UNIVERSIDADE FEDERAL DE GOIÁS
INSITUTO DE INFORMÁTICA

DOMINIC ROCHA DE PAULO

Detecção de traços de depressão e ansiedade em postagens do Twitter utilizando algoritmos de aprendizado de máquina

Goiânia
2022

UNIVERSIDADE FEDERAL DE GOIÁS

INSITUTO DE INFORMÁTICA

AUTORIZAÇÃO PARA PUBLICAÇÃO DE TRABALHO DE CONCLUSÃO DE CURSO EM FORMATO ELETRÔNICO

Na qualidade de titular dos direitos de autor, **AUTORIZO** o Insituto de Informática da Universidade Federal de Goiás – UFG a reproduzir, inclusive em outro formato ou mídia e através de armazenamento permanente ou temporário, bem como a publicar na rede mundial de computadores (*Internet*) e na biblioteca virtual da UFG, entendendo-se os termos "reproduzir" e "publicar" conforme definições dos incisos VI e I, respectivamente, do artigo 5º da Lei nº 9610/98 de 10/02/1998, a obra abaixo especificada, sem que me seja devido pagamento a título de direitos autorais, desde que a reprodução e/ou publicação tenham a finalidade exclusiva de uso por quem a consulta, e a título de divulgação da produção acadêmica gerada pela Universidade, a partir desta data.

Título: Detecção de traços de depressão e ansiedade em postagens do Twitter utilizando algoritmos de aprendizado de máquina

Autor(a): Dominic Rocha de Paulo

Goiânia, 11 de Abril de 2022.

Dominic Rocha de Paulo

Dominic Rocha de Paulo – Autor

Dra. Deborah Silva Alves Fernandes – Orientador

Márcio Giovane Cunha Fernandes – Co-Orientador

DOMINIC ROCHA DE PAULO

Detecção de traços de depressão e ansiedade em postagens do Twitter utilizando algoritmos de aprendizado de máquina

Trabalho de Conclusão apresentado à Coordenação do Curso de Sistemas de Informação do Insituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Área de concentração: Sistemas de Informação.

Orientador: Prof. Dra. Deborah Silva Alves Fernandes

Co-Orientador: Prof. Márcio Giovane Cunha Fernandes

Goiânia
2022

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

DOMINIC ROCHA DE PAULO

Detecção de traços de depressão e ansiedade em postagens do Twitter utilizando algoritmos de aprendizado de máquina

Trabalho de conclusão de curso apresentado à Universidade Federal de Goiás como parte dos requisitos para a obtenção do título de Bacharel em Sistemas de Informação.

Orientadora: Profa. Dra. Deborah Silva Alves Fernandes

Aprovado em 11/04/2022.

BANCA EXAMINADORA

Profa. Dra. Deborah Silva Alves Fernandes
Universidade Federal de Goiás
Instituto de Informática

Prof. Me. Márcio Giovane Cunha Fernandes
Universidade Estadual de Goiás

Profa. Dra. Luciana de Oliveira Berretta
Universidade Federal de Goiás
Instituto de Informática

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador(a).

Dominic Rocha de Paulo

Graduanda em Sistemas de Informação na UFG - Universidade Federal de Goiás.

Dedico este trabalho a minha querida avó Maria e ao meu filho que é o amor da minha vida.

Agradecimentos

Agradeço ao meu filho Bernardo por compreender as várias horas em que estive ausente devido ao desenvolvimento deste trabalho.

Agradeço também aos meus pais Silene e Evinaldo pelo carinho, atenção e apoio que eles me deram durante toda a minha vida.

Em seguida, quero agradecer aos meus avôs, Maria, Norminda e Ananias e minha irmã pentelha Geovana que estiveram sempre presentes ao meu lado e me apoiando ao longo de toda a minha trajetória.

Agradeço ao meu namorado, Ismael que esteve sempre ao meu lado durante o meu percurso acadêmico.

A minha professora orientadora Deborah pelas valiosas contribuições dadas durante todo o processo tornou-se essencial para que o projeto fosse concluído.

Por último, quero agradecer também à Universidade Federal de Goiás, ao Instituto de Informática e todo o seu corpo docente.

"Worry does not empty tomorrow of its sorrow. It empties today of its strength."

Corrie Ten Boom,

.

Resumo

de Paulo, R. Dominic. **Detecção de traços de depressão e ansiedade em postagens do Twitter utilizando algoritmos de aprendizado de máquina**. Goiânia, 2022. 60p. Relatório de Graduação. Insituto de Informática, Universidade Federal de Goiás.

Um estudo realizado pela OMS no ano de 2017, revela que no Brasil, há 11,5 milhões de pessoas diagnosticadas com distúrbios depressivos [42]. Além deste transtorno, o Brasil registra o maior índice de ansiedade ao ser comparado aos outros países, cerca de 18 milhões de brasileiros possuem esse problema. O que impacta diretamente na forma como o individuo enxerga a vida.

Este trabalho, pretende aplicar algoritmos de *machine learning* para detectar traços de ansiedade e depressão em *tweets* na língua portuguesa. Portanto, foram utilizados alguns classificadores independentes e *ensembles* para implementar este projeto.

Palavras–chave

Depressão, Ansiedade, *Machine Learning*, *Twitter*

Abstract

de Paulo, R. Dominic. **<Work title>**. Goiânia, 2022. 60p. Relatório de Graduação. Insituto de Informática, Universidade Federal de Goiás.

A study carried out by the WHO in 2017 reveals that in Brazil, there are 11.5 million people diagnosed with depressive disorders. In addition to this disorder, Brazil has the highest rate of anxiety when compared to other countries, about 18 million Brazilians have this problem. Which directly impacts the way the individual sees life.
This work aims to apply machine learning algorithms to detect traces of anxiety and depression in tweets in Portuguese. Therefore, some independent classifiers and ensembles were used to implement this project.

Keywords

Depression, Anxiety, Machine Learning, Twitter

Sumário

Lista de Figuras

[1] https://abracd.org/overfitting-e-underfitting-em-machine-learning/

Lista de Tabelas

CAPÍTULO 1

Introdução

1.1 Contextualização

Com os avanços tecnológicos, a internet proporcionou algumas mudanças de paradigmas na forma como as pessoas compartilhavam e recebiam as informações [43]. Um dos motivos para tal acontecimento deu-se em razão da evolução da Web para 2.0, na qual a comunidade começou a ser mais ativa [50], produzindo e compartilhando conhecimento entre si [52].

Considerando tais mudanças, surgiram alguns sites que permitiram às pessoas a construção de redes de conexões, podendo expressar diariamente suas emoções, experiências e opiniões sobre diversos assuntos [30]. Segundo um relatório realizado pelo *We Are Social* e *Hootsuite*[1], nos últimos 10 anos, os usuários que utilizam a internet dobraram de tamanho, de 2,18 bilhões para 4,95 bilhões no início de 2022. Nas mídias sociais houve um crescimento de 3 vezes maior no mesmo período, passando de 1,48 para 4,62 bilhões, o que sugere que a cada ano, menos usuários ficam *offline*.

No Brasil, de acordo com o Datareportal [2], existem aproximadamente 214,7 milhões de habitantes, cerca de 77% desta população possui acesso à internet e 79,0% possuem perfil em redes sociais, esses passam em média 3 horas e 41 minutos conectados por dia. Ao comparar os dados, é importante ressaltar que a taxa dos indivíduos nas mídias sociais podem não representar valores únicos, o que explicaria o percentual mais elevado em relação à população que utiliza somente a internet.

Com o crescimento desses usuários, é possível observar que as interações realizadas por eles geram uma grande coleção de dados, em sua maioria, não estruturados [52]. Conforme as estatísticas do site *Internet Live Stats*[3], produzido pelo projeto *Real Time Statistics Project*, o trafego da internet gira em torno de 145,893 GB de informações por segundo. Neste contexto, torna-se humanamente impossível analisar cada conteúdo

[1] https://datareportal.com/reports/digital-2022-global-overview-report
[2] https://datareportal.com/reports/digital-2022-brazil
[3] https://www.internetlivestats.com/

[52], devido à alta velocidade e o grande volume em que essas informações circulam. Portanto, atualmente existem vários estudos dentro e fora da academia, que utilizam técnicas de mineração de texto e inteligência artificial para analisar sentimentos, opiniões e obter outras informações a partir desses dados.

Neste trabalho, utilizaremos algoritmos de *machine learning* para detectar traços de depressão em postagens públicas no *Twitter*. Esta rede social foi escolhida, devido a sua popularidade no Brasil e algumas características específicas que facilitam a tarefa de mineração de texto:

- Conforme a pesquisa publicada pela *DataReportal*, o *Twitter* é considerado o décimo quinto site mais visitado, com um total de 19,2% milhões de usuários únicos, que permanecem cerca de 17 minutos e 29 segundos conectados na plataforma;
- Diferente de outras redes sociais, a maioria do conteúdo publicado no *Twitter* é de domínio público, produzindo mais conteúdos que podem ser utilizados em pesquisas;
- Por ser um microblog, os usuários conseguem transmitir uma opinião em uma mensagem curta, com o comprimento de até 280 caracteres; [31][51]. Isso é uma ótima característica, pois estamos trabalhando com uma grande quantidade de dados.

Por esses motivos, esta mídia fornece meios para capturar comportamentos relevantes que podem indicar traços de linguagem depressiva em suas postagens. O reconhecimento de linguagem depressiva em redes sociais pode ser útil para adoção de políticas públicas específicas bem como auxílio através do meio virtual para pessoas que se estejam sofrendo dentre outras possibilidades. Depressão e ansiedade é considerada um dos transtornos metais mais comuns atualmente e este contexto será abordado neste projeto final de curso.

1.2 Definição do Problema

A detecção automática de mensagens com traços de ansiedade e depressão em uma grande base de dados coletado da rede social Twitter é o problema abordado neste projeto final de curso.

1.3 Objetivos do Trabalho

O objetivo geral deste trabalho é realizar a análise de postagens coletadas no *Twitter* em língua portuguesa utilizando estrategias de *machine learning* e processamento de linguagem natural para detecção de traços de ansiedade e depressão.

Os objetivos específicos são:

1. Pesquisar e estudar trabalhos relacionados ao tema de detecção de traços de ansiedade e depressão em textos de língua portuguesa e outros idiomas; Implementar vários algoritmos de classificação;
2. Estudar e projetar, baseado no artigo *baseline* [23], um modelo para classificação automática de *tweets*;
3. Identificar, selecionar, estudar e implementar os algoritmos de *machine learning* mais adequados para compor a estratégia de classificação modo independente e em *ensemble*;
4. Comparar os modelos de classificação independente e *ensemble* implementados, identificando os melhores para o problema abordado.
5. Comparar os resultados obtidos neste projeto com os do artigo *baseline* [23].

1.4 Apresentação da Monografia

Este trabalho contém seis capítulos contando com este de introdução, os demais estão organizados da seguinte forma:

- **Capítulo 2 — Trabalhos Relacionados**: Será utilizado como base de conhecimento para a realização deste trabalho, apresentando pesquisas correlatas de alguns autores que realizaram experimentos em mídias sociais, especificamente no Twitter, onde tinham como objetivo utilizar técnicas de *machine learning* para detectar depressão ou ansiedade.
- **Capítulo 3 — Fundamentação Teórica**: Neste serão apresentados os conceitos utilizados para o desenvolvimento e entendimento do trabalho em relação ao tema ansiedade e depressão e também classificação automática de textos e algoritmos.
- **Capítulo 4 — Experimento**: Será apresentada a metodologia e ferramentas utilizadas durante a realização do projeto, contendo descrição de todas as etapas executadas.
- **Capítulo 5 — Resultados**: Neste serão mostrados e discutidos os resultados obtidos para o experimento. Por fim serão apontadas algumas diferenças entre os resultados do artigo utilizado como*baseline* e este.
- **Conclusão**: Finalmente, essa seção conterá uma sumarização das principais conclusões deste trabalho, além de sugestões para pesquisas futuras.

CAPÍTULO 2

Trabalhos Relacionados

2.1 Pesquisa de Literatura

Com a popularização das redes sociais, houve um grande aumento na quantidade de dados disponíveis na internet. Esses dados são utilizados por instituições acadêmicas e organizacionais com o intuito de extrair informações úteis que possam ser aprendidas e reconhecidas por máquinas através de padrões. Dentro desse contexto, as redes sociais possibilitaram várias pesquisas que estudam a saúde mental e as tendências comportamentais através de técnicas de inteligência artificial para detectar traços de ansiedade e depressão [15][12][61].

Wongkoblap et al.[62] propôs criar dois tipos de aprendizado supervisionado de *Multiple Instance Learning* (MIL), com o intuito de investigar se o conteúdo gerado no *Twitter* poderia detectar usuários com depressão. O primeiro modelo construído foi uma rede neural recorrente do tipo *Long Short Term Memory* (LSTM), denominada como MIL-SocNet. O segundo, foi nomeado como MILA-SocNet, este modelo era a extensão do primeiro, com o acréscimo de uma resolução anafórica, para garantir que o algoritmo se concentrasse nas postagens relacionadas ao autor. Wongkoblap et al.[62] analisou diferentes parâmetros e observou que as postagens com comprimento e *tokens* de palavras mais longas forneciam melhores resultados e os modelos com menos dimensões incorporadas tinham desempenho inferior comparados aos com mais dimensões. Além dessas observações, os pesquisadores constataram que o modelo MILA-SocNet, obteve os melhores desempenhos em comparação aos outros modelos implementados, com 92% para as métricas de acurácia, precisão, *recall* e *F1-score*.

Chen et al.[2] utilizou um sistema de ontologia (EMOTIVE) baseado em algoritmo de sentimento avançado para detectar e medir emoções básicas como, por exemplo: raiva, nojo, felicidade e outros em postagens do Twitter. Esse sistema tinha o intuito de investigar a eficácia dos recursos baseados na emoção para identificar usuários em risco de depressão. Para isso, os autores criaram dois conjuntos a partir das medições da ontologia. O conjunto EMO, representava os recursos atemporais utilizados para avaliar se as emoções carregavam informações preditivas. Já a base EMO_TS, representava as ca-

racterísticas temporais utilizadas para capturar as mudanças emocionais dos indivíduos ao longo do tempo, portanto neste conjunto, os *tweets* foram datados com o dia da publicação dos mesmos. Para a avaliação desses recursos, os pesquisadores combinaram vários classificadores com conjuntos de dados diferentes. Como resultados observaram que o conjunto EMO e EMO_TS obtiveram uma predição de 87,27% e 89,77% respectivamente com o classificador *Randon Forest*. Ao combinar os dois recursos, houve uma melhora na predição para 91,81%.

Tong et al.[61] criou um classificador, chamado *Inverse boosting pruning trees* (IBPT). Este é baseado em um algoritmo aprimorado de Adaboost + *pruning decision trees* para identificar usuários depressivos em dados ruidosos do *Twitter*, mitigando a influência dos ruídos e erros na classificação. Para a avaliação do modelo, os pesquisadores dividiram o experimento em três etapas. Na primeira etapa, foi analisado a correlação entre o número base de estimador M e a dobra de validação cruzada de *pruning*. Na segunda etapa, observaram a capacidade de generalização de classificação do modelo em comparação com 14 classificadores em dados reais de brinquedos. Por fim, na terceira etapa, Tong et al.[61] avaliou o desempenho do algoritmo de classificação para detectar traços de depressão em uma base de dados coletada por outra pesquisa que continha 2.558 e 5.304 amostras rotulando usuários como deprimidos e não deprimidos. Como resultados, observaram que o método proposto por eles foi superior aos demais e em específico na última etapa o modelo atingiu a maior pontuação de AUC com 93%. O classificador IBPT mostrou-se capaz de classificar conjuntos de dados complexos com multi-características e a pesquisa apontou que o recurso de alocação de *dirichlet* (*Latent Dirichlet Allocation* — LDA) para extrair distribuições de tópicos dos *tweets* seria o mais importante para detecção de depressão.

Shetty et al.[56] empregou métodos de *machine learning* para prever a depressão em usuários através da verificação de sinais deste em seus *tweets*. Para isso, os pesquisadores utilizaram dois modelos de *deep learning*: *Long short-term memory* (LSTM) e *Convolutional Neural Network* (CNN) para analisar os sentimentos das publicações no *Twitter* para prever as classes binárias deprimidos/não deprimidos. O modelo LSTM proposto utilizou um conjunto de dados do Kaggle que foi compilado usando o otimizador Adam, após a compilação o modelo foi ajustado aos recursos gerados anteriormente e validado nos rótulos especificados no conjunto de dados. Cada novo *tweet* obtido passava por um pré-processamento, com o intuito de preparar os dados para a etapa seguinte. Esses dados foram divido entre treino e teste e vetorizados com base na vetorização de contagem, TF-IDF e n-gramas. A predição obtida neste modelo, foi comparado ao modelo de rede neural convolucional compilado com as mesmas configurações, com a única diferença de que o argumento de peso recebia uma matriz de incorporação. No estágio dois, os pesquisadores tentaram melhorar o resultado utilizando classificadores de *ma-*

chine learning (Regressão Logística, SVC, *Multinomial Bayes Naïve*, *Bernoulli Bayes Naïve* com *Random Forest* e o classificador de aumento de gradiente). Como resultado o modelo LSTM obteve 70% de acurácia nos dados de validação e 93% nos dados de teste. Já, a rede neural CNN obteve 68% de acurácia nos dados de validação e 95% nos dados de teste. O classificador de regressão logística obteve a melhor pontuação com os vetores de contagem (75%) e TF-IDF (76,22%) de acurácia e o classificador multinomial *Naïve Bayes* obteve a melhor pontuação com (76,69%).

O estudo proposto por Asad et al. [40] aplicou a análise de sentimento para observar o nível de vulnerabilidade dos usuários à depressão e elucidar quais parâmetros indicariam esse transtorno nas redes sociais. Para atingir esse objetivo, foram coletados dados compartilhadas no *Twitter* e *Facebook*. Os dados foram pré-processados com a biblioteca NLTK (*Natural Language Toolkit*) e vetorizados pela medida estatística TF-IDF (*Term Frequency-Inverse Document Frequency*). O método proposto no estudo utilizou o classificador *Support Vector Machine* (SVM) e *Naive Bayes* para converter os sentimentos em seis intervalos de níveis de depressão: normal (1–25%), leve (26-40%), limítrofe (41-55%), moderado (56-70%), severo (71–85%) e extremo (85-100%). Caso o resultado do nível de depressão estivesse entre 1 — 55% o modelo assumiria o usuário como não deprimido em contraposição se o nível estivesse acima de 55% o mesmo seria assumido como deprimido. Para avaliar o modelo proposto, os pesquisadores utilizaram um método baseado em uma entrevista BDI-II (Inventário de Depressão de Beck) correlacionando o resultado dessa entrevista com o modelo proposto, onde obteve a acurácia de 74% e uma precisão de 100%.

Gui et al.[3] propôs criar um método de aprendizagem por reforço multiagente para detectar a depressão em postagens multimodais no *Twitter*. Para tal, os pesquisadores criaram um modelo cooperativo multiagente de operação indevida (*Cooperative Misoperation Multi-Agent* (COMMA)) que tenta resolver o problema de recompensas únicas globais em ambientes cooperativos. O COMMA utiliza dois agentes de gradiente de política para selecionar simultaneamente texto e imagem com uma abordagem ator-crítico (*Konda and Tsitsiklis* 2000) com vantagens diferenciadas, que compara a recompensa atual global com a recompensa recebida pelo agente a cada ação incorreta. A base de dados textual e visual foram extraídas por uma rede neural recorrente (RNN) e *Gated Recurrent Unit* (GPU) respectivamente, e divididas entre as classes “Depressivo” com 1402 usuários e “Não Depressivo” com 5160 usuários. O modelo foi treinado e testado utilizando *five-fold cross validation*. Através de vários experimentos, os pesquisadores concluíram que o método proposto obteve o melhor desempenho de acurácia, precisão, *recall* e F1 com 90% em comparação a todos os modelos avaliados pelos autores.

Almouzini et al. [4] identificou se os usuários árabes compartilhavam evidências de depressão e/ou sentimentos depressivos em publicações no *Twitter*. Para isso, os

pesquisadores criaram um *corpus* com rótulos de deprimidos e não deprimidos. Os usuários deprimidos foram identificados a partir do diagnóstico de transtorno depressivo compartilhado na rede social e seus *tweets* foram classificados pelas ferramentas de escalas de depressão CES-D e PHQ-9. Após o pré-processamento pelo pacote *Affective tweets* os dados foram submetidos a quatro classificadores supervisionados: *Random Forest*, *Naïve Bayes*, *AdaBoostM1* e *Liblinear* que foram implementados e treinados utilizando a ferramenta Weka com validação cruzada *(10-fold)*. Para avaliar o melhor modelo, os pesquisadores optaram por trabalhar com as métricas de acurácia, precisão, *recall* e F1. Como resultado, o classificador *AdaBoostM1* obteve a pontuação mais baixa com acurácia 55,2% e *recall* 55,3% em contraste com o classificador *Liblinear* que obteve a pontuação mais alta para acurácia 87,5% e com 87,5%. A partir do estudo, os pesquisadores observaram que os usuários que sofrem de depressão estão mais isolados socialmente, isso foi evidenciado ao examinarem como elas interagem com as *hashtags* e emojis populares usados em seus *tweets*.

Orabi et al. [41] investigou a possibilidade de utilizar as abordagens de *deep learning* em dados não estruturados para detectar usuários com sinais de depressão. Para isso os *embeddings* otimizados e produzidos no estudo foram avaliados a partir do *random trainable*, skip-gram e CBOW em dois conjuntos dados, CLPsych 2015 e Bell Let's Talk. O conjunto de treinamento e teste possuíam respectivamente uma quantidade média de 13.041 e 3.864 palavras por usuário, extraídas considerando os atributos de idade e gênero, diferente dos dados de teste.Para o pré-processamento desses dados, foi utilizado a biblioteca NLTK, um dicionário de palavras de termos fixos e 2 (dois) modelos de incorporação de palavras (Word2Vec, Rand). O modelo CNN, que usou a incorporação otimizada proposta pelos autores na base CLPsych 2015, obteve a maior acurácia (87,957%), F1 (86,967%), AUC (0,951), precisão (87,435%) e *recall* (87,029%) em comparação com outros modelos. Os modelos treinados que usaram os *embeddings* otimizados na base *Bell Let's Talk* tiveram as maiores pontuações de acurácia (83,11%), F1 (82,25%), AUC (92%), precisão (81,62%) e *recall* (84,43%) conseguindo manter o desempenho e capacidade de generalização.

Tiwari et al. [60] identificou a depressão e transtorno de estresse pós-traumático entre os usuários do *Twitter*, com o objetivo de descobrir qual a probabilidade de uma pessoa sofrer qualquer uma dessas doenças. Coletou dados do *Twitter* com as palavras chaves: "estresse", "depressão", "chateado", "abatido", "suicídio", "suicida", "trauma", "PTSD". Para realizar o pré-processamento dos dados utilizou as bibliotecas NLTK, STRING, RE, TIME & PANDAS, removeu os *emoticons* e sinais de pontuação, permanecendo apenas o texto e o ID do usuário. Em seguida o dicionário foi calculado o sentido das palavras do dicionário e comparado com o conjunto de dados. Após encontrar o sentimento, é calculado a polaridade do *tweet*, salvando o ID com a polaridade

correspondente em outro arquivo. Para o experimento os pesquisadores selecionaram os algoritmos de classificação:*Naïve Bayes*, Árvore de Decisão, SVM, KNN e *Random Forest*. O algoritmo que teve a melhor acurácia foi a Árvore de Decisão com 92,80%. O algoritmo com o melhor tempo de predição e uma acurácia de 87.13%. O algoritmo SVM obteve o pior tempo e o algoritmo *Random Forest* teve a pior acurácia com 51,35%. Segundo os autores, o projeto pode ser melhorado utilizando modelagens de léxico para remover palavras com frequência mais baixa. Sugerem ainda a análise de *emoticons* e sarcasmos para entender a real intenção dos *tweets*.

Zhang et al. [64] propôs desenvolver um método para tentar identificar usuários com depressão, com o propósito de descobrir quais foram os impactos mentais que o coronavírus trouxe para as pessoas devido às drásticas mudanças no cotidiano. Para alcançar o objetivo, os pesquisadores selecionaram 5.150 usuários do *Twitter*, metade deles rotulados com depressão e a outra metade grupo de controle. As informações dos *tweets* foram analisadas inicialmente sob 5 aspectos: traços de personalidade, sentimentos, demografia, consulta linguística(contagem de palavras) e engajamento na mídia social. Utilizaram dois modelos de *deep learning* para realizar a classificação: rede neural convolucional (*Convolutional neural network* — CNN) multicanal, e uma rede neural recorrente do tipo memória de curto prazo longa (*Long Short Term Memory* — LSTM) bidirecional. Foram treinados 3 (três) modelos representativos de classificação de sequência baseada em transformadores: BERT, RoBERTa e XLNet. Os modelos executaram os *tweets* do mesmo usuário para calcular a média dos resultados de conferência para obter a pontuação de confiança nos dados de treinamento e teste. Independentemente do conjunto de testes, os modelos baseados em transformadores BERT, RoBERTa e XLnet superaram invariavelmente BiLSTM e CNN. Em particular, o modelo XLNet registrou o melhor AUC com 84,4%, bem como a melhor precisão com 77,5% de todos os modelos quando treinados com o conjunto completo de treinamento. O modelo XLNet foi comparado com outras abordagens e o SVM atingiu os melhores escores em acurácia 78,9% e F1 79,2% e os resultados mostraram que usar dados demográficos e recursos de engajamento de mídia social não ajuda muito a classificação. Uma observação imediata é que, independentemente do modelo, o desempenho da classificação melhora à medida que aumenta o tamanho do conjunto de treino-validação.

CAPÍTULO 3

Fundamentação Teórica

3.1 Depressão e Ansiedade

A depressão e a ansiedade são doenças [49] [39] distintas, entretanto, é comum, pessoas experimentarem ambas as condições simultaneamente [42]. Segundo uma pesquisa feita pela OMS em 2017, 4,4% da população mundial possuía transtornos depressivos e cerca de 3,0% apresentava transtornos de ansiedade [42]. Diferenciar esses distúrbios psicológicos, pode ser uma tarefa difícil, porém, é necessário diagnosticar e tratar precocemente para não afetar o comportamento do indivíduo acarretando prejuízos no âmbito social, profissional e econômico. Estima-se que a depressão e a ansiedade sejam responsáveis por uma perda anual de produtividade de mais de um trilhão de dólares [45].

No Brasil, o estudo da OMS de 2017, revela que há 11,5 milhões de pessoas diagnosticadas com distúrbios depressivos, tornando-se o segundo país do continente americano com o maior número de casos, perdendo somente para os Estados Unidos da América [42]. Além disso, segundo a revista Ela[1] em 2019, o Brasil registrou o maior índice de ansiedade ao ser comparado aos outros países, com 18,6 milhões de brasileiros com algum tipo de transtorno de ansiedade.

3.1.1 Depressão

A depressão é considerada um distúrbio psicológico de natureza heterogênea com vários subtipos e sintomas diferentes, entretanto, a característica comum entre eles é o sentimento de humor triste, vazio ou irritabilidade acompanhada de alterações somáticas e cognitivas que afetam significativamente a capacidade de funcionamento do indivíduo. Essa doença pode ser categorizada em três níveis, leve, moderado e grave [6].

Um dos subtipos de transtornos depressivos é o Transtorno Depressivo Maior (TDM) diagnosticado em pacientes que apresentam quase diariamente episódios distintos de pelo menos 2 semanas de duração, causando prejuízo ou sofrimento no cotidiano. Para

[1] https://www.febrasgo.org.br/pt/revistas/revistaela/item/1154-ela-ansiedade

serem diagnosticados é necessário apresentar pelo menos cinco dos seguintes sintomas [6]:

1. Humor deprimido;
2. Perda de interesse;
3. Mudança no apetite ou peso;
4. Distúrbio do sono (insônia ou hipersonia);
5. Agitação ou deficiência psicomotora;
6. Fadiga ou perda de energia;
7. Sentimentos de inutilidade ou culpa excessiva, ou inapropriada;
8. Falta de concentração;
9. Pensamentos recorrentes de morte, ideação suicida ou tentativas de suicídio.

Caso esses sintomas se manifestem por no mínimo dois anos, esses indivíduos recebem um novo diagnóstico denominado Transtorno Depressivo Persistente [6].

3.1.2 Ansiedade

Segundo Darwin, ao longo de várias gerações, os nossos antepassados precisavam se defender, para garantir a preservação da espécie. Nessas situações de perigo, era necessário dispor de um grande esforço físico, que ocasionava um aumento no ritmo cardíaco, a respiração acelerava, o peito arqueava e as narinas dilatavam-se. No decorrer da evolução e da força hereditária, essas alterações fisiológicas, permaneceram nos nossos instintos e elas tendem há ressurgir toda vez que uma emoção de medo é fortemente sentida [18].

Existem evidências científicas que todas as pessoas possuem um nível de ansiedade e as causas estão relacionadas a fatores genéticos e ambientais. Estima-se que o fator genético influencia cerca de 20 a 40% da predisposição a um transtorno de ansiedade e 60% ou mais estão relacionados aos fatores ambientais [38].

A ansiedade é considerada uma resposta do nosso organismo relacionado ao sentimento de apreensão e aumento de vigilância em situações de ameaças ou perigo incerto [21]. Essa resposta torna-se patológico quando a frequência e a intensidade dessa emoção compromete a qualidade de vida do indivíduo, influenciando, os seus sentimentos e comportamentos [6]. Os sintomas dessa doença variam entre físico, emocional, comportamental e cognitivo [14]. A seguir é possível visualizar a tabela com esses sintomas:

Tabela 3.1: Sintomas de depressão.

Físicos	**Cognitivos**
Frequência cardíaca aumentada	Medo de perder o controle
Palpitações	Medo de ser incapaz de enfrentar
Sensação de desmaio	Hipervigilância para perigo
Falta de ar, respiração rápida	Medo de dano físico ou de morte
Náusea, estômago embrulhado, diarreia	Medo de “enlouquecer”
Dor ou pressão no peito	Medo de avaliação negativa dos outros
Sensação de asfixia	Pensamentos ou lembranças assustadoras
Vertigem, tontura	Percepções de irrealidade ou alheamento
Sudorese, calores, calafrios	Baixa concentração
Tremores, estremecimento	Estreitamento da atenção
Formigamento ou dormência nos braços, pernas	Memória fraca
Fraqueza, desequilíbrio	Dificuldade de raciocínio
Tensão muscular, rigidez	perda de objetividade
Boca seca	Confusão, distraibilidade

Tabela 3.2: Sintomas de depressão.

Comportamentais	**Emocionais**
Fuga, escape	Sentir-se nervoso, tenso, irritado
Busca de segurança, reasseguramento	Sentir-se assustado, temeroso, aterrorizado
Desassossego, agitação	Ser irascível, apreensivo, alvoroçado
Hiperventilação	Ser impaciente, frustrado
Paralisia, imobilidade	
Dificuldade para falar	
Andar nervosamente de um lado para o outro	

3.1.3 A ansiedade e depressão nas redes sociais

A depressão é uma das doenças que mais impossibilitam pessoas ao redor do mundo [53], pois, trata-se de um transtorno que tende a perdurar ao longo da vida. Embora existam tratamentos eficazes, uma pesquisa publicada no ano 2021 pelo instituto de estudos para políticas de saúde[2], informa que em 2019, 71,2% dos indivíduos com sintomas depressivos, não recebiam nenhum tipo de tratamento. Entre o período de 2013 a 2019, houve um crescimento de 36,7% de prevalência de depressão na população brasileira e os grupos mais propensos a está condição são os de baixa renda.

[2]https://ieps.org.br/wp-content/uploads/2021/10/Olhar$_I EPS_0$3.pdf

Esses altos índices podem ser explicados por várias razões, entre as quais, podemos citar a falta de recursos financeiros, profissionais qualificados, avaliação médica imprecisa e ao estigma social [1], onde as pessoas relutam em discutir sobre a depressão e ansiedade e muitas vezes não buscam tratamentos adequados.

Neste contexto, é de suma importância criar medidas preventivas que detectam esses distúrbios na fase inicial, para evitar o sofrimento dessas pessoas, garantindo que estas doenças não evoluam para condições extremas, onde levam ao suicídio.

Portanto, há pesquisas que estudam formas para detectar a linguagem depressiva em texto publicados em redes sociais. Essas pesquisas, apontam três características linguísticas marcantes em publicações de usuários classificados com depressão, que seriam:

- Uso excessivo do pronome de primeira pessoa;[13]
- Publicações com sentimentos negativos;
- Publicações de usuários depressivos possuem um fluxo maior durante a noite.[2]

Desta forma, esses dados compartilhados trazem informações valiosas que podem ser utilizadas para tentar reconhecer aspectos emocionais que indiquem transtornos de ansiedade e depressão em postagens do Twitter.

Para tal o trabalho seguirá o estudo de ferramentes de classificação automática em aprendizado supervisionado.

3.2 Classificação automática

Com a popularidade das redes sociais, há um excesso de dados sendo produzidos e armazenados diariamente através da internet [5]. A tendência é que esses dados cresçam exponencialmente, tornando-se humanamente impossível extrair informações sem a contribuição de tecnologias, devido ao grande volume. Portanto, técnicas computacionais de classificações automáticas tem se tornado cada vez mais importantes para instituições acadêmicas e organizacionais que pretendem utilizar esses dados para entender comportamentos sem a necessidade de visualizar uma grande quantidade de dados [5].

Com base nesse cenário, os profissionais que trabalham com *data science* desenvolveram ferramentas para obter informações e *insights* a partir desses dados. Uma dessas ferramentas é realizada através do processamento de linguagem natural (PNL), chamado análise de sentimento, que nada mais é que um estudo computacional que extrai a opinião e a emoção de uma frase ou um documento [5]. Essa análise pode ser realizadas em três níveis de granularidade.

- Documento: Este nível, assume que todo o texto (documento) expressa uma opinião geral sobre uma entidade. Um exemplo de aplicação dessa técnica é utilizada em

reviews de produtos ou filmes que classificam a polaridade da opinião como positivo ou negativo [29] [8].

- Sentença: Neste nível, cada frase pode expressar uma opinião a respeito de vários aspectos de uma entidade. Portanto, cada frase é analisada separadamente, com base na classificação da polaridade positivo, negativo ou neutro. Um exemplo de aplicação dessa técnica é utilizada no monitoramento do Twitter para previsão de bolsa de valores [29] [8].
- Entidades e aspectos: Neste nível, é realizado uma análise mais detalhada sobre a opinião do texto. Essas opiniões consistem em sentimentos (positivo ou negativo) e entidades alvo da opinião. Cada sentença pode ter várias entidades e aspectos associados a elas [29] [8].

A classificação automática de texto pode ser realizada por algoritmos de aprendizado de máquinas que extraem padrões a partir de conjuntos de dados para realizar inferências. A maioria desses algoritmos podem ser divididos nas categorias de aprendizado supervisionado e não supervisionado:

Aprendizado Supervisionado O aprendizado supervisionado é conceituado como um professor ou supervisor que apresenta um conjunto de dados [9] para treinamento, organizado por uma coleção de pares de entrada e saída esperada. Seu objetivo é criar um aluno capaz de receber um vetor de características "x" como entrada e deduzir a qual saída "y" a instância pertence [10].

Dependendo do problema, os algoritmos podem ser sub categorizados em dois tipos [26]:

- **Regressão**: Neste modelo, os algoritmos procuram uma função $r(x) := \mathbb{E}[Y|X = x]$ capaz de estimar uma resposta quantificativa "Y" para as covariáveis "X".
- **Classificação**: Ao contrário do modelo anterior, este possui o propósito de criar uma função capaz de estimar uma variável qualitativa.

Em ambos os casos, o objetivo do modelo é aprender uma aproximação da função desconhecida f(x) que estime o valor de f para novas observações de x. Dada essa aproximação, podem existir erros relacionados às estimativas, por isso, funções de custos são utilizadas para medir a qualidade de um modelo preditivo.

Aprendizado Não Supervisionado Neste aprendizado, o modelo é baseado na ausência de qualquer supervisor [9]. Portanto, os dados de entrada não são rotulados, pois, não se conhece a classe a qual a instância pertence. Desta forma, o algoritmo deve aprender sozinho o relacionamento e as características subjacentes dos dados disponíveis para poder categoriza-lós.

Dentre os algoritmos de aprendizado não supervisionado, podemos citar:

- *Clustering*: *k-Means*, *Hierarchical clustering*, Maximização da Expectativa.
- Visualização e redução da dimensionalidade: *Locally-Linear Embedding* (LLE), *t-distributed Stochastic Neighbor Embedding* (t-SNE),*Principal Component Analysis* (PCA), *Kernel* PCA,
- Aprendizado da regra da associação: Apriori, Eclat.

3.3 Algoritmos de Aprendizado Supervisionado

Os algoritmos de aprendizado supervisionado discriminados nesta seção serão implementados, testados e avaliados no experimento que será descrito no capítulo seguinte.

3.3.1 *Random Forest*

É um conjunto de classificadores (*ensemble*) representado por um vetor aleatório p-dimensional $(X_1, ..., X_P)^T$ que combina diversas árvores de decisões com o intuito de melhorar seu desempenho através de amostras *bootstraps* [17]. Essas amostras são um subconjunto dos dados originais utilizados para fornecer aleatoriedade na criação de cada árvore e suas divisões (nós) [55]. Na equação 3-1 é ilustrado a função preditora desse modelo [34]:

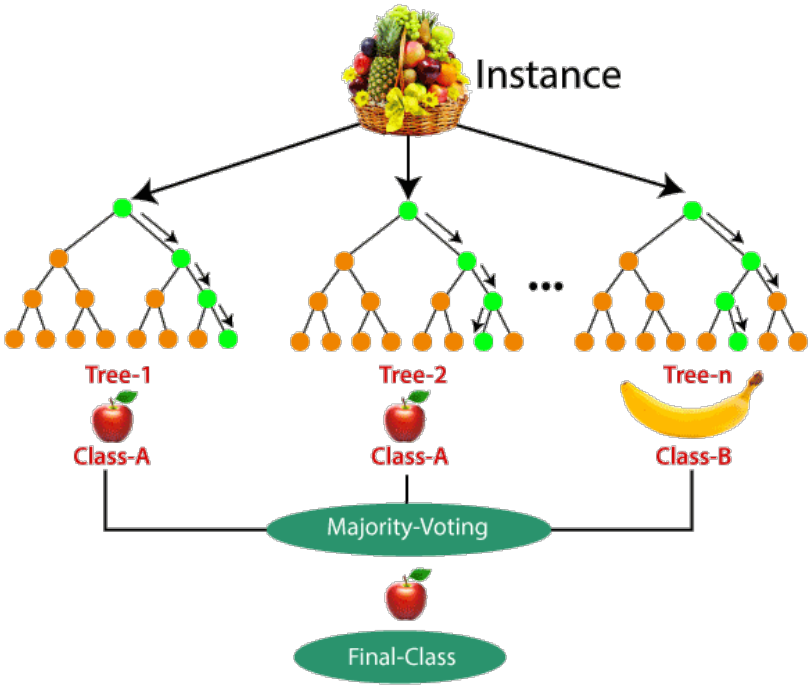
$$H(x) = arg_{max_y} \sum_{i=1}^{k} I(h_i(x) = Y) \tag{3-1}$$

Onde:

- $H(x)$ é a combinação dos classificadores independentes;
- h_i representa uma única árvore de decisão;
- Y é a variável de saída dessas árvores;
- $I()$ é uma função indicadora a qual determina se o elemento pertence ao conjunto;

A Figura 3.1 representa o funcionamento desse classificador, na qual as instâncias (cesta de frutas) são apresentadas ao modelo e o algoritmo cria de forma aleatória K arvores de decisões das amostras de dados, selecionando o melhor nó. A predição deste modelo, pode ser realizada tanto para problemas de regressão ou de classificação, entretanto nesta imagem, é apresentado um problema de classificação. Desta forma, o algoritmo soma todos os votos (fruta) das árvores de decisões, a classe (fruta) que tiver mais votos é eleita a classe vencedora.

Figura 3.1: Esquema de funcionamento do *Random Forest*.



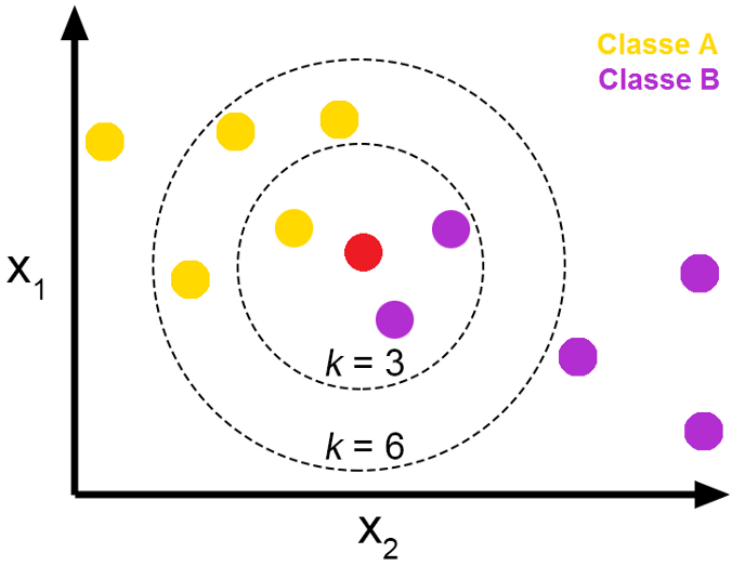
Fonte: (JAVATPOINT, 2021, p.5) - Disponível em:
https://www.javatpoint.com/machine-learning-random-forest-algorithm

3.3.2 K-Nearest Neighbours

É um método de classificação simples [27], bastante utilizado, que assume a existência da proximidade de objetos semelhantes. Esse algoritmo é considerado um típico aluno preguiçoso (*lazy*), visto que não assume premissas sobre os dados de treinamento, apenas os memoriza [22, 19]. No momento em que os dados de teste são apresentados, o modelo analisa a similaridade entre o novo elemento e os dados de treinamento [22]. Para isso ele calcula uma métrica de distância entre os pontos, identificando os vizinhos mais próximos e atribuindo a mesma rotulação dos K semelhantes [44].

Na imagem 3.2 é ilustrado um exemplo de funcionamento do algoritmo *K-Nearest Neighbours*. Nesta, o ponto vermelho representa a instância que deverá ser classificada e ao redor dela estão os dados de treinamento. Ao definir a variável K = 3 o classificador irá prever a instância como B, entretanto, ao modificar a variável K = 6 o modelo classificará a classe como A, visto que os vizinhos mais próximos de "K" pertencem a esta este grupo.

Figura 3.2: Esquema de funcionamento do algoritmo *k-NN*.



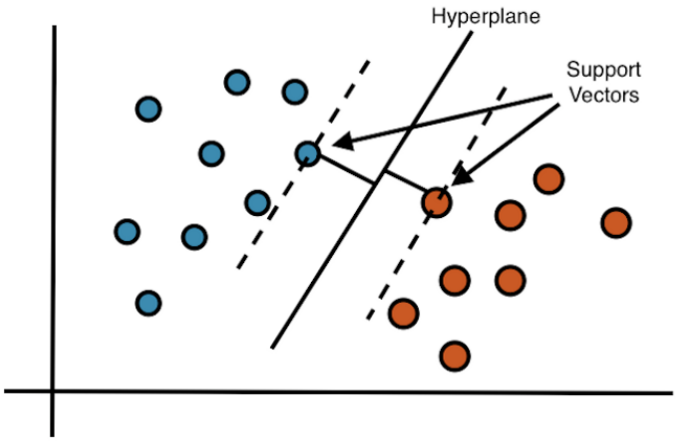
Fonte: (SURATKAR, 2020, p.3)
Disponível em: https://www.irjet.net/archives/V7/i4/IRJET-V7I4985.pdf

3.3.3 *Support Vector Machine* (SVM)

Trata-se de um modelo capaz de resolver problemas de classificação linear/não linear, regressão e detecção de *outliers*. Diferente de outros algoritmos, o SVM não utiliza a probabilidade $P(Y = c|x)$ para estimar as classes, sua técnica é baseada na indução de novas observações sobre o conjunto. Dado um conjunto de treinamento com "N" amostras $(X1, Y1), ..., (Xn, Yn)$ o Yi pode ter um valor de $f(x) < 0, g(x) = -1$ ou $f(x) \geq 0, g(x) = 1$ representando o rótulo da classe a qual o vetor X_i pertence.

A Figura 3.3 apresenta o funcionamento deste modelo. Para que esse algoritmo realize as classificações, cada instância é plotada como um ponto no espaço. Isso permite que os dados se agrupem de alguma forma. Se objetivo é identificar o melhor hiperplano para separar linearmente esse conjunto. Para isso, ele utiliza vetores de suporte para calcular a margem do hiperplano com as instâncias mais perto dessa reta.

Figura 3.3: Esquema de funcionamento do algoritmo *Support Vector Machine.*



Fonte: (ICHI.PRO, 2022, p.2)
Disponível em:
https://ichi.pro/pt/maquina-de-vetores-de-suporte-svm-explicada-97743104690915

3.3.4 *Naïve Bayes*

É um algoritmo baseado no teorema de Bayes, ele determina a probabilidade de um evento dado o conhecimento de sua ocorrência. Esse tipo de classificador é considerado ingênuo, pois pressupõe que as variáveis são independentes. Isso significa que ele ignora a relação entre as *features* portanto, não influencia nos valores de cada recurso [25]. Na equação 3-2 são ilustrados os termos matemáticos do Teorema de Bayes:

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} \tag{3-2}$$

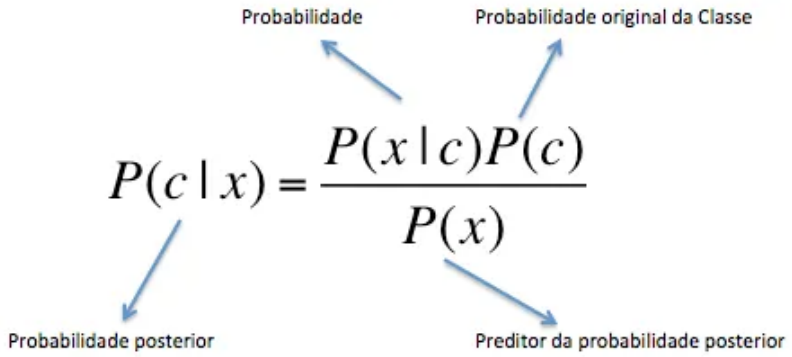
Onde A e B são eventos:

- $P(A)$ e $P(B)$ são probabilidades *priori*(anteriores) do evento A e evento B terem ocorrido, independentemente uma da outra. Para calcular essa probabilidade, basta dividir a quantidade de ocorrências de cada evento pelo total dos exemplos.
- $P(B|A)$ chamado probabilidade condicional, é a probabilidade de observar o evento B dado que A é verdadeiro.
- $P(A|B)$ chamado probabilidade posterior, é a probabilidade de observar o evento A dado que B é verdadeiro, esse é o evento que o estimador realmente quer saber.

Na Figura 3.4, é mostrado os eventos C e X. Na probabilidade posterior, responde à pergunta: “Qual a probabilidade de C ocorrer se X já ocorreu?”. No preditor da probabilidade, responde "a perguntar qual a probabilidade de X ocorrer se C já ocorreu?”.

Na probabilidade original e probabilidade posterior identifica qual a chance do evento ocorrer sozinho.

Figura 3.4: Esquema de funcionamento do *Naïve Bayes*.

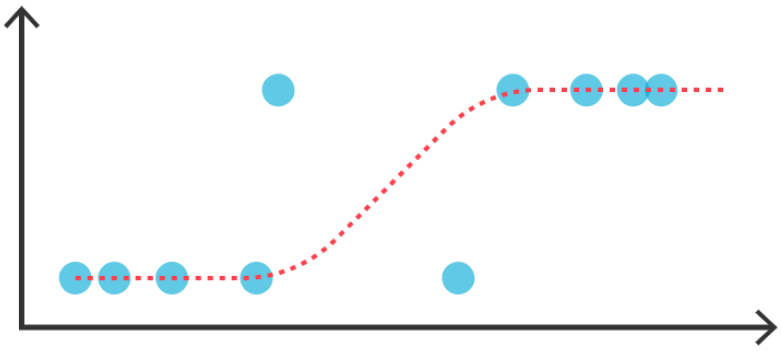


Fonte: (SUNIL RAY, 2022, p.3)
Disponível em: https://www.vooo.pro/insights/6-passos-faceis-para-aprender-o-algoritmo-naive-bayes-com-o-codigo-em-python/

3.3.5 *Logistic Regression*

O modelo de regressão logística estuda a probabilidade da ocorrência de um evento definido por Y, sendo Y apresentado na forma qualitativa dicotômica [24]. Esse algoritmo, possui outras denominações como modelo logístico, modelo logit ou classificador de máxima entropia. A diferença deste algoritmo para a regressão linear, é que os valores não são contínuos, ou seja, eles nunca iram ultrapassar os valores de 0 e 1. Para construir sua função, ele utiliza uma curva em formato de "S". Na figura 3.5 é apresentado sua modo de aprendizado.

Figura 3.5: Esquema de funcionamento do *Naïve Bayes*.



Fonte: (TIBCO, 2022, p.1)
Disponível em:
https://www.tibco.com/pt-br/reference-center/what-is-logistic-regression

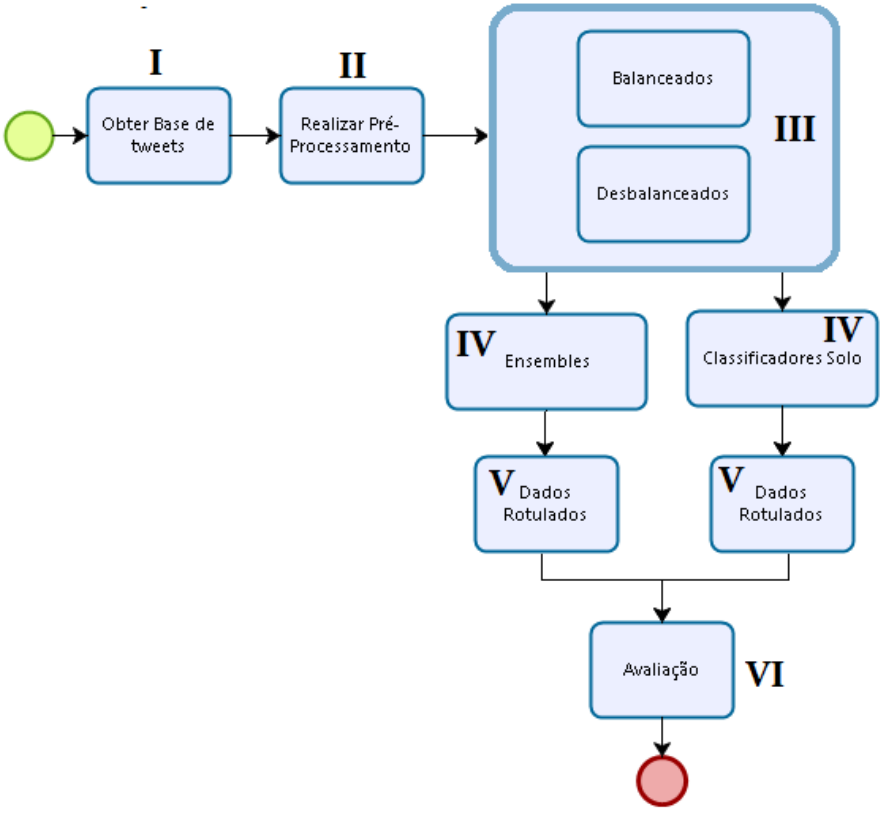
CAPÍTULO 4

Experimento

4.1 Desenho do Experimento

O presente capítulo tem como proposito, descrever o desenho do experimento ilustrado na Figura 4.1. Nesta imagem é apresentado todos os estágios implementados durante a reprodução do estudo dos pesquisadores Fauzi e Yuniarti [23] para este projeto de conclusão de curso, que se deu através da implementação dos modelos de classificadores para identificar trações de ansiedade e depressão em *tweets* de língua portuguesa. Suas etapas consistem na obtenção da base de *tweets* (1), pré-processamento (2), treinamento dos classificadores independentes (3), implementação dos *ensembles* (4) e avaliação dos modelos (5). Nas próximas subseções, cada uma dessas atividades serão detalhadas.

Figura 4.1: Desenho do Experimento.



4.1.1 Base de *tweets*

O conjunto de dados utilizado neste experimento foi obtido através do trabalho de conclusão de curso realizado em 2019 no Instituto de Informática da UFG. Nele, a pesquisadora Rodrigues [48] desenvolveu regras ontológicas de domínio para serem aplicados em um sistema de análise de sentimento para detectar traços de depressão.

Em seu trabalho, foram realizados estudos bibliográficos e entrevistas com profissionais da área de psicologia, para compreender o universo de palavras da linguagem depressiva. A partir de suas observações, a autora chegou a conclusão de que a depressão poderia ser classificada em três níveis: leve, moderada ou grave e, além disso, ela também identificou que pessoas depressivas tinham tendências a expressar-se de maneira negativa e utilizavam com frequência pronomes de primeira pessoa, como: eu, eu mesma e mim. Em 4.2 alguns exemplos de palavras associadas há graus de depressão. Para a realização de seu experimento, foram coletadas por três meses 217.606 mil *tweets* com menções às palavras associadas à linguagem depressivas utilizadas como filtro para coleta.

Figura 4.2: Classificações e termos sugeridos para a coleta de dados.

Profunda	Moderada	Leve
sofrer	não gosta de mim	estresse com pessoas
suicídio	vida	tempo desperdiçado
auto-estima baixa	sem amor próprio	se dar valor
sem vontade de viver	estou triste	
pensar em suicídio	insegura	
vacilar comigo	depressiva	
tudo errado	vivendo triste	
cansada de existir	solidão	
frustrado		
quero morrer		
vida sem sentido		
dificuldade de suportar		
sentindo mal		
depressão frescura		
ansiedade		
me cortar		

Fonte: RODRIGUES, 2019, p.39

Dando continuidade ao trabalho realizado por Rodrigues[48], Corrêa[16] avaliou o desempenho de um classificador ao usar um sistema de ontologia para detectar depressão e ansiedade. Para tal, realizou modificações nas regras ontológicas e testou algoritmos classificadores sobre uma base filtrada através da ontologia para verificar se haveria algum ganho no uso de ontologia com análise de sentimentos.

Para o experimento a ser relatado neste projeto final de curso, foram utilizados os dados de treinamento para classificadores explorados por Corrêa [16] em seu trabalho.

Na tabela 4.1.1 é apresentado três amostras de cada classe recolhidas manualmente da base de treinamento.

Tabela 4.1: Exemplos de *tweets* da base de treinamento.

Tweet	**Rótulos**
que perfume e esse que voce passa e fica com cheiro de amor da minha vida	0
grato por mais um dia de vida e mais uma oportunidade de fazer a diferenca	0
e tao bom ficar feliz pela felicidade dos outros	0
que inferno eu quero morrer	1
deitada morrendo de dor e sem expectativa de vida	1
nao acredito que vou ter que reviver a cena do dia mais triste da minha vida	1

Para a implementação deste projeto, foi escolhido um aplicativo *desktop* conhecido como *Orange Data Mining*[1]. Esse aplicativo foi desenvolvido em conjunto com a comunidade *open source* e algumas instituições de ensino.

Disponibilizado somente na língua inglesa, o *orange data mining*, é uma ferramenta simples e intuitiva que permite criar um *workflow* a partir de uma programação visual, onde vários *widget* são conectados para criar um projeto de mineração de dados.

No escopo deste trabalho, essa ferramenta viabilizou o desenvolvimento das etapas de pré-processamento (3), treinamento de classificadores (4) e avaliação dos modelos (6), ilustrados na imagem 4.1.

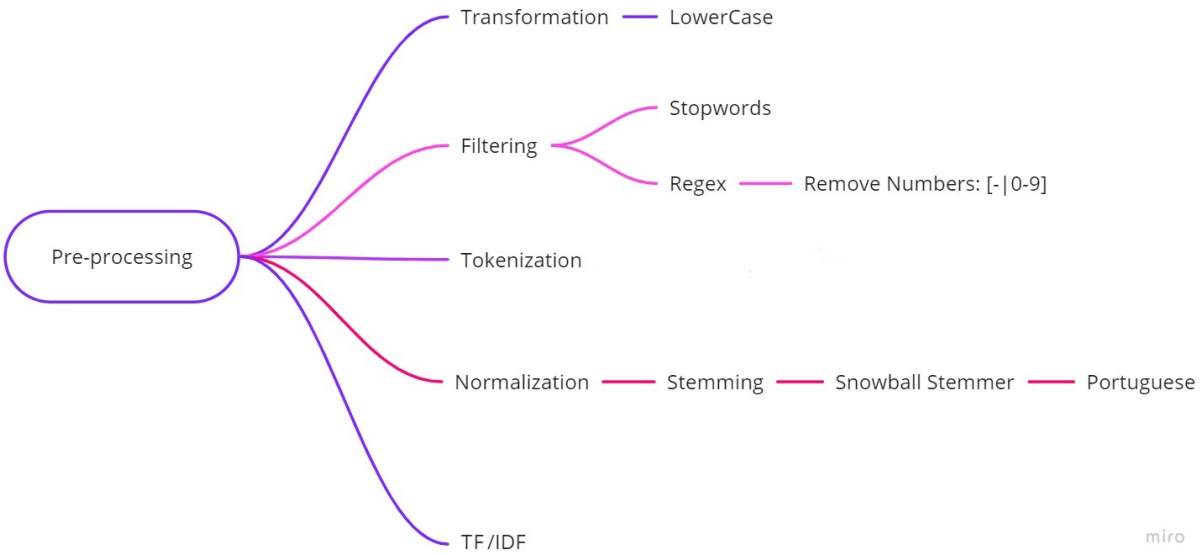
4.1.2 Pré-processamento de Dados

A etapa de pré-processamento é considerado uma das mais onerosas e importantes da mineração de texto, ela pode ser responsável tanto pelo sucesso quanto o fracasso de um projeto [63]. Seu objetivo consiste basicamente em utilizar várias técnicas para melhorar a qualidade dos dados. Essas técnicas consistem entre a limpeza dos dados, transformação, seleção e integração, podendo variar em cada projeto.

Nesta etapa do trabalho, os dados de treinamento citados na subseção anterior 4.1.1, foram submetidos as mesmas rotinas de pré-processamento utilizados na pesquisa de Fauzi [23], essas rotinas são ilustradas na Figura 4.3.

[1] https://orangedatamining.com/

Figura 4.3: Etapas de Pré-processamento.



Etapas de pré-processamento:

1. Transformação: Para evitar que o modelo diferencie as palavras em minúsculas e maiúsculas, aumentando o espaço vetorial sem necessidade, todas as letras foram convertidas para minúsculas (*lower case*).
2. Remoção de *stopwords*: Com o intuito de remover os ruídos menos evidentes, foi criado uma lista de palavras denominadas *stopwords*. Trata-se de um conjunto que contém as palavras mais irrelevantes para o contexto analisado [54], e são geralmente compostas por artigos, preposições, conjunções, pronomes e advérbios [20]. Neste trabalho a lista, foi selecionada ao utilizar um *widget* da ferramenta *orange data mining* para *ranquear* as *features* que possuíam menor ganho de informação. Além desse processo, foi utilizado uma representação visual das palavras mais frequentes (*cloud words*) para analisar e refinar ainda mais essa lista. A Figura 4.4 apresenta alguns termos inseridos na lista de *stopwords* utilizada para remover palavras consideradas ruidosas para o contexto deste trabalho.
3. Filtragem: Para evitar erros de classificação, os números presentes na base de dados, foram retirados.
4. Tokenização (*tokenization*): foi realizada nos *tweets* para quebrá-los em unidades menores chamadas *tokens* [4]. Esses *tokens* foram definidos através de uma expressão regular padrão \w+ que corresponde a um ou mais caracteres de palavras.
5. *Stemming*: Outro passo importante para a remoção de ruídos, foi a

utilização da técnica de *stemming* que reduzir as palavras para o seu radical [4, 54]. O algoritmo utilizado foi o *snowball stemmer* para a língua portuguesa. A Figura 4.5 mostra um exemplo do emprego de *stemming*, neste, o algoritmo reduziu o radical das palavras "casa", "casinha" e "casebre" para uma raiz comum "cas", visto que todos esses termos possuem o mesmo significado.

6. *TF-IDF*: Calcula a frequência do termo para medir quantas vezes uma palavra está presente em um documento e atribui um menor peso nas palavras mais frequentes. Essa técnica visa encontrar os termos chaves do *corpus*, criado um vetor esparso contendo todas as palavras (*features*) e seus valores correspondendo sua importância.

Figura 4.4: Exemplos de *stopwords* adotadas para o contexto do trabalho.

academia	cara	crianca	dona	fazendo	literalmente
passou	caramba	parar	dormi	fazer	alguns
acaso	casa	curtindo	engracado	fica	logo
acontecendo	caso	daqui	entendesse	filha	lugar
aconteceu	causa	dele	escolher	fisica	maravilha
acordei	cedo	deles	escrever	foda	medida
acredito	certeza	demais	escrevi	for	outro
ainda	certo	dentro	escutei	forcas	meio
ajuda	chamou	depois	estomago	realmente	menina
amanha	chega	passando	estudar	foto	menos
apenas	chegando	descobri	ponto	grande	mesma
aproveita	nessa	notas	expressao	grandes	mesmo
areas	colocar	dessa	extremamente	segunda	morar
assistir	comida	desse	faca	jeito	muita
aula	completamente	deve	faco	juro	muitas
aulas	conheco	diante	fala	lado	municipio
auto	conseguido	diferenca	falei	lavar	nasci
bonita	contato	direito	falou	lembrados	negro
calma	continuar	disso	fases	levar	nela
caminho	coracao	dizer	fazem	liga	nele

Figura 4.5: Exemplos de uso da técnica de pré-processamento *stemming*.

Suponhamos que temos um *corpus* com apenas dois documentos, sendo, que o conteúdo do documento 1 é “Se existe vida vamos celebrar” e o documento 2 é “vamos celebrar”. O cálculo do TF para o termo “celebrar” é realizado da seguinte forma:

1. TF(“celebrar”, documento 1) = $\frac{1}{5} = 0,2$
2. TF(“celebrar”, documento 2) = $\frac{1}{2} = 0,5$

Para medir a importância do termo no *corpus*, é realizado o cálculo do IDF da seguinte forma:

$$IDF(\text{"}celebrar\text{"}, Documentos) = \log \frac{2}{2} = 0 \tag{4-1}$$

Após essas estimativas é possível calcular o TF-IDF e perceber que este termo não é muito informativo, pois seus resultados são frequentes em todo o *corpus*:

1. TF-IDF("celebrar", documento 1) = $0.2 \cdot 0 = 0$
2. TF-IDF("celebrar", documento 2) = $0.5 \cdot 0 = 0$

4.1.3 Dados de Treinamento e balanceamento

Neste estágio do projeto, todos os algoritmos supervisionados adotados no artigo *baseline* [23] foram implementados, entretanto, é importante salientar que o classificador *Maximum Entropy* é destinado a problemas de multi classes, por conta disso, esse modelo foi substituído pelo seu equivalente em problemas binários *Logist Regression* [36].

Para avaliar a capacidade dos classificadores trabalhando em formato solo e em *ensemble*, o experimento foi realizado utilizando uma base de treinamento balanceada e outra desbalanceada.

- Dados balanceados: é representado por um *data set* que possui a mesma quantidade de instâncias para cada uma das classes adotadas no trabalho. No total foram utilizados 400 *tweets* para cada uma das classes, uma com traços e outra sem traços de ansiedade/depressão.
- Dados desbalanceados: São formados por *datasets* que não possuem uma proporção igual para todas as classes. Esse é o tipo de base que mais representa o domínio trabalho proposto, em razão a menor quantidade de *tweets* de usuários depressivos. Para desbalancear os dados, somente foi utilizado a quantidade de 60% dos *tweets* que representavam a classe 1 “com traços”.

Para a reprodução dos experimentos de Fauzi [23] no contexto de ansiedade e depressão, os dados de treinamento foram projetados para experimentos de avaliação em 70/30, ou seja, 70% para treinamento e 30% para teste e avaliação dos dados de classificação supervisionada *K Nearest Neighbor*, *Logistic Regression*, *Naïve Bayes*,

Random Forest e *Support Vector Machine* (descritos na seção 3.3) que foram utilizados no experimento.

A tabela 4.2 ilustra os dados estatísticos utilizado no treinamento de cada experimento. Para a base balanceada, os algoritmos foram treinados com 560 *tweets*, 285 sem traços e 275 sem traços, ou seja, 70% do total de 800 mensagens. Foram adotadas 1405 *features* criadas a partir da vetorização TF-IDF. Já nos dados desbalanceados, os classificadores foram treinados com 448 amostras + 1405 *features*.

Tabela 4.2: Estatística dos dados de treinamento.

Dados	**Classe**	**Feature**	**Treino**	**Total Treino**
Balanceados	0	1405	285	560
	1		275	
Desbalanceados	0	1405	286	448
	1		162	

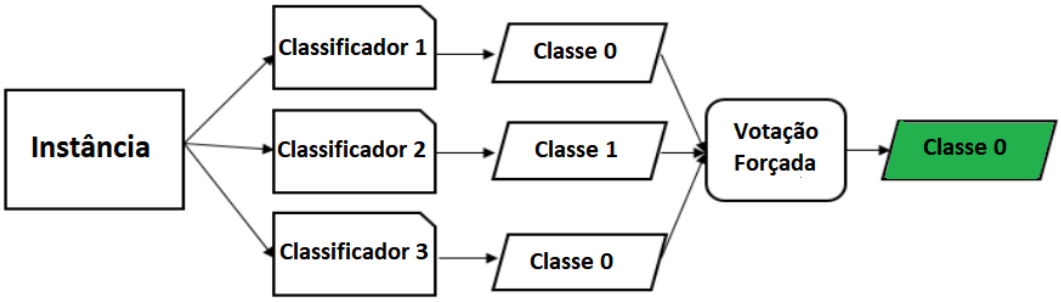
4.1.4 Modelos *Ensemble*

Prosseguindo no objetivo de reproduzir os experimentos realizados por [23] para o contexto de classificação de *tweets* em língua portuguesa com e sem traços de ansiedade e depressão, foram implementados dois modelos *ensemble*. No primeiro, adotou-se a combinação de todos os algoritmos de classificadores (*K Nearest Neighbor*, *Logistic Regression*, *Naïve Bayes*, *Random Forest* e *Support Vector Machine*) comentados na seção anterior 4.1.3. No segundo, realizou-se a junção dos três algoritmos que obtiveram o melhor desempenho na métrica *recall*. Esta métrica foi escolhida, pois, para a nossa base os valores de *false negativos* é mais nocivo, ou seja, é melhor classificar tweets de usuários saudáveis com depressão ou ansiedade do que deixar de classificar um *tweet* que realmente possui esses transtornos.

Para cada combinação de *ensemble* foram implementados dois tipos de votação:

1. Método *hard voting*: Na Figura 4.6 e ilustrado o modelo de *hard voting*, neste tipo de votação cada um dos classificadores (independentes) identifica uma classe provável para o *tweet*. A partir da soma do resultado de cada algoritmo, a classe que obtiver maior pontuação será considerada na saída do *ensemble*.

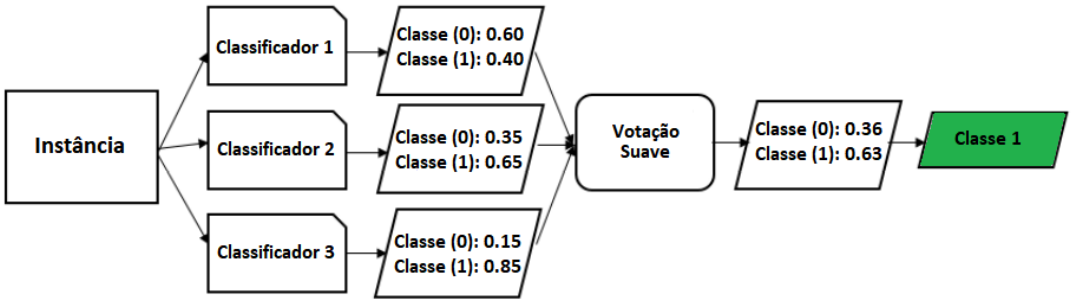
Figura 4.6: Exemplo da votação forçada com três classificadores.



Fonte: Imagem modificada do artigo *Ensemble method for indonesian twitter hate speech detection* [23]

2. Método *soft voting*: Neste regime, cada classificador solo atribuí a probabilidade de um *tweet* pertencer a uma determinada classe. A partir da média dos algoritmos, a classe que obtiver a maior pontuação será considerada a previsão final do modelo.

Figura 4.7: Exemplo da votação suave com três classificadores.



Fonte: Imagem modificada do artigo *Ensemble method for indonesian twitter hate speech detection* [23]

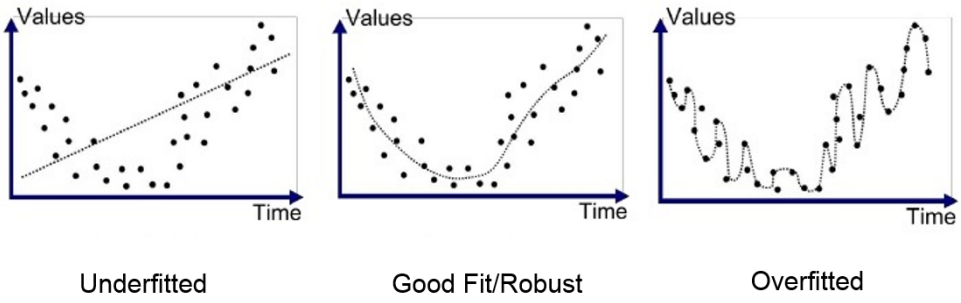
4.1.5 Avaliação dos modelos

Os métodos de aprendizado supervisionado devem ser avaliados quantitativamente para mensurar a capacidade de generalização dos dados não vistos. Dentre os algoritmos de classificação, existem duas grandes causas de má generalização, o *overfitting* e *underfitting*.

- *Overfitting*: Ocorre quando o modelo aprende demasiadamente sobre os dados de treinamento e por conta disso possui um ótimo desempenho com esses dados. Entretanto, quando novos dados são apresentados ao modelo, ele não consegue boa generalização, ou seja, seu desempenho para a classificação não é bom.
- *Underfitting*: Ocorre quando o algoritmo não aprende a relação entre os dados, ou seja, não funciona no treinamento e também na avaliação. Isso pode acontecer devido à falta de treinamento ou ao uso de variáveis pouco significativas para o modelo [7].

A Figura 4.8 ilustra o conceito dessas más generalizações, onde os classificadores representam suas curvas de aprendizado. Na primeira imagem, é possível observar um modelo com *underfitting*, onde ele não consegue compreender os padrões do problema e por conta disso produz um péssimo desempenho nos dados de treino, caso seja testado, reproduzirá o mesmo resultado. Na terceira imagem da Figura 4.8 é apresentado um exemplo de *overfitting*, nesta má generalização o classificador memoriza os dados de treinamento, atingindo uma boa *performance*, entretanto ao ser avaliado, não produzirá boas predições em novas observações. Na segunda imagem da Figura 4.8, é exposto o exemplo de um classificador que realmente aprendeu os padrões de entrada e consequentemente obteve um desempenho satisfatório.

Figura 4.8: Modelo com *overfitting* e *underfitting*.[2]



Fonte: (ABRACD, 2020, p.3)
Disponível em: https://abracd.org/overfitting-e-underfitting-em-machine-learning/

Para que esses problemas não ocorram, é importante avaliar os algoritmos antes de coloca-los em produção. Atualmente existem diferentes métricas capazes de avaliar o desempenho de cada modelo, entretanto nenhum delas é, capaz de medir todos os aspectos de interesse. Portanto, é necessário escolher o indicador que mais se adéque ao problema proposto. Nesta seção discutiremos as medidas de desempenho que extraem informações a partir das matrizes de confusões [58].

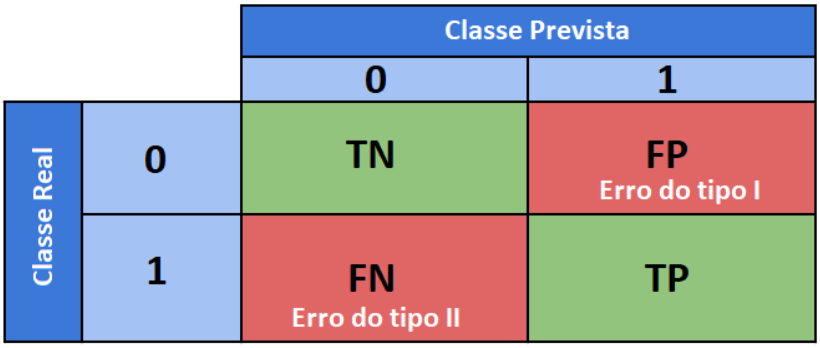
Matriz de confusão

É uma tabela de contingência bidimensional, representada pelo eixo dos rótulos previstos e o eixo dos rótulos reais. Em suma, ela apresenta a distribuição de todos os erros e acertos de um classificador, permitindo visualizar de forma simples e objetiva a confusão do modelo avaliado [59].

A imagem 4.9 ilustra a disposição dos elementos de uma matriz em uma classificação binária. Nessa Figura, a diagonal pintada pela cor verde representa os acertos do modelo, os valores que estiverem fora desse faixa, são considerados os erros. Apesar

de demonstrarmos essa matriz na forma binária, ela pode ser facilmente estendida para problemas de multi-classes.

Figura 4.9: Matriz de Confusão.



Fonte: Imagem modificada (MEDIUM, 2020, p.1)
Disponível em: https://medium.com/@mateuspdua/machine-learning-métricas-de-avaliaç~ao-acurácia-precis~ao-e-recall-d44c72307959/

No domínio do nosso problema a matriz de confusão retorna os seguintes valores:

- ***True Positive*** (TP): São as instâncias classificadas "com traços" de ansiedade e depressão e que realmente estão dentro dessa categoria.
- ***True Negative*** (TN): São as instâncias que o modelo classificou como "sem traços" e realmente estão dentro dessa categoria.
- ***False Positive*** (FP) ou **erro do Tipo I**: São as instâncias classificadas "com traços" de ansiedade e depressão e não fazem parte dessa categoria.
- ***False Negative*** (FN) ou **erro do Tipo II**: São as instâncias classificadas como "sem traços" de ansiedade e depressão e que não fazem parte dessa categoria.

Acurracy

É uma das métricas mais populadores e fáceis de serem implementadas [33, 58], ela é considera uma boa proposta para analisar o desempenho geral dos classificadores. Seu objetivo é medir a proximidade de uma estimativa com o valor verdadeiro [35]. Como se trata de uma média ponderada dos valores positivos (veja Eq. 4-2), ela é suscetível a *outliers*. Isso significa que ao trabalhar com dados desbalanceados o modelo fornecerá uma estimativa superotimista da classe majoritária [57, 35, 37, 11]. Outro problema em relação a essa medida, é que devido ao fato de ser um método de avaliação geral, ela não traz muitas informações sobre o tipo de erro que o classificador possa estar cometendo, portanto, modelos diferentes podem obter a mesma acurácia, entretanto atuam de formas distintas [58].

A acurácia é medida pela razão do número correto de predições, dividida pelo total de instâncias classificadas, conforme a equação [57] na qual TP, TN, FP e FN são respectivamente:

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{4-2}$$

Precision

Essa métrica originou-se nas avaliações médicas, ela consegue medir a proporção de todos os exemplos classificados como positivos, quantos realmente são [28, 46]. Seu objetivo é identificar e quantificar as instâncias classificadas como falso positivo, ou seja, trata-se dos exemplos que o modelo inferiu como contento traços de ansiedade e depressão quando, na verdade, não continham. Essa métrica deve ser evitada em dados desbalanceados, pois ela é sensível a eles [58].

Sua fórmula é baseada na proporção das instâncias que realmente pertencem à classe positiva dentre todas as instâncias recuperadas como positiva:

$$P = \frac{TP}{TP + FP} \tag{4-3}$$

Recall

Essa métrica possui várias denominações como *sensitivity*, taxa de verdadeiro positivo ou taxa de acerto. Ela é considerada uma medida de probabilidade condicional, na qual mede-se a taxa de acerto de um modelo através da proporção de verdadeiros positivos classificados corretamente [32, 58]. Ao contrário da *precision*, a *recall* pode ser utilizada para medir dados desbalanceados [58] e determina que os falsos negativos são mais prejudiciais ao modelo do que os falsos positivos, ou seja, nas circunstâncias do problema proposto neste trabalho, tal métrica, responde à pergunta: “De todos os *tweets* com traços de ansiedade e depressão, quantos foram detectados corretamente?”

A sua medida é calculada pela razão das previsões positivas realizadas corretamente e todas as previsões que realmente são positivas, conforme:

$$R = \frac{TP}{TP + FN} \tag{4-4}$$

F1-Score

Trata-se da média harmônica entre a *precision* e *recall* [32, 58], que calcula respectivamente a porcentagem dos positivos previstos e a porcentagem dos positivos reais. Ela foi originalmente introduzida em 1948 no domínio das estatísticas ecológicas,

porém, somente na década de 90 ganhou popularidade na comunidade de *machine learning* para as classificações binárias [58, 11].

Uma das grandes limitações presentes na *precision* e *recall* é que elas não avaliam o desempenho dos verdadeiros negativos [28], como a *F1-score* é uma extensão destas medidas, ele herdou o mesmo problema. Outro ponto negativo, desta métrica, é que ela é uma união de duas outras medidas, ela torna-se mais simples e menos informativa, devido a um único resultado [28]

A métrica *F1-Score* é calculada conforme:

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4-5}$$

Essas métricas serão utilizadas no próximo capítulo para avaliação do experimento realizado e descrito neste capítulo.

CAPÍTULO 5

Resultados

Nesta seção, serão avaliados os resultados dos classificadores independentes e *ensembles* para o contexto de detecção de *tweets* com traços de ansiedade e depressão em língua portuguesa. Este projeto utilizou um *corpus* de treinamento com 800 *tweets*, divididos aleatoriamente entre 70% das instâncias para treino e 30% para teste.

Os experimentos foram implementados através da ferramenta *Orange Data Mining* e planilha de Excel. Onde a base de teste utilizada foi composta por 240 *tweets* sendo 125 e 115 respectivamente rotulados com e sem traços de ansiedade e depressão para dados balanceados. Para os dados desbalanceados foram empregados 78 e 114 *tweets* rotulados, os primeiros com e os seguintes sem traços.

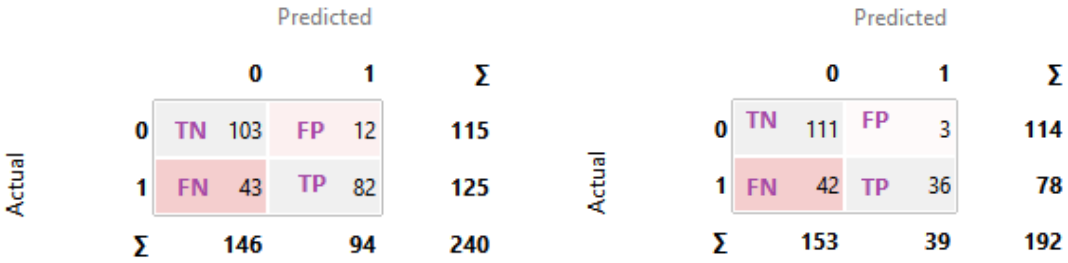
Os resultados serão expostos para os classificadores em modo solo (independentes), *ensemble* (*hard* e *soft voting*) e também *ensemble* com os três melhores classificadores (*hard* e *soft voting*) para o contexto apresentado com as bases de dados balanceada e desbalanceada. Para a análise desses modelos, serão adotadas as métricas apresentadas no capítulo 4, seção 4.1.5.

5.1 Matriz de Confusão

Durante a implementação deste trabalho, as classes “sem traços” e “com traços” de ansiedade e depressão, foram convertidas respectivamente para os algarismos 0 e 1, portanto, elas serão representadas por esses valores. Ao longo desta seção, as matrizes, ilustram as quantidades de *tweets* classificadas correta e incorretamente para os dados de testes nas bases balanceadas e desbalanceadas.

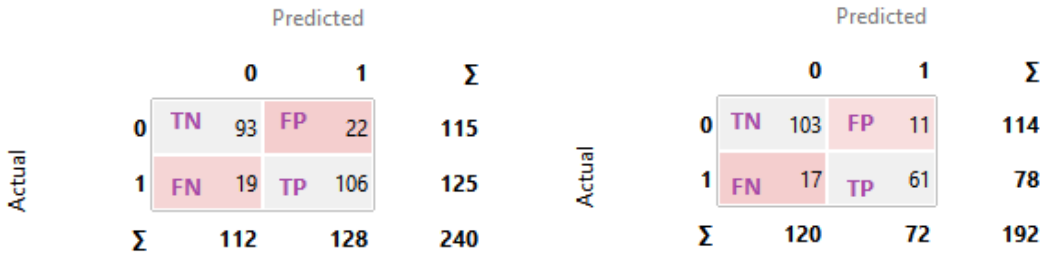
As Figuras 5.1(a) e 5.1(b) apresentam as matrizes de confusões do classificador *K Nearest Neighbor*. Nelas, em cada um dos modelos, é possível observar uma quantidade maior de acertos do tipo *True Negative*, com 103 instâncias classificadas corretamente nos dados balanceados e 111 nos desbalanceados. Em virtude da proporcionalidade inversa desses dados, houve uma menor incidência de erros do tipo I (*False Positive*), no qual o classificador atribui erroneamente a classe positiva para as instâncias negativas. Ao comparar as taxas de acertos dessas matrizes de confusões é notável que o classificador

com a base balanceada 5.1(a) obteve um melhor resultado, atingindo 77%, enquanto a base desbalanceada atingiu 76,5%.



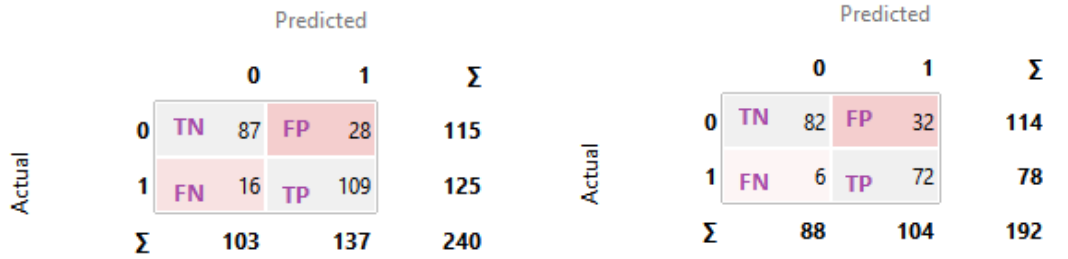
(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.1: Matriz de Confusão do *K Nearest Neighbor*.



(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.2: Matriz de Confusão do *Logistic Regression*.



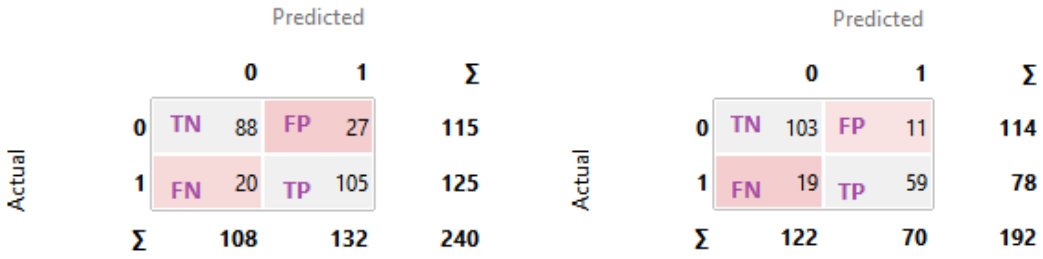
(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.3: Matriz de Confusão do *Naïve Bayes*.

As matrizes de confusões do algoritmo *Logistic Regression* são ilustrados nas imagens 5.2(a) e 5.2(b). Nelas, o modelo da Figura 5.2(a) passa uma falsa impressão de ter um melhor desempenho classificações verdadeiras (*True Positive* e *True Negative*), entretanto, isso é inválido, pois apesar de ter o maior número de acertos com 199 instâncias classificadas corretamente, o modelo dos dados desbalanceados 5.2(b) possui

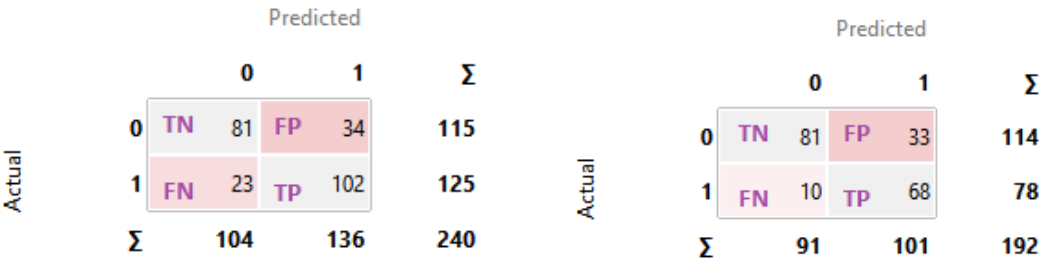
uma base menor, tornando proporcionalmente seus resultados melhores mesmo com uma quantidade mais baixa de acertos. Ao calcular as porcentagens diagonais das matrizes, os classificadores das Figuras 5.2(a) e 5.2(b), tiveram um total de acerto na diagonal verde de 82,9% e 85,4% e a taxa de erro na diagonal vermelha foi de 17% e 14,5% para os dados balanceados e desbalanceados respectivamente. Esses dados, revelam que o modelo da base 5.2(b) desequilibrada obteve o melhor desempenho.

As taxas de acertos das diagonais (*True Positive* e *True Negative*) do algoritmo *Naïve Bayes*, variaram respectivamente entre 80,2 e 81,6% para os dados desbalanceados 5.3(b) e balanceados 5.3(a). Ao analisar essas taxas, é possível verificar que o modelo da Figura 5.3(a) obteve o melhor desempenho devido a sua baixa taxa de erro de 18,3% comparada ao modelo 5.3(b) com 19,7%. Algo interessante a ser notado nesses dois classificadores é que ambos alcançaram uma menor quantidade de erro tipo II, que atribuí erroneamente a classe negativa para as instâncias positivas.



(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.4: Matriz de Confusão do algoritmo *Random Forest*.



(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.5: Matriz de Confusão do *Support Vector Machine*.

Ao comparar as matrizes de confusões dos classificadores *Random Forest* e *Support Vector Machine*, é possível observar que nas bases balanceadas das Figuras 5.4(a) e 5.5(a), houveram uma maior incidência de acertos das instâncias classificadas como positivas e que realmente eram positivas *(True Positive)*. Em contrapartida, nos modelos desbalanceados das imagens 5.4(b) e 5.5(b) aconteceu o inverso, houveram

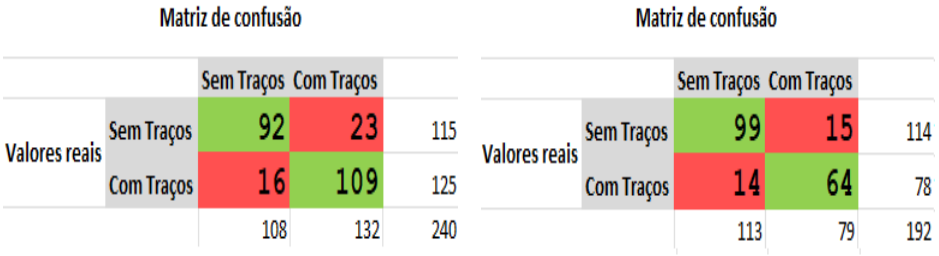
uma quantidade maior de acertos nas classes que eram negativas e foram realmente classificadas como tal *(True Negative)*. Além dessas similaridades, o *SVM* teve uma quantidade de menor de erros do tipo *(False Negative)*. O maior taxe de acerto dos modelos *Random Forest* e *Support Vector Machine* foram nas bases desbalanceadas 5.4(b), 5.5(b) com 84,3% e 77,6% respectivamente.

Para os modelos que classificam em conjunto os *tweets* (*Ensemble*), os resultados serão analisados. O modelo *hard voting* da base balanceada 5.6(a), conseguiu classificar corretamente 109 instâncias como verdadeiro positivo *(True Positive)* e 92 como verdadeiros negativos *(True Negative)*, totalizando uma taxa de acerto de 83%. Na base desbalanceada 5.6(b), o *ensemble* acertou 99 das instâncias como verdadeiro negativo *(True Negativa)* e 64 como verdadeiros positivos *(True Positive)*, acertando 85,7% do universo das classes positivas.

Apesar dos classificadores com *soft voting* possuírem praticamente as mesmas taxas de acertos e erros, respectivamente com 85% e 15% para os dados balanceados 5.7(a) e 85,4% e 14,5% para os desbalanceados 5.7(b), eles comportaram-se de formas diferentes. O classificador na base balanceada, teve mais instâncias classificadas como positiva ao contrário da base desbalanceada que classificou mais classes como negativa.

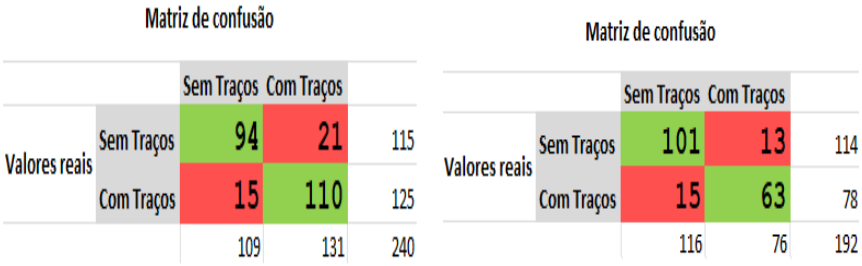
A Figura 5.8(b) representa a matriz de confusão da base desbalanceada do *ensemble* com os três melhores classificadores no sistema *hard voting*. Este modelo obteve uma taxa de acerto de 86,9%, classificando 67 instâncias corretamente como *true positive* (TP) e 100 como *true negative* (TN), assumindo respectivamente 52% e 34,8% para as classes verdadeiras. Para os dados balanceados, a Figura 5.8(a), ilustra um desempenho de 84% de acertos, com 110 *tweets* rotulados corretamente como *true positive* (TP) e 92 como *true negative* (TN), alcançando respectivamente 45,8% e 38,3% para as instâncias verdadeiras. Equivalente a esse modelo, o *ensemble* dos três melhores classificadores independentes no sistema de *soft voting* da base balanceada 5.9(a), alcançou um melhor desempenho para as instâncias verdadeiras positivas, atingindo uma porcentagem de 45,4% e nos dados desbalanceados 5.9(b) houve uma quantidade maior de acertos nas classes de verdadeiros negativos, acertando uma quantidade de 52,6%.

Ao comparar todas as matrizes de confusões dos classificadores independentes, é possível notar que o algoritmo *Suport Vector Machine* obteve o pior desempenho com 76,2% na base balanceada 5.5(a) e o *K Nearest Neighbor* com 76,5% na desbalanceada. O classificador *Logistic Regression* alcançou as melhores taxas de acertos, independente do tipo da base. Ao analisar o quantitativo de cada base, é notório que os algoritmos SVM, RF, LR se sobressaíram nos dados desbalanceados, está configuração de base obteve os melhores resultados.



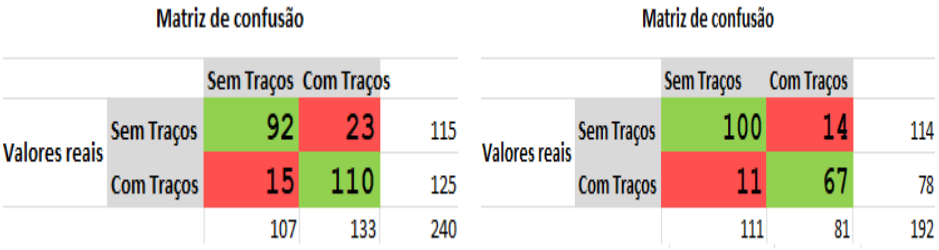
(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.6: Matriz de Confusão do modelo *ensemble* com *hard voting*.



(a) *Dados Balanceados* (b) *Dados Desbalanceados*

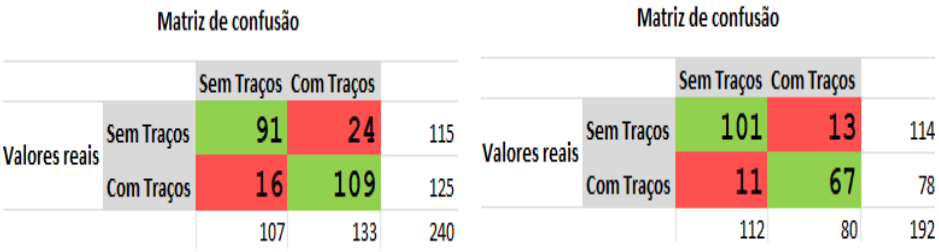
Figura 5.7: Matriz de Confusão do modelo *ensemble* com *soft voting*.



(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.8: Matriz de Confusão do modelo *ensemble* com os três melhores classificadores independentes com *hard voting*.

Nos classificadores em conjunto, o pior modelo foi o *ensemble* dos 5 algoritmos, tanto no sistema de *hard voting* para os dados balanceados com 83% quanto no de *soft voting* para a base desbalanceada com 85,4%. Já os melhores modelos utilizaram o sistema de *soft voting*, na base balanceada o *ensemble* com cinco classificadores atingiu o melhor desempenho com 85% e nos dados desbalanceados o classificador com os três melhores algoritmos obtiveram 87,5%. Algo interessante a ser apontado, é que nesses modelos de conjunto, todos tiveram um maior desempenho nos dados desbalanceados ao serem comparados com os balanceados.



(a) *Dados Balanceados* (b) *Dados Desbalanceados*

Figura 5.9: Matriz de Confusão do modelo *ensemble* com os três melhores classificadores independentes com *soft voting*.

5.2 Avaliação das métricas

Para a avaliação das métricas obtidas com os algoritmos classificadores independentes e em conjunto *ensemble*, serão apresentadas os resultados em gráficos para tornar possível a comparação visual das *performance* entre os modelos.

5.2.1 *Accuracy*

A Figura 5.10 demostra o resultado da métrica *Accuracy* para cada um dos modelos testados com dados balanceados e desbalanceados. Ao utiliza-la, ela representa o desempenho geral dos classificadores. Independente da configuração da base, os algoritmos *Random Forest* e *KNN* obtiveram o pior desempenho entre os modelos. O primeiro com acurácia de 81,7% para os dados balanceados e 78,1% para os desbalanceados. Já o *KNN* apresentou o segundo pior resultado, alcançando 82,8% de acurácia para a base balanceada e 82,8% para a desbalanceada. Além desses dados, é possível notar que *Logistic Regression* e o *ensemble* dos três melhores classificadores na *soft voting* foram os melhores algoritmos para a base desbalanceada com 87,5% e o *Naïve Bayes* atingiu o melhor desempenho de 89,2% na base balanceada. Algo interessante a ser notado, é que independente do tipo da base, *Logistic Regression* obteve a mesma pontuação de 87,5%.

5.2.2 *Precision*

A Figura 5.11 demostra o resultado da métrica *Precision* para cada um dos modelos testados com dados balanceados e desbalanceados, ao utiliza-lá à pergunta: “Quantos dos *tweets* que identificamos com traços de ansiedade e depressão estão realmente com esses traços?” é respondida. Independente da configuração da base, o algoritmo *Random Forest* obteve o pior desempenho com 81,7% para os dados balanceados e 78,1% para os desbalanceados. O segundo pior modelo foi o *ensemble* dos três melhores classificadores

Figura 5.10: Resultado da métrica de *accuracy* para os classificadores.

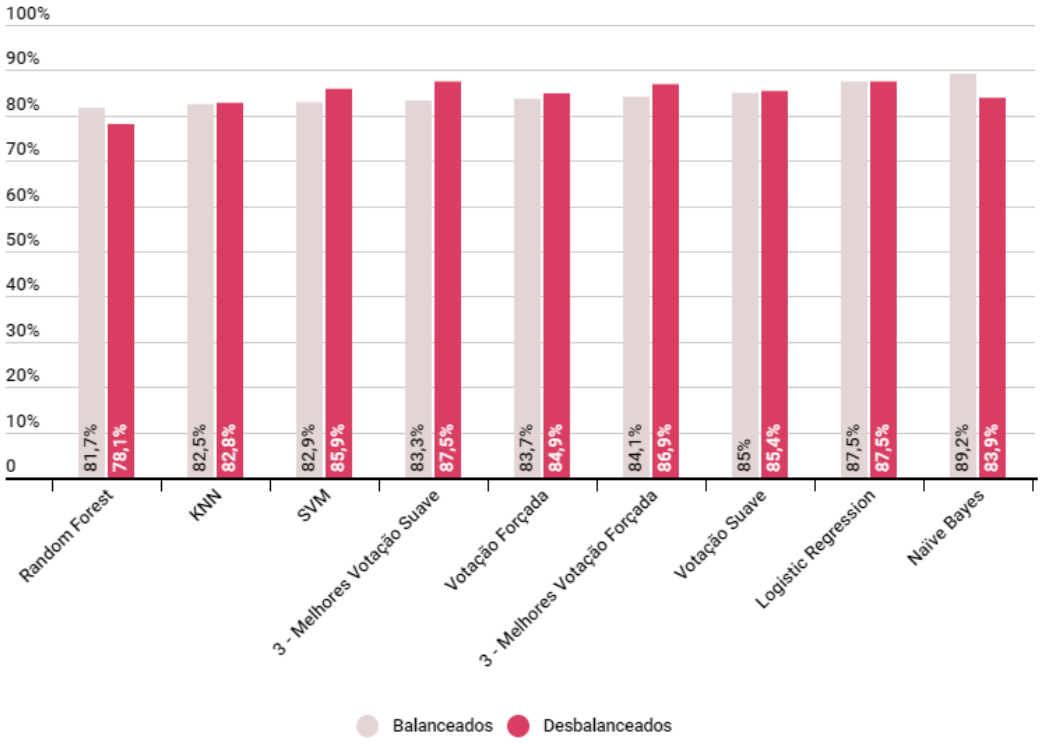
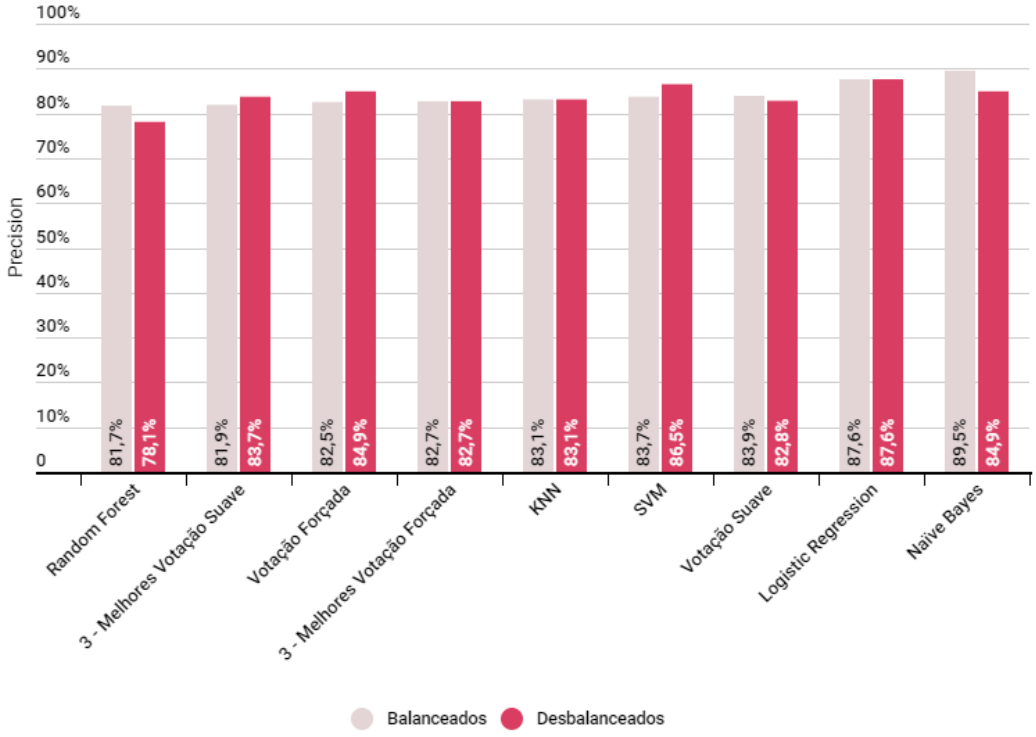


Figura 5.11: Resultado da métrica de *precision* para os classificadores.



com *soft voting* na base balanceada com 81,9% e o *ensemble* dos três melhores classificadores com *hard voting* para a base desbalanceada com 82,7%. O melhor algoritmo

para os dados balanceados foi o *Naïve Bayes* e o melhor nos dados desbalanceados foi o *Logistic Regression*. Os classificadores *KNN*, *Logistic Regression* e o *ensemble* dos três melhores classificadores com *hard voting* tiveram a mesma pontuação para os dados balanceados quanto os desbalanceados.

5.2.3 *Recall*

A Figura 5.12 demostra o resultado da métrica *Recall* para cada um dos modelos testados com dados balanceados e desbalanceados. Essa métrica é utilizada quando os falsos negativos são considerados mais prejudiciais que os falsos positivos. Independente da configuração da base, o algoritmo *Random Forest* obteve o pior desempenho com 81,7% para os dados balanceados e 78,1% para os desbalanceados. O segundo pior modelo foi o KNN e o *ensemble* de *hard voting*, com 82,5% e 82%, respectivamente para os dados balanceados e desbalanceados. O melhor desempenho entre os classificadores foi o *Naïve Bayes* para os dados balanceados e *Logistic Regression* para os desbalanceados. Além desses resultados, é possível observar houve um empate no segundo lugar com 85,9% para os modelos *SVM*, *ensemble* dos três melhores classificadores *soft voting* e *hard voting* para a base de dados desbalanceada. Para a base balanceada também houve um empate com 88% para os *ensemble* dos três melhores classificadores com *hard voting* e o *ensemble* com *soft voting*.

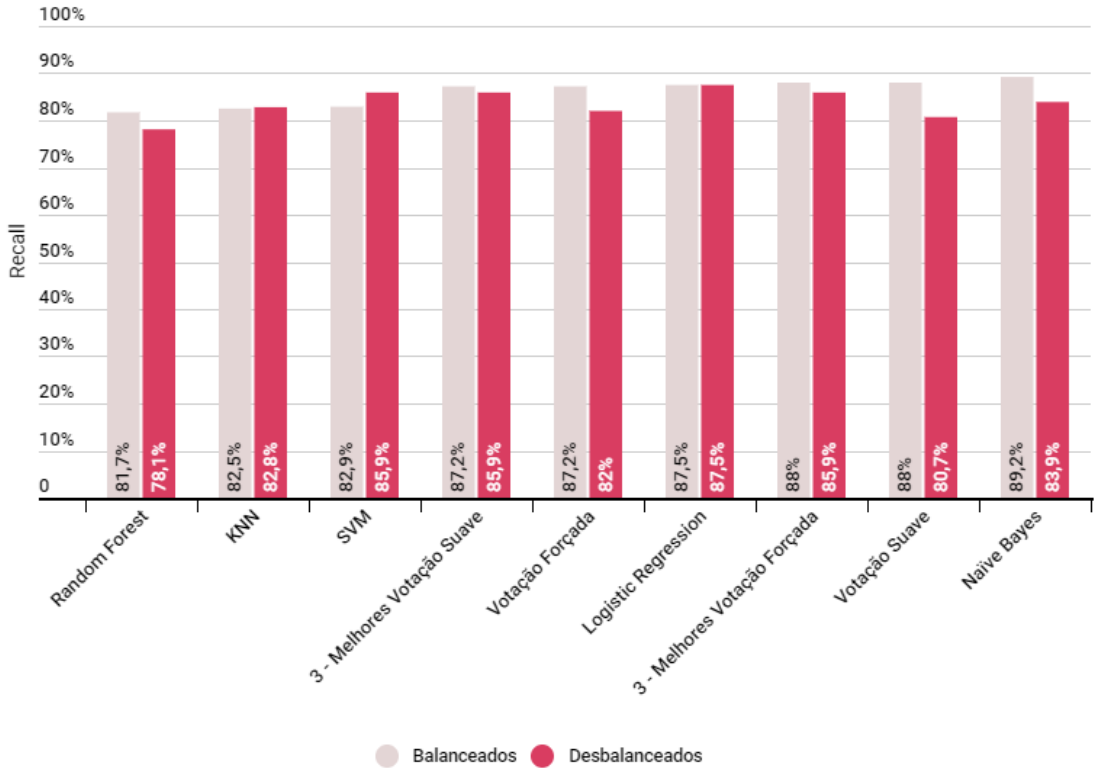
5.2.4 *F1-Score*

A Figura 5.13 demostra o resultado da métrica *F1-Score* para cada um dos modelos testados com dados balanceados e desbalanceados. Independente da configuração da base, o algoritmo *Random Forest* obteve o pior desempenho com 81,6% para os dados balanceados e 78% para os desbalanceados. O segundo pior modelo foi o KNN e o *ensemble* de *hard voting*, com 82,3% e 81,5%, respectivamente para os dados balanceados e desbalanceados. O primeiro e o segundo melhor classificador na base balanceada é o *Naïve Bayes* e *Logistic Regression*, respectivamente. Já nos dados desbalanceados o primeiro e o segundo melhor classificador é o *Logistic Regression* e *SVM*.

5.3 Comparação de resultados com o artigo *baseline*

Nesta seção serão comparados os perfis de teste e resultados de todos os modelos de classificações obtidos no artigo *baseline* [23] com os deste experimento. A métrica escolhida para a comparação das *performances* dos modelos foi a *F1-Score*. Essa métrica

Figura 5.12: Resultado da métrica de *recall*.

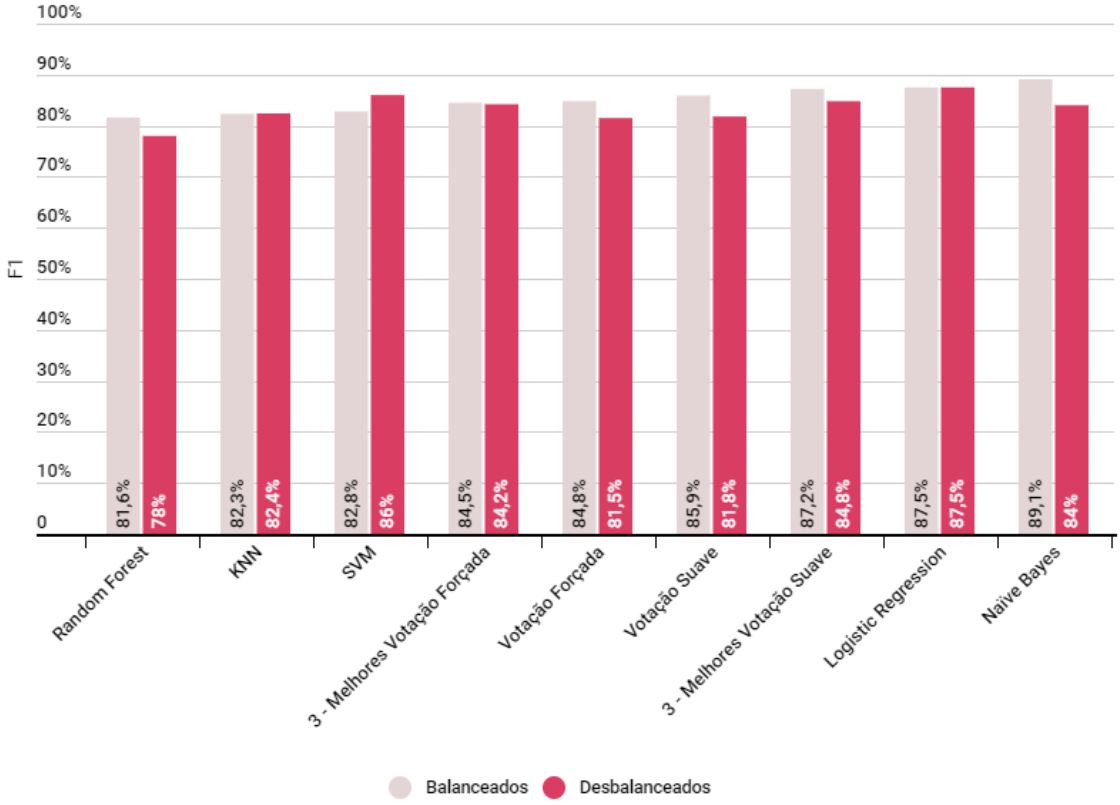


foi a utilizada no artigo original para expor os resultados.

Na Figura 5.14 é possível visualizar que dentre todos os classificadores independentes implementados neste experimento, o algoritmo *Naïve Bayes* obteve o melhor desempenho, com 89,1% e em segundo lugar, *Logistic Regression/maximum entropy* alcançou 87,5%. Os modelos *Random Forest*, *KNN* e *SVM* atingiram uma pontuação menor com 81,5%, 82,3% e 82,8% respectivamente. Em comparação com o artigo *baseline* [23], o algoritmo *Logistic Regression/maximum entropy* ficou em primeiro lugar com 83,2%, o*Naïve Bayes* passou para a segunda posição com 84,1% e o *KNN* foi considerado o pior classificador com 76,8%.

Independente do experimento realizado na Figura 5.14, todos os métodos de *ensemble* alcançaram uma pontuação maior do que os algoritmos RF, KNN e SVM. Na pesquisa de Fauzi [23] os algoritmos *Logistic Regression/maximum entropy* e *Naïve Bayes* atingiram uma pontuação de 84,1% e 83,2%, respectivamente, tornando esses modelos melhores do que o *ensemble* que uniu os classificadores NB, LR/ME e SVM com a estratégia de *hard voting*, cuja medida F1 foi de 84,0%. Um dos motivos, que podemos elencar para este resultado, é devido ao tipo de votação, pois torna-se difícil para o algoritmo atingir uma pontuação maior que o melhor modelo (NB 89,1), visto que

Figura 5.13: Resultado da métrica *F1-Scores*.



o ele é combinado com o classificador SVM que obteve um desempenho inferior com 81,3%. Diferente do artigo [23], no experimento deste trabalho, os algoritmos *Logistic Regression/maximum entropy* e *Naïve Bayes* ganharam de todos os *ensembles* e esses valores são explicados, pois, esses dois modelos tiveram um desempenho muito elevado em comparação ao RF, KNN E SVM.

Na Figura 5.15 é ilustrado os resultados do experimento com os dados desbalanceados. Nesta imagem, é possível verificar que no artigo *baseline* nenhum algoritmo atingiu o desempenho de 80% da métrica *F1-score*. Dentre os modelos de classificadores independentes, o melhor algoritmo foi o *Naïve Bayes* com 78,3% e em segundo lugar, o *SVM* com 78,1%.

Figura 5.14: Comparativo de resultados deste experimento com o do artigo *baseline* para *F1-Score* com dados balanceados.

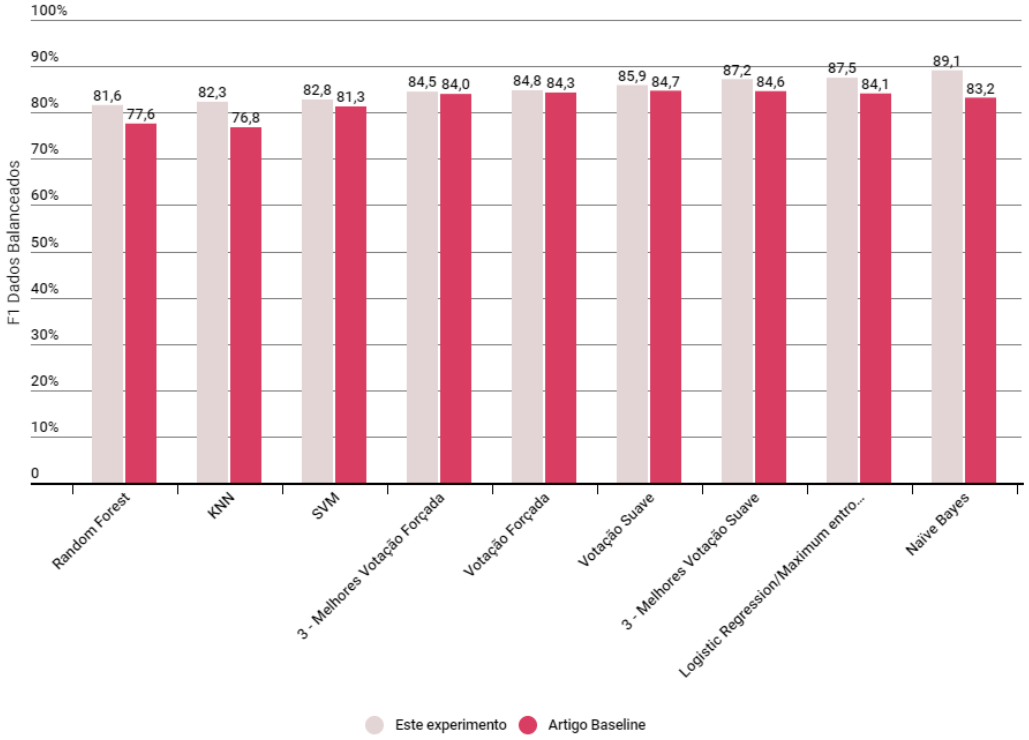
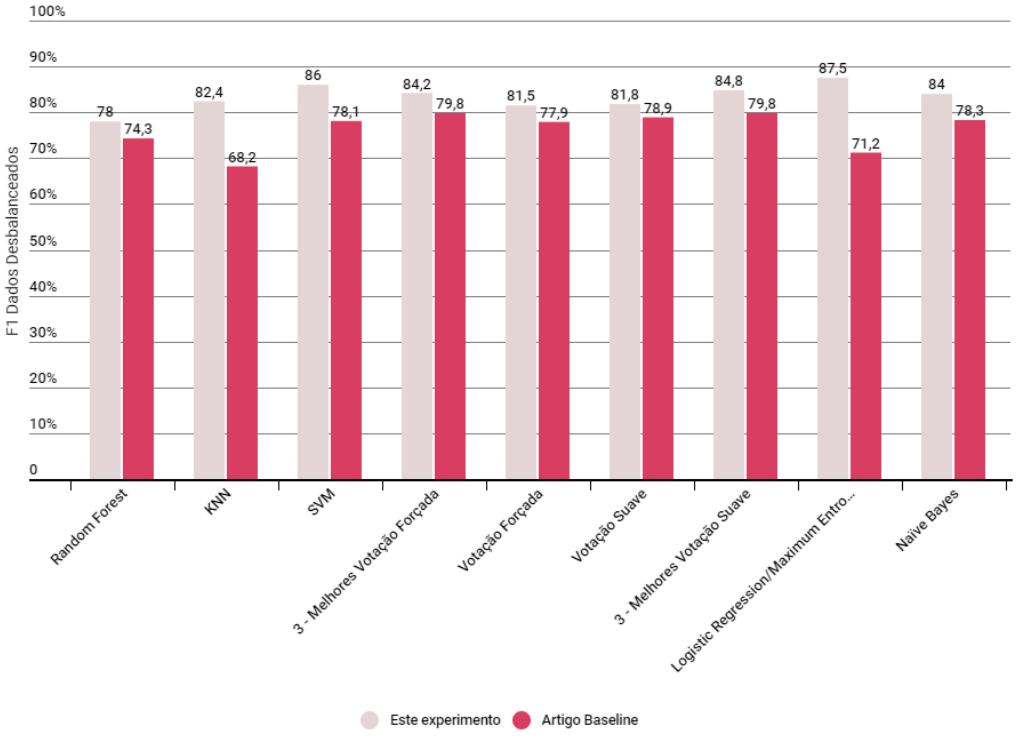


Figura 5.15: Comparativo de resultados deste experimento com o do artigo *baseline* para *F1-Score* com dados desbalanceados.



CAPÍTULO 6

Conclusão

Este trabalho final de curso abordou a detecção de traços de depressão e ansiedade em postagens do *Twitter* em língua portuguesa. Para tal foram implementados vários classificadores de *machine learning* em modo independente e em *ensemble* com a finalidade de verificar quais teriam o melhor desempenho na tarefa de classificação de *tweets*. A base de dados utilizada neste experimento foi coletada por Rodrigues [48] e estudada por Corrêa [16] para remoção de conteúdo ruidoso. Além das técnicas de pré-processamento dos dados empregados por Corrêa [16], aqui também foram realizadas outras como a tokenização, *stemming*, conversão de todas as letras para minúsculas, remoção de *stopwords* e números. Por último, a medida estatística TF-IDF foi empregada para destacar a importância das palavras no *corpus*.

Tendo como base o estudo e experimento realizado por Fauzi [23], neste projeto final de curso foram empregados os modelos *Random Forest, K-Nearest Neighbours, Support Vector Machine, Naïve Bayes* e *Logistic Regression*. Esses algoritmos foram utilizados independentes e em modo *ensemble*, para esse último foram implementados dois tipos: votação forçada e suave. O primeiro modelo *ensemble* foi construído com base na junção dos cinco classificadores independentes supracitados. Já o segundo, combinou os três melhores algoritmos avaliados na métrica *recall*. Todos esses modelos foram executados em bases balanceadas e desbalanceadas.

Ao utilizar o conjunto de dados balanceados os algoritmos *Random Forest* e *Naïve Bayes* obtiveram os melhores resultados em todas as métricas avaliadas (*Accuracy, Precision, Recall e F1-Scores*). Para os dados desbalanceados os melhores modelos foram: *K-Nearest Neighbor, Support Vector Machine*, *ensemble* completo com sistema de *Soft Voting* e *ensemble* dos três melhores com *Hard Voting*. *Logistic Regression*, *ensemble* completo com *Hard Voting* e o *ensemble* dos três melhores com *Soft Voting* ficaram empatados. Com base nesses resultados, observou-se que, para o domínio do problema proposto neste trabalho, com os dados desbalanceados uma quantidade maior de algoritmos obteve pontuações elevadas. Esta informação evidencia um dado muito importante, pois as publicações de usuários depressivos são bem menores comparados aos usuários saudáveis. Isso significa, que estes modelos, podem ser utilizados em dados

reais.

Comparando os resultados finais deste projeto com o artigo *baseline* [23], através da métrica *F1-Scores*, foi possível observar que independente da configuração da base, os valores apresentados aqui foram melhores. Apesar desta informação, os três melhores classificadores em ordem crescentes no estudo de Fauzi para os dados balanceados, foram os *ensembles* de *soft voting*, os três melhores com *soft voting* e *hard voting*. Nos dados desbalanceados, foram os *ensembles* dos três melhores com *soft voting*, os três melhores com *hard voting* e *soft voting*.

Como trabalho futuro, pretende-se utilizar uma base maior de treinamento com o objetivo de ampliar os testes realizados buscando aumentar o desempenho dos modelos propostos. Outra sugestão, é analisar a lista de *stop words* acrescentando novas palavras, para reduzir o tamanho da estrutura de indexação e aumentar a precisão do modelo [47].

Referências Bibliográficas

[1] **Folha informativa**. https://www.paho.org/pt/topicos/depressao. Accessed: 2021-10-01.

[2] **What about mood swings: Identifying depression on twitter with temporal measures of emotions**. In: *The Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018*, p. 1653–1660. Association for Computing Machinery, Inc, 4 2018.

[3] **Cooperative multimodal approach to depression detection in twitter**. 2019.

[4] ALMOUZINI, S.; KHEMAKHEM, M.; ALAGEEL, A. **Detecting arabic depressed users from twitter data**. *Procedia Computer Science*, 163:257–265, 01 2019.

[5] AMARAL, B. M.; DOS SANTOS SILVA, E. M.; DE ALMEIDA, A. M. G. **Análise de sentimentos/mineração de opinião: Uma revisão bibliográfica**, 2017.

[6] ASSOCIATION, A. P. **Manual diagnóstico e estatístico de transtornos mentais: DSM-5 - 5ª Edição**. 5 edition, 2014.

[7] BASHIR, D.; MONTANEZ, G.; SEHRA, S.; SEGURA, P.; LAUW, J. **An information-theoretic perspective on overfitting and underfitting**, 10 2020.

[8] BENEVENUTO, F.; RIBEIRO, F.; ARAÚJO, M. **Métodos para análise de sentimentos em mídias sociais**.

[9] BONACCORSO, G. **Machine learning algorithms: reference guide for popular algorithms for data science and machine learning**. Packt, 2017.

[10] BURKOV, A. **The hundred-page machine learning book**. 2019.

[11] CHICCO, D.; JURMAN, G. **The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation**. *BMC Genomics*, 21, 1 2020.

[12] CHOUDHURY, M.; COUNTS, S.; HORVITZ, E. **Social media as a measurement tool of depression in populations**. p. 47–56, 05 2013.

[13] CHOUDHURY, M. D. **Role of social media in tackling challenges in mental health**. *SAM 2013 - Proceedings of the 2nd International Workshop on Socially-Aware Multimedia, Co-located with ACM Multimedia 2013*, p. 49–52, 2013.

[14] CLARK, D. **Vencendo a Ansiedade e a Preocupação com a Terapia Cognitivo-Comportamental**. Artmed, 2014.

[15] COPPERSMITH, G.; DREDZE, M.; HARMAN, C. **Quantifying mental health signals in Twitter**. In: *Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, p. 51–60, Baltimore, Maryland, USA, jun 2014. Association for Computational Linguistics.

[16] CORRÊA, T. **Uso de ontologias para filtragem debase de dados para melhora de desempenho de classificadores em análise de sentimentos**, 2021.

[17] CUTLER, A.; CUTLER, D. R.; STEVENS, J. R. **Random forests**. *Ensemble Machine Learning*, p. 157–175, 2012.

[18] DARWIN, C.; LORENZ, K.; DE SOUZA LOBO GARCIA, L. **A expressão das emoções no homem e nos animais**. Campanhia de Bolso, 1965.

[19] DEY, L.; CHAKRABORTY, S.; BISWAS, A.; BOSE, B.; TIWARI, S. **Sentiment analysis of review datasets using naïve bayes' and k-nn classifier**. *International Journal of Information Engineering and Electronic Business*, 8:54–62, 07 2016.

[20] DIAS, M.; MALHEIROS, M. **Extração automática de palavras-chave de textos da língua portuguesa**. 01 2005.

[21] DOBSON, K. S. **The relationship between anxiety and depression**. *Clinical Psychology Review*, 5:307–324, 1 1985.

[22] ESCOVEDO, T.; KOSHIYAMA, A. **Introdução a Data Science Algoritmos de Machine Learning e Métodos de Análise by Tatiana Escovedo Adriano Koshiyama (z-lib.org) (1)**. Casa do Código, 2020.

[23] FAUZI, M. A.; YUNIARTI, A. **Ensemble method for indonesian twitter hate speech detection**. *Indonesian Journal of Electrical Engineering and Computer Science*, 11:294–299, 7 2018.

[24] FÁVERO, L. P. L.; BELFIORE, P. P. **Manual de análise de dados: estatística e modelagem multivariada com excel, SPSS e stata**. Elsevier, 2017.

[25] GUDIVADA, V. N.; RAO, C. R. **Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications**. Elsevier, 1 edition edition, 2018.

[26] IZBICK, R.; SANTOS, T. **Aprendizado de máquina: uma abordagem estatística**. 1 edition, 2020.

[27] JAIME, T. **Uso de algoritmos de aprendizado de máquina supervisionado para rotulação de dados**. 2019.

[28] JAPKOWICZ, N.; SHAH, M. **Evaluating Learning Algorithms: A Classification Perspective**. Cambridge University Press, 2011.

[29] KAUER, A. U. **Análise de sentimentos baseada em aspectos e atribuição de polaridade**, 2016.

[30] KUMAR, A.; SEBASTIAN, T. **Sentiment analysis on twitter**. *International Journal of Computer Science Issues*, 9:372–378, 07 2012.

[31] KUMAR, A.; SEBASTIAN, T. M. **Sentiment analysis on twitter**. 2012.

[32] LARNER, A. **The 2x2 Matrix: Contingency, Confusion and the Metrics of Binary Classification**. Springer International Publishing, 2021.

[33] LAVRAČ, N.; FLACH, P.; ZUPAN, B. **Rule evaluation measures: A unifying view**. In: *Inductive Logic Programming*, volume 1634, p. 174–185. Springer Verlag, 6 1999.

[34] LIU, Y.; WANG, Y.; ZHANG, J. **New machine learning algorithm: Random forest**. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7473 LNCS:246–252, 2012.

[35] MONICO, G.; DAL POZ, A.; GALO, M.; SANTOS, M.; OLIVEIRA, L. **Acurácia e precisão: revendo os conceitos de forma acurada / accuracy and precision: Reviewing the concepts by means of an accurate procedure**. *Boletim de Ciencias Geodesicas*, 15:469–483, 07 2009.

[36] MOUNT, J. **The equivalence of logistic regression and maximum entropy models**. 2011.

[37] MUNIZ, S. **Introdução à análise estatística de medidas**.

[38] NAUE, C.; WELTER, M. P. **Transtorno de ansiedade infantil**. Accessed: 2021-10-01.

[39] NURWIJAYANTI, W.; WIJAYA, D. S. **Determinants of depression**. *Journal of Holistic and Traditional Medicine*, 5, 2021.

[40] OF ELECTRICAL, I.; SECTION, E. E. B.; OF ELECTRICAL, I.; ENGINEERS, E. **Depression detection by analyzing social media posts of user**. 2019.

[41] ORABI, A. H.; BUDDHITHA, P.; ORABI, M. H.; INKPEN, D. **Deep learning for depression detection of twitter users**. In: *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*, p. 88–97. Association for Computational Linguistics, 2018.

[42] ORGANIZATION, W. H. **Depression and other common mental disorders global health estimates**, 2017.

[43] PRIMO, A. **O aspecto relacional das interações na web 2.0 1**, 2007.

[44] RASCHKA, S.; MIRJALILI, V. **Python machine learning : machine learning and deep learning with Python, scikit-learn, and TensorFlow**. Packt Publishing, 2017.

[45] RAZZOUK, D. **Por que o brasil deveria priorizar o tratamento da depressão na alocação dos recursos da saúde?** *Epidemiologia e servicos de saude : revista do Sistema Unico de Saude do Brasil*, 25:845–848, 10 2016.

[46] REEKER, L. H. **Theoretical constructs and measurement of performance and intelligence in intelligent systems | nist**. In: *Performance Metrics for Intelligent Systems (PerMIS 01)*, 9 2001.

[47] RIVAS, A. R.; IGLESIAS, E. L.; BORRAJO, L. **Study of query expansion techniques and their application in the biomedical information retrieval**. *The Scientific World Journal*, 2014, 2014.

[48] RODRIGUES, L. B. P. **Ontologia para detecção de traços de depressão em tweets**, 2019.

[49] ROKADE, P. **Release of endomorphin hormone and its effects on our body and moods: A review**. 2011.

[50] ROMANÍ, C. C.; KUKLINSKI, H. P. **Planeta web 2.0. inteligencia colectiva o medios fast food**. 2007.

[51] SAILUNAZ, K.; ALHAJJ, R. **Emotion and sentiment analysis from twitter text**. *Journal of Computational Science*, 36, 07 2019.

[52] SALLOUMSAID, A.; MOSTAFA, A.-E.; AZZA, A. M.; KHALED, S. **A survey of text mining in social media facebook and twitter perspectives**. *Advances in Science, Technology and Engineering Systems Journal*, 2:127–133, 2017.

[53] SANTOMAURO, D. F.; ET AL.. **Global prevalence and burden of depressive and anxiety disorders in 204 countries and territories in 2020 due to the covid-19 pandemic**. *The Lancet*, 398:1700–1712, 11 2021.

[54] SAUDE, U.; RODRIGUES, M. **SeleÇÃo de caracterÍsticas aplicada À moderaÇÃo automÁtica de comentÁrios de usuÁrios**. *Revosta Eletrônica Científica Inovação e Tecnologia*, 2, 2014.

[55] SCHONLAU, M.; ZOU, R. Y. **The random forest algorithm for statistical learning**. *Stata Journal*, 20:3–29, 3 2020.

[56] SHETTY, N. P.; MUNIYAL, B.; ANAND, A.; KUMAR, S.; PRABHU, S. **Predicting depression using deep learning and ensemble algorithms on raw twitter data**. *International Journal of Electrical and Computer Engineering*, 10:3751–3756, 2020.

[57] SOARES, G. **Sam-uma abordagem específica de mineração de dados socieconômicos de alunos do if amazonas para apoio ao processo de concessão de assistência estudantil**, 2020.

[58] THARWAT, A. **Classification assessment methods**. *Applied Computing and Informatics*, 17:168–192, 2018.

[59] THARWAT, A. **Classification assessment methods**. *Applied Computing and Informatics*, 17:168–192, 2018.

[60] TIWARI, P. K.; SHARMA, M.; GARG, P.; JAIN, T.; VERMA, V. K.; HUSSAIN, A. **A study on sentiment analysis of mental illness using machine learning techniques**. *IOP Conference Series: Materials Science and Engineering*, 1099:012043, 3 2021.

[61] TONG, L.; LIU, Z.; JIANG, Z.; ZHOU, F.; CHEN, L.; LYU, J.; ZHANG, X.; ZHANG, Q.; SENIOR, A. S.; WANG, Y.; LI, L.; ZHOU, H. **Inverse boosting pruning trees for depression detection on twitter**. 6 2019.

[62] WONGKOBLAP, A.; VADILLO, M.; CURCIN, V. **Depression detection of twitter posters using deep learning with anaphora resolution: Algorithm development and validation**. *JMIR Mental Health*, Apr. 2021.

[63] YAMADA, A. K.; MOURA, M. F.; CRUZ, S. A. B.; HIGA, R. H. **Uma solução flexível para a etapa de pré-processamento em mineração de textos**. Embrapa Agricultura Digital, 8 2012.

[64] ZHANG, Y.; LYU, H.; LIU, Y.; ZHANG, X.; WANG, Y.; LUO, J. **Monitoring depression trend on twitter during the covid-19 pandemic**. 7 2020.