Instituto Federal do Norte de Minas Gerais
campus Montes Claros
Curso de Bacharelado em Ciência da Computação

DETECÇÃO AUTOMÁTICA DE TEXTOS COM IDEAÇÃO SUICIDA VIA ALGORITMOS DE APRENDIZADO DE MÁQUINA

ANNE OLIVEIRA ALMEIDA
Orientador: Laércio Ives Santos

Montes Claros
Novembro de 2021

ANNE OLIVEIRA ALMEIDA

DETECÇÃO AUTOMÁTICA DE TEXTOS COM IDEAÇÃO SUICIDA VIA ALGORITMOS DE APRENDIZADO DE MÁQUINA

Monografia apresentada ao Curso de Graduação em Ciência da Computação do Instituto Federal do Norte de Minas Gerais – Campus Montes Claros, como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Laércio Ives Santos

Montes Claros

Novembro de 2021

Oliveira Almeida, Anne
D1234p Detecção Automática de Textos com Ideação Suicida via Algoritmos de Aprendizado de Máquina / Anne Oliveira Almeida. — Montes Claros, 2021
xxii, 73 f. : il. ; 29cm

Monografia (graduação) — Instituto Federal do Norte de Minas Gerais
Orientador: Laércio Ives Santos

1. Aprendizagem de Máquina. 2. Processamento de Linguagem Natural. 3. PLN. 4. Ideação Suicida. 5. Inteligência Artificial. 6. Suicídio. I. Título.

CDU 519.6*82.10

[Folha de Aprovação]

Quando a secretaria do Curso fornecer esta folha,
ela deve ser digitalizada e armazenada no disco em formato gráfico.

Se você estiver usando o `pdflatex`,
armazene o arquivo preferencialmente em formato PNG
(o formato JPEG é pior neste caso).

Se você estiver usando o `latex` (não o `pdflatex`),
terá que converter o arquivo gráfico para o formato EPS.

Em seguida, acrescente a opção `approval={`*nome do arquivo*`}`
ao comando `\ppgccufmg`.

Dedico esse trabalho às inúmeras pessoas que lutam todos os dias para permanecerem vivas e principalmente àquelas que aqui já não estão na pessoa da minha querida Amiga Luciana Castilho.

Agradecimentos

Gostaria de agradecer imensamente aos meus queridos Professores do Instituto Federal do Norte de Minas Gerais (IFNMG) - campus Montes Claros, meu orientador Professor Laércio Ives por dedicar o seu tempo, conhecimentos e experiência para me auxiliar nessa caminhada e também, expressar a minha gratidão ao Professor Lúcio Dutra pela paciência e ajuda na primeira etapa do trabalho, atendimentos individuais que muito acrescentaram ao desenvolvimento e organização desse trabalho. Agradeço ainda aos Colegas que me apoiaram nessa jornada e sempre torceram pelo meu sucesso acadêmico, em especial, ao Igor Alberte que me ajudou na revisão do documento.

"A lei da mente é implacável.
O que você pensa, você cria;
O que você sente, você atrai;
O que você acredita, torna-se realidade."
(Shakyamuni)

Resumo

A Organização Mundial da Saúde (OMS) afirma que a prevenção ao suicídio é um imperativo global e que, com a identificação precoce da ideação suicida, é possível implementar tratamentos que ajudem o indivíduo a repensar e desistir do plano de acabar com a própria vida. A detecção da ideação suicida consiste em investigar comportamentos ou intenções que possam desencadear a tentativa de suicídio. Nessa realidade, a escrita pessoal em meios digitais ou físicos é uma forma de o indivíduo comunicar, conscientemente ou não, o seu desejo de morte. No que tange à tecnologia, algoritmos de Processamento de Linguagem Natural (PLN) vêm sendo implementados para que a máquina possa processar a comunicação humana e, assim, fazer a detecção automática da ideação, sendo útil principalmente no monitoramento digital de possíveis vítimas e na telemedicina. O estudo desse domínio usando PLN é presente na literatura através de diversos trabalhos usando como base mensagens de redes sociais. Devido a questões éticas e humanas, é raro encontrar trabalhos desenvolvidos com dados de uma única vítima bem-sucedida de suicídio. No geral, os dados são de diversas pessoas e não se sabe se elas ao menos chegaram a tentar o ato. Consequentemente, a natureza desses dados é um fator complexo para os rotuladores que decidem sobre a intenção expressa no texto sem um referencial linear para concluir se há ou não ideação suicida. Neste presente trabalho foram utilizados dois conjuntos de dados públicos, de duas pessoas que faleceram por suicídio, Virginia Woolf e Victoria McLeod. Esses dados foram processados usando duas técnicas de seleção de características, validação cruzada e cinco técnicas de aprendizado de máquina com o objetivo de melhorar os resultados dos estudos existentes na literatura ao agregar ferramentas de seleção de características aos modelos de predição. O *Naive Bayes* combinado com a ferramenta *chi-square* (χ^2) obteve o melhor resultado do experimento para o conjunto de dados da Virginia Woolf. Enquanto que no conjunto da Victoria McLeod, o mesmo classificador combinado com o TF-IDF obteve a melhor performance. A análise de sentimentos dos dois conjuntos mostrou similaridade entre as classes positivas (1) e negativas (0), podendo isso ter dificultado a separabilidade das classes pelos modelos. As núvens de palavras dos dois conjuntos destacou o uso de palavras temporais e também a presença de palavras com

frequências parecidas na classe positiva de ambos conjuntos: *time, life* e *will*. Com esses resultados, amplia-se a discussão não apenas sobre a performance de modelos para identificação de ideação suicida, mas também, a possibilidade de generalização além dos limites etários, sociais, culturais e econômicos.

Palavras-chave: Machine Learning, Aprendizado Supervisionado, Inteligência Artificial, Suicídio, Processamento de Linguagem Natural.

Abstract

The World Health Organization (WHO) states that suicide prevention is a global imperative and that, with the early identification of suicidal ideation, it is possible to implement treatments that help the individual to rethink and give up on the plan to end his own life. The detection of suicidal ideation consists of investigating behavior or intentions that may trigger the suicide attempt. In this reality, personal writing in digital or physical media is a way to communicate, consciously or not, your death wish. Regarding technology, Natural Language Processing (NLP) algorithms have been implemented so that the machine can process human communication and, thus, automatically detect ideation, being mainly useful in digital monitoring of possible victims and in telemedicine. The study of this domain using NLP is present in the literature through several works using messages from social networks as a baseline. Due to ethical and human issues, it is rare to find works developed with data from a single successful suicide victim. Often, the data are from different people and it is not known if they even attempted the act. Consequently, the nature of these data is a complex factor for labelers who decide on the intention expressed in the text without a linear framework to conclude whether or not there is suicidal ideation. In this present work, two sets of public data were used, from two people who died by suicide, Virginia Woolf and Victoria McLeod. These data were processed using two feature selection techniques, cross validation and five machine learning techniques with the aim of improving the results of existing studies in the literature. Naive Bayes combined with the chi-square (χ^2) tool obtained the best result of the experiment for the dataset of Virginia Woolf. While in the set of Victoria McLeod, the same classifier combined with the TF-IDF had the best performance. The analysis of feelings of the two sets showed similarity between the positive (1) and negative (0) classes, which may have hindered the separability of the classes by the models. The word clouds of both sets highlighted the use of temporal words and also the presence of words with similar frequencies in the positive class of both sets: time, life and will. With these results, the discussion is broadened not only about the performance of models to identify suicidal ideation, but also the possibility of generalization of the models beyond the limits of age, social, cultural and

economic limits.

Keywords: Machine Learning, Supervised Learning, Artificial Intelligence, Suicide, Natural Language Processing.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

O suicídio não é um fenômeno dos tempos modernos. O primeiro texto considerado por especialistas como suicida foi encontrado no Egito e data do ano 2040 A.C. [Erman, 1978]. Na cultura ocidental, os primeiros registros vêm da Grécia, onde o suicídio era considerado por Platão como um comportamento negativo, com poucas exceções, a questão não estava na existência do indivíduo e sim no seu papel para a comunidade e estado (dever e obrigações). Para outros, como o estoico romano Sêneca, o aspecto individual também deveria ser considerado — "viver por viver não é bom, o importante é viver bem", pensamento enfatizado com a seguinte frase: "uma pessoa sábia vive o quanto é necessário e não o quanto ela pode" [Cholbi, 2017]. Em se tratando de um tema de origem subjetiva com motivos difusos, ao longo dos anos, especialistas vêm elaborando pensamentos diversos oriundos de estudos desenvolvidos [Kaplan et al., 2008]:

- Thomas Szasz, psiquiatra austríaco, se posicionou, no início da década de 70, contra a ideia de que o suicídio fosse fruto de doença mental. A vontade de se matar deveria ser respeitada pelos médicos, policiais e demais pessoas envolvidas. Ele ainda se mostrou contra a infantilização ou desumanização do suicida.

- Richard Brandt, ex-presidente da sociedade de filosofia americana, difere o suicídio racional do irracional. No caso do primeiro, ele defende a não intervenção e até o suicídio assistido, enquanto no segundo ele mostra que estratégias de prevenção ao suicidio são necessárias.

- Erwin Ringel, fundador da Associação Internacional para a Prevenção do Suicídio, argumentou, no início da década de 80, que toda vida importa e posicionou-se contra atitudes libertárias no que tange ao suicídio — inclusive para pacientes em estágios terminais de doenças. Ele defendeu a prevenção ao suicídio como uma estratégia de revigoração da vida humana através do tratamento terapêutico.

Estratégias de tratamento são possíveis de serem implementadas quando se identifica a vulnerabilidade do indivíduo nesse domínio — ideação suicida é o termo que descreve a tendência de pôr fim à própria vida. Indivíduos com ideação suicida podem ser reconhecidos como planejadores (planejam o ato), tentadores (implementam o plano) e completadores, aqueles que obtiveram sucesso na sua estratégia e que hoje estão mortos [Tadesse et al., 2020]. As vítimas com ideação suicida podem expressar os seus desejos de morte através de várias formas no seu dia-a-dia. A detecção dessa ideação consiste em investigar intenções ou comportamentos perigosos antes que o suicídio venha a acontecer [Ji et al., 2019].

Com base no que foi dito acima, pode-se inferir que a predição do suicídio e a distinção da ideação racional ou irracional pode ser considerado um desafio tanto para os profissionais da área clínica quanto para os da área de ciência de dados. Ainda, o processo cognitivo humano é complexo e este se desenvolve em ambientes diversificados, nos tornando pessoas únicas que vivenciam experiências comuns de forma inédita. Problemas psiquiátricos, experiências traumáticas na vida, traços de personalidade, fatores estressantes pontuais ou não, questões culturais e religiosas, relacionamentos interpessoais, dentre outros, são fatores que convergem para a decisão consciente ou inconsciente do indivíduo em cometer o sucídio [Kaplan et al., 2008].

A razão pela qual as pessoas atentam contra as suas próprias vidas não é simples de entender e identificar. A depressão é um fator de risco alto, todavia, pessoas sem esse quadro psiquiátrico também podem apresentar pensamentos suicidas [O'Connor & Nock, 2014]. Sendo assim, é essencial o desenvolvimento de métodos que aumentem a assertividade dos resultados e assim possam diminuir a incerteza dos profissionais da área em relação à detecção automática, pela máquina, da ideação suicida como forma de auxílio às estratégias de prevenção [Ji et al., 2019]. Em tempos de ferramentas virtuais e aplicativos, o desenvolvimento de um modelo preditor suficientemente confiável para o uso no atendimento psicoterapêutico presencial ou remoto significa uma alternativa de melhoria nos dados de saúde pública nesse setor.

1.1 Motivação

O suicidio é uma trágedia que não só termina uma vida prematuramente, mas que tem efeito devastador para os familiares e amigos da vítima. A cada 40 segundos alguém se mata no mundo. A cada 3 segundos alguém atenta contra a própria vida e cada ação suicida impacta seriamente pelo menos 6 pessoas [Berni et al., 2018]. As consequências sociais e econômicas para a comunidade são significativas, já que o sistema de saúde precisa tratar os indivíduos que não obtiveram sucesso na sua estratégia de morte e que,

em alguns casos, podem ficar com deficiências físicas permanentes e também tratar o seu círculo social que tem a saúde abalada pelo ato do ente querido [Parekh & Phillips, 2014]. A Organização Mundial da Saúde (OMS), em um relatório de 2014, afirma que a prevenção ao suicídio é um imperativo global e estabelece como meta reduzir o índice em 10% até 2020.

Esse mesmo relatório mostra que o número de mortes por suicídio no mundo, estimado em 2012, foi de 804.000 — 11,4 por 100.000 habitantes — e, para cada morte, existem muitos outros indivíduos que tentaram e não obtiveram sucesso. Segundo a OMS, os indicadores, muito provavelmente, estão subestimados pelo fato de que muitos suicídios são registrados, devido a questões legais e culturais, como acidente ou outros motivos diversos [Parekh & Phillips, 2014]. Em termos de comparação, dados globais apontam que o número de mortes, anual, por acidente de trânsito é em torno de 1,3 milhões. Sendo que, nos Estados Unidos constam 38.000 mortes — 12,4 por 100.000 habitantes — [Organization, 2019].

A OMS ainda afirma que a prevenção ao suicidio é possível com os serviços de saúde incorporando, como seu componente principal, a identificação precoce da ideação suicida e oferecendo os cuidados adequados ao indivíduo. Cria-se, assim, a oportunidade para que as vítimas possam refletir sobre o ato, suas consequências e superar a crise, já que a maioria tem um sentimento ambivalente em relação ao desejo de morrer e o ato muitas vezes é uma resposta impulsiva a um fator estressante [Parekh & Phillips, 2014].

A ideação suicida pode ser observada e detectada de diversas formas, dentre elas, pela escrita. A comunicação faz parte da natureza humana e, segundo Tadesse et al. [2020], mais de 20% das pessoas que tentam suicídio e mais de 50% das que conseguem terminar com a própria vida deixam "carta de despedida". Por conseguinte, qualquer sinal de ideação suicida em textos deve ser visto como um alerta considerável.

Nas últimas décadas o Processamento de Linguagem Natural (PLN), que é o uso de métodos que tornam possível o acesso dos computadores à linguagem humana, vem sendo cada vez mais utilizado [Eisenstein, 2019]. O PLN usa algoritmos de aprendizado de máquina que fazem parte da área de inteligência artificial na computação: são técnicas que automaticamente detectam padrões que são utilizados para fazer predição de eventos em dados [Silge & Robinson, 2017]. A capacidade da máquina em processar e aprender a linguagem humana faz com que ferramentas como: tradução entre idiomas, filtragem de emails e classificação de textos possam existir [Eisenstein, 2019]. Na detecção da ideação suicida em textos, a técnica de classificação verifica se o texto tem características condizentes com o domínio investigado e o classifica como ideação suicida ou não.

A integração de técnicas de PLN com a prática clínica para a identificação automática de fatores de risco de suicídio (pensamentos e comportamentos) a fim de que alertas apoiem a tomada de decisão do corpo clínico é defendida por Bantilan et al. [2021], que ainda afirmam a importância de plataformas de telemedicina[1], que favorecem o uso do PLN e corroboram para uma rápida resposta ao risco de suicídio e implementação de tratamento. A telemedicina, como prevenção ao suicídio, proporciona um aumento na segurança do indivíduo em risco e é de grande importância quando se trata de pacientes jovens adultos por ser o suicídio a segunda causa mais frequente de morte nessa categoria. Ademais, essa classe de indivíduos são típicos usuários precoces de sistemas de telemedicina, facilitando assim o uso das técnicas de PLN como estratégia de saúde pública [Bantilan et al., 2021].

O estudo social do fenômeno através da tentativa de identificar a ideação suicida em textos coletados nas redes sociais é explorada por diversos autores, dentre eles: Tadesse et al. [2020], Chiroma et al. [2018] e Malini & Tan [2016]. Todavia, não se sabe se a ideação suicida identificada nos textos culminou na tentativa de fato e se esses indivíduos obtiveram sucesso ou não na suas estratégias. Ademais, as técnicas de aprendizado de máquina utilizadas são do tipo supervisionado, em que se faz necessária a utilização de dados rotulados. A rotulagem humana dos dados não é uma tarefa trivial e objetiva, sendo considerada, pelos profissionais que a realizam, como difícil [O'Dea et al., 2015]. No caso de textos extraídos das redes sociais, onde os dados são originados de vários usuários diferentes, essa tarefa se torna ainda mais complexa por não haver linearidade e, no caso, implica inferir o contexto e a natureza da mensagem de um único pequeno texto para determinar a classe à qual ele pertence.

Diferentemente, o trabalho de Berni et al. [2018] utilizou dados provenientes de um único indivíduo que faleceu vítima de suicídio. A disponibilidade desse tipo de dado é raro — por questões éticas — e, sendo assim, são poucos estudos na literatura com essa abordagem. Apesar de dificultar a generalização do modelo, devido a sua especificidade, esse conjunto de dados apresenta, ainda assim, a oportunidade de ser melhor explorado, visto que utilizou apenas o modelo Naive Bayes e o *Term Frequency* (TF) como seleção de características. Existem outras técnicas mais avançadas do que a utilizada no trabalho em questão e outras ferramentas de seleção de características que podem trazer resultados melhores do que os encontrados na metodologia testada.

[1] "Telemedicina, em sentido amplo, pode ser definida como o uso das tecnologias de informação e comunicação na saúde, viabilizando a oferta de serviços ligados aos cuidados com a saúde (ampliação da atenção e da cobertura), especialmente nos casos em que a distância é um fator crítico [Maldonado et al., 2016]".

1.2 Objetivos

O objetivo deste estudo foi construir e avaliar modelos capazes de identificar padrões condizentes com a ideação suicida e comparar a performance deles. Para alcançar esse objetivo, os seguintes objetivos específicos foram necessários:

1. Desenvolver modelos de aprendizado de máquina para a detecção da ideação suicida em textos, utilizando as técnicas de: *Naive Bayes*, *Support Vector Machine* (SVM), Árvore de Decisão, Floresta Aleatória e Rede Neural.

2. Avaliar a performance dos modelos implementados sem a utilização de métodos de seleção de características.

3. Avaliar a performance dos modelos desenvolvidos quando fazem uso de métodos de seleção de características: TF-IDF e χ^2 (*chi-square*).

1.3 Organização do documento

Este trabalho está dividido da seguinte forma: no Capítulo 2 discute-se o processamento de linguagem natural, apresentando as etapas desse processo: extração de características 2.1, seleção de características 2.2, técnicas de classificação 2.3 e métricas de avaliação 2.4. No Capítulo 3 são apresentados os trabalhos relacionados e no Capítulo 4 é apresentada a metodologia utilizada. Finalizando, o Capítulo 5 com os resultados encontrados e o Capítulo 6 com a conclusão.

Capítulo 2

Processamento de Linguagem Natural

Alguns eventos e publicações científicas na área de processamento de linguagem natural (PLN) se referem ao tema como linguística computacional [Othero, 2006]. Portanto, podem ser tratados como sinônimos. Essa área da computação é focada na criação e análise de algoritmos com o propósito de prover ferramentas para que a máquina possa melhorar a sua capacidade de trabalhar com a linguagem humana natural: extrair informações de textos, tradução entre línguas, etc. As técnicas de aprendizado de máquina (*machine learning*) são comumente implementadas nesse campo de estudo por permitirem o uso de algoritmos de reconhecimento de padrões. Porém, o processamento de linguagem natural tem algumas peculiaridades que o diferem dos outros domínios que usam aprendizado de máquina. Por exemplo, os dados são gerados através de arranjos combinatórios de símbolos, sendo assim, são dados discretos. Os algoritmos precisam ser robustos na observação de ocorrências que não estejam presentes nos dados de treinamento, visto que a distribuição de palavras e outros elementos linguísticos no texto são conhecidos por conter poucas palavras muito frequentes e muitas palavras raras pouco presentes, como exemplo: preposições e palavras como angústia, medo, suicídio, etc. A composição é outra característica: uma unidade (palavra) se combina com outras unidades para formar frases e que, por sua vez, também se combinam para formar parágrafos. Por fim, o uso de referencial, como palavras adjacentes, é muitas vezes utilizado para que não haja ambiguidade durante o processamento do texto [Eisenstein, 2019] .

Os algoritmos de aprendizado de máquina podem desempenhar diversas tarefas: identificação de objetos na área de visão computacional, predição de função de proteínas na biologia computacional, detecção de fraudes financeiras, identificação de *spam* em emails e conteúdo impróprio ou explícito na *web*, reconhecimento de fala, sintetização de discurso e outros. Sendo assim, a análise de sentimentos (classificação de texto) é apenas uma das tarefas que os algoritmos podem desempenhar [Mohri et al., 2012].

Segundo Goldberg [2017], a classificação de sentimento é um tanto desafiadora por envolver a correta identificação e tratamento de metáforas e sarcasmo. Sendo assim, a definição de sentimentos não é algo claro e direto. No processamento de linguagem natural relacionado a sentimentos, frequentemente se usa uma representação mais simplista, binária — são apenas duas classes (ideação suicida e não ideação suicida) em que se define os valores positivo e negativo para o domínio analisado. Além disso, frases positivas podem conter palavras negativas e vice-versa, o que implica na necessidade de avaliar todo o contexto em que a frase se encontra: construções linguísticas como a negação e a estrutura da senteça de modo geral [Goldberg, 2017]. A análise de sentimentos pode usar os recursos léxicos, semânticos, sintáticos e contextuais para um melhor entendimento do conjunto de dados analisados.

Para que o processamento de linguagem natural seja realizado por computadores, são necessárias duas fases: a extração de características do texto (segmentação da entrada em unidades significativas) e a classificação do vetor gerado na primeira fase de acordo com o domínio em estudo [Haddi et al., 2013].

Nas subseções a seguir, todo o processamento será abordado de forma mais detalhada: extração de características, seleção de características, técnicas de classificação e métricas de avaliação.

2.1 Extração de Características

Os documentos, quando são pré-processados (submetidos à extração de características), se tornam mais relevantes para o contexto, uma vez que sofrem alguns procedimentos, como: remoção das palavras menos significativas e pontuações; segmentação das sentenças e tokenização. Esses procedimentos reduzem o tamanho dos dados reais, fazendo com que o trabalho de classificação seja realizado apenas sobre dados que apresentam algum impacto no domínio estudado. O principal objetivo do pré-processamento é aprimorar a relevância entre palavra e documento e a relevância entre palavra e categoria [Ogada, 2016].

A fase de extração de características é composta por duas etapas: a extração propriamente dita e a avaliação da relevância dessas características para a análise do grupo de dados, seleção de características. Na primeira, as palavras que compõem o texto são extraídas e colocadas em um vetor que é denominado vetor de características — processo chamado de tokenização. A tokenização permite contabilizar as características do texto de forma organizada. Na última fase, o estudo da relevância das palavras para o contexto é importante para descartar os itens que não agregam significado para a análise ou estão no texto com finalidade, apenas, de coesão contextual. Essas pala-

vras, quando levadas para o processo de classificação, apresentam um custo-benefício alto: o processamento é oneroso pela grande quantidade de palavras e o benefício é pequeno pela pouca ou nenhuma significância de grande parte dessas palavras [Haddi et al., 2013]. Sendo assim, a ferramenta *Term Frequency-Inverse Document Frequency* ajuda na seleção das palavras de maior relevância para o contexto, exercendo um papel importante na otimização inicial do estudo.

A preparação dos dados de entrada para o processamento dos algoritmos de classificação é uma etapa importante que envolve procedimentos como: tokenização, definição de n-grams, representação vetorial (*bag of words*) e a relevância de cada palavra na classificação (TF-IDF).

2.1.1 Tokenização

A tokenização é descrita como um processo de segmentar o texto e extrair as palavras. O token é uma sequência de caracteres, uma unidade semântica definida do documento. Segundo Baeza-Yates & Ribeiro-Neto [2011], a tokenização pode, durante o processo, ignorar pontuação e outras características do texto, como, por exemplo, artigos — as palavras retiradas do texto são denominadas *stopwords*. O autor ainda afirma que a retirada das *stopwords* pode reduzir em até 40% o tamanho final do vetor de tokens.

Quando se trata do processo de tokenização, é importante definir de forma clara a unidade semântica que será usada no estudo. No presente trabalho, uma palavra é definida como uma sequência alfanumérica de caracteres com espaço em ambos os lados, podendo ocorrer hífen ou apóstrofos, excluindo pontuação. Ademais, uma sentença pode ser entendida como uma sequência de palavras que terminam com “ . ”, “ ? ” ou “ ! ”, sendo que tal heurística atende a 90% dos casos de representação de textos [Manning & Schütze, 1999].

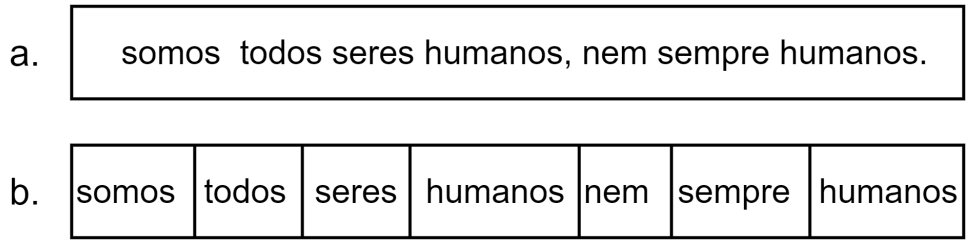

Figura 2.1: Geração de tokens.

A Figura 2.1 mostra (a) uma entrada antes da tokenização e o (b) resultado do processo onde sete tokens foram gerados. O tipo do token está relacionado com a classe semântica à qual ele pertence (verbo, substantivo, adjetivo, numeral, etc.). O termo é a nomenclatura que se dá ao token normalizado que está presente no dicionário

utilizado pelo sistema. A normalização consiste na retirada de inflexões, formatação específica de data e números, substituição de caracteres maiúsculos por minúsculos ou *stemming* - retirada de afixos, reduzindo as palavras às suas raizes [Horn, 2008]. Em alguns casos complexos, a eliminação de afixos pode causar erros e o uso de *Lemmatizers* se faz necessário. Esses sistemas identificam alguns erros que ocorrem no *stemming* e ainda, tratam transformações mais complexas peculiares a cada língua, por exemplo: *geese* para *goose* no inglês [Eisenstein, 2019].

Silge & Robinson [2017] ressaltam que para a mineração de texto de forma organizada, o token é, geralmente, uma única unidade semântica. Mas, em alguns casos, dependendo do documento analisado e do objetivo do projeto, o token armazenado pode ser um n-gram, sentença ou até parágrafo.

2.1.2 N-grams

Os n-grams são uma sequência de n-itens em um texto que podem ser caracteres, unidades semânticas, palavras ou frases. No presente trabalho o gram é usado para representar uma única palavra. A sequência de dois grams é denominada bigram, de três grams são trigrams, de quatro grams são 4-grams e assim por diante. A escolha de quantos grams usar na representação do texto pode ser uma ferramenta de referencial no processamento para amenizar ambiguidades nos dados. Segundo o trabalho desenvolvido por Burnap et al. [2017], que analisou a comunicação relacionada ao suicídio na rede social Twitter, foram testados 1-5 grams, sendo que o melhor resultado alcançado foi com a utilização de 1-3 grams.

2.1.3 Bag of Words

A *bag of words* (BOW) é considerada uma representação clássica para a mineração de dados em documento de texto. Após o processo de tokenização, n-grams resultantes da entrada são contabilizados e o número de ocorrências de cada n-gram extraído do texto é armazenado em um vetor denominado BOW. Esse modelo de representação de características de um texto não considera a posição da palavra na entrada nem analisa o contexto em que é usada, apenas registra a quantidade de vezes que determinada ocorrência aparece no texto. Caso seja usada a representação total do texto, todos os tokens serão indexados [Srivastava & Sahami, 2009].

No entanto, outra alternativa é usar uma visão em que apenas palavras relevantes ao contexto serão indexadas. Essa escolha é feita por especialistas na área estudada que constroem o dicionário léxico do domínio. Os tokens indexados são chamados de palavras-chave e passam a exercer função indexadora. Usualmente, os tokens indexa-

dores são escolhidos de acordo com a sua função sintática. Substantivos carregam mais semântica do que verbos, advérbios ou adjetivos. A principal motivação para criar esse dicionário é a possibilidade de operar a indexação e a busca usando uma ferramenta controlada [Baeza-Yates & Ribeiro-Neto, 2011].

Após a definição dos elementos indexadores que irão fazer parte do dicionário, cada termo no dicionário é associado com pesos representando a sua positividade ou negatividade [Alshari et al., 2018]. O vetor de tokens é submetido ao dicionário léxico e, assim, a BOW é constituída pelos termos presentes no dicionário, descartando os ausentes.

A classificação usando a BOW leva em conta a medida de compatibilidade da palavra com a classe. No caso da detecção de ideação suicida, quando se analisa as palavras tristeza e alegria em relação a classe positiva (ideação suicida), por ser mais compatível com o contexto, a primeira terá um peso positivo atribuído a ela e a segunda, alegria, um peso negativo [Eisenstein, 2019]. Ainda segundo Eisenstein [2019], o BOW, por si só, já é bastante efetivo para classificação de texto. As Figuras 2.1 e 2.2 mostram a representação do *bag of words* de duas entradas: "Eu estou chorando de tristeza" e "Eu estou chorando de felicidade".

$$X_1 = \begin{bmatrix} eu & estou & chorando & de & tristeza \end{bmatrix}^t \tag{2.1}$$

$$X_2 = \begin{bmatrix} eu & estou & chorando & de & felicidade \end{bmatrix}^t \tag{2.2}$$

$$I = \begin{bmatrix} chorando : 3 & felicidade : -7 & morrer : 7 & tristeza : 6 & ... \end{bmatrix}^t \tag{2.3}$$

$$I_{x_1} = \begin{bmatrix} 3 & 0 & 0 & 6 & ... \end{bmatrix}^t \tag{2.4}$$

$$I_{x_2} = \begin{bmatrix} 3 & -7 & 0 & 0 & ... \end{bmatrix}^t \tag{2.5}$$

Os vetores 2.1 e 2.2 representam os tokens das duas entradas, X_1 e X_2. Enquanto o vetor I, 2.3, é a representação dos tokens indexadores com os seus respectivos pesos.

Os vetores 2.4 e 2.5 mostram os vetores de saída após o processamento, I_{x_1} e I_{x_2}, onde consta apenas os tokens presentes no vetor indexador, os ausentes foram descartados. É efetuada então a contagem dos tokens do dicionário presentes nas entradas, multiplicados pelos seus pesos atribuídos. O resultado final da somatória dos valores do vetor definem a qual classe a entrada pertence. No caso do domínio ideação suicida, o trabalho usa duas classes para separar os dados: ideação suicida e não ideação suicida. Os resultados: $I_{x_1} = 9$ e $I_{x_2} = -4$ mostram que a entrada I_{x_1} (com valor final positivo) apresenta um resultado mais compatível com a classe " ideação suicida ", enquanto a entrada I_{x_2} (com valor final negativo) faz parte do agrupamento da classe " não ideação suicida ".

O *bag of n-grams* é uma variação do BOW que usa n-grams ao invés de palavras. Goldberg [2017] afirma que essa versão seria mais poderosa do que a original por ser mais informativa do que os seus componentes individuais. O autor diz ainda que a melhor estratégia é testar vários tamanhos de n-grams durante a implementação por ser difícil definir, *a priori*, com quantos n-grams o algoritmo terá melhor resultado.

2.1.4 Term Frequency-Inverse Document Frequency (TF-IDF)

Os tokens comuns e raros que aparecem no texto mostram extremos importantes que são partes significativas na análise de sentimento. O TF-IDF é uma ferramenta relevante para a análise do impacto do token no domínio, faz parte da avaliação estatística com base em informações externas e é comumente usada como um complemento do BOW [Goldberg, 2017].

- TF é a frequência que o token aparece em um documento que faz parte de uma base de dados a ser analisada. Como forma de normalizar o dado, a TF (frequência do termo) é dividida pelo número de termos no documento.

  TF(t) = (quantidade de ocorrências do termo no documento) / (número total de termos presentes no documento).

- IDF mostra a relevância do termo para o domínio. Essa ferramenta atenua o impacto de tokens comuns e aumenta o peso de tokens raros. É medido pelo número de vezes que o token aparece em todas as entradas do universo analisado.

  IDF(t) = log_e(número total de dados / número de dados onde o termo t está presente).

$$W_{i,j} = tf_{i,j} \times \log(N/df_j) \tag{2.6}$$

Em que:

- W: resultado da análise da palavra i no texto j.
- $tf_{i,j}$: frequência da palavra i na entrada (texto) analisado j.
- N: número total de textos na base.
- df_j: número de entradas (textos) j onde há ocorrência da palavra analisada.

Considerando como exemplo:

- Entrada: um texto com 100 palavras.
- Palavras a serem analisadas: "suicídio" palavra que apresenta 3 ocorrências no texto de entrada e a palavra "e" que está presente 15 vezes no texto.
- tf_i: $3/100 = 0.03$
- tf_e: $15/100 = 0.15$
- Tamanho da base de dados(N): 1000 textos.
- df_i: 300 textos.
- df_e: 1000 textos.
- $IDF_{(i)}$: $log(1000/300) = 0.52$
- $IDF_{(e)}$: $log(1000/1000) = 0$

Sendo assim:

$$\textbf{TF-IDF}_{(i)} = 0.03 \times 0.52 = 0.0156 \tag{2.7}$$

$$\textbf{TF-IDF}_{(e)} = 0.15 \times 0 = 0.0 \tag{2.8}$$

A Equação 2.7 obteve um resultado maior do que a 2.8, mostrando assim que a palavra "suicídio" é mais relevante para o domínio. Ainda, o resultado obtido em 2.8 com a análise de "e" aponta a insignificância da palavra para o resultado da classificação do texto. Sendo assim, poderia ser adicionada ao grupo de *stopwords*. Esse processo de análise é feito com todas as palavras do BOW para que, com base nos números, seja possível identificar quais características são mais importantes para a classificação e, assim, continuam no processo, e quais serão retiradas — *stopwords*.

Como apresentado acima, o uso do TF sozinho apresenta um problema crítico para o PLN: todos os termos são considerados igualmente importantes quando se trata de avaliar a sua relevância para o contexto. O que ocorre de fato é que alguns termos têm pouco ou nenhum impacto na classificação. O IDF é um mecanismo que atenua os termos com alta frequência. Esse procedimento reduz o TF de um termo por um fator que aumenta de acordo a frequência dele na base de dados [Horn, 2008].

2.2 Seleção de Características

Quando se trabalha com um número grande de características em uma quantidade pequena de instâncias, a tarefa de classificação se torna um tanto desafiadora para os pesquisadores de aprendizado de máquina. Sendo assim, a seleção de características se torna relevante para diminuir a dimensionalidade da entrada enquanto melhora o desempenho do classificador [Bolón-Canedo et al., 2014]. A alta dimensionalidade pode melhorar o desempenho do classificador devido ao uso de mais características, contudo, alguns n-grams irrelevantes e redundantes não impactam os acertos do algoritmo na classificação, podem aumentar significativamente a complexidade computacional e o requisito de armazeamento de memória, o que é ainda mais preocupante em dados com milhares de características e apenas algumas dezenas de entradas [Hu et al., 2018].

Segundo Souza [2017], o desafio é responder à pergunta: "dado um número de características, como selecionar dentre elas as mais importantes, de modo a reduzir seu conjunto e, ao mesmo tempo, manter a maior gama possível de informações de seus dados?". A autora ainda complementa: a seleção de características com reduzido poder de discriminação entre dados acarreta um baixo desempenho do classificador, contudo, quando as características são bastante significativas para o contexto, o modelo de classificação poderá ser bastante simplificado. Existem 3 categorias principais de seleção de características: *wrapper, embedded* e o filtro [Bolón-Canedo et al., 2014].

O *wrapper* tem como base a performance de um algoritmo de classificação pré-definido pelo qual é avaliada a qualidade das características selecionadas. Dentre as heurísticas utilizadas, a busca exaustiva, os algoritmos genéticos e o *simulated annealing* são exemplos de técnicas de *wrapper* [Santoro & Nicoletti, 2005]. *Embedded* (embutido) usa algoritmos de aprendizado de máquina e então um subgrupo de características ótimo é construído pelo classificador. Chama-se "embutido" devido ao procedimento acontecer na construção do algoritmo de classificação, ou seja, faz parte do algoritmo — um exemplo típico é o procedimento de seleção dos atributos nos nós da árvore de decisão [Dias, 2015]. A filtragem, ao contrário dos dois primeiros, não interage com o algoritmo de classificação, sendo essa uma desvantagem segundo o autor. Porém, tem

como vantagem o custo computacional mais baixo [Hu et al., 2018].

O desempenho do PNL com utilização de seleção de características pelo método *wrapper* é superior ao que utiliza o filtro devido ao primeiro avaliar a qualidade do grupo de características selecionadas, etapa ignorada pelo filtro, o que resulta, no entanto, em um menor custo computacional para este último método [Souza, 2017]. Desde os anos 90, a comunidade científica tem preferido o uso de recursos mais baratos como filtros aos computacionalmente mais caros [Bolón-Canedo et al., 2014].

Um exemplo de filtro usado em seleção de características para PLN é o método estatístico *Chi-Square* (χ^2) que testa a relação entre as variáveis e as categorias. A hipótese nula do teste significa que as variáveis são independentes da classe — não há relação entre elas [Aldehim et al., 2014]. O papel dessa função na seleção de características é nortear a retirada dos n-grams que são mais prováveis de serem independentes — sem relação com a categoria e assim irrelevantes para o processo de classificação [Pedregosa et al., 2011]. Um exemplo do funcionamento dessa ferramenta seria perguntar se a variável "parque" tem alguma relação significante com a classe ideação suicida.

Vale ressaltar que outras técnicas citadas ao longo do texto também são consideradas como filtro de seleção de características: TF, TF-IDF e ganho de informação que será abordado mais à frente por fazer parte (embutido) do algoritmo de classificação árvore de decisão (subseção 2.3.3). Entretanto, em um estudo realizado por Forman et al. [2003], os métodos Ganho de Informação e χ^2 superaram de forma consistente os métodos TF e TF-IDF para seleção de características na classificação de textos. Os autores atribuem isso ao fato de que os métodos TF e TF-IDF não utilizam informações das classes no processo de filtragem.

2.3 Técnicas de Classificação de Texto

Classificar textos e extrair informações de forma automatizada se tornam cada vez mais importantes pela velocidade e quantidade de material digital produzido no mundo e a incapacidade de processamento humano nessas condições. As informações extraídas são, por sua vez, transformadas em conhecimentos que são empregados em diversas áreas como saúde, educação e economia. O uso de algoritmos de aprendizagem de máquina na classificação de textos tem se mostrado bastante promissor pela possibilidade de generalização quando se constrói uma ferramenta robusta, com bons resultados nas métricas de avaliação de qualidade [Esteva et al., 2019].

O aprendizado de máquina é subdividido em dois tipos: não supervisionado e supervisionado. O não supervisionado é geralmente usado em conjunto no qual não há dados de treinamento disponíveis — dados que não apresentam rótulo, enquanto que

o supervisionado precisa da disponibilidade de dados rotulados para o treinamento [Baeza-Yates & Ribeiro-Neto, 2011]. Modelos não supervisionados são comumente aplicados em domínios de investigações de fraudes em sistemas bancários para identificar movimentações não usuais e também no domínio de análise de afinidade para identificar a associatividade nos itens comprados pelos consumidores. No caso do supervisionado, os dados rotulados permitem que o grupo de dados seja subdividido em dados de treinamento e dados de teste. O algoritmo usa os dados de treinamento para aprender as similaridades e assim, relacionar as características do texto com o rótulo pré definido. Com esse procedimento, a máquina consegue transformar o seu aprendizado em uma função de classificação que é testada em dados ainda não submetidos ao algoritmo (conjunto de teste). Após a avaliação da qualidade da classificação do algoritmo, ele pode ser usado para classificar dados não rotulados dentro do mesmo contexto.

A classificação de dados em linguagem natural pode ser entendida como um problema de otimização, no qual, um texto, que pertence ao grupo de dados estudado, é a entrada para o algoritmo e a saída é um rótulo resultante do mapeamento realizado pelo algoritmo [Eisenstein, 2019]. Ainda, segundo Eisenstein [2019], os algoritmos de processamento de linguagem natural (PNL) apresentam 2 módulos distintos: busca — responsável pelo *argmax* da função Ψ, neste caso, o módulo de busca encontra a saída $\widehat{y}$ que representa o melhor resultado para a entrada X; aprendizagem — é responsável por encontrar os parâmentros no vetor Θ e processá-los.

$$\widehat{y} = argmax_{y \in Y(x)} \Psi(x, y; \Theta) \tag{2.9}$$

Na Equação 2.9, em que: x: entrada, pertence à base de dados χ; y: saída que é um elemento do conjunto $Y(x)$; Ψ: função (modelo) que mapeia $\chi \rightarrow Y$; Θ: um vetor de parâmetros para Ψ; $\widehat{y}$: saída, prevista para a entrada de acordo com o modelo usado.

A seguir, serão abordados os principais algoritmos de aprendizagem de máquina supervisionado que são usados para o estudo de dados relacionados ao tema suicídio: *Naïve Bayes*, *Support Vector Machine*, Árvore de Decisão, Floresta Aleatória e Rede Neural.

2.3.1 Naïve Bayes (NB)

O Naïve Bayes é um algoritmo muito usado em mineração de dados e aprendizagem de máquina por ser simples, eficiente e por sua complexidade de tempo ser linear $\theta(n)$ — em que n é o tamanho do grupo de treinamento. O espaço computacional utilizado é o produto da multiplicação do número de atributos, número de classes e número de

valores por atributos — sendo esse produto o espaço ocupado pela tabela de frequência [Zhang et al., 2000]. A eficiência do NB possibilita, por exemplo, a filtragem de e-mails através de ferramentas anti-spam [Hovold, 2005]. Outra aplicação interessante desse classificador, que mostra sua robustez, é em estratégias mercadológicas, por possibilitar o monitoramento de *feedback* de clientes, funcionários e fornecedores [Sánchez-Franco et al., 2019].

Considerado um algoritmo de classificação probabilístico baseado no teorema de Bayes, a versão mais comum do NB assume que o peso dos tokens é binário (0,1), presente ou ausente. Essa representação é conhecida como Bernoulli. No entanto, o classificador pode ser modificado para incluir a frequência dos termos e assim melhorar a qualidade do resultado, variação conhecida como multinomial [Baeza-Yates & Ribeiro-Neto, 2011]. No caso do uso da frequência, palavras que não estão presentes no documento não apresentam impacto na análise e, sendo assim, na multiplicação das probabilidades recebem o valor 1 [Eisenstein, 2019].

Esse método opera classificando previamente as informações contidas no grupo de treinamento, construindo uma tabela de frequência com todas as palavras do vetor (BOW). Por não considerar nem avaliar a relação entre os atributos do vetor, cada n-gram é tratado como uma unidade independente. Essa peculiaridade faz com que seja conhecido como um algoritmo ingênuo — *naive*.Nesse sentido, vale ressaltar que tratando os n-grams como unidades independentes dificulta o processamento do texto quando há ambiguidades, como já citado no início do capítulo, as palavras adjacentes são usadas como referencial diminuindo assim a dificuldade do algoritmo na tomada de decisão de classificação. A teoria de decisão de Bayes trata da probabilidade condicional. Para entender, suponha dois eventos independentes A e B. A probabilidade condicional, dada por P(A/B) é a probabilidade de o evento A ocorrer dado que o evento B ocorreu. Essa probabilidade pode ser obtida com:

$$P_{(A/B)} = \frac{P_{(B/A)} \times P_{(A)}}{P_{(B)}} \tag{2.10}$$

A Equação 2.10 mostra: P(A/B): probabilidade de A acontecer, dado B; P(A): probabilidade da classe A; P(B/A): Probabilidade de B acontecer, dado A; P(B): probabilidade da classe B.

2.3.1.1 Exemplificando o funcionamento do algoritmo Naïve Bayes.

A base de dados após o pré-processamento é subdividida em dois grupos: o de treinamento para construção do modelo e o de teste para avaliação da qualidade do classifi-

cador. No exemplo a ser usado, o conjunto de treinamento é composto por 12 textos, sendo 8 da classe 0 (não ideaçao suicida) e 4 da classe 1 (ideação suicida). Nesse universo hipotético existem apenas 4 tokens que são apresentados abaixo na tabela de frequência.

classe	cansado	vida	dinheiro	viajar
0	1	2	5	6
1	3	2	2	1

Tabela 2.1: Frequência dos tokens nos dados de treinamento.

A Tabela 2.1 mostra a contagem dos tokens nas classes. O leitor poderá observar que nos 8 textos da classe 0, a palavra dinheiro apresenta 5 ocorrências, por exemplo.

Considerando as classes 0 e 1 é possível calcular $P_{(0)}$ e $P_{(1)}$:

- $P_{(0)} = P_{(0)}/(P_{(0)} + P_{(1)}) = 8/12$
- $P_{(1)} = P_{(1)}/(P_{(0)} + P_{(1)}) = 4/12$

Calculando as frequências dos tokens nas classes, palavra "cansado":

- 1/14: são 14 palavras na classe 0 e a palavra cansado ocorre apenas 1 vez.
- 3/8: das 8 ocorrências na classe 1, 3 delas são o token cansado.

classe	cansado	vida	dinheiro	viajar
0	0,071	0,143	0,357	0,428
1	0,375	0,25	0,25	0,125

Tabela 2.2: Frequência dos tokens nas suas respectivas classes.

As frequências dos tokens nas classes, mostradas na Tabela 2.2, serão usadas para calcular a probabilidade relativa de a entrada pertencer à classe 0 ou 1. Após o treinamento, segue a etapa de teste do modelo. No caso, a frase escolhida para avaliar o modelo é "Estou cansado da vida", que faz parte da classe 1 do conjunto de dados.

Avaliando a probabilidade relativa de a frase ser da classe 0: o algoritmo usa a probabilidade $P_{(0)}$ e a frequência dos tokens acima na classe 0.

$$8/12 \times 0,071 \times 0,143 = 0,0068$$

Avaliando a frase em relação $P_{(1)}$, classe 1 :

$$4/12 \times 0,375 \times 0,25 = 0,0312$$

Os resultados mostram que a probabilidade da entrada "Estou cansado da vida." ser da classe 1 é maior em relação à classe 0. Sendo assim, o modelo rotulou corretamente a entrada.

Apesar de ser um algoritmo popular pelas suas características de implementação e eficiência, existem algumas limitações no uso dessa ferramenta: não consegue operar dados de forma satisfatória no domínio binário quando esses não são separáveis por uma função linear; também, não consegue ser eficiente para todos os grupos de dados que podem ser classificados por funções lineares [Zhang et al., 2000].

2.3.2 Support Vector Machine (SVM)

O SVM é um método de aprendizagem de máquina supervisionado desenvolvido por Boser et al. [1992], que usa espaço vetorial com o objetivo de separar o conjunto de dados em classes. O algoritmo trabalha com hiper-planos com o objetivo de encontrar uma superfície de decisão final que maximiza a separação das classes. O nome do algoritmo tem origem nos vetores de suporte que são a representação da amostra no plano. Os dois hiperplanos paralelos são tangenciais a pelo menos um dado de cada classe. A distância entre os dois hiperplanos é a margem que o classificador deve maximizar [Berry & Kogan, 2010].

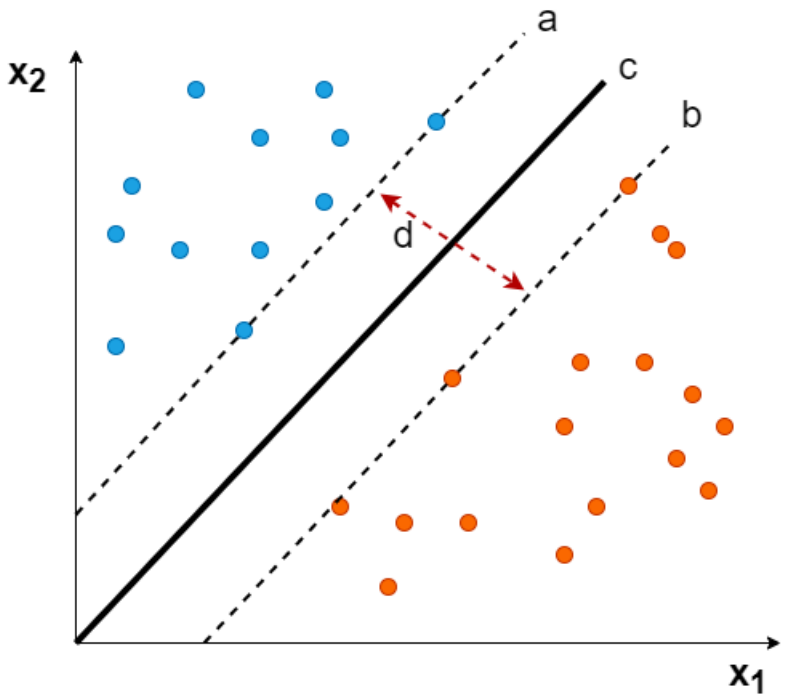

Figura 2.2: Support Vector Machine (SVM)

A Figura 2.2 mostra as classes 0 (azul) e 1 (laranja) sendo separadas por um hiper-

plano (c) que foi auxiliado pelos dois hiperplanos (a e b) de forma que a margem (d) fosse maximizada.

Os n-grams compõem um espaço t-dimensional. O número de dimensões corresponde à quantidade de n-grams, onde os documentos são representados como pontos ou vetores [Baeza-Yates & Ribeiro-Neto, 2011]. A função do *kernel* (componente principal do algoritmo, núcleo) é transformar os tokens em entidades algébricas, tornando as entradas estruturadas no espaço multidimensional que independentemente do seu tamanho pode ser treinado em tempo polinomial. Portanto, podemos citar duas propriedades fundamentais do processo: o acesso à alta dimensionalidade a um baixo custo computacional e análise de padrões com otimização do cálculo convexo de forma eficiente sem ter o mínimo local como obstáculo [Srivastava & Sahami, 2009]. Ainda segundo o autor, a função que traça os hiperplanos pode ser linear, polinomial, sigmoidal ou uma combinação dessas. A escolha do tipo a ser utilizado está intrinsecamente relacionada à natureza e distribuição dos dados no plano. O SVM linear é reconhecido como uma das melhores ferramentas de classificação de texto [Srivastava & Sahami, 2009].

Os vetores de suporte são eficientes na classificação binária, definindo se uma entrada pertence a uma determinada classe ou não. Quando o universo avaliado apresenta múltiplas classes, um classificador diferente para cada classe precisa ser aprendido. A estratégia é reduzir o problema à classificação binária: o texto pertence a essa classe ou a uma das k-1 outras classes? Essa estratégia é conhecida como "um contra todos". Se o número total de classes é k, o processo de classificação é repetido k vezes [Baeza-Yates & Ribeiro-Neto, 2011]. Ainda falando sobre eficiência do SVM, estudos mostram que quando o token é constituído de apenas uma palavra (unigram), o SVM tem uma performance melhor do que o NB [Pu et al., 2017].

2.3.3 Ávore de Decisão (AD)

A arquitetura desse algoritmo é similar, como o nome indica, a uma árvore. Ao longo do tronco são tomadas decisões que levam às folhas que representam a classe do dado. A AD é um algoritmo de classificação supervisionado que usa um grupo de treinamento para construir um conjunto de regras organizadas de classificação que indicam o caminho do nó raiz através dos nós de decisão até os nós terminais, que são as classes. Ao longo desse caminho, as ligações entre os nós intermediários são os valores que o atributo, n-gram, instanciado pode assumir, sim (1) e não (0). Ainda, o conjunto de regras construído, após a avaliação da qualidade da classificação, pode ser generalizada para a rotulagem de outros dados que fazem parte do mesmo domínio. As vantagens, desse modelo consistem na visualização das decisões da árvore e, assim, facilita a in-

terpretação dos resultados do processo de classificação [Baeza-Yates & Ribeiro-Neto, 2011].

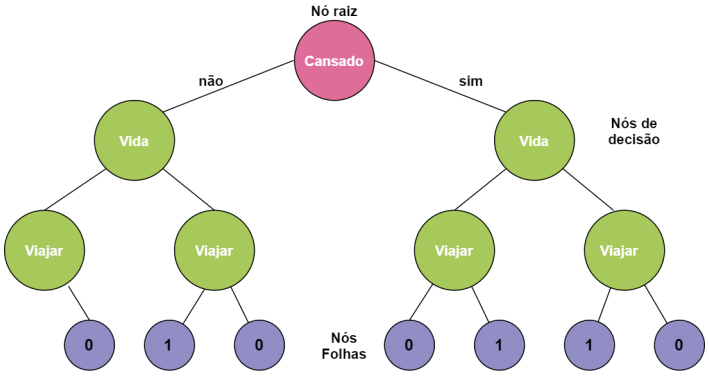

Figura 2.3: Árvore de Decisão

A Figura 2.3 ilustra o uso do algoritmo em análise de sentimento. Os n-grams são representados como raiz (cor rosa) e nós intermediários (verde). Cada nó tem duas saídas: não(0), indicando a ausência daquele token no texto ou sim (1), indicando a sua presença. Assim, a árvore vai sendo construída até os nós terminais (folhas, em azul) que são as representações das classes: sem ideação suicida (0) e com ideação suicida (1). A árvore foi construída de forma simplista e teórica com apenas 3 atributos. Na prática, são inúmeros tokens que são instanciados como nós intermediários até chegar nos nós que representam as classes.

No processo de utilização dessa ferramenta, o passo fundamental é a construção da árvore inicial com os dados de treinamento. Ele pode ser feito através de alguns métodos como o recursivo e o ganho de informação [Quinlan, 1993].

Quando se constrói a árvore recursivamente usando o método "dividir para conquistar", o conjunto de dados de treinamento será subdividido continuamente até que a partição resultante contenha dados de apenas uma das classes ou até que a subdivisão não ofereça nenhum melhoramento na classificação — o resultado, usualmente, é uma árvore grande e complexa, com mais estrutura do que o justificável para a base de treinamento [Quinlan, 1993].

As árvores de decisão são primeiramente construídas usando um número grande de atributos para em seguida usar a poda (eliminação de tokens) como forma de redimensionar a estrutura. A poda é necessária para evitar o *overfitting*, que ocorre quando o classificador toma decisões baseadas em tokens que ocorrem raramente no grupo de treinamento, se ajustando demais aos dados e prejudicando a generalização do modelo, tendo como consequência erros no conjunto de teste [Manning & Schütze, 1999]. A simplificação da árvore pode ser feita através de duas formas: decisão de não prosseguir com a subdivisão das instâncias de treinamento ou remover retrospectivamente

partes da estrutura que foi construída previamente; é importante observar a melhor forma de particionar os dados e avaliar a partição sob o ponto de vista de significância estatística, ganho de informação, redução de errors e outros [Quinlan, 1993].

Observar o balanceamento da árvore é relevante para o desempenho do modelo e o principal desafio na implementação é ter uma estrutura o mais balanceada possível. Os recursos mais usados para controlar a arquitetura (forma e tamanho) da árvore é o Ganho de Informação (GI) e a Entropia (E) [Baeza-Yates & Ribeiro-Neto, 2011]. A Entropia mede o nível de incerteza de um determinado atributo, ou seja, quanto maior a entropia menor o ganho de informação de um determinado atributo. A seleção de termos com alto GI tende a gerar árvores menores e menos complexas. Portanto, quanto maior o GI, maior prioridade na inclusão na árvore. A Equação 2.11 mostra o cálculo da Entropia e, em seguida, 2.12 mostra o GI para o n-gram A:

$$Entropia(S) \equiv -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus} \tag{2.11}$$

Na equação 2.11: S é um conjunto de dados escolhidos aleatoriamente no grupo de treinamento; $p_{\oplus}$ é a proporção de dados positivos em S (classe 1); $p_{\ominus}$ é a proporção de dados negativos em S (Classe 0);

$$GI(S, A) \equiv Entropia(S) \; - \sum_{v \in Valores(A)} \frac{|S_v|}{|S|} Entropia(S_v) p_{\ominus} \tag{2.12}$$

Em que, em 2.12: A é o atributo de interesse, S_v representa a frequência do valor v em S.

Um dos principais métodos utilizados para construir uma árvore de decisão a partir de uma base de instâncias é o ID3 [Quinlan, 1979]. O ID3 constrói a árvore considerando a questão "Qual atributo é o mais importante e, portanto, deve ser colocado na raiz da árvore?" e esse será o nó raiz. Para isso, cada atributo é testado e sua capacidade para se tornar nó raiz é avaliada. Em seguida, criam-se tantos nós filhos da raiz quantos valores possíveis esse atributo puder assumir (caso discreto). Repete-se o processo para cada nó filho da raiz e assim sucessivamente. Para decidir qual nó é o mais importante, utiliza-se o Ganho de Informação (GI), calculado em função da Entropia (E).

O uso do cálculo do GI para melhorar a performance do algoritmo deve ser observado juntamente com a escolha do tamanho do subconjunto de n-grams da BOW que farão parte da construção da árvore. Segundo Eisenstein [2019], escalar a AD para o tamanho das entradas do BOW é difícil — esse modelo tem uma taxa de eficiência maior quando um grupo de atributos mais compacto é utilizado.

Corroborando com a observação acima, um estudo com o objetivo de identificar

comportamento predatório *online* utilizou 25 transcritos de diálogos: 349 a 1500 linhas de texto retirados da internet. Os predadores eram homens com idade entre 23 a 58 anos e todos condenados por pedofilia. O estudo era de classificação binária (predador ou vítima) onde 16 trechos dos transcritos foram utilizados para cada classe que contou com oito atributos. Nessas condições, o classificador AD teve sucesso em 60% da rotulagem contra 50% do BOW usado como classificador em experimento muito similar [Berry & Kogan, 2010].

2.3.4 Floresta Aleatôria (FA)

A Floresta Aleatória é um modelo que faz previsões através da média dos resultados obtidos dos inúmeros modelos base (árvores) independentes. É um algoritmo usado para classificação genérica [Denil et al., 2014]. Essa forma de trabalhar escolhendo o rótulo mais popular dentre as árvores para definir a classe final da instância acarretou em um melhoramento significativo da acurácia na classificação em relação à árvore de decisão [Breiman, 2001].

O método de construção da floresta é aleatória devido ao fato de que a escolha dos dois componentes (instâncias e atributos),usados na construção das árvores serem feitos de forma randomica através de sorteio. Segundo Breiman [2001], esse também é um fator que influencia positivamente na acurácia. O algoritmo segue os seguintes passos:

- Um subconjunto (x) dos dados de treinamento (X) é escolhido aleatoriamente; a unicidade não é obrigatória: $x \subset X$.
- Se o número total de atributos é M, um número $m < M$ é especificado. A cada nó, m atributos são selecionados aleatoriamente. Os m atributos são avaliados pelo GI e o melhor dentre eles é escolhido para ser o nó. E assim sucessivamente as árvores da floresta vão sendo criadas.
- Não há poda nas estruturas das árvores, elas são as maiores possíveis.
- As x árvores construídas a partir dos dados de treinamento são utilizadas para validação do algoritmo no conjunto de teste. Cada instância desse grupo é avaliada pela floresta. Cada árvore individual define um rótulo para a instância. O rótulo final da instância é aquele que teve o maior número de votos — foi a classe escolhida pelo maior número de árvores.

Na Figura 2.4, é feita uma representação simplista e teórica do funcionamento do algoritmo, em que são usados 5 atributos na construção das 4 árvores da floresta. O

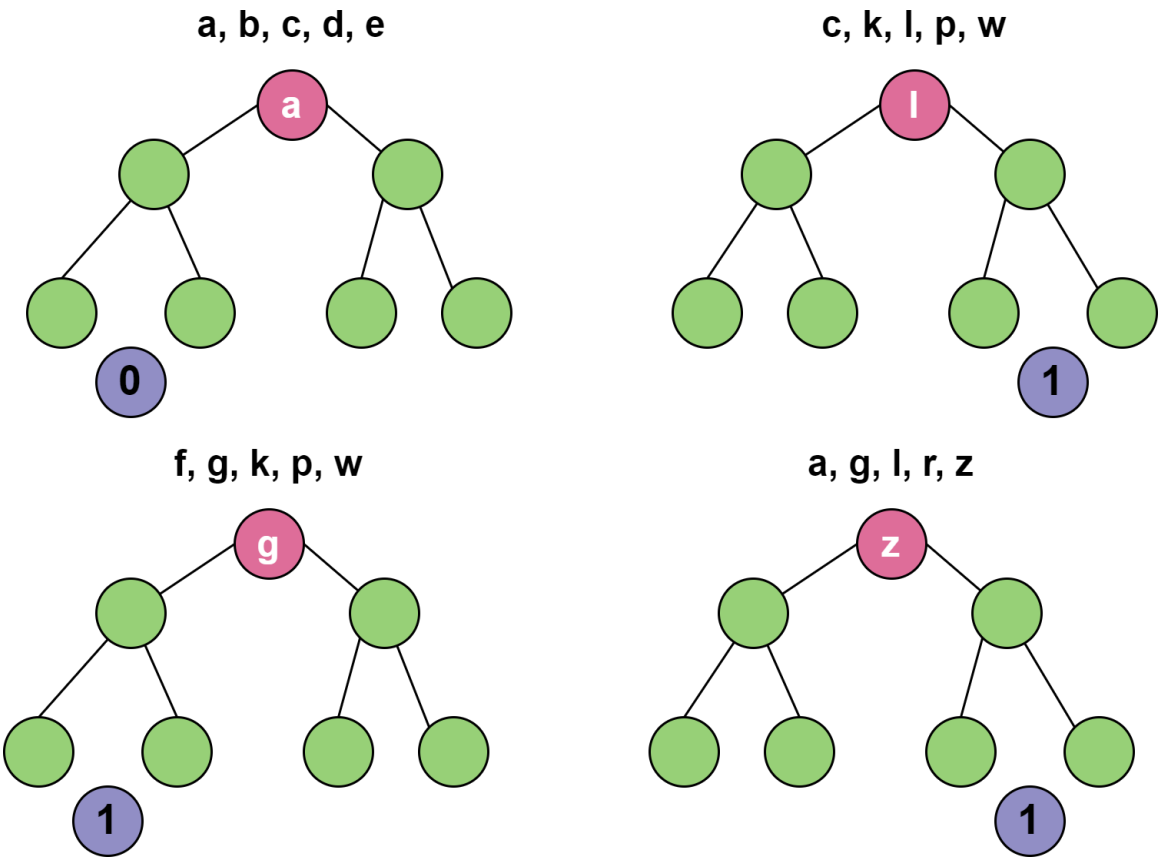

Figura 2.4: Floresta Aleatória

nó raiz e os nós de decisão subsequentes são escolhidos através do sorteio aleatório de 3 dos atributos e a avaliação de GI deles em cada nó, sendo o de maior relevância instanciado naquele ponto.

Após a criação da floresta, usando quatro instâncias como treinamento, um dado de teste foi submetido à floresta, tendo o resultado de 75% da floresta sido a classe 1 (com ideação suicida) e 25% classe 0 (sem ideação suicida). O rótulo final é definido pela maioria das árvores, sendo, nesse caso, a classe 1 a vencedora.

A floresta aleatória é um modelo eficaz na previsão de rótulos por ter a característica de não favorecer ao *overfitting* devido ao grande número de árvores prevenir o ajuste excessivo dos dados; ademais, a aleatoridade a torna um classificador mais preciso [Breiman, 2001].

Apesar de ser um método mais robusto do que a AD, devido ao *overfitting* raramente acontecer, algumas das vantagens do modelo anterior se tornam desvantagens no presente: a interpretação dos resultados é muito mais difícil devido à quantidade de árvores; também, é um modelo mais lento por a entrada ter que ser processada por todos os elementos da floresta e não apenas por uma árvore como no algoritmo exposto anteriormente.

2.3.5 Redes Neurais (RN)

As redes neurais artificiais fazem parte da família de técnicas que foram historicamente inspiradas na forma com que o cérebro trabalha o raciocínio para resolver problemas e que podem ser caracterizadas em aprendizagem de funções matemáticas diferenciadas e parametrizadas. Em uma rede, podem existir várias camadas encadeadas dessas funções [Goldberg, 2017]. As camadas são formadas por neurônios e, por conseguinte, a aprendizagem pode ser definida de forma mais simplista como uma entrada que passa por várias camadas encadeadas que, após todo o processamento, devolve uma saída como resultado. Abaixo, segue a imagem representativa de uma unidade básica da rede neural, o neurônio.

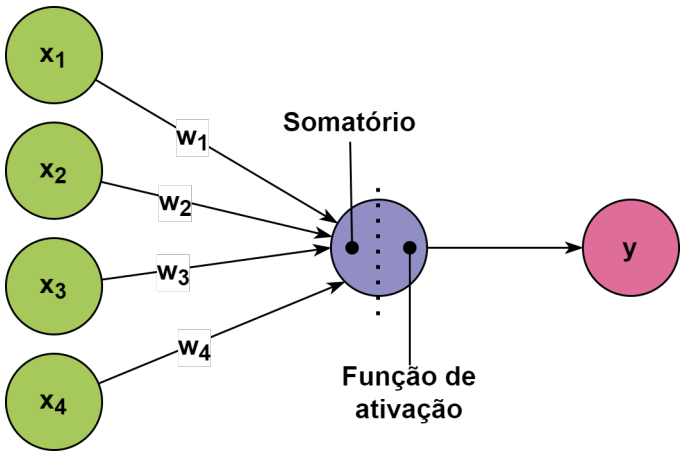

Figura 2.5: Neurônio

A Figura 2.5 mostra um neurônio com 4 entradas (x_i), pesos (w_i) nas sinapses, somatório ($\sum_{i=1}^{4} x_i \times w_i$), função de ativação e uma saída (*output*).

Gurney [2004] explica que, nas sinapses (que são as ligações de entrada nos neurônios), é atribuído um peso para multiplicação da entrada antes dela atingir o procedimento de ativação. Uma função processa as entradas modificadas e o resultado da ativação, então é comparado com um limite: se excede esse limite, a saída é o valor 1; caso contrário, recebe o valor 0. As funções usadas na ativação podem ser diversas, como, por exemplo: linear, degrau, sigmoide, tanh, ReLu (*Rectified Linear Unit*) e suas derivadas. Portanto, a rede neural artificial que é formada por neurônios, depende de 3 aspectos fundamentais: entradas, pesos nas sinapses e função de ativação. Como as entradas e a função de ativação são fixas, o comportamento da rede é definido pelos pesos [Akinsola, 2017].

A rede neural mais simples opera com uma função de ativação linear e é conhecida como *Perceptron*. Indo além desse modelo, com várias camadas escondidas e podendo

apresentar funções não lineares, temos o **Multi Layer Perceptron** [Goldberg, 2017]. O fluxo do processamento pode ocorrer como *back-propagation* ou *feed-forward*.

Quando o algoritmo de treinamento depende de erros que aconteceram anteriormente na rede, em neurônios passados, ele é referido como *back-propagation*. Ao contrário, o design *feed-forward* não busca erros anteriores ao neurônio atual para processar a entrada recebida [Gurney, 2004].

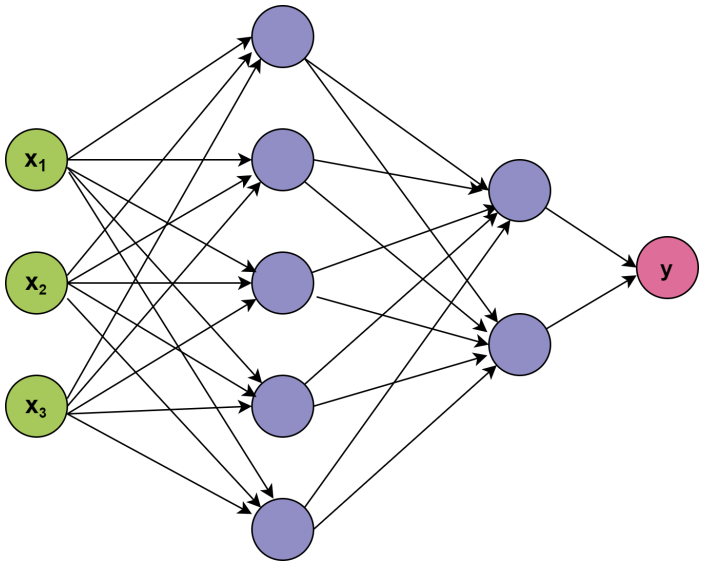

Figura 2.6: Rede neural

A Figura 2.6 mostra uma rede neural *feed-forward* com 3 entradas (x_i), 5 neurônios na primeira camada, 2 neurônios na segunda e saída (y).

Embora o vetor resultante do processamento seja difícil de ser interpretado, essa técnica vem sendo aplicada com grande êxito na área de processamento de linguagem natural [Li et al., 2016].

A rede neural é considerada uma ferramenta poderosa no que tange ao tamanho da amostra a ser classificada e, por conseguinte, vem se popularizando ao longo das décadas com crescente interesse da comunidade científica em trabalhar com camadas profundas, ultrapassando métodos mais clássicos na área — especialmente quando se trata de reconhecimento de padrões [Albawi et al., 2017].

O *deep learning* é o termo usado para redes neurais complexas com muitos neurônios dispostos em camadas profundas. Existem dois tipos básicos de arquitetura de redes neurais profundas: redes recorrentes/recursivas e as não recorrentes/não recursivas (*feed-forward networks*). As técnicas mais comuns para o presente domínio são LSTM (*long-short term memory*) e CNN (*Convolutional Neural Networks*) [Goldberg, 2017].

A LSTM usa a estrutura de redes neurais recorrentes (RNN) e é uma técnica que tem como ponto forte a sua habilidade de priorizar mais as palavras-chave do que os

outros modelos. A composição das orações são processadas de forma competitiva para que o modelo consiga capturar a assimetria negativa que é uma propriedade semântica importante para compreender a linguagem natural. Há um entendimento nítido de localidade dimensional nesse algoritmo com certas dimensões marcando negação e qualificando de forma muito localista [Li et al., 2016].

Apesar de as redes com estrutura CNN apresentarem um melhor desempenho na análise de sentimento, as duas arquiteturas funcionam de forma complementar. Enquanto a RNN calcula uma combinação ponderada de todas as palavras da frase, a CNN trabalha com os tokens mais significativos, considerando apenas o resultados das suas ativações [Yin et al., 2017].

Agregando mais poder de processamento a essas técnicas, avanços recentes na área mostram resultados animadores no uso do *Bidirectional Encoder Representations from Transformers* (BERT) que atua na fase de treinamento do modelo [de Carvalho et al., 2020]. Esse algoritmo, baseado em *transformers*, foi criado para pré-treinar textos sem classificação, não rotulados. O *transformer* é um tipo de arquitetura de rede neural composto por codificadores e decodificadores que, por sua vez, são RNN compostos por apenas uma unidade de recorrência. Esse tipo de algoritmo alcançou o estado da arte nas traduções de inglês-alemão e inglês-francês [de Carvalho et al., 2020].

Embora os trabalhos publicados na área com o uso dessa nova tecnologia apontarem para um novo direcionamento de processamento de linguagem natural, o presente trabalho abordará apenas os métodos clássicos pela quantidade de dados disponíveis para o estudo, o custo de processamento e o aspecto prático que foi estabelecido.

2.4 Métricas de Avaliação

A avaliação é muito importante para métodos de classificação. É através desses resultados e também do *benchmarking* (comparação com os resultados de outros métodos de classificação) que se determina se o algoritmo escolhido teve um bom desempenho ou não. As métricas de avaliação são necessárias para validar o método de classificação proposto [Baeza-Yates & Ribeiro-Neto, 2011].

Quando se trata de medir o desempenho do modelo de classificação, uma matriz de confusão é uma ferramenta bastante útil para uma melhor visualização de como as métricas foram construídas. Antes de elencar as métricas usadas no trabalho, por uma melhor linearidade, a Figura 2.7 mostra um modelo de matriz de confusão que será exemplificada com o domínio ideação suicida:

- **Verdadeiro Positivo (VP):** nesse quadrante da matriz, encontram-se os dados que são positivos para o domínio e foram classificados como tal. Textos que eram

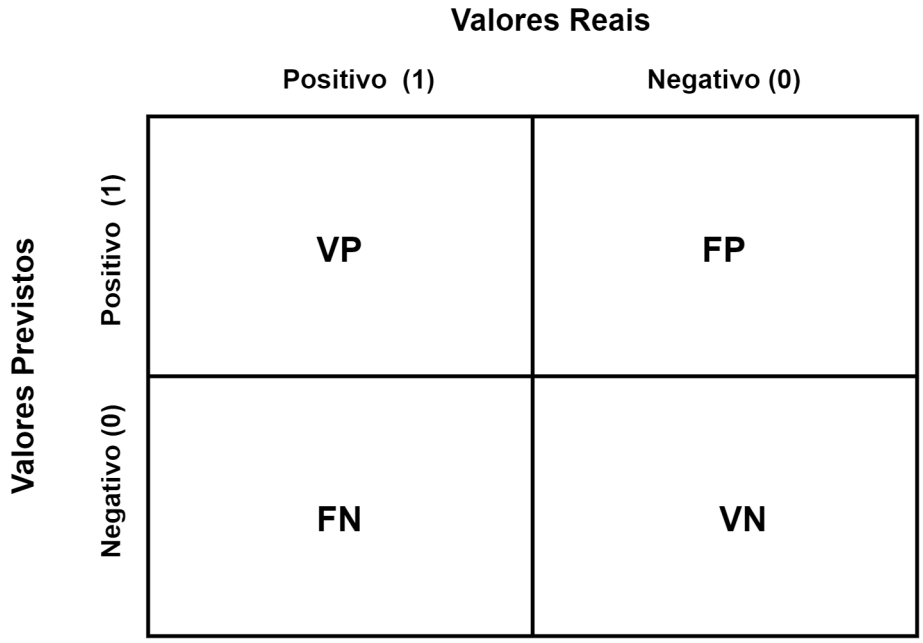

Figura 2.7: Matriz de Confusão

de ideação suicida e foram previstos como ideação suicida pelo modelo.

- **Falso Positivo (FP):** nesse quadrante da matriz, encontram-se os dados que são negativos para o domínio e foram classificados erroneamente como positivos. Textos que não eram de ideação suicida e foram previstos como ideação suicida pelo modelo.

- **Verdadeiro Negativo (VN):** nesse quadrante da matriz, encontram-se os dados que são negativos para o domínio e foram classificados como tal. Textos que não eram de ideação suicida e foram previstos corretamente pelo modelo.

- **Falso Negativo (FN):** nesse quadrante da matriz, encontram-se os dados que são positivos para o domínio e foram classificados erroneamente como negativos. Textos que eram de ideação suicida e foram previstos o contrário, não ideação suicida, pelo modelo.

2.4.1 Acurácia

Essa métrica mostra a capacidade de o algoritmo identificar corretamente as classes dos dados, ou seja, qual a frequência de acerto na classificação. Segundo Baeza-Yates & Ribeiro-Neto [2011], é a fração dos dados de treinamento que são rotulados corretamente pelo algoritmo. A matriz de confusão na Figura 2.8 e a Equação 2.13 mostram como deve ser feito o cálculo da acurácia:

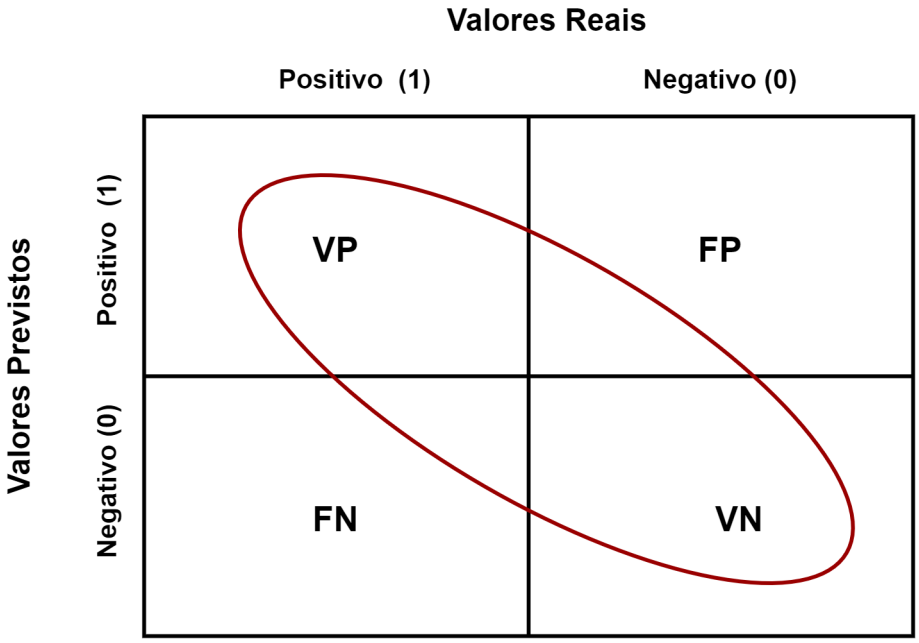

Figura 2.8: Matriz de Confusão - acurácia

$$Acuracia = \frac{VP + VN}{VP + VN + FP + FN} \tag{2.13}$$

Considerando uma classificação binária onde a composição dos dados tem a seguinte distribuição: 30 textos de ideação suicida (1) e 70 textos de não ideação suicida (0). Suponha-se que o algoritmo de classificação tenha identificado 50 textos para cada classe, da seguinte forma:

- **VP = 10**, textos da classe de ideação suicida que foram classificados corretamente como ideação suicida.
- **FP = 40**, textos da classe de não ideação suicida que foram classificados erroneamente como ideação suicida.
- **VN = 30**, textos da classe de não ideação suicida e que foram classificados corretamente como não ideação suicida.
- **FN = 20**, textos da classe de ideação suicida e que foram classificados erroneamente como não ideação suicida.

$$Acuracia = \frac{10 + 30}{10 + 30 + 40 + 20} = 0,4 \tag{2.14}$$

A equação 2.14 Acurácia de 0,4 significa que o algoritmo de classificação obteve sucesso em 40% da rotulagem, ele conseguiu identificar a classe correta de 40% das instâncias .

A interdependência do método de aprendizagem e do resultado da acurácia no grupo de treinamento é uma importante característica de muitos algoritmos de classificação. Os procedimentos de treinamento são, muitas vezes, caros computacionalmente. Sendo assim, evitar grupos grandes pode ser uma vantagem. Porém, dados insuficientes para o treinamento podem resultar em uma acurácia abaixo do esperado. Sintetizando, um grupo de treinamento muito complexo (muitas variáveis) pode ter o seu resultado de acurácia baixo devido ao *overfitting* nos dados de treinamento. No caso inverso, um grupo com complexidade muito baixa não consegue fazer o máximo uso do algoritmo, o que novamente acarreta uma baixa acurácia em dados de teste. É preciso encontrar o equilíbrio e o uso do recurso de *cross-validation* é uma possibilidade de melhorar a performance do algoritmo [Manning & Schütze, 1999]. O *overfitting* acontece quando o algoritmo apresenta uma alta acurácia no conjunto de dados com o qual foi treinado, mas não consegue repetir a mesma performance em outros dados, mostrando que houve um sobreajuste no treinamento e inviabilizando assim a reutilização e generalização do algoritmo. Implementar a estratégia de dividir o conjunto de dados em dados de treinamento e dados de teste (*cross-validation*) é uma forma de tratar o *overfitting* e ajustar o modelo.

Medir a performance do algoritmo é importante, todavia, o valor da acurácia em si não significa muito. A análise do valor obtido na Equação 2.14 deve levar em conta o grau de dificuldade do contexto estudado. Um bom exemplo seria a tradução de textos em inglês: apesar de a acurácia de 90% ser facilmente alcançada por humanos, esse valor ainda está além da capacidade dos tradutores automáticos [Manning & Schütze, 1999].

Eisenstein [2019] relata a dificuldade dessa métrica quando se trata de avaliar modelos que trabalham com dados não balanceados, exemplificando com um caso extremo: avaliar a acurácia de um algoritmo de classificação binária para as classes ideação suicida que conta com 1% dos dados e não ideação suicida que conta com os restantes 99% dos dados. Caso o modelo classifique todos os dados como negativos, pertencentes à classe não ideação suicida, ele teria uma acurácia de 99%. Quando se aplica um modelo de classificação em um conjunto de dados, espera-se que esse modelo seja eficiente na discriminação das classes e que a métrica utilizada para avaliação dele seja capaz de detectar essa habilidade ainda que o conjunto de dados seja desbalanceado. No exemplo citado, a acurácia pode ser descartada, não é confiável. Portanto, como na prática o universo de dados não balanceados é extenso, são necessárias métricas adicionais para

medir o desempenho dos classificadores.

2.4.2 Precisão

Essa métrica de avaliação identifica o número de positivos verdadeiros do total de todos os positivos obtidos, ou seja, a proporção dos itens da classe que o sistema identificou corretamente dentre todos os itens dessa classe. Caso o resultado da análise seja uma precisão muito baixa, isso indica um grande número de falsos positivos. Do contrário, uma alta identificação de positivos verdadeiros. Abaixo, a matriz de confusão na Figura 2.9 seguida pela Equação 2.15 mostram como deve ser o cálculo da precisão:

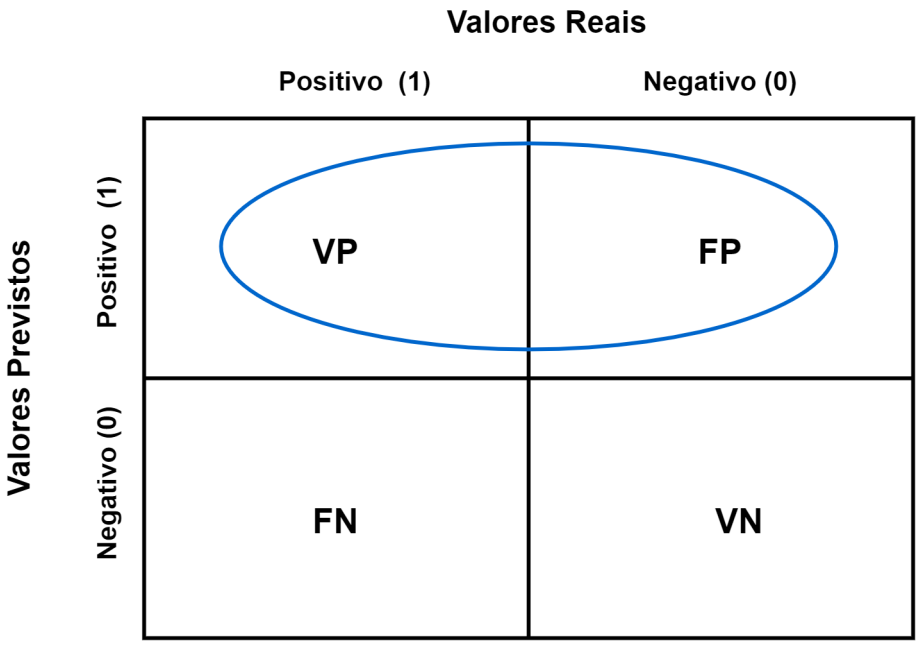

Figura 2.9: Matriz de Confusão - precisão

$$Precisao = \frac{VP}{VP + FP} \tag{2.15}$$

Trabalhando com a classificação binária do exemplo acima: 30 textos de ideação suicida (1) e 70 textos de não ideação suicida (0). Abordando o cálculo da precisão:

- **VP = 10**, textos da classe de ideação suicida e que foram classificados corretamente como ideação suicida.
- **FP = 40**, textos da classe de não ideação suicida e que foram classificados erroneamente como ideação suicida.

$$Precisao = \frac{10}{10 + 40} = 0,2 \tag{2.16}$$

A Equação 2.16 tem como valor final 0,2 de precisão, significando que o algoritmo de classificação conseguiu rotular com sucesso apenas 20% dos dados que foram identificados por ele como pertencentes à classe ideação suicida.

Um modelo de classificação com alta precisão é útil em casos onde a relevância do dado é importante. Por exemplo, nas buscas por conteúdo na internet, é um parâmetro de qualidade ter os temas mais compatíveis com a busca na primeira página. Sendo assim, a porcentagem de dados corretamente identificados no grupo de dados com a mesma característica é importante.

2.4.3 Recall

Essa ferramenta mostra a proporção dos dados que deveriam ser identificados e foram identificados corretamente pelo algoritmo. A matriz de confusão na Figura 2.10 e a Equação 2.17 abaixo mostram como deve ser o cálculo da recall:

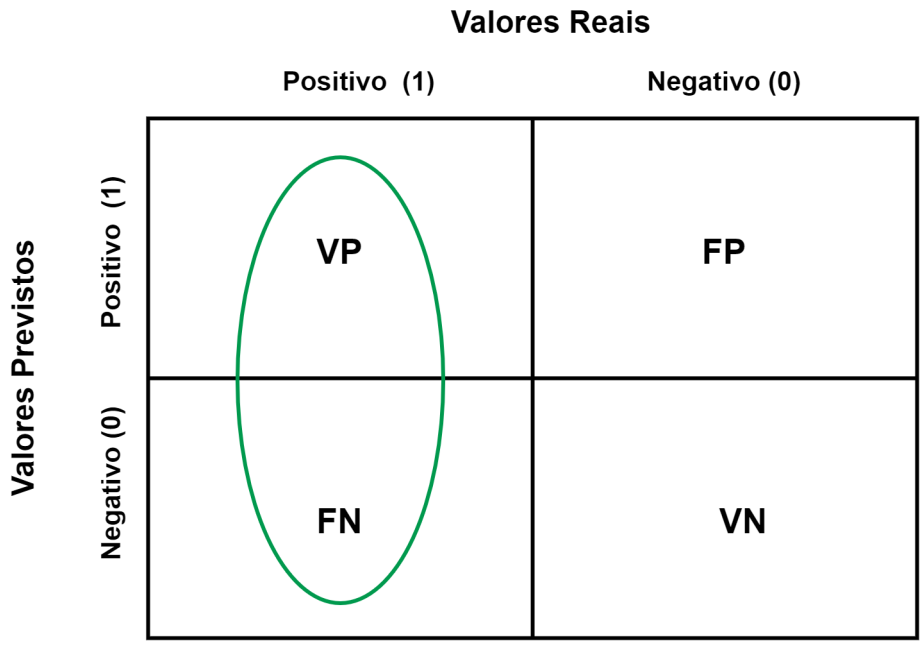

Figura 2.10: Matriz de Confusão - recall

$$Recall = \frac{VP}{VP + FN} \tag{2.17}$$

Ainda trabalhando com o exemplo anterior, 30 textos de ideação suicida (1) e 70 textos de não ideação suicida (0). Abordando o cálculo do recall:

- **VP = 10**, textos da classe de ideação suicida e que foram classificados corretamente como ideação suicida.

- **FN = 20**, textos da classe de ideação suicida e que foram classificados erroneamente como não ideação suicida.

$$Recall = \frac{10}{10 + 20} = 0,33 \tag{2.18}$$

A Equação 2.18 mostra um recall de 0,33, apontando que quando o dado encontrado pertencia à classe ideação suicida, em 33% das vezes ele foi rotulado corretamente pelo algoritmo.

Quando se trata de escolha do classificador, é importante avaliar o domínio e os impactos da ineficiência em classificar determinados dados [Eisenstein, 2019]. Por exemplo, um recall alto é relevante quando o falso positivo é mais barato do que o falso negativo: no caso de doenças, um falso positivo pode ter o custo de um teste adicional ou de um tratamento para uma patologia inexistente. Enquanto que o falso negativo pode levar a uma doença existente evoluir sem tratamento.

2.4.4 F-Score

Essa métrica combina, através da média harmônica, precisão e recall de modo a trazer um número único que indique a qualidade geral do modelo utilizado. É eficiente para avaliar o algoritmo mesmo quando o conjunto de dados é desbalanceado. Ressaltando que classificadores com muitos dados no quadrante falso positivo apresentam uma precisão baixa e classificadores com muitos falsos negativos apresentam um baixo recall, quando se combinam as duas métricas, o resultado consegue identificar o desempenho do algoritmo de forma mais realista.

$$F\_Score = 2 \times \frac{Precisao \times Recall}{Precisao + Recall} \tag{2.19}$$

$$F\_Score = 2 \times \frac{0,2 \times 0,33}{0,2 + 0,33} = 0,249 \tag{2.20}$$

A Equação 2.20 apresenta valor de F-Score = 0,249. Portanto, a avaliação geral da performance do modelo de classificação foi de aproximadamente 25% .

2.4.5 ROC-AUC

A visualização da performance do algoritmo através de gráfico é feita pela curva ROC (*Receiver operating characteristics*) em que o eixo x mostra a taxa de falsos positivos e o eixo y mostra a taxa de verdadeiros positivos. Uma performance perfeita do classificador pode ser identificada quando o caminho da curva segue os seguintes pontos:

(0,0), (0,1) e (1,1), linha verde na Figura 2.11. Enquanto que em um classificador com baixa performance a curva é uma linha diagonal do ponto (0,0) ao (1,1), classificação aleatória (linha pontilhada). A performance de classificadores ocilam entre esses dois extremos, considerando na Figura 2.11 a seta que indica um melhoramento na performance e a outra (-) que indica performance que deve ser desconsiderada. Ainda, o eixo y indica o *Recall* que mostra a sensibilidade do classificador. A curva ROC pode ser resumida, através de integral, em um único número indicador de performance — *Area under the Curve* (AUC). O resultado da integral da curva de um classificador perfeito seria $AUC = 1$ e o outro extremo seria $AUC = 0,5$ [Eisenstein, 2019].

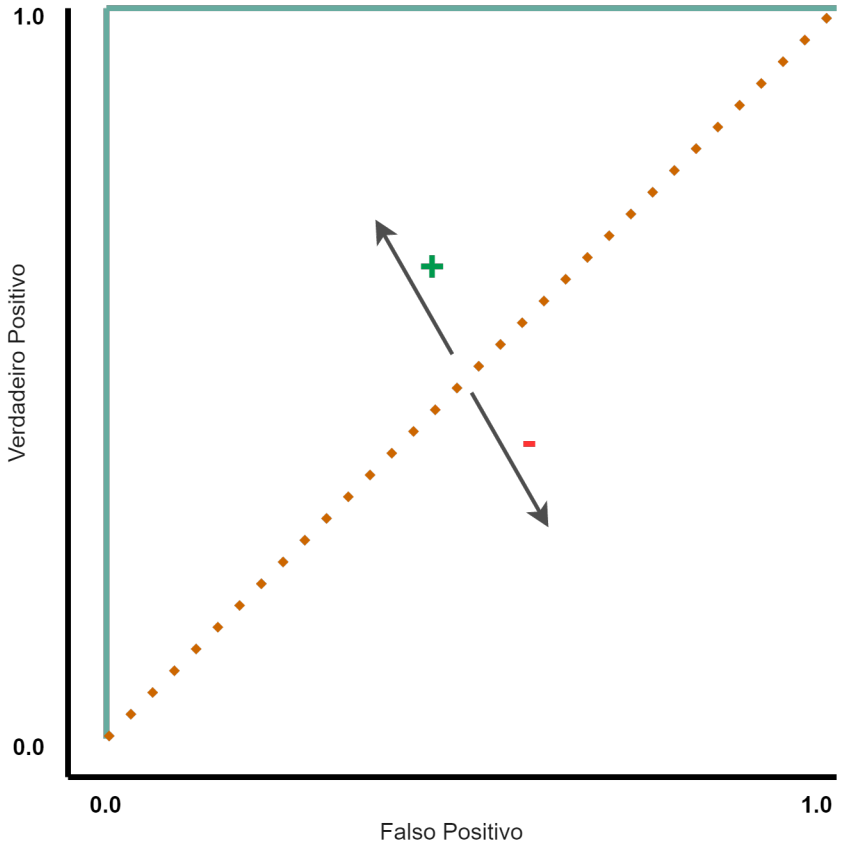

Figura 2.11: Roc-Auc

Capítulo 3

PLN em domínio de textos de ideação suicida

O suicídio como um problema de saúde pública mundial vem atraindo o interesse de diversos pesquisadores como Desmet & Hoste [2018] e Tadesse et al. [2020], que consideram o PLN uma ferramenta importante e estratégica para identificar a ideação suicida a tempo de prevenir o seu desfecho final.

As redes sociais, nos dias de hoje, são relevantes por muitas vezes ocuparem o lugar dos diários pessoais de tempos atrás: são nelas em que as pessoas expressam o seu pensamento e sentimento a respeito de uma variedade de fatos. Sendo assim, algumas fontes de dados usadas por diversos trabalhos são: Netlog[1] [Tadesse et al., 2020], Reddit[2] [Malini & Tan, 2016], outros usam o Twitter[3]. Há ainda trabalhos realizados com dados que não foram extraídos de redes sociais, por exemplo: Sohn et al. [2012] que utilizou notas suicidas fornecidas por uma competição[4] de PLN; Berni et al. [2018] que utilizou os escritos (fragmentos de cartas e diários) da romancista Virginia Wolff, vítima de suicídio em 1941; assim como Joh & hun Lee [2019] que utilizou obras literárias da mesma autora.

Segundo Chiroma et al. [2018], estudos mostram uma correlação entre pessoas vulneráveis nas redes sociais e suicídio. Expressar sentimentos, algo abstrato, de forma exata em texto se torna uma tarefa desafiadora para aqueles que não gozam de saúde mental. Portanto, o uso de palavras adjacentes na tentativa de estabelecer um referencial e aumentar a significância do termo isolado pode ajudar o desempenho do classificador, como mostram os autores Sohn et al. [2012], Burnap et al. [2017] e Tadesse et al. [2020], que fizeram uso da ferramenta *bag of n-grams* ao invés do BOW.

[1] http://nl.netlog.com/
[2] https://www.reddit.com
[3] https://twitter.com
[4] 2011 I2B2/VA/Cincinnati

As emoções impressas nos textos suicidas contribuem para a análise do estado psicológico do indivíduo e com base nessa premissa o trabalho realizado por Sohn et al. [2012] estudou a presença de 15 tipos de sentimentos em cartas suicidas. O resultado estatístico identificou um grupo de 8 sentimentos em 90% das entradas analisadas enquanto as outras emoções raramente estavam presentes. O NB obteve uma precisão de 61% na identificação das seguintes características nos textos: acusação, culpa, desesperança, informação, instruções e gratidão. Corroborando com os resultados desse trabalho, O'Dea et al. [2015] apontam caminhos para avanços nesse campo de pesquisa, como, por exemplo: a expansão da gama de termos relacionados ao suicídio, para que mais expressões sejam incluídas; inclusão da técnica de normalização de palavras (retirando prefixos e sufixos) assim como análise de palavras adjacentes — chamando a atenção para palavras raras ou palavrões que, segundo os autores, podem apontar para um aumento no risco de suicídio.

O uso de seleção de características como forma de melhorar a performance dos classificadores foi a opção de todos os trabalhos investigados com exceção do Chiroma et al. [2018]. Contudo, o autor aponta para um futuro uso da ferramenta para aprimorar o preditor utilizado. A ferramenta estatística TF (*Term frequency*) está presente no estudo de Berni et al. [2018] enquanto os outros preferiram explorar o TF-IDF.

Na tentativa de obter resultados mais contundentes nesse domínio abstrato (mente humana), os atributos linguísticos vêm sendo o foco para o desenvolvimento de ferramentas para seleção de características que possam auxiliar o classificador, tal como o trabalho de Desmet & Hoste [2018] que obteve um bom resultado ao comparar o uso do SVM puro com o SVM aliado ao algoritmo genético como seletor de características preditoras.

Apesar de existirem inúmeros trabalhos já publicados na literatura usando textos de microblogs (publicação de mensagens pequenas) para comparar classificadores, Chiroma et al. [2018] observa que não há nenhuma conclusão definitiva sobre qual é o melhor classificador para se trabalhar com entradas pequenas e informais de contexto suicida.

A evolução dos algoritmos de aprendizado de máquina proporcionou que classificadores mais complexos fossem estudados. Publicações recentes como o Tadesse et al. [2020] mostram resultados promissores com o uso de *deep learning* não apenas na detecção da ideação suicida, mas também na observação dos 3 estágios pscicológicos (ansiedade, reclusão e planejamento) que antecedem o sucídio. Contudo, esse tipo de ferramenta exige um volume grande de entradas (dados), tornando a sua utilização inviável no atendimento clínico individualizado ou de pequenos grupos. Não obstante o bom resultado obtido pelo estudo (93,2% de precisão), os autores apontam para li-

mitações — recorrentes nesse tipo de pesquisa — relacionadas à anotação humana dos dados. A rotulação é um trabalho complexo e as técnicas de classificação utilizadas nessa área de estudo são, na sua maioria, aprendizado supervisionado que depende da rotulagem correta de dados [Tadesse et al., 2020].

O suicídio é um tema sensível para as famílias das vítimas e carrega consigo estigmas sociais ainda difíceis de serem trabalhados pela incompreensão do ato de atentar contra a própria vida. Nesse cenário, por questões éticas, são incomuns estudos que utilizam base de dados de um único indivíduo e ainda mais raro de indivíduos que obtiveram sucesso na sua estratégia de morte. Posto isto, os trabalho realizados por Joh & hun Lee [2019] e Berni et al. [2018] são um diferencial na literatura por utilizarem como base de dados textos de uma única pessoa e essa ter sido bem sucedida na sua estratégia suicida. Enquanto o primeiro tem como objetivo mostrar como atributos linguísticos podem ajudar a detectar notas suicidas em textos ordinários, o segundo se concentra em identificar ideação suicida em escritos pessoais. Apesar do bom resultado encontrado por Berni et al. [2018] com a implementação do classificador NB, sensibilidade de $69,23\%$ na detecção da classe positiva, alguns procedimentos que poderiam potencializar o algoritmo não foram testados: TF-IDF e outras ferramentas de seleção de características preditoras. Assim como a representação do vetor de termos (*bag of grams*), além do BOW presente no trabalho. Concluindo assim que o diferencial do grupo de dados usado é importante nessa área de estudo e que existem lacunas para que seja explorado por outras ferramentas mais robustas de seleção de características e técnicas de classificação não tão ingênuas como o NB.

A Tabela 3.1 apresenta um resumo dos principais estudos abordados nesse capítulo de acordo com três principais características, a saber: representação de vetor de termos, algoritmos de classificação e uso ou não de seleção de caracteristicas.

Estudo	**Representação de vetor do termos**	**Algoritmos de Classificação**	**Seleção de Características**
Sohn et al. [2012]	Bag of grams (1 a 3 grams)	NB, Ripper (expressões regulares) e NB-Ripper	GI (ganho de informação)
O'Dea et al. [2015]	BOW	SVM	TF,TF-IDF e filtro
Burnap et al. [2017]	Bag of grams (1 a 5 grams)	NB, SVM e AD	TF-IDF
Chiroma et al. [2018]	BOW	NB, SVM, AD e FR	Não
Desmet & Hoste [2018]	BOW	SVM	wrapper (algoritmo genético)
Berni et al. [2018]	BOW	NB	TF
Joh & hun Lee [2019]	BOW	SVM	não
Tadesse et al. [2020]	BOW e Bag of grams (1 a 2 grams)	Deep Learning (LSTM-CNN), NB, SVM e FR	TF-IDF

Tabela 3.1: Trabalhos Relacionados

Capítulo 4

Metodologia

O trabalho foi desenvolvido seguindo o fluxo da Figura 4.1 que ilustra a metodologia composta por 2 grandes módulos: o primeiro (3) é aplicado aos dados de treinamento e o segundo (4) aos dados de teste, e finalizando com o módulo de análise de sentimento (5). No início do trabalho, o conjunto de dados (1) foi separado aleatoriamente em dois subconjuntos distintos (2): 80% subconjunto de treinamento e os outros 20% como o subconjunto de teste. Os procedimentos de pré-processamento (a, d) são comuns aos dois subconjuntos, são eles:

- normalização das entradas, onde todas as letras dos textos serão colocadas em minúsculas e serão retirados das frases os caracteres especiais, pontuação e espaços;
- tokenização: as entradas foram segmentadas e as palavras transferidas para um vetor de termos na forma de BOW ou *bag of n-grams*.

O caminho distinto que segue o subconjunto de treinamento começou com a seleção de características (b) para que, após a análise, apenas os tokens que carregassem significância para o contexto fossem considerados e, assim, a dimensionalidade do vetor de termos fosse reduzida. As ferramentas de seleção de características utilizadas foram: TD-IDF e *chi-square*. A etapa de treinamento (c) dos modelos foi precedida pela divisão dos dados em 5 *folds* (blocos), para que fosse possível fazer a validação cruzada do procedimento utilizando 4 blocos para treinamento e 1 para validação. O procedimento de validação cruzada é realizado em grandes conjuntos de dados, todavia, pode ser adaptado para grupos menores — como foi no presente caso. Sendo assim, o procedimento foi repetido até que todos os *k-folds* tivessem participado tanto do processo de treinamento quanto do de validação do modelo, as k-iterações então tiveram suas estimativas de performance resumidas, usualmente, pelo cálculo da média e do erro padrão [Santos, 2018]. Os hiperparâmetros, variáveis que controlam os algoritmos

de classificação, foram ajustados de acordo com as suas performances preditivas nos dados de validação em relação à métrica ROC-AUC. Na otimização dos hiperparâmetros foi utlizado o BayesSearchCV (*Sci-kit learn*) que se mostrou mais eficiente do que outros algoritmos de otimização em desafios de otimização de funções de *bechmarking* segundo [Snoek et al., 2012]. A tabela do Apêndice B mostra os valores utilizados na calibração dos hiperparâmetros.

Após todos os procedimentos do bloco de treinamento, a rotina do bloco de teste se iniciou. Os dados (20%) separados para o teste passaram pelo pré-processamento (d) e seguiram para alimentar os modelos preditivos (e). Na etapa de teste é avaliado o erro de generalização do modelo selecionado [Santos, 2018]. As métricas de avaliação (f) utilizadas para comparar os algoritmos do trabalho foram: precisão definida na Eq. 2.15, por ser uma métrica de avaliação comum nos trabalhos da seção 3, o F-score, ROC-AUC e o recall definido na Eq.2.17 devido ao custo alto do falso negativo para o contexto estudado: ideação suicida não detectada pode determinar a não assistência ao indivíduo que consequentemente pode vir a culminar num desfecho trágico. Sendo assim, o recall alto é relevante devido ao falso positivo ter um custo financeiro, emocional e social menor no domínio.

Para comparar os resultados em termos de AUC utilizamos o teste U de Mann-Whitney e consideramos o nível de significância de 95%. Esse teste é não-paramétrico usado para variáveis quantitativas ou qualitativas e pode ser usado em amostras pequenas. O teste U é utilizado para verificar se existe diferença estatística entre dois grupos, aponta se os grupos independentes pertencem ou não a mesma população. Esse teste ultiliza a mediana, testando a sua igualdade e indicando o grau de entrelaçamento dos dados após a sua ordenação. Um resultado de $p <= 0.05$ mostra que há uma diferença estatística significante entre os dois grupos, confiança $>= 95\%$, apontando para amostras distintas e assim descartando a hipótese de igualdade das medianas (hipótese nula) [MacFarland & Yates, 2016].

Com o objetivo de verificar o nível de discriminação das classes, foi realizada uma análise de sentimento de forma automática utilizando a biblioteca *TextBlob* do python que utiliza um dicionário léxico para avaliar e separar as instâncias em três categorias: positivo, neutro e negativo. Com isso, esperou-se investigar o grau de dificuldade da tarefa desempenhada pelos classificadores, assim como buscar uma correlação entre o rótulo do texto analisado com o sentimento apontado pelo resultado das polaridades das palavras contidas nele.

O código do trabalho foi desenvolvido na linguagem Python usando, dentre outras bibliotecas, a *scikit-learn* e a *Natural Language Toolkit*, que oferecem ferramentas para o processamento da linguagem humana.

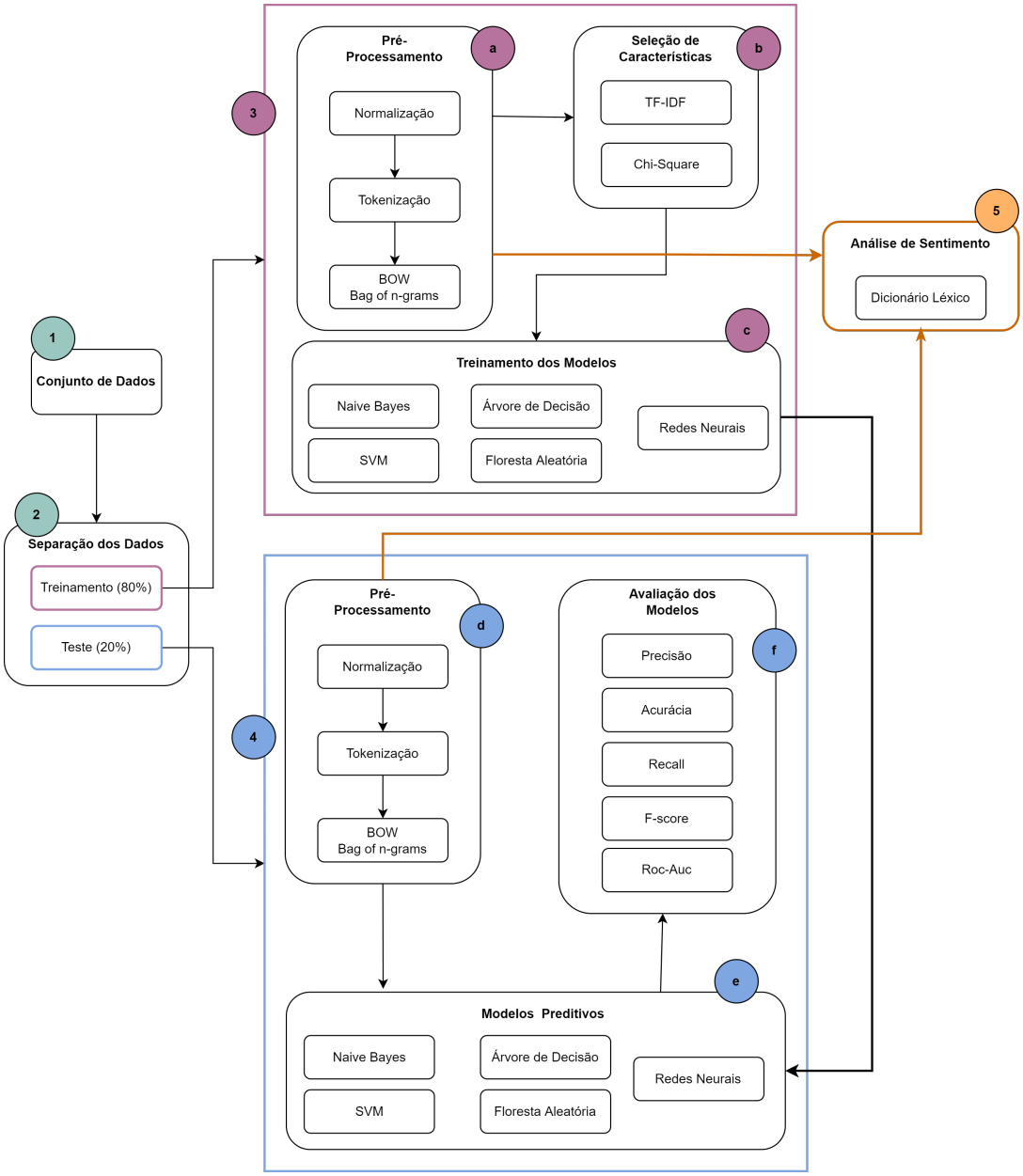

Figura 4.1: Fluxo da Metodologia.

4.1 Conjuntos de dados

4.1.1 Escritos pessoais de Virgínia Woolf

Virgínia Woolf foi uma escritora inglesa modernista que faleceu em 1941, aos 57 anos de idade, por suicídio, após ter fracassado em pelo menos outras 3 tentativas. Segundo Berni et al. [2018], a escolha dos textos da Virgínia se deu por 2 motivos: além das cartas de despedida endereçadas ao marido e à irmã, a autora deixou um vasto repertório de diários onde se sentia livre para escrever seus sentimentos e angústias.

> " [...] *I am doing what seems the best thing to do. You have given me the greatest possible happiness. You have been in every way all that anyone could be. I don't think two people could have been happier 'til this terrible disease came. I can't fight any longer*[...] "

O trabalho de organização dos textos e rotulagem foi feito por Berni et al. [2018], que estabeleceu um prazo de 60 dias para definir se o texto tinha ou não relação com o evento: classe positiva para textos de 60 dias antes da morte; e classe negativa para textos fora da janela de 60 dias. A autora justifica esse período devido ao comportamento da Virgínia antes de tirar a própria vida e a necessidade de se estabelecer um prazo alvo.

O conjunto de dados está caracterizado da seguinte forma:

- 100 textos únicos.
- Rotulagem:
  - Antes de 60 dias (0): 69 textos (69%).
  - No período de 60 dias (1): 31 textos (31%).

4.1.2 Diário digital de Victoria McLeod

A Victoria McLeod foi uma jovem neozelandesa que por suicidou-se no ano de 2014 aos 17 anos. O seu diário digital, escritos de um período de 4 meses que antecederam à sua morte, foi cedido pela família para que especialistas pudessem estudá-lo como forma de ajudar a entender a mente suicida.

> " *Sat in the shower. Did the whole crying bit ... Sat in bed. Did the whole sad songs and crying bit. . . . PLEASE MAKE THIS SAD STOP. FUCKING MAKE IT STOP. God, something out there, please make it stop.*"

Os textos foram retirados do livro do psicólogo Jesse Bering, *Suicidal: Why We Kill Ourselves* [Bering, 2020]. As entradas foram rotuladas pelo autor em 6 estágios progressivos: sendo o primeiro *"falling short of expectations"* (fracasso, expectativas não cumpridas) e o último *"disinhibition"* (desinibição), no qual aconteceu o evento suicida. O objetivo do livro, segundo o autor, seria ajudar às pessoas a melhor entender o problema e talvez, reconhecer alguns sinais e enfrentá-los. O projeto ainda o ajudou a lidar com as próprias tendências suicidas e que espera que as pessoas que contemplam o

suicídio possam ter outra perspectiva e que as famílias, vítimas do ato, possam melhor compreender o estado mental da pessoa que partiu [Bering, 2020].

Este conjunto de dados está caracterizado da seguinte forma:

- 62 entradas únicas.
- Rotulagem:
  - Rótulo 0: nenhum estágio identificado, 11 entradas (18%).
  - Estágio 1: Expectativas que não se cumprem, 8 entradas (13%).
  - Estágio 2: Atribuições internas, culpar a si mesmo, 7 entradas (11%).
  - Estágio 3: Percepção exacerbada de si mesmo, 7 entradas (11%).
  - Estágio 4: Negatividade, 7 entradas (11%).
  - Estágio 5: Destruição Cognitiva, 11 entradas (18%).
  - Estágio 6: Desinibição, 11 entradas (18%).
- Agrupamento definido para transformar o contexto em binário: não ideação suicida $[0, 1, 2, 3]$ e ideação suicida $[4, 5, 6]$.

Capítulo 5

Resultados e Discussão

Os resultados serão apresentados nesta sessão separadamente: primeiro os resultados da Virginia Woolf e em seguida os resultados da Victoria McLeod. A análise foi feita usando o parâmetro ROC-AUC e também avaliando a performance segundo o *Recall* que, para o domínio estudando, é igualmente relevante. Como citado na metodologia, os métodos foram executados 10 vezes para cada procedimento de seleção de característica e também para o método puro, sem seleção de característica.

5.1 Resultados - Virginia Woolf

5.1.1 Naive Bayes

O método Naive Bayes teve uma performance melhor utilizando o χ^2 para seleção das características, o teste estatístico também apontou como significante (*p-value* 0.019) essa performance em relação ao método puro. A Tabela de resultados 5.1 mostra que o modelo acertou, em média, apenas 45% das ocorrências positivas em todas as tentativas de identificação. No entando, o classificador foi capaz de classificar corretamente 65% das entradas e apresentou um *Recall* de 0.750, mostrando uma sensibilidade acima da média (50%) para detectar corretamente as classes positivas. Portanto, nesse experimento com os dados da Virgínia Woolf, concluímos que o método Naive Bayes apresenta uma performance melhor quando combinado com χ^2 para seleção de características.

5.1.2 SVM

O resultado de *p-value* (0.013) no modelo aponta para um resultado significante quando este é combinado com a ferramenta χ^2 para seleção de características. Entretanto, os dados na Tabela 5.2, mostram que o método SVM teve dificuldades em identificar as classes: a precisão do modelo puro acusa que nenhuma classificação positiva correta foi

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.867	0.517	0.453
Recall	0.217	0.233	0.750
F-Score	0.337	0.314	0.560
Acurácia	0.750	0.705	0.650
ROC-AUC	0.598	0.570	**0.679***

Tabela 5.1: Resultados do Naive Bayes.
* Relevância estatística ao teste *p-value*

feita e sendo assim mostra uma sensibilidade 0.000 em detectar corretamente a classe onde há presença de ideação suicida. Quando se observa a acurácia, essa mostra que ainda assim, o modelo obteve 70% de sucesso na sua tarefa. Quando se leva em conta a distribuição dos dados (30% classe 1 e 70% classe 0), o resultado da acurácia nos três tipos de testes, assim como o baixo resultados das outras métricas de avaliação, podemos concluir que o SVM não foi capaz de realizar o trabalho de identificação das classes nesse conjunto de dados. Chamando a atenção para o método puro onde o classificador atribuiu apenas um rótulo (0) para todo o conjunto de teste.

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.000	0.100	0.250
Recall	0.000	0.017	0.217
F-Score	0.000	0.029	0.231
Acurácia	0.700	0.700	0.700
ROC-AUC	0.500	0.505	**0.562 ***

Tabela 5.2: Resultados do SVM
* Relevância estatística ao teste *p-value*

5.1.3 Árvore de Decisão

O teste estatístico não mostrou resultado significante entre o método puro e o χ^2, melhor resultado de ROC-AUC do experimento. A análise dos dados da tabela 5.3, apesar de mostrar a métrica ROC-AUC acima da média (64%), pode-se afirmar que o modelo foi mediano na tarefa: apenas em 49% das tentativas de identificar ocorrências positivas (Precisão) ele foi bem sucedido e combinando esse resultado com a acurácia de 69%, percebe-se que o método foi melhor em identificar a classe negativa do que a positiva. Portanto, é possível afirmar que esse modelo apresentou uma porcentagem considerável de falsos negativos, o que é bastante preocupante para esse domínio já que o impacto e o custo do falso negativo apresenta consequências ruins, como já citado no texto.

Métricas	Sem Seleção	TF-IDF	χ^2
Precisão	0.393	0.520	0.490
Recall	0.383	0.433	0.517
F-Score	0.384	0.452	0.485
Acurácia	0.635	0.690	0.690
ROC-AUC	0.563	0.617	0.640

Tabela 5.3: Resultados da Árvore de Decisão

5.1.4 Floresta Aleatória

O método Floresta Aleatória, assim como a AD, também não apresentou resultados significativos no teste estatístico. Os resultados para o conjunto de dados da Virgínia Woolf, Tabela 5.4, podem ser analisados como os resultados do SVM, os classificadores se comportaram de maneira análoga na tarefa.

Métricas	Sem Seleção	TF-IDF	χ^2
Precisão	0.000	0.200	0.200
Recall	0.000	0.033	0.033
F-Score	0.000	0.057	0.057
Acurácia	0.700	0.710	0.710
ROC-AUC	0.500	0.517	0.517

Tabela 5.4: Resultados da Floresta Aleatória

5.1.5 Rede Neural

A Rede Neural, diferentemente dos outros modelos, mostra uma melhor performance no método puro. Corroborando com os dados da tabela 5.5, o resultado do uso de ferramenta de seleção de características não se mostrou significante no teste estatístico (*p-value* 0.677). A precisão mostra que o χ^2 teve mais sucesso na identificação das entradas corretas da classe positiva do que o método puro, porém este apresentou uma sensibilidade maior ao identificar a classe positiva — o que poderia ser utilizado como critério decisório, nesse domínio, para modelos com performances similares.

Métricas	Sem Seleção	TF-IDF	χ^2
Precisão	0.590	0.439	0.629
Recall	0.583	0.483	0.517
F-Score	0.509	0.414	0.497
Acurácia	0.715	0.685	0.710
ROC-AUC	0.677	0.627	0.655

Tabela 5.5: Resultados da Rede Neural

Diante do exposto acima, concluímos que o modelo Naive Bayes combinado com a ferramenta χ^2 para seleção de características foi o mais adequado para o trabalho de identificação de ideação suicida nesse conjunto de dados pois apresentou melhor resultado tanto para o ROC-AUC (0.679) quanto para o *Recall* (0.750). Apenas o NB e o SVM apresentaram resultados de **p-value** satisfatórios ao uso de ferramentas de seleção de características quando submetidos ao teste estatístico.

5.2 Resultados - Victoria McLeod

5.2.1 Naive Bayes

O modelo teve uma performance melhor na identificação da classe positiva quando utilizando uma ferramenta de seleção de características, sendo o seu melhor resultado com o TF-IDF (0.636), seguido pelo χ^2 (0.617). Na Tabela 5.6, também é possível observar que a ferramenta TF-IDF apresentou um resultado ROC-AUC (0.638) melhor do que o χ^2 (0.625) e esse último mostra uma pequena diferença positiva em relação ao *Recall* do anterior. É possível concluir que as duas ferramentas de seleção de características obtiveram resultados muito próximos nas métricas de avaliação e um pouco acima do método puro. Vale ressaltar que o teste estatístico não mostrou resultado significante (0.27) na utilização do TF-IDF ou χ^2 para a performance do modelo.

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.513	0.636	0.617
Recall	0.517	0.533	0.550
F-Score	0.508	0.576	0.548
Acurácia	0.577	0.646	0.631
ROC-AUC	0.573	0.638	0.625

Tabela 5.6: Resultados do Naive Bayes

5.2.2 SVM

Corroborando com o teste estatístico que não apontou diferença significativa no uso de ferramentas de seleção de características, o melhor resultado para esse modelo, como mostra a Tabela 5.7, foi o método puro. No entanto, é importante observar que apesar do modelo ter apresentado um resultado ROC-AUC acima da média (0.608) o *Recall* teve um resultado baixo, apenas 0.317, mostrando pouca sensibilidade em detectar a classe positiva, apesar de ter conseguido sucesso em 69% das vezes que identificou as ocorrências como sendo dessa classe. O resultado mais alto do *F-score* (0.433) sendo

obtido com a ferramenta Chi2 aponta para uma baixa performance do modelo SVM no conjunto de dados da Victoria McLeod.

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.692	0.667	0.567
Recall	0.317	0.267	0.383
F-Score	0.414	0.363	0.433
Acurácia	0.631	0.577	0.585
ROC-AUC	0.608	0.555	0.570

Tabela 5.7: Resultados do SVM

5.2.3 Árvore de Decisão

Os resultado da Tabela 5.8 mostram, em uma análise geral, que esse modelo não obteve uma boa performance nesse conjunto de dados. O resultado ROC-AUC ficou abaixo de 0.60 para todos os modelos e *Recall* máximo foi de 0.400, mostrando uma baixa sensibilidade do método em identificar a classe positiva. Ao ser submetido ao teste estatístico, o resultado não foi significante para implementação das ferramentas de seleção de características.

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.503	0.325	0.650
Recall	0.400	0.367	0.300
F-Score	0.410	0.300	0.377
Acurácia	0.531	0.462	0.577
ROC-AUC	0.521	0.455	0.557

Tabela 5.8: Resultados do Árvore de Decisão

5.2.4 Floresta Aleatória

O teste estatístico mostrou que há uma diferença significativa (0.003) ao combinar o modelo puro com ferramentas de seleção de características. No entanto, os dados da Tabela 5.9 apontam para uma dificuldade do classificador em analisar os dados corretamente. Analisando o χ^2, melhor resultado obtido, observa-se que: dentre todos os apontamentos como classe positiva, o modelo acertou em 64% das vezes que definiu o rótulo, o baixo *Recall* mostra que muitos dados que eram positivos foram considerados negativos pelo classificador (falso negativo) — sendo isso um resultado preocupante para o domínio — e a média harmônica que mostra o desempenho do classificador em relação a classe positiva ficou muito abaixo da média (0.369), observando aqui uma

deficiência do modelo em relação a classe positiva. Entretanto, a Acurácia mostra um resultado acima da média (0.623), o que permite concluir que o modelo foi mais assertivo na classificação dos dados relacionados a classe negativa.

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.117	0.472	0.642
Recall	0.050	0.333	0.267
F-Score	0.069	0.387	0.369
Acurácia	0.515	0.515	0.623
ROC-AUC	0.482	0.502	**0.598 ***

Tabela 5.9: Resultados da Floresta Aleatória
* Relevância estatística ao teste *p-value*

5.2.5 Rede Neural

A Rede Neural também foi um modelo que não apresentou diferença estatística significante ao uso de ferramentas de seleção de características, *p-value* 0.47. No entanto, a Tabela 5.10 mostra uma melhor performance do modelo quando combinado com a ferramenta TF-IDF: uma sensibilidade em detectar a classe positiva acima da média (0.633) e a média harmônica entre os marcadores de desempenho em relação a classe positiva também acima da média (0.581).Nota-se também que o classificador conseguiu identificar corretamente a classe de 62% das ocorrências.

Métricas	**Sem Seleção**	**TF-IDF**	χ^2
Precisão	0.515	0.591	0.534
Recall	0.517	0.633	0.567
F-Score	0.485	0.581	0.542
Acurácia	0.562	0.623	0.554
ROC-AUC	0.558	0.624	0.555

Tabela 5.10: Resultados da Rede Neural

Enquanto que o teste estatístico nos dados da Virginia Woolf apresentou resultados relevantes para o NB e o SVM quando combinados com ferramentas de seleção de características, nos dados da Victoria McLeod, diante do exposto acima, observa-se que apenas o modelo Floresta Aleatória apresentou resultado satisfatório no teste estatístico para o uso de ferramenta de seleção de características, *p-value* abaixo de 5%. A RN foi melhor no que tange o domínio estudado por apresentar resultados melhores para os marcadores de desempenho na classificação positiva: *Recall* e F-score.

5.3 Comparação dos Classificadores

Essa seção irá mostrar a comparação dos classificadores segundo o teste U de Mann-Whitney que foi efetuado sobre os resultados do ROC-AUC com seleção de características e consideramos o nível de significância de 95%. Lembrando que um resultado $<= 0.05$ mostra que há uma diferença estatística significante entre os modelos comparados, a confiança de $>= 0.95$, aponta para amostras distintas e assim descartando a hipótese de igualdade das medianas.

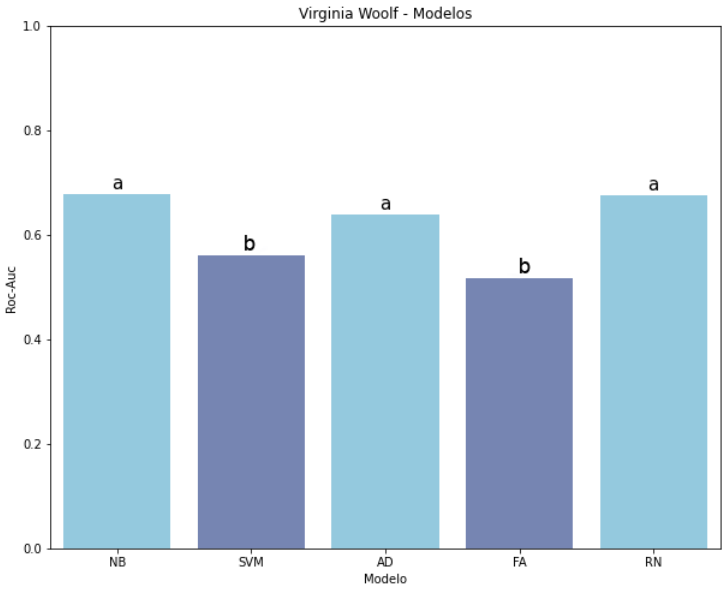

Figura 5.1: Comparação dos Classificadores - Virginia Woolf

A Figura 5.1 mostra a comparação dos classificadores em relação aos dados da Virginia Woolf. Os classificadores foram agrupados de acordo os seus resultados, o grupo a, assim como o b, não apresentam diferenças estatísticas entre eles, por exemplo: NB e AD com resultado de *p-value* de 0.225 e SVM e FA com resultado de *p-value* de 0.11. Enquanto classificadores em grupos diferentes obtiveram resultados significativos no teste estatístico, por exemplo: NB e FA com resultado de *p-value* de 0.001 e AD e FA com resultado de *p-value* de 0.008. Podemos concluir que o grupo a teve uma performance significativamente melhor do que os modelos do grupo b.

A Figura 5.2 mostra a comparação dos classificadores em relação aos dados da Victoria McLeod. Os classificadores não apresentaram diferença estatística significante entre si, sendo assim, estão todos no mesmo grupo a. Citando como exemplo: NB e RN com resultado de *p-value* de 0.569 e AD e RN com resultado de *p-value* de 0.17 no

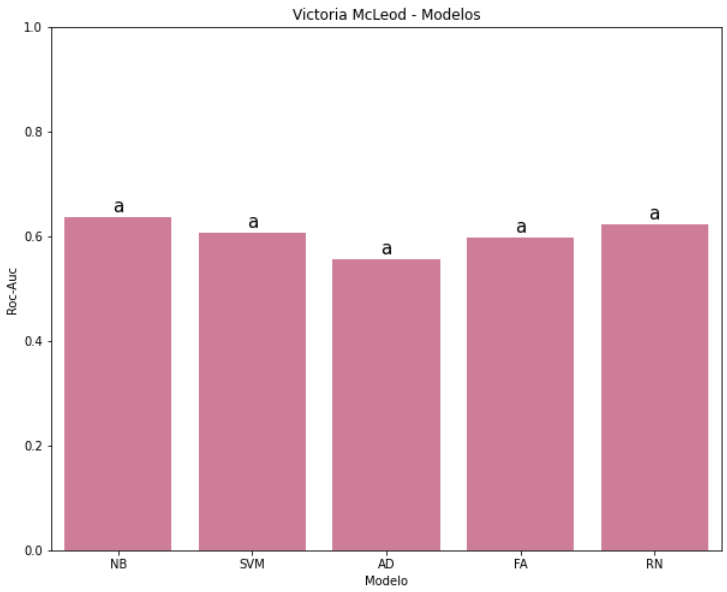

Figura 5.2: Comparação dos Classificadores - Victoria McLeod

teste U de Mann-Whitney. Podemos concluir que não houve diferença significativa na performance dos modelos.

5.4 Análise de Sentimento

Considerando os resultados dos testes dos modelos, se viu necessário investigar a separabilidade das classes e a correlação entre rótulo e sentimento expressado no texto. A análise léxica das palavras é feita de forma dissociada do domínio estudado, a polaridade é definida com base em qual sentimento a palavra invoca e a intensidade deste, por exemplo: feliz apresenta uma conotação positiva, enquanto triste apresenta uma conotação negativa. A vantagem desse procedimento é que não requer uma análise semântica profunda ou desambiguação da palavra para atribuir uma pontuação coerente com o sentimento expressado pelo autor [Guerini et al., 2013].

5.4.1 Virginia Woolf

A análise de sentimentos nos dados da Virgínia Woolf, Figura 5.3, mostra os dados positivos (rosa), os dados neutros (azul) e os dados negativos (verde) separados nas classes 0 e 1, que aponta ideação suicida. É possível observar que a distribuição dos dados é bastante homogênea nas duas classes. É observado também, que a classe posi-

tiva para ideação suicida apresenta mais textos positivos do que negativos, lembrando que o rótulo das entradas foi definido de forma temporal.

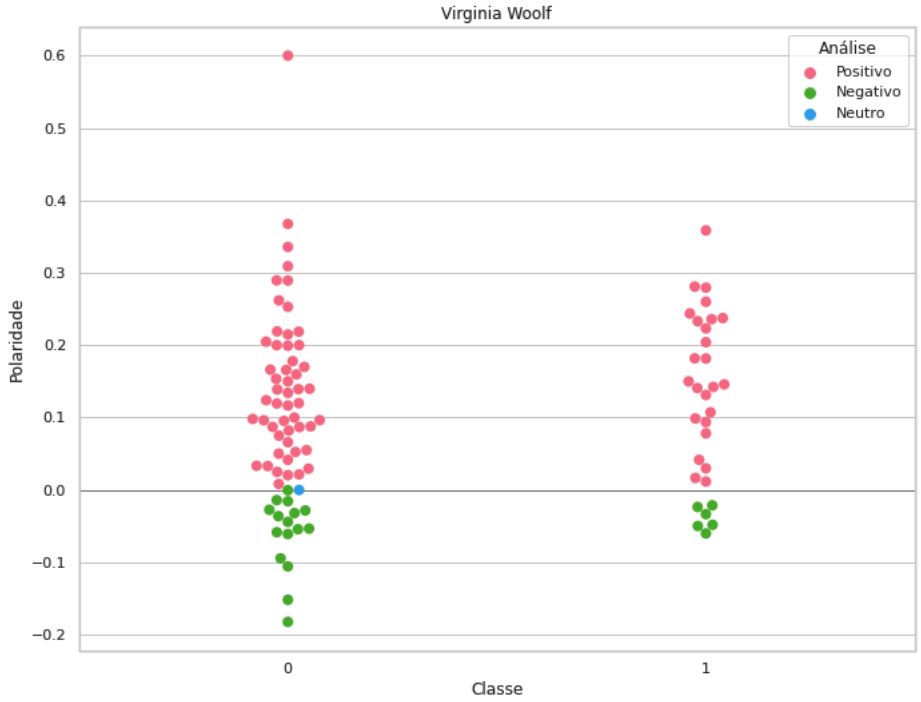

Figura 5.3: Análise de Sentimento - Virginia Woolf

Corroborando com o observado na Figura 5.3, a linha de tempo do conjunto de dados na Figura 5.4 mostra os dados concentrados entre o intervalo -0.1 a 0.3 com alguns picos fora dele. Importante observar que o sentimento mais negativo e mais positivo acontecem na classe 0, sem ideação suicida. A Tabela com a análise de sentimento desse conjunto de dados: polaridade e seu respectivo sentimento, se encontra no Apêndice A.1.

A núvem de palavras mostra de forma concisa e resumida o conteúdo das entradas. A Figura 5.5 mostra uma representação visual das palavras mais usadas pela escritora nos textos analisados. O tamanho das letras é relativo à frequência ou à importância da palavra no conjunto de dados: *letter, book* e *write* podem ser relacionadas à profissão da autora. "Leonard" mostra a presença do marido nos textos e também é possível notar palavras temporais no texto (*never, time* e *last*).

5.4.2 Victoria McLeod

Na análise de sentimentos dados da Victoria McLeod da Figura 5.6 foi usada a classificação original com os seis estágios progressivos da deterioração mental que antecederam

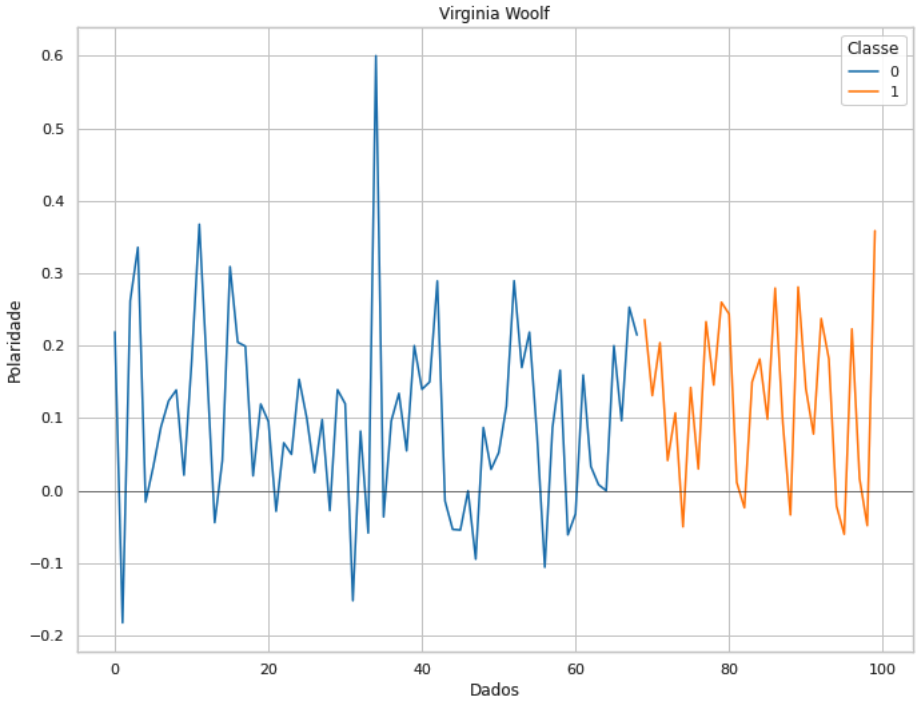

Figura 5.4: Análise de Sentimento - Virginia Woolf

ao suicídio, ressaltando que a morte aconteceu no estágio 6 (último). Como mostrado no conjunto de dados anterior, os dados positivos estão em rosa, os dados neutros em azul e os dados negativos em verde. É possível observar que a distribuição dos dados é muito parecida, assim como foi observado nos dados da Virginia Woolf, contudo, dessa vez, os dois extremos encontram-se justamente no estágio em que ocorreu o suicídio — estágio 6. É possível obsevar também uma menor desigualde entre a presença de palavras positivas e negativas nos estágios rotulados pelo profissional.

A linha do tempo no caso dos dados da Victoria McLeod, é a linha dos estágios por não haver indicativo temporal nos dados. Na Figura 5.7 os estágios de 0 a 3 estão na classe 0 e os estágios de 4 a 6 na classe 1 que indica ideação suicida. Observa-se os dados concentrados entre o intervalo -0.2 a 0.4 com poucos picos fora dele. Importante observar que, ao contrário dos dados anteriores, o sentimento mais negativo e mais positivo acontecem na classe 1, com ideação suicida. A Tabela com a análise de sentimento desse conjunto de dados: polaridade e seu respectivo sentimento, se encontra no Apêndice A.2.

A nuvem de palavras do conjunto de dados, Figura 5.8, ilustra a frequência das palavras mais usadas pela adolescente no período dos 3 meses que antecederam o evento suicida. Algumas ocorrências ilustram a vida de um típico adolescente: preocupação com escola e notas, assim como a presença de “palavrões” na escrita, outras favorecem

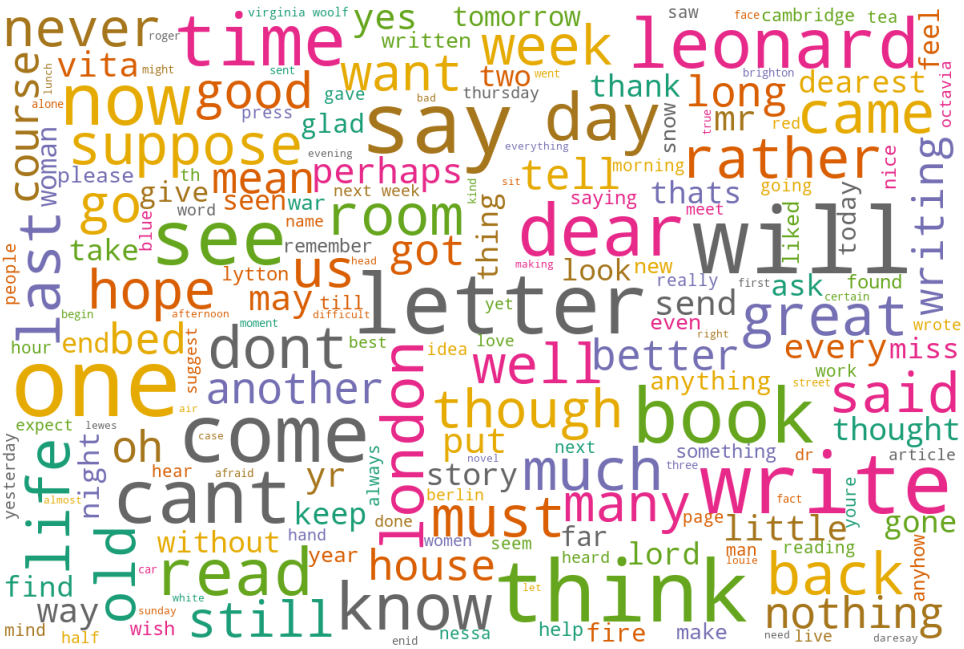

Figura 5.5: Núvem de Palavras - Virginia Woolf

a reflexão como: respirar, futuro, sentir, medo, coração, etc. Assim como nos dados da Virginia Woolf, é também possível identificar marcadores temporais no texto: *end, now, future, never, time, last*, etc. Um fato interessante é observar que, apesar de serem de gerações diferentes e de terem vivido em épocas diferentes, ainda assim é possível relacionar uma à outra através de algumas palavras comuns e com frequências parecidas, por exemplo: *time, life* e *will*.

Quando observamos a Figura 5.9 com as nuvens de palavras apenas das classes positivas dos dois conjuntos de dados é possível identificar os marcadores temporais e as mesmas palavras comuns identificadas acima: *time, life* e *will*. A nuvem da Virgínia (a) com apenas os dados positivos é muito similar à nuvem contendo todos os dados. De maneira oposta, a nuvem positiva da Victoria (b) apresenta palavras referentes ao medo (*fear*), falha/derrota (*failure*), desesperança (*hopeless*) e suicídio (*suicidal*). Nesse sentido, vale a observação que a Virgínia Woolf já era uma mulher madura e que, de certa forma, já estava acostumada a lidar com as crises depressivas que apresentava desde muito jovem. Ademais, a sua morte foi planejada, já que vinha de pelo menos 3 outras tentativas de suicídio mal sucedidas. Enquanto que a Victoria McLeod era uma adolescente tendo que conviver com a depressão que ainda era um fator novo na sua vida.

Sendo assim, concluímos que a dificuldade em separação dos dados e identificação correta dos rótulos provavelmente advém da similaridade entre as classes que mostrou a análise de sentimentos.

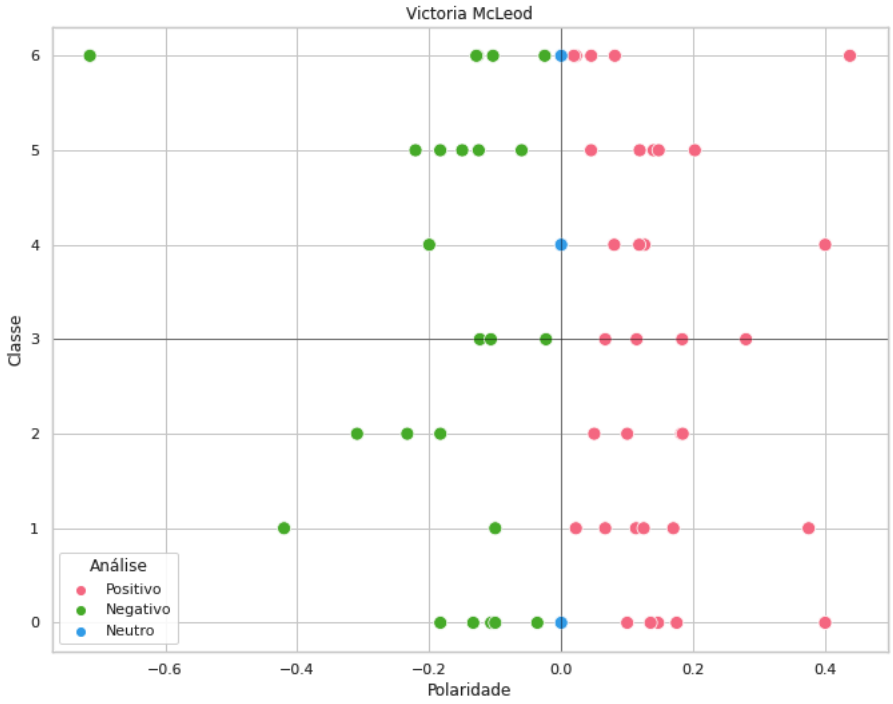

Figura 5.6: Análise de Sentimento - Victoria McLeod

Quando se trata do processamento de linguagem natural relacionada a estados depressivos e ao suicídio, a escolha do vocabulário a ser monitorado e usado como entrada para os algoritmos é complexo. Estudos não mostram uma correlação entre a severidade da depressão e as palavras que demonstram emoções positivas ou negativas — raiva, tristeza, etc. No entanto, é claro que pessoas acometidas pela doença experenciam mais emoções negativas e direcionam o foco para este universo, do que pessoas não afetadas pela depressão. Porém, o resultado nulo para palavras que expressam emoções negativas em dados (linguagem escrita/falada) de pessoas que apresentam sintomas de depressão, sugerem que esses indivíduos inibem os marcadores mais óbvios da doença através da regulação da sua linguagem emocional. Possivelmente, uma tentativa de evitar as consequências sociais da depressão. Pessoas depressivas, muitas vezes, tendem a mascarar a doença com o objetivo de não prejudicar os relacionamentos sociais (*networking*) [Holtgraves, 2014].

A nuvem de palavras como ferramenta coadjuvante na análise de sentimentos ofereceu uma perspectiva diferente dos conjuntos de dados, favorecendo assim a identificação de similiaridades entre eles que pode ser objeto de um estudo mais aprofundado.

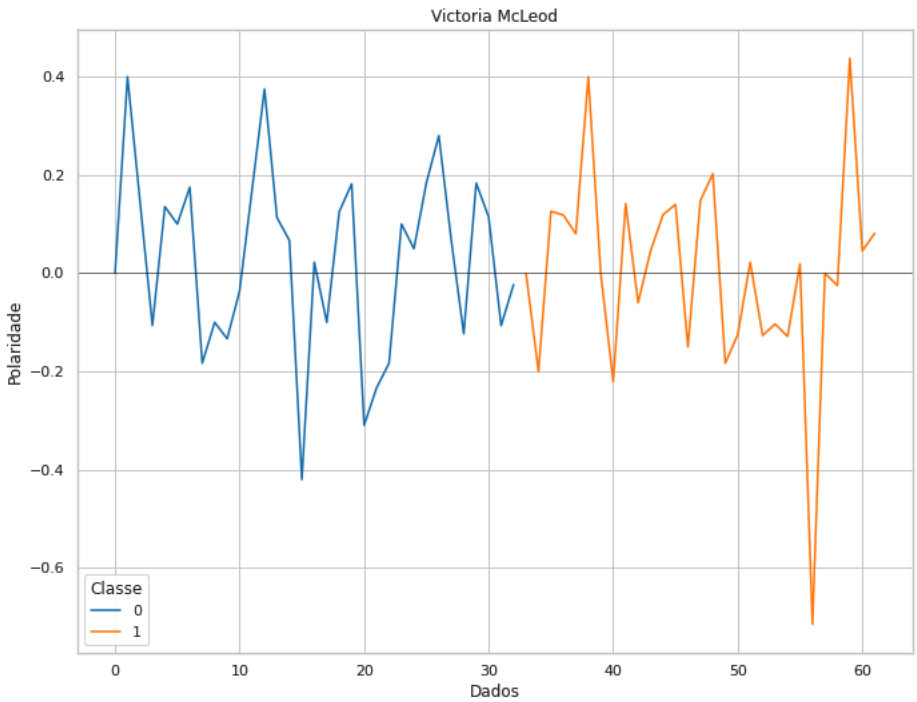

Figura 5.7: Análise de Sentimento - Victoria McLeod

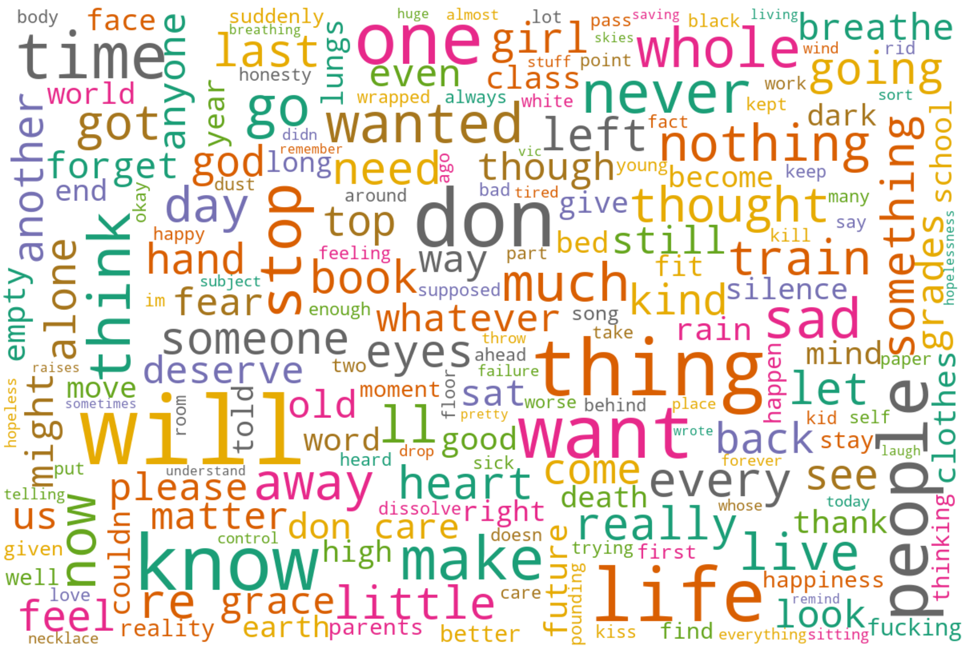

Figura 5.8: Núvem de Palavras - Victoria McLeod

(a) Virginia Woolf

(b) Victoria McLeod

Figura 5.9: Núvem de palavras - Classe Positiva

Capítulo 6

Conclusão

O presente estudo explorou o uso das ferramentas de seleção de caracteríscas TF-IDF e χ^2 como forma de melhorar a performance de modelos treinados para identificar textos com ideação suicida. Os resultados mostraram que o modelo que obteve o melhor resultado Roc-Auc (68%) na classificação dos dados da Virginia Woolf foi a NB usando o χ^2 como ferramenta de seleção de características. Quando o uso de ferramentas de seleção de característica teve o resultado *p-value* relevante ao teste estatístico, esse foi com o modelo combinado com o χ^2, no conjunto da Virgínia Woolf NB e SVM e nos dados da Victoria McLeod apenas o modelo FA. Os resultados também apontaram para uma performance melhor dos modelos NB, AD e RN em relação ao SVM e FA para os dados da Virgina Woolf. Experimento realizado por Berni et al. com o mesmo conjunto de dados e utilizando o modelo NB puro mostrou que o classificador apresentou uma sensibilidade de $69,23\%$ na detecção da classe positiva. O presente trabalho utilizando o χ^2 para seleção de características obteve 75% para essa mesma métrica (*Recall*). No caso dos dados da Victoria McLeod o melhor resultado Roc-Auc (64%) foi obtido com a combinação do modelo NB. Apesar do uso da ferramenta de seleção de características não ter se mostrado significativa ao teste estatístico para alguns modelos, quando usada, houve uma resposta positiva da métrica Roc-Auc para todos os modelos, exceto: RN nos dados da Virgina Woolf e SVM nos dados da Victoria McLeod.

Embora o estudo tenha apresentado resultados positivos para o uso de ferramentas de seleção de características, ele apresenta algumas limitações em relação aos conjuntos de dados utilizados:

- Em relação ao tamanho da amostra, como a proposta foi identificar a ideação suicida em texto de um indivíduo que completou o suicídio (morreu de fato), os dados públicos são raros e pequenos.

- Em relação à rotulagem dos dados, no conjunto da Virgínia Woolf, a rotulagem

foi feita de forma temporal e não considerou o estado mental da escritora. A similaridade dos dados nas duas classes, mostrada na análise de sentimentos, pode ser um indício de que a rotulagem pode não ter sido muito adequada. Dessa forma, é difícil afirmar que os textos fora do período de 60 dias que antecederam o suicídio estão livres de ideação suicida. No caso da Victoria McLeod, os dados são de um período curto, 3 meses, que antecedem o evento suicida — podendo ela já estar imersa no estado psicológico que a levou a concretizar o ato — mesmo os dados tendo passado por rotulagem de um especialista, isso poderia ser uma das causas da homogeneidade apontada na análise de sentimentos.

- A terceira questão a ser levantada é sobre a generalização do modelo: com dados tão específicos, de uma única pessoa, talvez não seja possível o uso dos modelos treinados para outros dados.

Sobre o último item acima, o estudo mostra algumas palavras comuns e com FA-equências parecidas entre duas pessoas com diferenças etárias e temporais: a primeira morreu aos 59 anos de idade em 1941 e a segunda faleceu com 17 anos de idade em 2014. Sendo assim, um estudo mais abrangente sobre a similaridade do estado mental entre pessoas com ideação suicida poderia apontar — caso tenha — para o limite na generalização de modelos para esse domínio. Seria interessante avaliar se classificadores treinados em gerações diferentes, grupos de indivíduos com culturas não similares ou até mesmo com diferenças socio-econômicas, teriam performances semelhantes quando aplicados em conjuntos diversos. Seria interessante também explorar outra abordagem na seleção de característica como por exemplo o *wrapper* utilizando o algoritmo genético como ferramenta.

Como a ideação suicida é um elo comum em todas as gerações e classes sociais, a prevenção ao suicídio é uma prioridade global devido a todos os fatores já abordados no início do texto. Este estudo espera contribuir para o avanço da utilização das técnicas de aprendizado de máquina nesse domínio.

Referências Bibliográficas

Akinsola, J. E. T. (2017). Supervised machine learning algorithms: Classification and comparison. *International Journal of Computer Trends and Technology (IJCTT)*, 48:128 – 138.

Albawi, S.; Mohammed, T. A. & Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pp. 1–6.

Aldehim, G.; De La Iglesia, B. & Wang, W. (2014). Heuristic ensemble of filters for reliable feature selection. International Conference on Pattern Recognition Applications and Methods (ICPRAM 2014) ; Conference date: 10-05-2014.

Alshari, E.; Azman, A.; Doraisamy, S. & Alksher, M. (2018). Effective method for sentiment lexical dictionary enrichment based on word2vec for sentiment analysis. pp. 1–5.

Baeza-Yates, R. A. & Ribeiro-Neto, B. (2011). *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., USA.

Bantilan, N.; Malgaroli, M.; Ray, B. & Hull, T. D. (2021). Just in time crisis response: suicide alert system for telemedicine psychotherapy settings. *Psychotherapy Research*, 31(3):289–299. PMID: 32558625.

Bering, J. (2020). *Suicidal: Why We Kill Ourselves*. University of Chicago Press.

Berni, G.; Rabelo-da Ponte, F.; Librenza-Garcia, D.; Boeira, M.; Kauer-Sant'Anna, M.; Passos, I. & Kapczinski, F. (2018). Potential use of text classification tools as signatures of suicidal behavior: A proof-of-concept study using virginia woolf's personal writings. *PLoS ONE*, 13.

Berry, M. W. & Kogan, J., editores (2010). *Text Mining: Applications and Theory*. Wiley, Chichester, UK.

Bolón-Canedo, V.; Sánchez-Maroño, N.; Alonso-Betanzos, A.; Benítez, J. & Herrera, F. (2014). A review of microarray datasets and applied feature selection methods. *Information Sciences*, 282:111–135.

Boser, B. E.; Guyon, I. M. & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, p. 144–152, New York, NY, USA. Association for Computing Machinery.

Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

Burnap, P.; Colombo, G.; Amery, R.; Hodorog, A. & Scourfield, J. (2017). Machine classification of suicide-related communication on twitter. In *Online Social Networks and Media*.

Chiroma, F.; Liu, H. & Cocea, M. (2018). Suicide related text classification with prism algorithm. *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*, 2:575–580.

Cholbi, M. (2017). Suicide. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, fall 2017 edição.

de Carvalho, V. F.; Giacon, B.; Nascimento, C. & Nogueira, B. M. (2020). Machine learning for suicidal ideation identification on twitter for the portuguese language. In Cerri, R. & Prati, R. C., editores, *Intelligent Systems*, pp. 536–550, Cham. Springer International Publishing.

Denil, M.; Matheson, D. & Freitas, N. D. (2014). Narrowing the gap: Random forests in theory and in practice. In Xing, E. P. & Jebara, T., editores, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 665–673, Bejing, China. PMLR.

Desmet, B. & Hoste, V. (2018). Online suicide prevention through optimised text classification. *INFORMATION SCIENCES*, 439:61–78.

Dias, T. N. (2015). *Desenvolvimento de Técnicas de Seleção de Atributos no Contexto da Classificação Hierárquica Monorrótulo*. PhD thesis, Universidade Federal de Ouro Preto.

Eisenstein, J. (2019). *Introduction to Natural Language Processing*. Adaptive Computation and Machine Learning series. MIT Press.

Erman, A. (1978). *The Ancient Egyptians: A Sourcebook of Their Writings*. Peter Smith.

Esteva, A.; Robicquet, A.; Ramsundar, B.; Kuleshov, V.; DePristo, M. A.; Chou, K.; Cui, C.; Corrado, G.; Thrun, S. & Dean, J. (2019). A guide to deep learning in healthcare. *Nature Medicine*, 25:24–29.

Forman, G. et al. (2003). An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.*, 3(Mar):1289–1305.

Goldberg, Y. (2017). *Neural Network Methods in Natural Language Processing (Synthesis Lectures on Human Language Technologies)*. Morgan Claypool Publishers.

Guerini, M.; Gatti, L. & Turchi, M. (2013). Sentiment analysis: How to derive prior polarities from sentiwordnet. *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*.

Gurney, K. (2004). *An introduction to neural networks*. Taylor Francis Group.

Haddi, E.; Liu, X. & Shi, Y. (2013). The role of text pre-processing in sentiment analysis. *Procedia Computer Science*, 17:26 – 32. First International Conference on Information Technology and Quantitative Management.

Holtgraves, T. M., editor (2014). *The Oxford Handbook of Language and Social Psychology*. Oxford University Press, Oxford, New York.

Horn, B. K. P. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Hovold, J. (2005). Naive bayes spam filtering using word-position-based attributes. In *Proceedings of the Second Conference on Email and Anti-Spam, http://www. ceas. cc*.

Hu, L.; Gao, W.; Zhao, K.; Zhang, P. & Wang, F. (2018). Feature selection considering two types of feature relevancy and feature interdependency. *Expert Systems with Applications*, 93:423–434.

Ji, S.; Pan, S.; Li, X.; Cambria, E.; Long, G. & Huang, Z. (2019). Suicidal ideation detection: A review of machine learning methods and applications. *ArXiv*, abs/1910.12611.

Joh, G. & hun Lee, Y. (2019). Identifying suicide notes using forensic linguistics and machine learning. volume 27, pp. 171–191.

Kaplan, K.; Schwartz, M. & Wolterstorff, N. (2008). *A Psychology of Hope: A Biblical Response to Tragedy and Suicide*. Eerdmans Publishing Company.

Li, J.; Chen, X.; Hovy, E. & Jurafsky, D. (2016). Visualizing and understanding neural models in NLP. Association for Computational Linguistics.

MacFarland, T. & Yates, J. (2016). *Mann–Whitney U Test*, pp. 103–132.

Maldonado, J. M.; Marques, A. B. & Cruz, A. (2016). Telemedicine: challenges to dissemination in brazil.

Malini, N. & Tan, V. (2016). Forensic linguistics analysis of virginia woolf's suicide notes. *International Journal of Education*, 9:50.

Manning, C. D. & Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, USA.

Mohri, M.; Rostamizadeh, A. & Talwalkar, A. (2012). *Foundations of Machine Learning*. Adaptive Computation and Machine Learning series. MIT Press.

O'Connor, R. C. & Nock, M. K. (2014). The psychology of suicidal behaviour. *The Lancet Psychiatry*, 1(1):73–85.

O'Dea, B.; Wan, S.; Batterham, P. J.; Calear, A. L.; Paris, C. & Christensen, H. (2015). Detecting suicidality on twitter. *Internet Interventions*, 2(2):183–188.

Ogada, K. O. (2016). *N-grams for Text Classification Using Supervised Machine Learning Algorithms*. PhD thesis, Jomo Kenyatta University Of Agriculture and Technology, P.O. Box 62 000 – 00200 NAIROBI, KENYA.

Organization, W. H. (2019). *Global Status Report on Road Safety 2018*. Nonserial Publication. World Health Organization.

Othero, G. d. (2006). Lingüística computacional: uma breve introdução. *Letras de Hoje*, v.41(n.2).

Parekh, A. & Phillips, M. (2014). *Preventing suicide: a global imperative*, p. 89.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M. & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pu, X.; Wu, G. & Yuan, C. (2017). Exploring overall opinions for document level sentiment classification with structural svm. *Multimedia Systems*, 25:21–33.

Quinlan, J. R. (1979). Discovering rules by induction from large collections of examples. In Michie, D., editor, *Expert Systems in the Micro-Electronic Age*, pp. 168–201. Edinburgh University Press, Edinburgh.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Santoro, D. M. & Nicoletti, M. D. C. (2005). Investigating a wrapper approach for selecting features using constructive neural networks. In *International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*, volume 2, pp. 77–82 Vol. 2.

Santos, H. G. (2018). *Comparação da performance de algoritmos de machine learning para análise preditiva em saúde pública e medicina*. PhD thesis, Universidade de São Paulo.

Silge, J. & Robinson, D. (2017). *Text Mining with R: A Tidy Approach*. O'Reilly Media, Inc., 1st edição.

Snoek, J.; Larochelle, H. & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms.

Sohn, S.; Torii, M.; Li, D.; Wagholikar, K. B.; Wu, S. T.-I. & Liu, H. (2012). A hybrid approach to sentiment sentence classification in suicide notes. In *Biomedical informatics insights*.

Souza, R. S. d. (2017). *Algoritmo Evolutivo com Representação Inteira para Seleção de Características*. PhD thesis, Universidade Federal de Goiás.

Srivastava, A. & Sahami, M. (2009). *Text mining. Classification, clustering, and applications*. Chapman Hall/CRC.

Sánchez-Franco, M. J.; Navarro-García, A. & Rondán-Cataluña, F. J. (2019). A naive bayes strategy for classifying customer satisfaction: A study based on online reviews of hospitality services. *Journal of Business Research*, 101:499–506.

Tadesse, M. M.; Lin, H.; Xu, B. & Yang, L. (2020). Detection of suicide ideation in social media forums using deep learning. *Algorithms*, 13:7.

Yin, W.; Kann, K.; Yu, M. & Schütze, H. (2017). Comparative study of cnn and rnn for natural language processing.

Zhang, H.; Ling, C. X. & Zhao, Z. (2000). The learnability of naive bayes. In Hamilton, H. J., editor, *Advances in Artificial Intelligence*, pp. 432–441, Berlin, Heidelberg. Springer Berlin Heidelberg.

Apêndice A

Análise de Sentimento

A.1 Virgínia Woolf

	Dia	Rótulo	Polaridade	Sentimento
0	4469	0	0.219	Positivo
1	4459	0	-0.182	Negativo
2	4456	0	0.262	Positivo
3	4443	0	0.336	Positivo
4	4442	0	-0.015	Negativo
5	4441	0	0.033	Positivo
6	4440	0	0.088	Positivo
7	4438	0	0.124	Positivo
8	4436	0	0.139	Positivo
9	4432	0	0.021	Positivo
10	4429	0	0.178	Positivo
11	4407	0	0.368	Positivo
12	4103	0	0.166	Positivo
13	4102	0	-0.044	Negativo
14	4093	0	0.042	Positivo
15	4078	0	0.309	Positivo
16	4075	0	0.205	Positivo
17	4074	0	0.199	Positivo
18	4071	0	0.021	Positivo
19	4070	0	0.119	Positivo
20	4066	0	0.095	Positivo
21	4065	0	-0.028	Negativo

	Dia	Rótulo	Polaridade	Sentimento
22	4063	0	0.066	Positivo
23	4063	0	0.050	Positivo
24	4061	0	0.154	Positivo
25	4061	0	0.100	Positivo
26	4060	0	0.025	Positivo
27	4058	0	0.098	Positivo
28	4057	0	-0.027	Negativo
29	4050	0	0.139	Positivo
30	3369	0	0.120	Positivo
31	3362	0	-0.152	Negativo
32	3361	0	0.082	Positivo
33	3360	0	-0.058	Negativo
34	3354	0	0.600	Positivo
35	3349	0	-0.036	Negativo
36	2152	0	0.096	Positivo
37	2151	0	0.134	Positivo
38	2151	0	0.055	Positivo
39	2147	0	0.200	Positivo
40	2146	0	0.140	Positivo
41	2131	0	0.150	Positivo
42	2000	0	0.289	Positivo
43	1998	0	-0.014	Negativo
44	1981	0	-0.053	Negativo
45	1972	0	-0.054	Negativo
46	450	0	0.000	Negativo
47	423	0	-0.094	Negativo
48	413	0	0.087	Positivo
49	369	0	0.029	Positivo
50	367	0	0.053	Positivo
51	325	0	0.117	Positivo
52	324	0	0.290	Positivo
53	320	0	0.170	Positivo
54	86	0	0.219	Positivo
55	83	0	0.075	Positivo
56	78	0	-0.106	Negativo

	Dia	Rótulo	Polaridade	Sentimento
57	78	0	0.087	Positivo
58	77	0	0.166	Positivo
59	75	0	-0.061	Negativo
60	72	0	-0.032	Negativo
61	69	0	0.160	Positivo
62	68	0	0.033	Positivo
63	67	0	0.008	Positivo
64	64	0	0.000	Neutro
65	62	0	0.200	Positivo
66	62	0	0.097	Positivo
67	61	0	0.253	Positivo
68	61	0	0.215	Positivo
69	55	1	0.236	Positivo
70	54	1	0.131	Positivo
71	54	1	0.204	Positivo
72	53	1	0.042	Positivo
73	52	1	0.107	Positivo
74	49	1	-0.050	Negativo
75	48	1	0.142	Positivo
76	46	1	0.030	Positivo
77	40	1	0.233	Positivo
78	39	1	0.146	Positivo
79	37	1	0.260	Positivo
80	33	1	0.244	Positivo
81	30	1	0.012	Positivo
82	27	1	-0.023	Negativo
83	24	1	0.150	Positivo
84	24	1	0.182	Positivo
85	20	1	0.099	Positivo
86	20	1	0.279	Positivo
87	18	1	0.094	Positivo
88	15	1	-0.033	Negativo
89	12	1	0.281	Positivo
90	10	1	0.141	Positivo
91	8	1	0.078	Positivo

	Dia	Rótulo	Polaridade	Sentimento
92	7	1	0.237	Positivo
93	7	1	0.182	Positivo
94	6	1	-0.021	Negativo
95	5	1	-0.060	Negativo
96	5	1	0.223	Positivo
97	4	1	0.017	Positivo
98	1	1	-0.048	Negativo
99	0	1	0.359	Positivo

A.2 Victoria Mcleod

	Rótulo	Polaridade	Sentimento
0	0	0.000	Neutro
1	0	0.400	Positivo
2	0	0.147	Positivo
3	0	-0.106	Negativo
4	0	0.135	Positivo
5	0	0.100	Positivo
6	0	0.175	Positivo
7	0	-0.183	Negativo
8	0	-0.100	Negativo
9	0	-0.133	Negativo
10	0	-0.036	Negativo
11	0	0.170	Positivo
12	0	0.375	Positivo
13	0	0.113	Positivo
14	0	0.067	Positivo
15	0	-0.420	Negativo
16	0	0.022	Positivo
17	0	-0.100	Negativo
18	0	0.125	Positivo
19	0	0.182	Positivo
20	0	-0.309	Negativo
21	0	-0.233	Negativo
22	0	-0.183	Negativo

	Rótulo	Polaridade	Sentimento
23	0	0.100	Positivo
24	0	0.050	Positivo
25	0	0.184	Positivo
26	0	0.280	Positivo
27	0	0.067	Positivo
28	0	-0.123	Negativo
29	0	0.183	Positivo
30	0	0.114	Positivo
31	0	-0.107	Negativo
32	0	-0.023	Negativo
33	1	0.000	Neutro
34	1	-0.200	Negativo
35	1	0.126	Positivo
36	1	0.118	Positivo
37	1	0.080	Positivo
38	1	0.400	Positivo
39	1	0.000	Neutro
40	1	-0.221	Negativo
41	1	0.142	Positivo
42	1	-0.060	Negativo
43	1	0.045	Positivo
44	1	0.119	Positivo
45	1	0.140	Positivo
46	1	-0.150	Negativo
47	1	0.148	Positivo
48	1	0.203	Positivo
49	1	-0.183	Negativo
50	1	-0.125	Negativo
51	1	0.022	Positivo
52	1	-0.127	Negativo
53	1	-0.103	Negativo
54	1	-0.129	Negativo
55	1	0.019	Positivo
56	1	-0.714	Negativo
57	1	0.000	Neutro

	Rótulo	Polaridade	Sentimento
58	1	-0.025	Negativo
59	1	0.438	Positivo
60	1	0.045	Positivo
61	1	0.081	Positivo

Apêndice B

Hiperparâmetros

Modelo	Hiperparâmetros	Virgínia Woolf	Victoria McLeod
NB	Grams	Uni - Bigram	Uni - Bigram
	TFIDF	500 - 1500	50 - 600
	χ^2	500 - 1500	50 - 600
SVM	Grams	Uni - Bigram	Uni - Bigram
	TFIDF	500 - 1500	50 - 600
	χ^2	500 - 1500	50 - 600
	Kernel	Linear, Polinomial,	Linear, Polinomial,
	.	Sigmoide, RBF	Sigmoide, RBF
AD	Grams	Uni - Bigram	Uni - Bigram
	TFIDF	500 - 1500	50 - 600
	χ^2	500 - 1500	50 - 600
	Profundidade Máxima	5 - 15	5 - 15
	Mínimo para Divisão	5 - 10	5 - 10
FA	Grams	Uni - Bigram	Uni - Bigram
	TFIDF	500 - 1500	50 - 600
	χ^2	500 - 1500	50 - 600
	Profundidade Máxima	15 - 300	15 - 300
	Número de Árvores	15 - 150	15 - 150
RN	Grams	Uni - Bigram	Uni - Bigram
	TFIDF	500 - 1500	50 - 600
	χ^2	500 - 1500	50 - 600
	Número de Camadas	10 - 100	10 - 100
	Taxa de Aprendizagem	0.01 - 0.3	0.01 - 0.3
	Função de Ativação	Tanh, ReLu,	Tanh, ReLu,
	.	Logística	Logística