3.

A outra etapa onde a barreira da limitação computacional fez-se presente foi durante a validação, onde realiza-se a aferição dos sentimentos diante do montante de teste. Nesse caso porém, como não utiliza de computação de alto desempenho, os maiores impeditivos foram o *runtime* máximo e o número limitado de instâncias simultâneas permitidas. Foi necessário dividir as execuções em frações bem reduzidas do volume total de amostras de teste para executar de forma sequencial e incremental conforme as execuções fossem sendo finalizadas, salvando em intervalos de tempo seguros dentro do limite de *12 horas* imposto. Esses empecilho somada a falta de tempo de desenvolvimento restante, fizeram com que não houvesse tempo hábil para

completar as predições de todas as amostras de grupos com maiores volumes extraídos, como *Finanças* e *Entretenimento*, antes de encerrar a etapa de execução para partir à análise dos resultados.

Do ponto de vista pessoal, os maiores desafios enfrentados durante a realização foram quanto a concepção e modelagem do projeto em suas etapas inicialmente, isso é, definição do cronograma e melhores abordagens para cada passo do projeto, e quanto ao estudo aprofundado de referências e aplicação em código para uma avaliação coerente e precisa do modelo. Para o primeiro caso, a análise de outros projetos de ajuste fino e treinamento de modelo foram fundamentais para estruturar corretamente o plano de ação. Já para a segunda dificuldade apontada, a leitura de diversos artigos e materiais didáticos, bem como a familiarização com as bibliotecas de validação em *Python* foram essenciais para contornar e chegar no resultado final obtido com esse projeto.

Capítulo 4

CONCLUSÃO

4.1 Contribuições

Este projeto apresenta como contribuição a publicação de um modelo de análise de sentimento para a língua portuguesa, especialmente para mídias digitais e de avaliações. Esse modelo apresentou, dado os testes realizados e dispostos nessa monografia, um grande potencial de assertividade na classificação de excertos textuais em conotações *Positivas* ou *Negativas*. Os resultados obtidos nos testes demonstram que seus valores de acurácia, precisão, *F1* e demais propriedades são superiores a 90%, e para algumas categorias como contexto chegando a 95%. Quando comparado a outros modelos que desempenham esse papel para o idioma português, os valores obtidos demonstram a capacidade do mesmo em servir como base para outros projetos em nível de aplicação. Dessa forma, pode concluir que o projeto disponibiliza à comunidade uma ótima ferramenta com vasto potencial para atender de maneira adequada futuros projetos que fomentem o cenário brasileiro de análise de sentimentos.

Somado a isso, o *dataset* obtido durante a realização do experimento também foi disponibilizado atuando como outro recurso, validado e com comprovação de bons resultados, a disposição de futuros projetos.

Dessa forma, o objetivo inicial de desenvolvimento de um modelo base para auxiliar no desenvolvimento do ambiente de mineração de opiniões no cenário brasileiro foi atingido.

4.2 Trabalhos Futuros

O modelo obtido e o *dataset* com avaliações dos aplicativos extraídas durante o desenvolvimento do projeto encontram-se disponibilizados em repositórios no site *Hugging Face*, de modo a permitir o uso dos mesmos para todas as ideias de projetos que necessitem de ferramentas baseadas em *word embedding* para análise de sentimento na língua portuguesa, como pretendia-se com a realização deste. Assim, as futuras contribuições possuem um vasto horizontes de possibilidades para serem exploradas, partindo desde aplicações mais direta no uso de um classificador - como uma ferramenta de avaliação e recomendação ou um código para prever curvas de tendências de certos objetos e serviços baseado no gosto de um publico-alvo, a projetos mais complexos, onde torna-se apenas uma ferramenta associada a outro modelo com

determinado objetivo - por exemplo uma IA que cria um excerto que representa um sentimento específico.

Além disso, é importante destacar que, como um modelo refinado para esse domínio. disponibilizado para funcionalidades futuras, faz-se interessante uma ampliação e possível maior ajuste do treinamento do mesmo, podendo realizar uma expansão para novos sistemas de *reviews* de produtos e sites diferentes, por exemplo. Podendo melhorar, assim, a assertividade e desempenho do mesmo.

REFERÊNCIAS

ABRAO, B. S. **A História da Filosofia**. [S.l.]: Editora Nova cultura, 2004. 15–68 p. Citado na página 17.

ALECRIM, E. Facebook é acusado de analisar sentimentos de adolescentes para direcionar anúncios. tecnoblog, 2017. Disponível em: <https://tecnoblog.net/noticias/2017/05/02/facebook-sentimentos-adolescentes/>. Citado na página 17.

ALMEIDA, R. J. A. **LeIA - Léxico para Inferência Adaptada**. [S.l.]: GitHub, 2018. <https://github.com/rafjaa/LeIA>. Citado na página 41.

ARAUJO, A. F.; GOLO, M. P. S.; MARCACINI, R. M. Opinion mining for app reviews: An analysis of textual representation and predictive models. Kluwer Academic Publishers, USA, v. 29, n. 1, may 2022. ISSN 0928-8910. Disponível em: <https://doi.org/10.1007/s10515-021-00301-1>. Citado 2 vezes nas páginas 18 e 24.

BARROS, Á. G. D.; SOUZA, C. H. M. D.; TEIXEIRA, R. F. Evolução das comunicações até a internet das coisas: A passagem para uma nova era da comunicação humana. **Cadernos de Educação Básica**, 2020. Citado na página 17.

BISWAS, M.; TANIA, M. H.; KAISER, M. S.; KABIR, R.; MAHMUD, M.; KEMAL, A. A. Accu3rate: A mobile health application rating scale based on user reviews. **PLOS ONE**, Public Library of Science, v. 16, n. 12, p. 1–24, 12 2021. Disponível em: <https://doi.org/10.1371/journal.pone.0258050>. Citado na página 33.

BROWNLEE, J. What are word embeddings for text? 2017. Disponível em: <https://machinelearningmastery.com/what-are-word-embeddings/>. Citado na página 26.

CARVALHO, P.; SARMENTO, L.; SILVA, M. J.; OLIVEIRA, E. de. Clues for detecting irony in user-generated contents: Oh...!! it's "so easy";-). In: **Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion**. New York, NY, USA: Association for Computing Machinery, 2009. (TSA '09), p. 53–56. ISBN 9781605588056. Disponível em: <https://doi.org/10.1145/1651461.1651471>. Citado na página 26.

CASTRO, P. H. P. Formalismo e funcionalismo: uma análise da complementariedade dessas correntes linguísticas. **Artigo publicado pela Universidade Federal do Ceará**, 2015. Citado na página 18.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**. arXiv, 2018. Disponível em: <https://arxiv.org/abs/1810.04805>. Citado 4 vezes nas páginas 18, 28, 29 e 30.

FONSECA, C. Word embedding: fazendo o computador entender o significado das palavras. 2021. Disponível em: <https://medium.com/turing-talks/word-embedding-fazendo-o-computador-entender-o-significado-das-palavras-92fe22745057>. Citado na página 26.

FREIRE, P. M. S.; GOLDSCHMIDT, R. R. Combate automático às fake news nas mídias sociais virtuais: uma revisão do estado da arte. In: . [S.l.]: Instituto Militar de Engenharia, 2021. Citado na página 17.

GANTZ, J.; REINSEL, D. The digital universe decade - are you ready? **IDC (Analyse the Future) Information and Data**, 2010. Citado na página 21.

GILBERT, C. H. E. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: **Eighth International Conference on Weblogs and Social Media (ICWSM-14)**. [S.l.: s.n.], 2014. Citado na página 41.

KARIN, B.; TUMITAN, D. Introdução à mineração de opiniões: Conceitos, aplicações e desafios. In: **Lectures of the 28th Brazilian Symposium on Databases**. [S.l.]: Programa de Pós-Graduação em Ciência da Computação - Instituto de Informática Universidade Federal do Rio Grande do Sul (UFRGS), 2013. p. 27–52. Citado 4 vezes nas páginas 21, 22, 23 e 25.

MAIYA, A. S. **ktrain: A Low-Code Library for Augmented Machine Learning**. arXiv, 2020. Disponível em: <https://arxiv.org/abs/2004.10703>. Citado na página 37.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. **Efficient Estimation of Word Representations in Vector Space**. arXiv, 2013. Disponível em: <https://arxiv.org/abs/1301.3781>. Citado 2 vezes nas páginas 27 e 28.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G.; DEAN, J. **Distributed Representations of Words and Phrases and their Compositionality**. arXiv, 2013. Disponível em: <https://arxiv.org/abs/1310.4546>. Citado na página 27.

MISURACA, M.; SCEPI, G.; SPANO, M. Using opinion mining as an educational analytic: An integrated strategy for the analysis of students' feedback. **Studies in Educational Evaluation**, v. 68, p. 100979, 2021. ISSN 0191-491X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0191491X21000055>. Citado na página 17.

PANG, B.; LEE, L.; VAITHYANATHAN, S. Thumbs up? sentiment classification using machine learning techniques. In: **Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)**. Association for Computational Linguistics, 2002. p. 79–86. Disponível em: <https://aclanthology.org/W02-1011>. Citado na página 22.

PIRES, T.; SCHLINGER, E.; GARRETTE, D. How multilingual is multilingual BERT? In: **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**. Florence, Italy: Association for Computational Linguistics, 2019. p. 4996–5001. Disponível em: <https://aclanthology.org/P19-1493>. Citado na página 30.

PRAGER, J. Open-domain question–answering. **Foundations and Trends® in Information Retrieval**, v. 1, n. 2, p. 91–231, 2007. ISSN 1554-0669. Disponível em: <http://dx.doi.org/10.1561/1500000001>. Citado na página 21.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Bertimbau: Pretrained bert models for brazilian portuguese. In: CERRI, R.; PRATI, R. C. (Ed.). **Intelligent Systems**. Cham: Springer International Publishing, 2020. p. 403–417. ISBN 978-3-030-61377-8. Citado na página 30.

TSYTSARAU, M.; PALPANAS, T. **Survey on Mining Subjective Data on the Web**. [S.l.]: Data Min Knowl Disc 24, 2012. 478–514 p. Citado 2 vezes nas páginas 21 e 22.

TURNEY, P. D. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. arXiv, 2002. Disponível em: <https://arxiv.org/abs/cs/0212032>. Citado na página 22.

WIEBE, J. M.; BRUCE, R. F.; O'HARA, T. P. Development and use of a gold-standard data set for subjectivity classifications. In: **Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics**. College Park, Maryland, USA: Association for Computational Linguistics, 1999. p. 246–253. Disponível em: <https://aclanthology.org/P99-1032>. Citado na página 22.

XU, H.; LIU, B.; SHU, L.; YU, P. S. **BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis**. arXiv, 2019. Disponível em: <https://arxiv.org/abs/1904.02232>. Citado na página 30.

YUGOSHI, I. P. M. Mineração de opiniões baseadas em aspectos para revisões de produtos e serviços. In: **Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos.** [s.n.], 2018. Disponível em: <https://teses.usp.br/teses/disponiveis/55/55134/tde-17102018-112458/pt-br.php>. Citado 4 vezes nas páginas 18, 21, 22 e 23.

ZHANG, L.; LIU, B. Sentiment analysis and opinion mining. In: ____. **Encyclopedia of Machine Learning and Data Mining**. Boston, MA: Springer US, 2017. p. 1152–1161. Disponível em: <https://doi.org/10.1007/978-1-4899-7687-1_907>. Citado na página 21.

ZWEIG, M. H.; CAMPBELL, G. Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. **Clinical Chemistry**, v. 39, n. 4, p. 561–577, 04 1993. ISSN 0009-9147. Disponível em: <https://doi.org/10.1093/clinchem/39.4.561>. Citado na página 44.