UNIVERSIDADE FEDERAL DE MINAS GERAIS

Instituto de Ciências Exatas

Programa de Pós-Graduação em Estatística

Gabriel Augusto Narciso Barreiros

CLASSIFICAÇÃO E ANÁLISE DE VERBETES DA ENCICLOPÉDIA DA CONSCIENCIOLOGIA COM PROCESSAMENTO DE LINGUAGEM NATURAL E MÉTODOS DE MACHINE LEARNING

Belo Horizonte

2025

UNIVERSIDADE FEDERAL DE MINAS GERAIS

Instituto de Ciências Exatas

Programa de Pós-Graduação em Estatística

Gabriel Augusto Narciso Barreiros

CLASSIFICAÇÃO E ANÁLISE DE VERBETES DA ENCICLOPÉDIA DA CONSCIENCIOLOGIA COM PROCESSAMENTO DE LINGUAGEM NATURAL E MÉTODOS DE MACHINE LEARNING

Monografia apresentada ao Programa de Pós-Graduação em Estatística na Universidade Federal de Minas Gerais como requisito parcial para obtenção do título de Especialista em Estatística.

Orientador: Marcos Antonio da Cunha Santos

Belo Horizonte

2025

Barreiros, Gabriel Augusto Narciso.

B271c Classificação e análise de verbetes da Enciclopédia da Conscienciologia com processamento de linguagem natural e métodos de machine learning [recurso eletrônico] / Gabriel Augusto Narciso Barreiros – 2025.
1 recurso online (46 f. il., color.): pdf.

Orientador: Marcos Antonio da Cunha Santos.

Monografia (Especialização) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Estatística.
Referências: f. 41-45.

1. Estatística. 2. Análise de regressão logística. 3. Classificação (Computadores). 4. Aprendizado do computador. 5. Processamento de linguagem natural. 6. Redes neurais. I. Santos, Marcos Antônio da Cunha. I II. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Estatística. III. Título.

CDU 519.2(043)

Ficha catalográfica elaborada pela bibliotecária Irénquer Vismeg Lucas Cruz CRB 6/819 - Universidade Federal de Minas Gerais – ICEx

Universidade Federal de Minas Gerais
Instituto de Ciências Exatas
Departamento de Estatística
P Programa de Pós-Graduação / Especialização
Av. Pres. Antônio Carlos, 6627 - Pampulha
31270-901 – Belo Horizonte – MG

E-mail: pgest@ufmg.br
Tel: 3409-5923 – FAX: 3409-5924

ATA DO 355ª. TRABALHO DE FIM DE CURSO DE ESPECIALIZAÇÃO EM ESTATÍSTICA DE GABRIEL AUGUSTO NARCISO BARREIROS.

Aos cinco dias do mês de setembro de 2025, às 09:30, com utilização de recursos de videoconferência a distância, reuniram-se os professores abaixo relacionados, formando a Comissão Examinadora homologada pela Comissão do Curso de Especialização em Estatística, para julgar a apresentação do trabalho de fim de curso do aluno **Gabriel Augusto Narciso Barreiros**, intitulado: "Classificação e Análise de Verbetes da Enciclopédia da Conscienciologia com Processamento de Linguagem Natural e Métodos de Machine Learning", como requisito para obtenção do Grau de Especialista em Estatística. Abrindo a sessão, o Presidente da Comissão, Professor Marcos Antonio da Cunha Santos – Orientador, após dar conhecimento aos presentes do teor das normas regulamentares, passou a palavra ao candidato para apresentação de seu trabalho. Seguiu-se a arguição pelos examinadores com a respectiva defesa do candidato. Após a defesa, os membros da banca examinadora reuniram se sem a presença do candidato e do público, para julgamento e expedição do resultado final. Foi atribuída a seguinte indicação: o candidato foi considerado Aprovado. O resultado final foi comunicado publicamente ao candidato pelo Presidente da Comissão. Nada mais havendo a tratar, o Presidente encerrou a reunião e lavrou a presente Ata, que será assinada por todos os membros participantes da banca examinadora. Belo Horizonte, 05 de setembro de 2025.

Prof. Marcos Antonio da Cunha Santos (orientador)
DEST/UFMG

Prof. Luiz Henrique Duczmal
DEST/UFMG

gov.br
Documento assinado digitalmente
URIEL MOREIRA SILVA
Data: 07/09/2025 15:55:15-0300
Verifique em https://validar.iti.gov.br

Prof. Uriel Moreira Silva
DEST/UFMG

RESUMO

Atualmente, existe grande interesse em desenvolver análises estatísticas de texto. Extrair palavras-chave e criar vetores de forma eficiente para possibilitar a aplicação de métodos de análise estatística, algoritmos de classificação e detecção de padrões são desafios frequentes nessa área. A análise de sentimentos, que avalia o grau de positividade, neutralidade ou negatividade em textos, é um campo de pesquisa em expansão.

Com a finalidade de entender melhor técnicas de processamento de linguagem natural e desenvolver a análise de sentimentos, a presente pesquisa utiliza-se da linguagem de programação *Python* e suas diversas bibliotecas de processamento textual, processamento de dados e aprendizagem de máquina como *PyPDF, Pandas, NumPy, SpaCy, NLTK, Scikit-learn* e *SciPy*.

O método empregado envolve a extração de texto de arquivos PDF, a limpeza de dados para eliminar ruídos, informações ausentes e duplicadas, o pré-processamento dos dados para convertê-los ao formato adequado para entrada nos modelos e, por fim, a aplicação de modelos de aprendizagem de máquina para a classificação dos arquivos PDF.

A base de dados foi criada a partir de 2019 verbetes da *Enciclopédia da Conscienciologia*, cada um contendo informações como o título (ou tema de pesquisa) e classificação que pode ser positiva, neutra ou negativa.

O objetivo desta pesquisa é classificar os verbetes da *Enciclopédia da Conscienciologia* utilizando modelos de aprendizagem de máquina, como *naive bayes, regressão logística, máquina de vetores de suporte, florestas aleatórias* e *redes neurais* e encontrar o melhor método através da comparação de resultados.

Para validação dos modelos utilizou-se técnica de amostragem aleatória como *validação cruzada estratificada* e *f1-score* como métrica de classificação para classes desbalanceadas.

Palavras-chaves: machine learning; processamento de linguagem natural; redes neurais; algoritmos de classificação; análise de sentimentos.

ABSTRACT

Currently, there is a great interest in developing statistical text analysis. Extracting keywords and efficiently creating vectors to enable the application of statistical methods, classification algorithms, and pattern detection are frequent challenges in this field. Sentiment analysis, which assesses the degree of positivity, neutrality, or negativity in texts, is a growing research area.

To better understand natural language processing techniques and develop sentiment analysis, this research utilizes the *Python* programming language and its various libraries for text processing, data processing, and machine learning, such as *PyPDF, Pandas, NumPy, SpaCy, NLTK, Scikit-learn* and *SciPy*.

The method employed involves extracting text from PDF files, cleaning the data to eliminate noise, missing information, and duplicates, preprocessing the data to convert it into the appropriate format for model input, and finally, applying machine learning models to classify the PDF files.

The dataset was created using 2019 entries from the *Encyclopedia of Conscientiology*, each containing information such as the title (or research topic) and a classification that can be positive, neutral, or negative.

The objective of this research is to classify the entries from the *Encyclopedia of Conscientiology* using machine learning models such as *Naïve Bayes, Logistic Regression, Support Vector Classifiers, Random Forests* and *Neural Networks*. Additionally, a descriptive analysis of the results was performed using statistical techniques.

To validate the models, a random sampling technique was used, such as *stratified cross-validation*, and the *f1-score* was used as a classification metric for imbalanced classes.

Keywords: machine learning; natural language processing; neural networks; classification algorithms; sentiment analysis.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1. INTRODUÇÃO

Vivemos em uma era na qual a quantidade de informações geradas diariamente na internet, especialmente em redes sociais, fóruns e plataformas de avaliação, cresce de maneira exponencial. Nesse contexto, compreender e extrair valor de textos escritos por usuários torna-se uma tarefa estratégica para diversas áreas, como marketing, política, saúde, atendimento ao cliente e desenvolvimento científico. É nesse cenário que a análise de sentimentos e a classificação de textos ganham relevância como ferramentas capazes de transformar dados não estruturados em informações úteis para a tomada de decisão.

O presente trabalho explora técnicas de processamento de linguagem natural (PLN) e métodos de aprendizagem de máquina aplicados à análise de sentimentos. A partir de uma base de dados textual, busca-se construir modelos capazes de identificar automaticamente o sentimento predominante de um texto — como positivo, negativo ou neutro — bem como classificá-lo de acordo com categorias pré-definidas.

O estudo está inserido no campo da *estatística computacional aplicada*, integrando conhecimentos estatísticos, linguísticos e computacionais para lidar com desafios relacionados ao tratamento de linguagem natural. Por meio da aplicação de algoritmos de aprendizado supervisionado, além de técnicas de pré-processamento textual, pretende-se avaliar o desempenho de diferentes abordagens e discutir suas vantagens e limitações.

A base de dados foi construída com 2.019 verbetes da *Enciclopédia da Conscienciologia*, verbetes esses construídos através da análise minuciosa da consciência humana dentro de inesgotáveis temas, contextos, abordagens e especialidades. A *Enciclopédia da Conscienciologia* conta atualmente com cerca de 6.500 verbetes; 1.001 verbetógrafos; 727 especialidades (Ano-base: 2023).

O método empregado segue a sequência lógica que inicia-se na extração de dados dos arquivos PDF, passa por pré-processamento textual com objetivo de limpar e organizar o conjunto de dados, inserção dos dados nos modelos de aprendizagem de máquina, ajuste de parâmetros e hiperparâmetros e, por fim, análise do melhor modelo de classificação supervisionado.

Apesar do trabalho utilizar uma base de dados de enciclopédia conscienciológica, o trabalho não é sobre *Conscienciologia*, mas sim sobre aprendizagem de máquina e processamento de linguagem natural, suas aplicações, desafios e possibilidades.

2. JUSTIFICATIVA E FORMULAÇÃO DO PROBLEMA

Uma das particularidades deste trabalho é justamente a natureza dos dados. O primeiro aspecto é a *neociência Conscienciologia* proposta por Waldo Vieira (1932-2015) e o segundo a própria *Enciclopédia da Conscienciologia* que ditam o conteúdo e a forma da base de dados coletada.

Com finalidade de entender melhor o conjunto de dados utilizado na pesquisa segue breve histórico da *Conscienciologia* e enciclopédia.

A neociência foi proposta pelo médico e pesquisador brasileiro Waldo Vieira (1932-2015) e a ideia central da conscienciologia, como o próprio nome destaca, é o estudo (*lógos*) da consciência humana (*conscientia*) de forma integral, fundamentada em paradigma próprio (*paradigma consciencial*).

Em 1986 foi publicado o tratado *Projeciologia: Panorama da Experiência da Consciência fora do Corpo Humano* (Waldo Vieira) e em 1994 o tratado *700 Experimentos da Conscienciologia* (Waldo Vieira) que formam a base da ciência da consciência. Em 1995 o *Instituto Internacional de Projeciologia e Conscienciologia* recebe doação de terreno de 22.500m² em Foz do Iguaçu, PR, onde atualmente encontra-se o campus *Centro de Altos Estudos da Conscienciologia* (CEAEC) e o bairro *Cognópolis* (bairro do conhecimento) com diversas outras instituições e campi conscienciológicos, que conta com laboratórios de pesquisa, biblioteca, café e livraria, restaurante, ambientes de debates, cursos e palestras, tudo voltado ao estudo da consciência.

Figura 01: Centro de Altos Estudos da Conscienciologia, Foz do Iguaçu, PR.

Atualmente (dados de 2023) a conscienciologia conta com 29 Instituições Conscienciocêntricas, 1.986 voluntários, 3.872 atividades gratuitas, 3.765 cursos realizados, 23 revistas científicas, 200 autores de livros, 223 livros publicados e 1064 verbetógrafos da *Enciclopédia da Conscienciologia*.

A conscienciologia conta com novos termos (*neologismos*), conceitos e formas particulares de escrita. Tudo isso afeta o processamento de dados textuais, como os dados serão inseridos no modelo, quais técnicas e métodos podemos utilizar. Então, a primeira pergunta a se fazer é: As técnicas e métodos tradicionais de processamento de linguagem natural e de classificação textual funcionam nesse contexto, ou são necessárias adaptações?

Além disto, a *Enciclopédia da Conscienciologia*, em sua 10ª edição completa (2023), conta com cerca de 6.500 verbetes; 1.001 verbetógrafos; 727 especialidades em suas 34.612 páginas. Os temas são variados do simples ao complexo, desde títulos como *rotina útil*, *ansiedade* e *alcoolismo*, passando por *medo de errar*, *compaixão discernida* e *biblioteca de alexandria*, até *cultura da amizade evolutiva*, *ano de aplicação de técnicas projetivas* e *infopesquisa da bibliografia conscienciológica*.

Além da diversidade de temas, cada verbete da *Enciclopédia da Conscienciologia* conta com forma específica de escrita e chapa verbetográfica na qual o verbete deve ser desenvolvido, após a etapa de escrita a pesquisa passa por rigoroso processo de revisão pela equipe da *Encyslossapiens* (Associação Internacional de Enciclopediologia Conscienciológica) e por fim, defendido ao vivo, online e presencial, e aberto ao público no *Tertuliarium*, ambiente de debate do campus CEAEC. Todos esses detalhes são descritos no *Manual de Verbetografia da Enciclopédia da Conscienciologia* (2012).

Figura 02: Tertuliarium no campus CEAEC, Foz do Iguaçu, PR.

Como pode-se observar, com tal diversidade de temas, especificidade nos conteúdos e detalhes minuciosos na forma, surge a segunda pergunta: Quais os desafios na extração, organização, limpeza e tabulação desses dados? Os procedimentos são replicáveis para outros documentos e linhas específicas de conhecimento?

Ademais, a pesquisa também estrutura método para classificação textual que perpassa por diversas etapas: extração textual de PDFs, limpeza e correção textual, pré-processamento, classificação e análise. Cada etapa conta com diversas técnicas, conceitos e ferramentas visando o melhor resultado possível.

3. OBJETIVOS

Qualquer pesquisa científica qualificada preza por um bom método que seja claro e objetivo nos procedimentos e ao mesmo tempo flexível e adaptável a novas situações. O método permite a padronização da pesquisa e desenvolvimento de soluções para que outros pesquisadores possam replicar em seus objetos de estudo.

O método é o programa, processo, estratégia, lógica ou organização sistemática de técnicas, procedimentos e ações, visando a solução de problemas. No contexto desta pesquisa o método tem como finalidade a organização sistemática dos procedimentos do processamento de linguagem natural.

Seguindo essa lógica, o principal objetivo desta pesquisa é desenvolver um método de processamento de linguagem natural que inicia-se na extração textual até a análise dos resultados utilizando linguagem de programação, técnicas estatísticas e métodos de aprendizagem de máquina.

Os objetivos secundários tem como norte encontrar o modelo de aprendizagem de máquina que melhor atende às necessidades da base de dados, da pesquisa e da análise dos resultados obtidos ao longo do trabalho.

4. FUNDAMENTAÇÃO TEÓRICA

O processamento de linguagem natural (PLN) tem suas origens na década de 1950, marcadas pela publicação do artigo *Computing Machinery and Intelligence* de Alan Turing (1950). Este trabalho seminal introduziu o *Teste de Turing*, proposto como um padrão para avaliar a inteligência de máquinas.

No artigo Turing apresenta um jogo chamado *O Jogo da Imitação* (The Imitation Game) no qual propõe que, resumidamente, se uma máquina conseguisse se passar por um ser humano de forma convincente em uma proporção significativa das vezes, então poderíamos considerar que a máquina demonstra um comportamento inteligente, ou, nas palavras de Turing, que ela "pensa".

A ideia central do jogo da imitação é a seguinte:

- **Cenário:** Um interrogador humano se comunica por texto (sem contato visual ou auditivo) com dois participantes ocultos. Um desses participantes é um ser humano e o outro é uma máquina (um computador).
- **Objetivo do Interrogador:** Determinar qual dos participantes é a máquina e qual é o ser humano, baseando-se unicamente nas respostas escritas às suas perguntas.
- **Objetivo da Máquina:** Tentar enganar o interrogador, fazendo-o acreditar que ela é o ser humano.
- **Objetivo do Humano:** Ajudar o interrogador a fazer a identificação correta.

As interações ocorrem por texto, ou seja, o processamento de linguagem natural. Hoje o *Teste de Turing*, com as *inteligências artificiais generativas* e *large language models*, é problema da vida real presente em nosso dia a dia, artigos científicos, conteúdos de redes sociais, atendimento médico e terapêutico, conversas são gerados inteiramente por IAs com pouca ou nenhuma intervenção humana.

Em 1954, a *Universidade de Georgetown* juntou-se com a *IBM* para demonstrar a primeira máquina de tradução realizada em 7 de janeiro (John Hutchins, 2006). O experimento envolveu a tradução totalmente automática de mais de sessenta frases russas para o inglês e foi uma das primeiras tentativas de tradução por máquina.

Russo (romanizado)	Tradução para inglês
Mi pyeryedayem mislyi posryedstvom ryechyi.	We transmit thoughts by means of speech.
Vyelyichyina ugla opryedyelyayetsya otnoshyenyiyem dlyini dugi k radyiusu.	Magnitude of angle is determined by the relation of length of arc to radius.
Myezhdunarodnoye ponyimanyiye yavlyayetsya vazhnim faktorom v ryeshyenyiyi polyityichyeskix voprosov.	International understanding constitutes an important factor in decision of political questions.

Tabela 01: Tradução do russo para inglês por Georgetown–IBM system.

O algoritmo do sistema de tradução da IBM tinha como base regras, a abordagem era principalmente lexicográfica fundamentada em um dicionário onde uma palavra específica tinha uma conexão com regras e etapas específicas.

Palavra em russo	Equivalente em inglês	Código	Número da regra
vyelyichyina	magnitude	/-/-/	6
ugl-	coal, angle	121-/-25	2
-a	of	131-222-25	3

Tabela 02: Regras lexicográficas do Georgetown-IBM system.

O início dos chatbots ocorreu a partir de alguns sistemas de PLN desenvolvidos nos anos 60, entre eles um merece destaque, o *SHRDLU*. O nome *SHRDLU* foi derivado de ETAOIN SHRDLU, a ordem das teclas das letras em uma máquina *Linotype*, organizadas em ordem decrescente de frequência de uso em inglês.

SHRDLU foi um programa de computador de compreensão de linguagem natural desenvolvido por Terry Winograd no *MIT* em 1968-1970 (Stanford Human-Computer Interaction (HCI) Group). No programa, o usuário mantém uma conversa com o computador, movendo objetos, nomeando coleções e consultando o estado de um "blocks world" simplificado, essencialmente uma caixa virtual preenchida com blocos diferentes.

Segue uma demonstração do sistema:

- **Pessoa:** Pegue um bloco vermelho grande.
- **Computador:** OK.
- **Pessoa:** Segure a pirâmide.
- **Computador:** NÃO ENTENDO QUAL PIRÂMIDE VOCÊ QUER DIZER.
- **Pessoa:** Encontre um bloco mais alto que o que você está segurando e

coloque-o na caixa.

- **Computador:** POR "ELE", ASSUMO QUE VOCÊ QUER DIZER O BLOCO MAIS ALTO QUE O QUE ESTOU SEGURANDO.
- **Computador:** OK.
- **Pessoa:** O que a caixa contém?
- **Computador:** A PIRÂMIDE AZUL E O BLOCO AZUL.

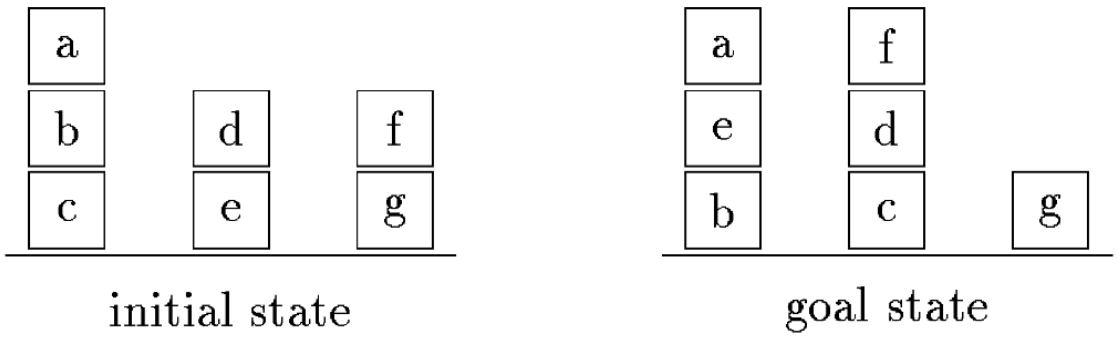
Outro destaque entre 1964 e 1966, Joseph Weizenbaum desenvolveu *ELIZA*, uma simulação de psicoterapeuta. Apesar de usar informações limitadas sobre cognição e emoção humanas, o programa gerava interações que, em certas ocasiões, pareciam notavelmente humanas. *ELIZA* é um exemplo relevante nessa área.

Blocks World era o principal algoritmo utilizado nestes sistemas, um problema clássico e largamente utilizado na área de inteligência artificial (IA), especialmente no estudo e desenvolvimento de algoritmos de planejamento automatizado. Ele serve como um ambiente simplificado para testar e demonstrar a capacidade de um agente de IA em raciocinar sobre ações e alcançar um estado desejado a partir de um estado inicial.

Neste cenário, o "mundo" consiste tipicamente em um conjunto de blocos de diferentes cores ou letras (por exemplo, A, B, C) que estão dispostos sobre uma mesa. Um agente, geralmente representado por um braço robótico com uma garra, pode interagir com esses blocos. O processo completo é exaustivamente detalhado no artigo *Blocks World revisited* (Slaney, 2000)

As regras e ações permitidas no *Blocks World* são geralmente as seguintes:

- **Pegar (Pickup):** O agente pode pegar um bloco que está sobre a mesa, desde que nada esteja em cima dele e a garra esteja vazia.
- **Colocar (Putdown):** O agente pode colocar um bloco que está segurando sobre a mesa.
- **Empilhar (Stack):** O agente pode colocar um bloco que está segurando em cima de outro bloco, desde que o bloco de destino esteja livre (nada em cima dele).
- **Desempilhar (Unstack)**: O agente pode pegar um bloco que está em cima de outro bloco, desde que nada esteja em cima do bloco a ser pego e a garra esteja vazia.



1. move a to table	2. move b to table	3. move d to c
4. move e to b	5. move a to e	6. move f to d

Figura 03: Planejamento e plano ideal com Blocks World.

Sistemas com base em aprendizado de máquina e métodos estatísticos começaram a surgir apenas na década de 1990, tais algoritmos revolucionaram o processamento de linguagem natural (Hutchins, 2014, p. 3). O aumento do poder computacional em conjunto com a computação linguística foi responsáveis pela diminuição gradual da dominância das teorias da linguística.

Um dos algoritmos de aprendizado de máquina mais antigos utilizados em PLN foi o *árvores de decisão* e a partir daí começaram a surgir novos modelos estatísticos com bases probabilísticas e atribuição de peso aos corpos textuais. Pesquisas atuais direcionam seus esforços para algoritmos de aprendizagem semi-supervisionados ou sem supervisão, pois são capazes de aprender com dados que não foram categorizados manualmente com as respostas desejadas ou usando combinação de dados categorizados e não categorizados.

Outra opção são os algoritmos de aprendizagem supervisionada nos quais já se sabe a saída desejada do modelo a partir de rótulos e assim torna-se possível treinar para inferir em dados nunca vistos. Por fim, modelos de aprendizagem profunda, uma abordagem recente, empregam redes neurais com múltiplas camadas para realizar tarefas como classificação e regressão. Inspirado pela neurociência biológica, esse campo foca em empilhar neurônios artificiais em camadas e treiná-los para o processamento de dados. A aprendizagem supervisionada e profunda são a base da presente pesquisa.

O processamento de linguagem natural tem aplicações diversas, desde a tradução de idiomas até análise de discurso, tais técnicas e métodos vêm sendo utilizadas na vida real em diversos casos, atualmente (Ano-base 2025) os agentes de inteligência artificial merecem

destaque pois através da linguagem natural é possível construir sistemas autônomos para realizar tarefas específicas, interagir com o ambiente e tomar decisões com pouca ou nenhuma intervenção humana, por exemplo: chatbots, assistentes virtuais, sistemas de recomendação, análise de dados, robótica, as possibilidades são infinitas. A seguir listaremos, em ordem alfabética, algumas das principais aplicações na área da PLN.

Uma das aplicações mais comuns é o *reconhecimento de entidade nomeada* (NER) que consiste em extrair e classificar entidades a partir de um texto. Esse processo emprega conhecimentos de linguística computacional para manipular palavras e textos conforme suas classes gramaticais, inferindo os limites e a classificação das entidades.

Segundo o artigo *Processing Named Entities in Text* (McNamee, 2011, p. 33), NER consiste em identificar em um texto sequências de palavras que correspondem a uma taxonomia predefinida de entidades, como pessoas, organizações e locais. Tal como acontece com a tecnologia relacionada da *part-of-speech tagging* (marcação de classe gramatical), a maioria das abordagens de NER tenta rotular cada palavra numa frase com a sua classe apropriada.

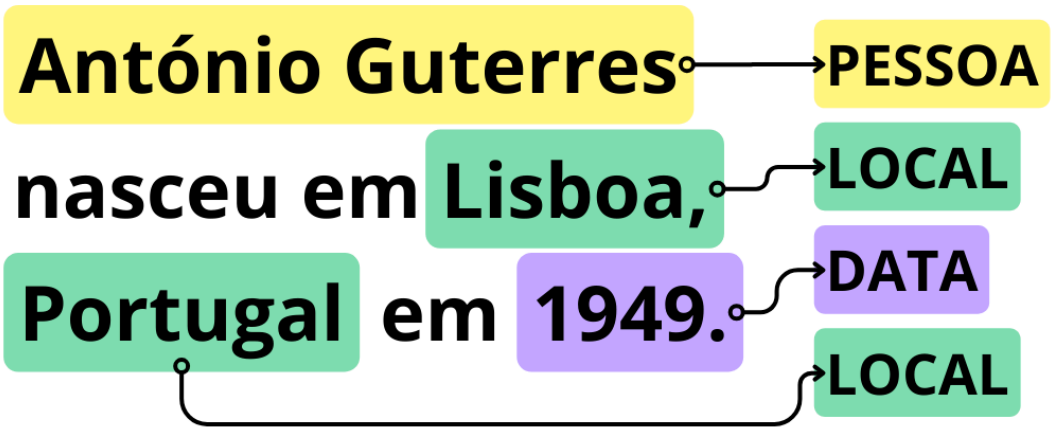


Figura 04: Reconhecimento de entidade nomeada.

Outra aplicação comum é o *reconhecimento óptico de caracteres* (OCR), uma tecnologia comum para identificar caracteres em arquivos de imagem ou bitmap, incluindo documentos escaneados, manuscritos, datilografados ou impressos. O OCR permite converter esses arquivos em texto editável por computador. A tecnologia OCR, aliada à inteligência artificial, possibilita a automação de processos empresariais como cadastro, onboarding e formalização em diversos setores. Isso é feito através da extração de dados relevantes

presentes em documentos como identificações pessoais, contratos e comprovantes de residência.

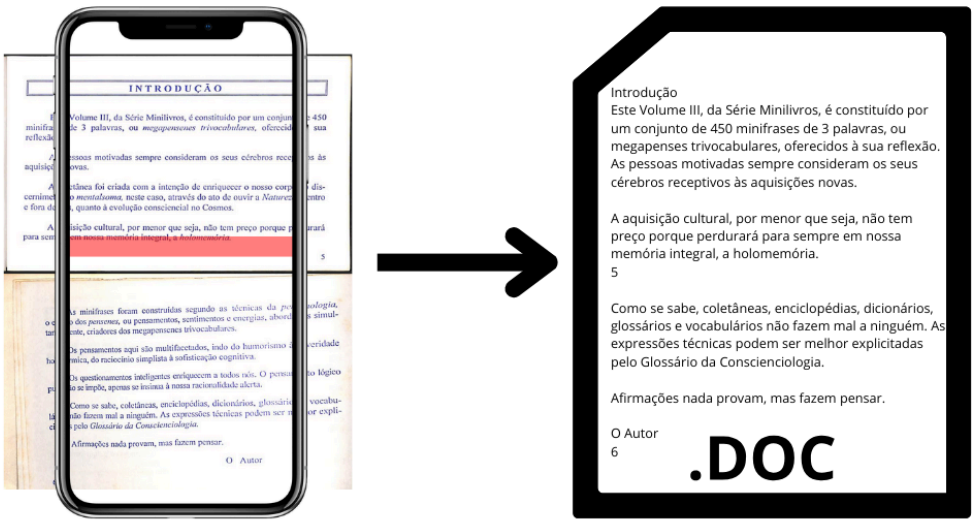


Figura 05: Reconhecimento óptico de caracteres (OCR).

Dentre muitas outras aplicações, ainda temos *recuperação de informação* (IR) e *extração de informação* (IE), ambas utilizam métodos de PLN para extrair, armazenar, pesquisar e recuperar informações a partir do texto como *stopwords removing*, *punctuation removing*, *unidecoding*, *stemming*, *lemmatization* e *tokenization*.

Método	Texto original
Texto original	Com a PNL, máquinas decifram a complexidade da linguagem humana, abrindo portas para a comunicação intuitiva entre nós e a inteligência artificial.
Stopwords removing	PNL, máquinas decifram complexidade linguagem humana, abrindo portas comunicação intuitiva nós inteligência artificial.
Punctuation removing	Com a PNL maquinas decifram a complexidade da linguagem humana abrindo portas para a comunicação intuitiva entre nos e a inteligencia artificial
Unidecoding	Com a PNL, maquinas decifram a complexidade da linguagem humana, abrindo portas para a comunicacao intuitiva entre nos e a inteligencia artificial.
Tokenization	['Com', 'a', 'PNL', ', ', 'máquinas', 'decifram', 'a', 'complexidade', 'da', 'linguagem', 'humana', ', ', 'abrindo', 'portas', 'para', 'a', 'comunicação', 'intuitiva', 'entre', 'nós', 'e', 'a', 'inteligência', 'artificial', '.']
Stemming	com a pnl , máquina decifram a complexidad da linguagem humana , abrindo porta para a comunicação intuitiva entr nó e a inteligência artifici .

Lemmatization	Com a PNL , máquinas decifram a complexidade da linguagem humana , abrindo porta para a comunicação intuitiva entre nós e a inteligência artificial .

Tabela 03: Extração e recuperação de informações.

O processamento de linguagem natural (PLN) abrange métodos quantitativos para o tratamento automatizado da linguagem, como modelagem probabilística, teoria da informação e álgebra linear. A base tecnológica do PLN estatístico reside principalmente no aprendizado de máquina e na mineração de dados, campos da inteligência artificial focados no aprendizado a partir de dados.

O processamento estatístico em língua natural emprega métodos estocásticos, probabilísticos e estatísticos para mitigar desafios, notadamente a ambiguidade de frases extensas processadas com gramáticas complexas, as quais podem gerar inúmeras análises possíveis.

A possibilidade de métodos, técnicas e aplicações são infinitas, listamos algumas para um panorama geral e parte delas usaremos nos próximos capítulos.

5. METODOLOGIA

A presente pesquisa é fundamentalmente descritiva, pois busca atribuir rótulos a textos com base em padrões aprendidos. Mas, em determinadas situações, também utiliza métodos exploratórios e explicativos.

Por exemplo, a centralidade do estudo é a classificação textual dos verbetes entre positivo, negativo ou neutro, esse ponto traz o caráter descritivo, mas também utiliza-se método exploratório para o processamento dos dados necessário para a inserção no modelo de aprendizagem de máquina. Por fim, a análise utiliza-se método explicativo para aferir o melhor modelo e as razões por trás das classificações do modelo.

Na seção de fundamentação teórica buscou-se ao máximo utilizar fontes primárias para explicitar breve histórico do processamento de linguagem natural, mas a principal fonte da pesquisa é o próprio experimento com modelos de aprendizagem de máquina, técnica de manipulação textual, métodos de amostragem e métricas de classificação para definir o modelo com maior precisão e acurácia.

Ao longo da análise, direciona-se os esforços para resultados quantitativos mais confiáveis e objetivos, mas se tratando de dados textuais e processamento de linguagem natural, não há como deixar de fora a análise qualitativa, tornando o trabalho qualiquantitativo.

A população de verbetes da *Enciclopédia da Conscienciologia* conta atualmente com cerca de 6.500 verbetes; 1.001 verbetógrafos; 727 especialidades (Ano-base: 2023). A seleção da amostra não foi aleatória, mas sim intencional, foram selecionado 2019 verbetes e o critério foi utilizar todos os verbetes do propositor da enciclopédia (Waldo Vieira) pois estes serviriam como base para toda a enciclopédia e outros verbetógrafos que viriam a escrever futuramente.

Além das informações acima, o conjunto de dados conta com 836 textos positivos, 827 neutros e 356 negativos. Apesar do desbalanceamento de valores negativos, não foi utilizada nenhuma técnica de balanceamento de dados, ao invés optou-se por utilizar validação cruzada estratificada que preserva a proporção percentual de cada classe para treino e teste. A técnica *K-Fold* estratificada é preferível para lidar com problemas de classificação com distribuições de classes desbalanceadas (PRUSTY, 2022). A porcentagem de dados para treino e teste é definida pelo parâmetro *n_split*, que nesse caso foi 5 ou seja, 2019/5 = 405,8 (20%).

A principal característica do *StratifiedKFold* é que ele garante que a proporção de classes no conjunto de teste (e, consequentemente, no de treinamento) seja aproximadamente a mesma que a proporção de classes no conjunto de dados original. Isso é crucial, especialmente para conjuntos de dados desbalanceados, onde algumas classes têm muito menos amostras que outras. Sem a estratificação, um *fold* de teste poderia acabar com pouquíssimas ou nenhuma amostra da classe minoritária, levando a uma avaliação enviesada do modelo.

O método segue a sequência lógica que inicia-se na extração de dados dos arquivos PDF e finaliza na análise do melhor modelo de classificação supervisionado. Eis, em ordem lógica de desenvolvimento, 3 passos que compõem o método da presente pesquisa:

1. **Extração de dados:** o primeiro passo é a extração dos dados textuais de arquivos PDF. Para isso, utilizou-se técnicas de manipulação textual como *expressões regulares* (RegEx) e a biblioteca *PyPDF*.

2. **Pré-processamento:** o segundo passo consiste no pré-processamento dos dados, ou seja, transformá-los e organizá-los no formato ideal para inserir nos modelos de aprendizagem de máquina. Para tal, também foi utilizado *expressões regulares* e bibliotecas como *unidecode*, *NLTK* e *spacy*.

3. **Ajuste dos modelos:** o terceiro passo é o ajuste dos modelos de aprendizagem de máquina e técnicas para chegar a parâmetros com maior acerto para o modelo escolhido. Para isso foram utilizado técnicas de amostragem aleatória, divisão treino-teste e métricas para escolha do modelo com melhor desempenho. Utilizou-se a biblioteca *scikit-learn*.

6. RESULTADOS

A apresentação dos resultados foi dividida em três partes de acordo com o método utilizado durante a pesquisa, ou seja, extração de dados, pré-processamento e ajuste de modelos. Para servir de exemplo real, as tabelas a seguir serão baseadas no conjunto de dados utilizado na pesquisa, porém, apenas com as 5 primeiras linhas de todo o conjunto.

Extração de Dados

A extração de dados é o processo de coletar dados de diversas fontes e convertê-los para um formato utilizável para análise, relatórios ou armazenamento. É uma etapa crucial em fluxos de trabalho de dados, como ETL (Extrair, Transformar, Carregar) e ELT (Extrair, Carregar, Transformar), que preparam dados para insights e tomada de decisões.

A biblioteca utilizada nesta etapa foi a *PyPDF* que sua função é manipular e extrair dados de arquivos PDF. Em conjunto com expressões regulares foi extraído o texto dos 2019 PDFs, removendo cabeçalho e rodapé, dividido em duas colunas, uma para o texto e outra para o sentimento do textual.

Texto	Sentimento
Conformática\n\n Definologia. O megafoco autopensênico é a manutenção da autopensenidade da cons-\nciência em determinado ponto ideativo, específico, com a fixação da vontade,...	neutro
Conformática\n\n Definologia. A inteligência técnica é a aptidão, talento, habilidade, discernimento, pers-\npicácia, intelecção, interpretação e acuidade desenvolta da lucidez pess...	neutro
Conformática\n\n Definologia. A megatolice é ato máximo, manifestação pensênica tola, impensada ou\ninepta, de alta expressão de ignorância, capaz de paralisar a evolução da consc...	negativo
Conformática\n\n Definologia. A megapolivalência é o conjunto versátil dos multivalores intraconscien-\nciais, ou megatrafores máximos, da conscin de nível superior quanto à vivência exem...	positivo
Conformática\n\n Definologia. A hiperacuidade pancognitiva é a qualidade de lucidez máxima da conscin\nalcançada pela recuperação possível dos cons ou das unidades de autoconsciência quan...	positivo

Tabela 04: Extração textual dos PDFs.

Pré-processamento

O pré-processamento de dados é um conjunto de técnicas utilizadas para transformar dados brutos em um formato adequado para análise e modelagem, especialmente em aprendizado de máquina. Essa etapa é crucial para garantir a qualidade dos dados e melhorar a performance de modelos.

O próximo passo foi remover alguns elementos desnecessários para o modelo em nosso texto e fazer algumas transformações, por exemplo, deixar todas palavras em minúsculos, remover pontuações, remover numerações, espaçamentos duplos e muitos mais. Para isso, a técnica utilizada foi expressões regulares.

Texto	Sentimento
o megafoco autopensenico e a manutencao da autopensenidade da consciencia em determinado ponto ideativo especifico com a fixacao da vontade da concentracao mental e da atencao o primeiro elemento...	neutro
a inteligencia tecnica e a aptidao talento habilidade discernimento perspicacia inteleccao interpretacao e acuidade desenvolta da lucidez pessoal capaz de demonstrar criatividade no emprego teatic... neutro	neutro
a megatolice e ato maximo manifestacao pensenica tola impensada ou inepta de alta expressao de ignorancia capaz de paralisar a evolucao da consciencia o elemento de composicao mega deriva do idio...	negativo
a megapolivalencia e o conjunto versatil dos multivalores intraconscienciais ou megatrafores maximos da conscin de nivel superior quanto a vivencia exemplarista da inteligencia evolutiva ie o pri...	positivo
a hiperacuidade pancognitiva e a qualidade de lucidez maxima da conscin alcancada pela recuperacao possivel dos cons ou das unidades de autoconsciencia quanto a autocognicialidade o elemento de c...	positivo

Tabela 05: Pré-processamento textual dos textos.

Stopwords, ou palavras de parada, são palavras comuns em um idioma que geralmente são ignoradas em análises de texto, como em buscas na internet ou processamento de linguagem natural (PNL). Exemplos comuns em português incluem "o", "a", "os", "as", "e", "de", "em", "para", "um", "uma". A remoção de *stopwords* pode melhorar a eficiência de algoritmos de busca e análise de texto, pois concentra-se nas palavras mais significativas. Nesta etapa foram utilizadas as bibliotecas *NLTK* e *spacy* que contém listas de *stopwords* definidas.

Texto	Sentimento
megafoco autopensenico manutencao autopensenidade consciencia determinado ideativo especifico fixacao vontade concentracao mental atencao elemento composicao mega idioma grego megas megale grandem...	neutro
inteligencia tecnica aptidao talento habilidade discernimento perspicacia inteleccao interpretacao acuidade desenvolta lucidez pessoal capaz demonstrar criatividade emprego teatico desenvolvimento...	neutro
megatolice ato maximo manifestacao pensenica tola impensada inepta alta expressao ignorancia capaz paralisar evolucao consciencia elemento composicao mega deriva idioma grego megas megale grandeme...	negativo
megapolivalencia conjunto versatil multivalores intraconscienciais megatrafores maximos conscin nivel superior vivencia exemplarista inteligencia evolutiva ie elemento composicao mega deriva idiom...	positivo
hiperacuidade pancognitiva qualidade lucidez maxima conscin alcancada recuperacao possivel cons unidades autoconsciencia autocognicialidade elemento composicao hiper idioma grego hyper acima acima...	positivo

Tabela 06: Pré-processamento textual na remoção de stopwords.

Stemmer e *Lemmatizer* são duas *técnicas de pré-processamento* de texto muito importantes no *processamento de linguagem natural* (PLN), ou seja, são transformadores morfológicos das palavras. Ambas têm o objetivo de reduzir as diferentes formas de uma palavra à sua forma base, o que ajuda a padronizar o texto e a melhorar a análise. No entanto, elas fazem isso de maneiras diferentes:

1. **Stemmer:** "correndo", "corre", "correu" = "corr".
2. **Stemmer:** "amigos", "amizade", "amigável" = "amig".
3. **Stemmer:** "gato", "gata", "gatos", "gatas" = "gat".
4. **Lemmatizer:** "correndo", "corre", "correu" = "correr".
5. **Lemmatizer:** "melhor", "melhores" = "bom" (se considerarmos que "melhor" é o comparativo de "bom").
6. **Lemmatizer:** "fui", "era", "serei" = "ser".

Stemmer	Lemmatizer	Sentimento
megafoco autopensenico manutencao autopensenidad consciencia determinado ideativo especifico fixacao vontad concentracao mental atencao elemento composicao mega idioma grego mega megal grandement ...	megafoco autopensenico manutencao autopensenidade consciencia determinar ideativo especifico fixacao vontade concentracao mental atencao elemento composicao mega idioma grego mega megale grandemen...	neutro
inteligencia tecnica aptidao talento habilidad discernimento perspicacia	inteligencia tecnico aptidao talento habilidade discernimento perspicacia	neutro

inteleccao interpretacao acuidad desenvolta lucidez pessoal capaz demonstrar criatividad emprego teatico desenvolvimento te...	inteleccao interpretacao acuidade desenvolto lucidez pessoal capaz demonstrar criatividade emprego teatico desenvolvimento...	
megatolic ato maximo manifestacao pensenica tola impensada inepta alta expressao ignorancia capaz paralisar evolucao consciencia elemento composicao mega deriva idioma grego mega megal grandement ...	megatolice ato maximo manifestacao pensenico tolo impensado inepto alto expressao ignorancia capaz paralisar evolucao consciencia elemento composicao mega derivar idioma grego mega Megale grandeme...	negativo
megapolivalencia conjunto versatil multivalor intraconscienciai megatrafor maximo conscin nivel superior vivencia exemplarista inteligencia evolutiva ie elemento composicao mega deriva idioma greg...	megapolivalencia conjunto versatil multivalores intraconsciencial megatrafor maximo conscin nivel alto vivencia exemplaristo inteligencia evolutivo ie elemento composicao mega derivar idioma grego...	positivo
hiperacuidad pancognitiva qualidad lucidez maxima conscin alcancada recuperacao possivel con unidad autoconsciencia autocognicialidad elemento composicao hiper idioma grego hyper acima acima super...	Hiperacuidade pancognitivo qualidade lucidez maximo conscin alcancar recuperacao possivel cons unidade autoconscienciar autocognicialidade elemento composicao hiper idioma grego hyper acima acima ...	positivo

Tabela 07: Pré-processamento textual com stemmer e lemmatizer.

Em processamento de linguagem natural*, Bag of Words* (BoW), ou "saco de palavras", é uma das técnicas mais fundamentais e amplamente utilizadas para representar documentos de texto como dados numéricos, que podem ser processados por algoritmos de aprendizado de máquina.

O processo de criação de um modelo Bag of Words geralmente segue estes passos:

1. **Coleta do vocabulário:** Primeiramente, todas as palavras únicas (o vocabulário) de um conjunto de documentos (chamado de corpus) são identificadas.
2. **Criação de um vetor para cada documento:** Para cada documento, é criado um vetor numérico. A dimensão desse vetor é igual ao tamanho do vocabulário.
3. **Contagem de frequência:** Cada posição no vetor corresponde a uma palavra específica do vocabulário. O valor nessa posição indica a frequência com que aquela palavra aparece no documento.

Existem diversas técnicas para se criar um “saco de palavras”, nesta pesquisa utilizamos 3 mais conhecidas. A primeira delas é o *CountVectorizer* da biblioteca scikit-learn

com o parâmetro binário ativado (*binary=True*), ou seja, o resultado final será apenas 0 e 1 indicando se a palavra existe ou não no texto.

abarcando	abertismo	abordagens	voliciolina	voliciologia	voluntaria	Sentimento
0	0	1	0	1	1	0
0	0	1	0	0	1	0
0	1	1	0	0	0	1
1	0	1	0	0	1	1
0	1	1	1	0	1	2

Tabela 08: Pré-processamento textual com BoW e contagem de vetores binário.

A segunda técnica é o *CountVectorizer* da biblioteca scikit-learn com o parâmetro binário desativado (*binary=False*), ou seja, o resultado final será, literalmente, uma contagem indicando quantas vezes a palavra aparece no texto.

abarcando	abertismo	abordagens	voliciolina	voliciologia	voluntaria	Sentimento
0	0	1	0	1	1	0
0	0	1	0	0	1	0
0	1	1	0	0	0	1
1	0	1	0	0	1	1
0	1	2	1	0	1	2

Tabela 09: Pré-processamento textual com BoW e contagem de vetores.

Por último, foi utilizada a *técnica TfidfVectorizer* (Term Frequency-Inverse Document Frequency), usada para avaliar a importância de uma palavra em um documento em relação a um conjunto de documentos (*corpus*). Pensa assim: algumas palavras são muito comuns em qualquer texto, como "o", "a", "e". Essas palavras, mesmo aparecendo muito, não carregam muito significado sobre o conteúdo de um documento específico. O *TF-IDF* resolve isso.

Ele faz isso combinando duas métricas e segue os seguintes passos da fórmula:

1. Frequência do Termo **(TF - *Term Frequency*)**.
    a. O Term Frequency (TF) mede a frequência com que um termo (*t*) aparece em um documento (*d*).

   b. A fórmula mais comum para o TF é: TF(t,d) = frequência de t em d
2. Frequência Inversa do Documento **(IDF - *Inverse Document Frequency*)**.
   a. O Inverse Document Frequency (IDF) mede a raridade de um termo em todo o corpus. Quanto mais rara uma palavra for no conjunto de documentos, maior será o seu IDF.
   b. A fórmula mais comum para o IDF é: IDF(t,D) = log(N/DF(t))
   c. N é o número total de documentos no corpus (D).
   d. DF(t) é o número de documentos no corpus que contêm o termo t (Document Frequency).

O resultado final do TF-IDF para uma palavra em um documento é o produto dessas duas métricas:

$$TF - IDF(t,\ d,\ D) = TF(t,\ d) \times IDF(t,\ D)$$

Em Resumo:

1. Se um termo aparece frequentemente em um documento (alto TF) e é raro em todo o corpus (alto IDF), ele terá um alto TF-IDF, indicando que é uma palavra muito relevante para aquele documento específico.
2. Se um termo aparece frequentemente em um documento (alto TF) mas também é muito comum em todo o corpus (baixo IDF), ele terá um baixo TF-IDF, pois não ajuda a distinguir aquele documento de outros.
3. Se um termo não aparece em um documento (TF = 0), seu TF-IDF será zero.

abarcando	abertismo	abordagens	voliciolina	voliciologia	voluntaria	Sentimento
0.00000	0.00000	0.00996	0.00000	0.02090	0.01178	0
0.00000	0.00000	0.01564	0.00000	0.00000	0.01849	0
0.00000	0.03085	0.01822	0.00000	0.00000	0.00000	1
0.02747	0.00000	0.01309	0.00000	0.00000	0.01548	1
0.00000	0.01358	0.01604	0.01683	0.00000	0.00948	2

Tabela 10: Pré-processamento textual com BoW e vetorizador TF-IDF.

Ajuste dos Modelos

Ajuste de modelo em aprendizagem de máquina se refere ao processo de otimizar os hiperparâmetros de um modelo para que performe da melhor forma possível em um dado específico. É como afinar um instrumento musical para que ele produza o som mais harmonioso.

Se estes hiperparâmetros não forem ajustados, o modelo pode ter um desempenho abaixo do ideal. Por exemplo, pode apresentar :

- **Underfitting:** O modelo é muito simples e não consegue capturar os padrões complexos nos dados de treinamento, resultando em baixa performance tanto nos dados de treinamento quanto nos dados novos. É como se a melodia fosse muito simples e sem graça.
- **Overfitting:** O modelo é muito complexo e "memoriza" os dados de treinamento, incluindo o ruído e as particularidades irrelevantes. Ele performa bem nos dados de treinamento, mas muito mal em dados novos e não vistos. É como se a melodia fosse tão complexa que ninguém consegue acompanhar.

O ajuste de modelo busca encontrar o equilíbrio perfeito, onde o modelo aprende o suficiente para fazer boas previsões em dados novos, sem memorizar demais os dados de treinamento.

Para atingir tal equilíbrio é crucial entender a diferença entre hiperparâmetros vs. parâmetros:

- **Parâmetros do Modelo:** São as variáveis que o modelo aprende a partir dos dados durante o treinamento. Por exemplo, em uma regressão linear, os coeficientes (inclinação e intercepto da linha) são os parâmetros. Em uma rede neural, os pesos das conexões são os parâmetros. Você não define esses valores, o algoritmo os descobre.
- **Hiperparâmetros:** São as configurações externas do modelo que você define antes do treinamento começar. Eles controlam como o modelo aprende e a estrutura do modelo.

Esta etapa de ajuste dos modelos de aprendizagem de máquina seguiu a seguinte ordem lógica:

1. Escolha do modelo de aprendizagem de máquina.
2. Divisão da base de dados entre treino e teste com validação cruzada estratificada *K-Fold*.

3. Transformação textual com *Porter Stemmer, Snowball Stemmer* e *Lemmatizer*.
4. Criação da *bag of words* (BoW) para inserção dos dados no modelo com e sem padronização dos dados utilizando *StandardScaler*.
5. Ajuste do modelo com diversas variações de hiperparâmetros.
6. Escolha do modelo que melhor atende às necessidades da pesquisa com base na métrica *f1-score*.
7. Repetir as etapas anteriores até encontrar os melhores hiperparâmetros do modelo.
8. Repetir as etapas anteriores para todos modelos de aprendizagem de máquina escolhidos.

Para facilitar esse processo foi utilizada duas funções da bibliotecas *scikit-learn: GridSearchCV* para iteração entre os hiperparâmetros do modelo e dos vetorizadores e *pipeline* para criar uma rotina lógica para cada loop do algoritmo.

Os modelos escolhidos foram *Naive Bayes, Regressão Logística, C-Máquina de Vetores de Suporte, Florestas Aleatórias* e *Redes Neurais* com 1, 2 e 3 camadas ocultas. Para melhor compreensão do processo, apresentamos no apêndice uma lista com os parâmetros e seus valores que foram testados durante o ajuste de modelos.

Ao avaliar o desempenho de modelos de classificação em *machine learning*, é crucial entender as diferenças e vantagens de métricas como acurácia, precisão, sensibilidade e *f1-score*. Cada uma delas oferece uma perspectiva única sobre quão bem o modelo está performando, especialmente em cenários com classes desbalanceadas.

Para compreender essas métricas, é útil pensar em termos de Verdadeiros Positivos (VP), Verdadeiros Negativos (VN), Falsos Positivos (FP) e Falsos Negativos (FN), que são os resultados possíveis da classificação de um modelo:

- **Verdadeiro Positivo (VP):** o modelo previu positivo e a instância realmente é positiva
- **Verdadeiro Negativo (VN):** modelo previu negativo e a instância realmente é negativa
- **Falso Positivo (FP):** o modelo previu positivo, mas a instância realmente é negativa (erro tipo I)
- **Falso Negativo (FN):** o modelo previu negativo, mas a instância realmente é positiva (erro tipo II).

A acurácia é a métrica mais intuitiva e amplamente utilizada. Ela mede a proporção de previsões corretas (tanto positivas quanto negativas) em relação ao total de previsões. Em

conjuntos de dados onde uma classe é muito mais frequente que a outra (desbalanceamento de classes), a acurácia pode ser enganosa. Por exemplo, se 95% das instâncias são negativas, um modelo que sempre prevê negativo terá 95% de acurácia, mas não detecta nenhum positivo.

A precisão foca nos Verdadeiros Positivos entre todas as instâncias que o modelo classificou como positivas. Ela responde à pergunta: "De todas as instâncias que o modelo previu como positivas, quantas eram realmente positivas?".

A sensibilidade (também conhecido como recall) foca nos Verdadeiros Positivos entre todas as instâncias que são realmente positivas. Ele responde à pergunta: "De todas as instâncias que são realmente positivas, quantas o modelo identificou corretamente?".

O *f1-score* é a média harmônica da precisão e da sensibilidade. Ele é uma métrica que busca equilibrar a Precisão e a Sensibilidade, sendo útil quando você precisa de um balanço entre a minimização de Falsos Positivos e Falsos Negativos.

Quando se trata de dados desbalanceados, o *f1-score* é uma ótima escolha para validação dos resultados do modelo, desde que usado com uma técnica para lidar com esse problema, por exemplo, a validação cruzada estratificada. Além disso, a taxa de desequilíbrio de classes também deve ser reportada (LIU, 2023).

$$\textbf{Fórmula} = \frac{2(Precisão \times Sensibilidade)}{Precisão + Sensibilidade}$$

Após toda avaliação os maiores valores de *f1-score* foram: regressão logística (*f1-score:* 0.79), máquinas de vetores de suporte (*f1-score:* 0.79), *MLPClassifier* ou redes neurais (*f1-score:* 0.78). Segue uma tabela com os melhores ajustes de cada modelo ranqueados de acordo com *f1-score* em ordem decrescente:

Modelo	Transformador	Vetorizador	Melhores Parâmetros	F1-Score
Logistic Regression	Porter Stemmer	TfidfVectorizer	{'model__C': 1.5, 'model__class_weight': 'balanced', 'model__solver': 'lbfgs', 'vectorizer__max_df': 1.0, 'vectorizer__min_df': 1, 'vectorizer__norm': 'l2', 'vectorizer__smooth_idf': False, 'vectorizer__sublinear_tf': True}	0.79
Support Vector Classifier	Porter Stemmer	CountVectorizer	{'model__C': 10,	0.79

			'model__class_weight': 'balanced', 'model__kernel': 'rbf', 'vectorizer__binary': True, 'vectorizer__max_df': 1.0, 'vectorizer__min_df': 1}	
MLPClassifier	Porter Stemmer	CountVectorizer	{'model__activation': 'logistic', 'model__alpha': 0.01, 'model__hidden_layer_sizes': (50,), 'model__learning_rate': 'constant', 'vectorizer__binary': False, 'vectorizer__max_df': 1.0, 'vectorizer__min_df': 1}	0.78
Naive Bayes	Porter Stemmer	CountVectorizer	{'model__alpha': 1.0, 'vectorizer__binary': False, 'vectorizer__max_df': 0.8, 'vectorizer__min_df': 1}	0.76
Random Forest	Porter Stemmer	CountVectorizer	{'model__n_estimators':100, 'model__class_weight': None, 'model__criterion': 'gini', 'model__max_depth': None, 'model__max_features': None, 'model__min_samples_leaf': 4, 'model__min_samples_split': 5, 'vectorizer__binary': False, 'vectorizer__max_df': 1.0, 'vectorizer__min_df': 1}	0.76

Tabela 11: Melhor ajuste de cada modelo.

Apenas com base no *f1-score,* não foi possível definir quais dos modelos foram os melhores, a diferença entre um e outra é muito baixa e não tão significante quando limitada apenas a uma métrica. No entanto, considerando que a *regressão logística* obteve o maior valor de *f1-score* e, é um modelo significativamente mais fácil de explicar as decisões, utilizaremos como base para entender quais palavras têm maior influência na classificação de cada classe. A *regressão logística* é um modelo linear interpretável por natureza:

- **Interpretabilidade:** Os coeficientes de cada característica na equação da Regressão Logística indicam a direção e a magnitude da influência dessa característica no resultado. Por exemplo, se o coeficiente para "idade" é positivo, isso significa que, quanto maior a idade, maior a probabilidade do evento acontecer (mantendo outras variáveis constantes). Isso torna a Regressão Logística muito transparente.
- **Probabilidades:** Ela gera probabilidades de pertencer a uma classe, o que pode ser diretamente interpretado como o grau de confiança da previsão.

- **Simplicidade:** A sua base matemática é relativamente simples de entender, mesmo para públicos não técnicos, permitindo uma explicação clara sobre como cada fator contribui para a decisão final.
- **Transparência:** É considerada um modelo "caixa branca" devido à sua estrutura linear e à clareza de como as variáveis de entrada influenciam o resultado.

Para tal análise, foi transformado os coeficientes da regressão logística em *odds ratio* (OR), ou razão de chances (RC), uma medida de associação utilizada em estudos e em análises estatísticas para quantificar a força da associação entre uma exposição e um resultado.

A interpretação da Odds Ratio é feita em relação ao valor 1 de modo usual:

- **OR = 1:** Não há associação entre a variável e a classificação
- **OR > 1:** Estar exposto à categoria está associado a um aumento nas chances de classificação positiva .
- **OR < 1:** A exposição está associada a uma diminuição nas chances de classificação positiva..

Associando o métrica *odds ratio* às palavras e suas classes pode-se criar um gráfico de barras indicando as principais palavras e sua influência na classificação dos textos.

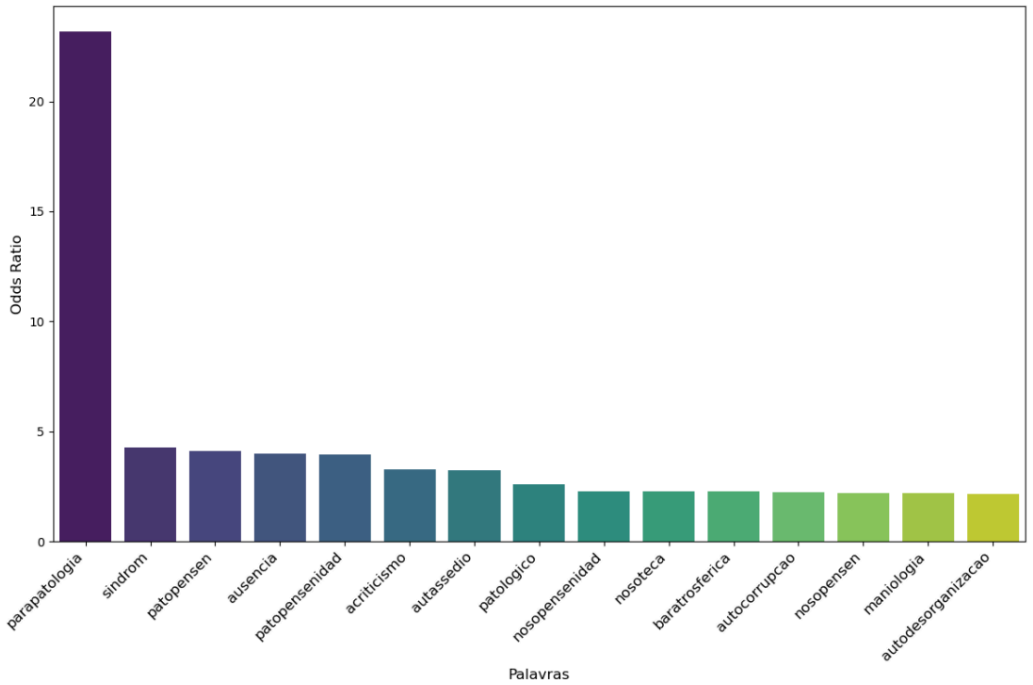


Figura 06: 15 palavras mais influentes para classificação do texto como negativo.

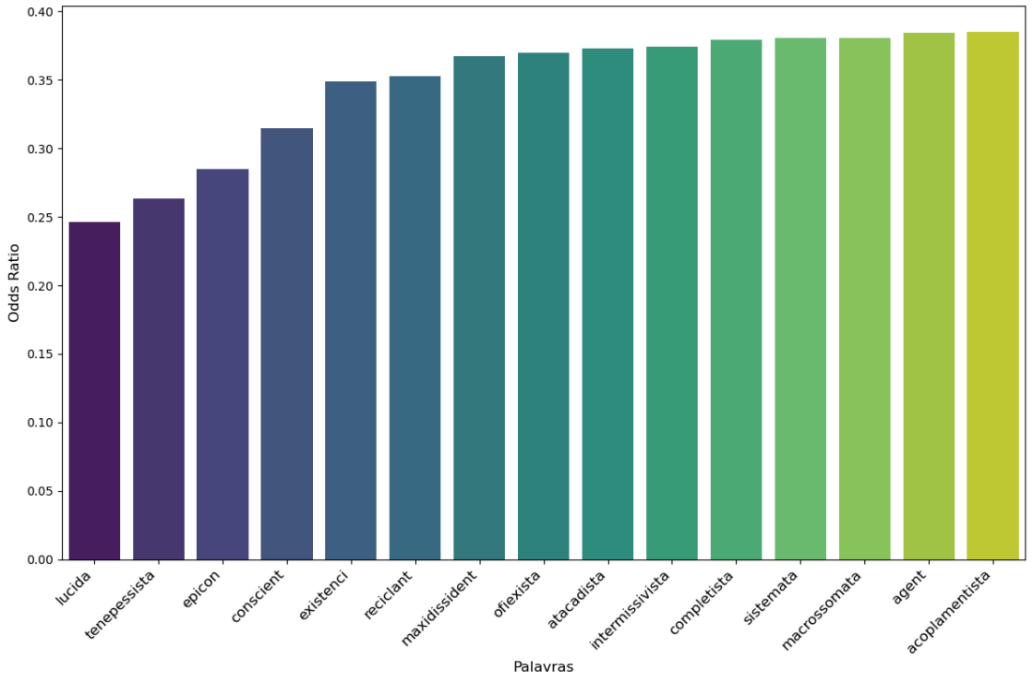


Figura 07: 15 palavras menos influentes para classificação do texto como negativo.

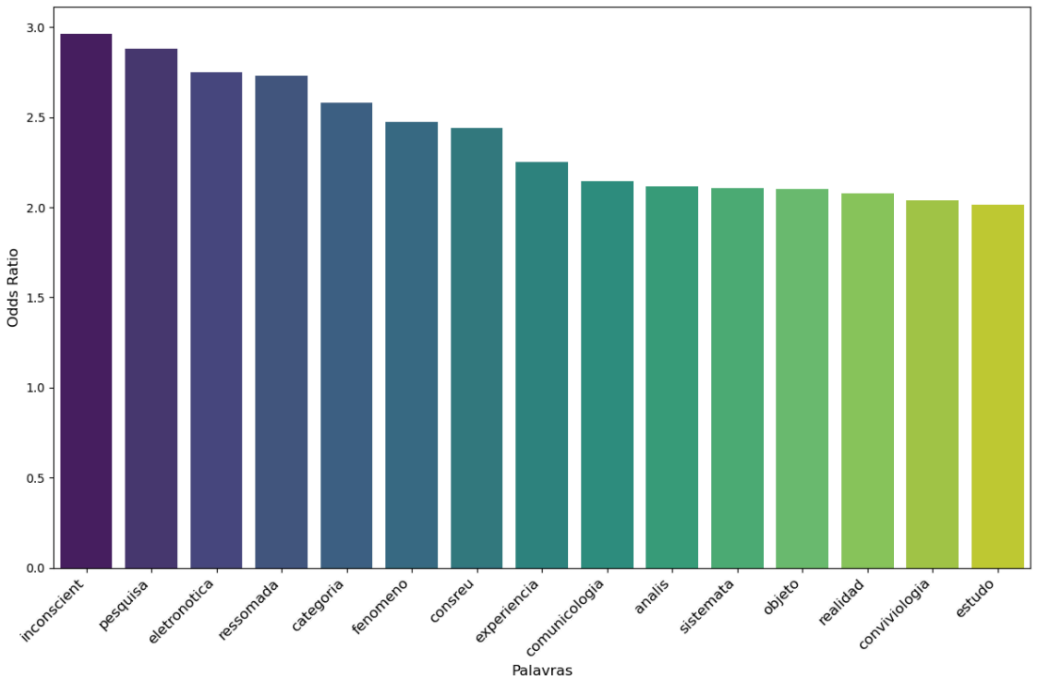


Figura 08: 15 palavras mais influentes para classificação do texto como neutro.

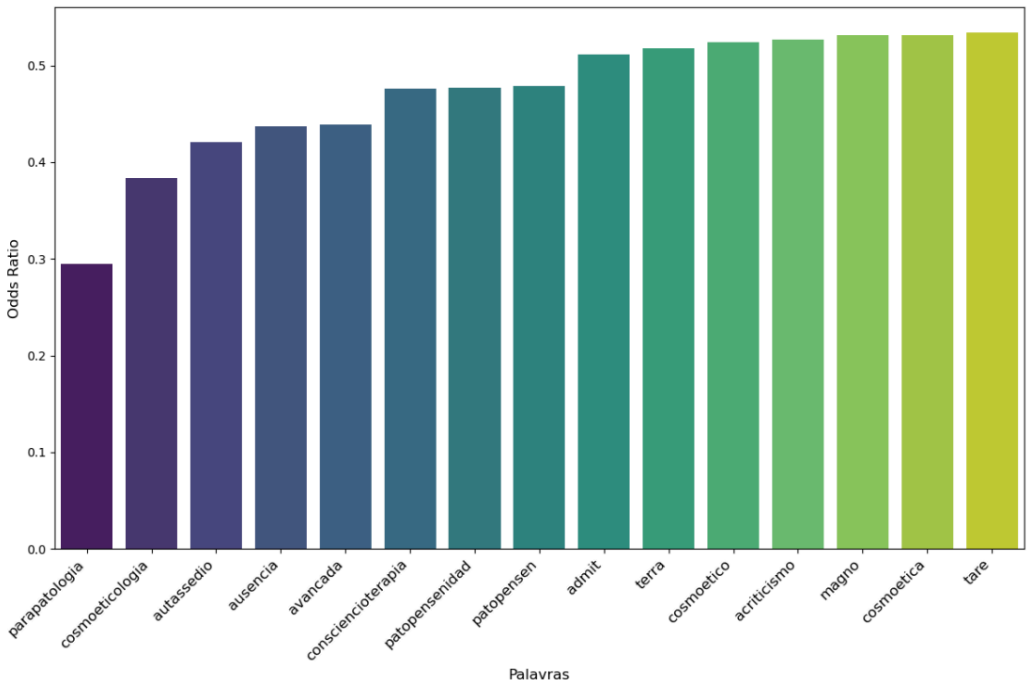


Figura 09: 15 palavras menos influentes para classificação do texto como neutro.

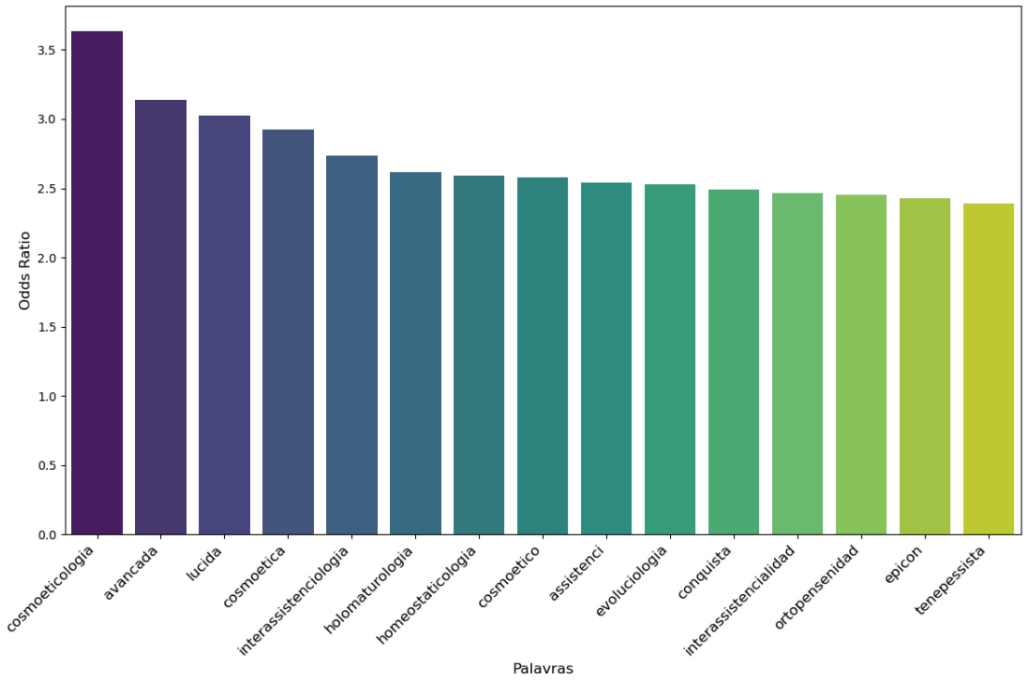


Figura 10: 15 palavras mais influentes para classificação do texto como positivo.

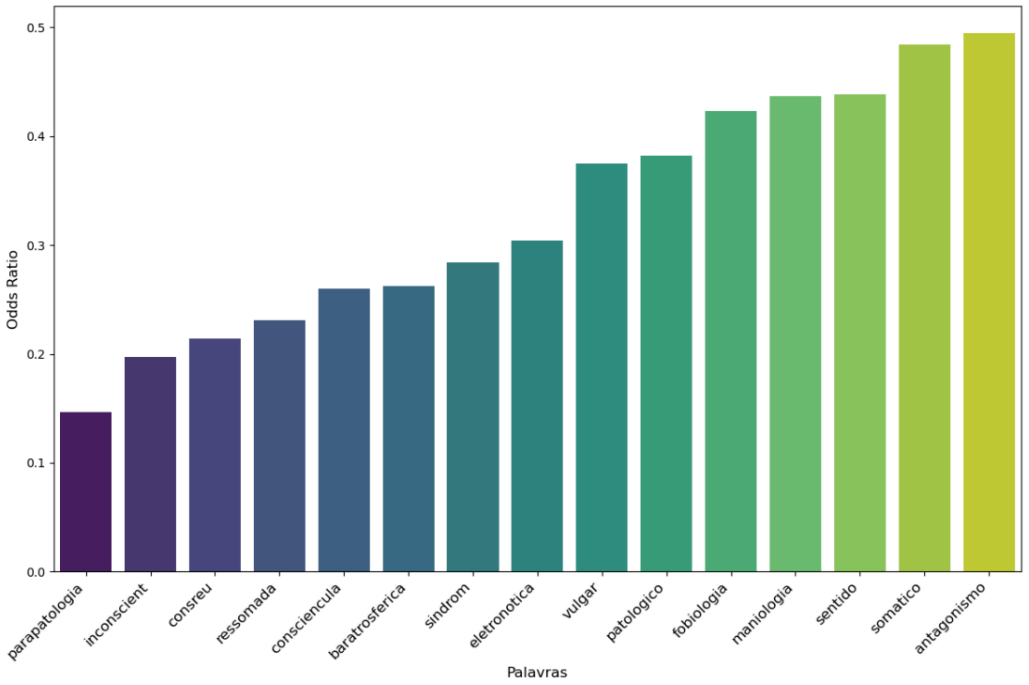


Figura 11: 15 palavras menos influentes para classificação do texto como positivo.

Outra técnica interessante para visualizar a distribuição de dados bidimensionais ou tridimensionais é a *t-Distributed Stochastic Neighbor Embedding* (t-SNE), um método estatístico para visualizar dados de alta dimensão, dando a cada ponto de dados uma localização em um mapa. É um algoritmo de redução de dimensionalidade, proposto por Laurens van der Maaten e Geoffrey Hinton (2008). O objetivo é representar dados de várias dimensões em 2D ou 3D para efeito de visualização, procurando preservar as relações locais de vizinhança presente nos dados.

Esta técnica foi utilizada em um dos modelos estudados, a Regressão Logística. Na biblioteca sklearn utilizamos os seguintes parâmetros:

1. n_components=3.
2. random_state=0.
3. perplexity=30.
4. n_iter=1000.
5. init='random'.

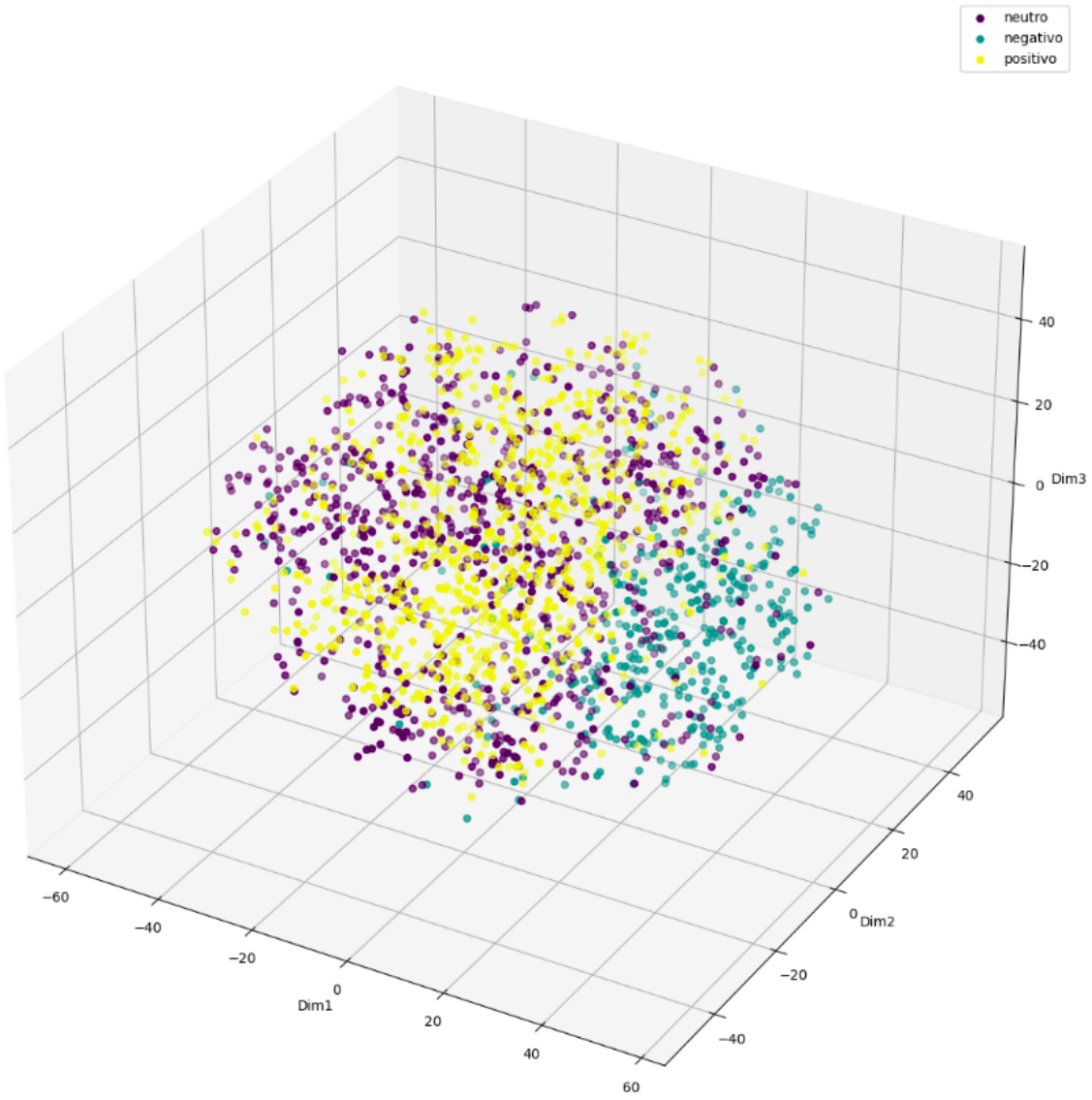


Figura 12: Incorporação estocástica de vizinhos distribuídos em t (t-SNE).

O gráfico acima nos traz algumas ideias interessantes sobre a classificação dos textos entre neutro, negativo e positivo:

1. **Distribuição Geral:** Os pontos estão distribuídos em uma forma aproximadamente esférica ou no espaço 3D. Isso sugere que não há clusters extremamente separados, mas sim uma sobreposição e transição entre as diferentes classificações.
2. **Negativo (verde-água):** Parece haver uma concentração mais clara de pontos verdes-água em uma região específica, mais para o lado direito do gráfico (valores mais altos em Dim2 e Dim3, e possivelmente mais baixos em Dim1). Embora não seja um cluster perfeitamente isolado, ele demonstra uma tendência a agrupar-se.

3. **Positivo (amarelo):** Os pontos amarelos estão amplamente dispersos e misturados com os pontos roxos (neutros) no centro e em grande parte do volume da esfera. Há uma presença forte de pontos amarelos em todas as regiões, mas sem formar um agrupamento denso e exclusivo.
4. **Neutro (roxo):** Assim como os pontos amarelos, os pontos roxos estão bastante dispersos e misturados com os pontos amarelos. Eles parecem ocupar a maior parte do espaço central, misturando-se com os positivos.
5. **Grau de Mistura/Sobreposição:** Existe uma sobreposição significativa entre os rótulos "positivo" (amarelo) e "neutro" (roxo). Isso sugere que os dados subjacentes para essas duas classes são bastante semelhantes. Mesmo assim, é possível perceber leve predominância dos pontos amarelos no centro da esfera e predominância dos pontos roxos na esquerda da esfera, indicando possíveis pontos de diferenciação entre os dois.
6. **Tendência 1:** A dificuldade em separar claramente os "positivo" e "neutro" pode indicar que as características que definem essas classes são muito similares. Há certa tendência de textos neutros serem mais próximos de serem positivos e menos negativos.
7. **Tendência 2:** O fato de o "negativo" mostrar um agrupamento um pouco mais distinto pode sugerir que os dados negativos possuem características mais únicas ou que são mais facilmente separáveis dos outros dois.

7. CONCLUSÃO

Com base na análise dos verbetes da *Enciclopédia da Conscienciologia*, esta pesquisa demonstrou a viabilidade e eficácia da aplicação de técnicas de processamento de linguagem natural (PLN) e métodos de aprendizagem de máquina para a classificação de sentimentos em um corpus textual com neologismos e estruturas particulares. Neste estudo o objetivo principal foi desenvolver um método sistemático que abrange desde a extração de dados de arquivos PDF até a análise e ajuste de modelos de classificação.

Os resultados indicam que, mesmo diante da complexidade e especificidade do conteúdo da *Conscienciologia*, as ferramentas tradicionais de PLN, quando devidamente ajustadas, são capazes de performar com eficiência. A metodologia empregada, que incluiu a extração textual com a biblioteca *PyPDF* e *expressões regulares*, um rigoroso pré-processamento para limpeza e normalização dos dados e a vetorização dos textos, mostrou-se fundamental para o sucesso da classificação.

A avaliação de múltiplos algoritmos de aprendizagem de máquina revelou que os modelos de *regressão logística* e C-Máquinas de Vetores de Suporte (SVC) alcançaram o maior desempenho, ambos com um *f1-score* de 0.79, seguidos de perto pelas *redes neurais* (MLPClassifier) com um *f1-score* de 0.78. Um achado relevante foi a superioridade consistente do *Porter Stemmer* como técnica de transformação textual e do *TFIDFVectorizer* para a criação da *bag of words* na obtenção desses resultados.

O processo de ajuste fino dos hiperparâmetros, utilizando *GridSearchCV* e *Pipeline*, foi crucial para otimizar a performance dos modelos, encontrando o equilíbrio entre o subajuste e o sobreajuste. A utilização da *validação cruzada estratificada* e do *f1-score* como métrica principal garantiu a robustez e a confiabilidade dos resultados, especialmente ao lidar com classes desbalanceadas.

Apesar dos resultados satisfatórios, é importante frisar algumas limitações da pesquisa:

- **Seleção da Amostra:** A amostra de 2.019 verbetes não foi aleatória, mas sim intencional. O critério foi selecionar todos os verbetes do propositor da enciclopédia, Waldo Vieira, por servirem de base para futuros escritos. Isso significa que os resultados podem não ser generalizáveis para o universo total de aproximadamente 6.500 verbetes da Enciclopédia da Conscienciologia.

- **Natureza dos Dados:** Uma particularidade e desafio do projeto é a própria natureza dos dados, provenientes da *neociência Conscienciologia*. O uso de neologismos (termos novos), conceitos e formas de escrita particulares afeta o processamento textual. Isso levou ao questionamento se os métodos tradicionais de PLN seriam eficazes ou se precisariam de adaptações.
- **Escopo da Metodologia:** A pesquisa se concentrou em um conjunto específico de modelos de aprendizado de máquina (naive bayes, regressão logística, máquina de vetores de suporte, florestas aleatórias e redes neurais). Embora tenha realizado uma comparação entre eles, a análise não se estendeu a outras arquiteturas de modelos, como modelos de aprendizado profundo mais complexos que se tornaram proeminentes.
- **Processo de Extração:** O método se inicia com a extração de dados textuais de arquivos PDF. A extração a partir deste formato, embora abordada com o uso da biblioteca *PyPDF* e *expressões regulares*, pode ser uma fonte de ruídos e erros (como quebras de linha indevidas ou problemas de formatação) que necessitam de uma limpeza rigorosa na etapa de pré-processamento.

Conclui-se que o método desenvolvido é replicável e pode ser adaptado para a análise de outros corpus de conhecimento específico, enfrentando os desafios de extração, organização e classificação de dados. A pesquisa contribui não apenas com a classificação dos verbetes da *Enciclopédia da Conscienciologia*, mas também oferece um roteiro metodológico para futuras investigações na área de análise de sentimentos e PLN aplicado a domínios especializados.

REFERÊNCIAS

1. ASSOCIAÇÃO INTERNACIONAL DE ENCICLOPEDIOLOGIA CONSCIENCIOLÓGICA. **Enciclopédia da Conscienciologia**. Foz do Iguaçu, PR: [ENCYCLOSSAPIENS], 2023. Disponível em: https://encyclossapiens.org/ec/. Acesso em: 30 jun. 2025.
2. AUGUSTO, Gabriel. **Ano de Aplicação de Técnicas Projetivas (N. 6.932; 26.01.2025).** In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Verbete defendido no Tertuliarium do Centro de Altos Estudos da Conscienciologia (CEAEC), Foz do Iguaçu, PR. Disponível em: https://encyclossapiens.space/buscaverbete. Acesso em: 30 jun. 2025, 15h43.
3. BAG-OF-WORDS model. In: **WIKIPEDIA: The Free Encyclopedia**. [S. l.]: Wikimedia Foundation, [2025]. Disponível em: https://en.wikipedia.org/wiki/Bag-of-words_model. Acesso em: 30 jun. 2025.
4. Campus CEAEC. **Conheça a Conscienciologia.** Disponível em: https://campusceaec.org/conheca-a-conscienciologia/. Acesso em: 30 jun. 2025.
5. FENNIAK, Mathieu et al. **The pypdf library.** 2024. Disponível em: https://pypi.org/project/pypdf/. Acesso em: 12 jun. 2025.
6. FREITAG, Dayne. **Machine Learning for Information Extraction in Informal Domains**. Norwell: Kluwer Academic Publishers, 2000. Disponível em: https://www.cs.cmu.edu/~dayne/thesis.pdf. Acesso em: 30 jun. 2025.
7. GOOGLE CLOUD. **What is ETL?**. Disponível em: https://cloud.google.com/learn/what-is-etl. Acesso em: 30 jun. 2025.
8. GUIMARÃES, Daniela. **Medo de errar.** In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Apres. Coordenação da Encyclossapiens; revisores Equipe de Revisores da Encyclossapiens. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica; Associação Internacional Editares, 2023. Verbete. Vol. digital único (PDF), p. 22.206–22.211. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h38.
9. HUTCHINS, John. **The history of machine translation in a nutshell.** [S.l.: s.n.], 2014.
10. HUTCHINS, John. **The first public demonstration of machine translation: the Georgetown-IBM system, 7th January 1954.** [S.l.: s.n.], 2006.

11. INSTITUTO COGNOPOLITANO DE GEOGRAFIA E ESTATÍSTICA (ICGE). **ICGE**. Disponível em: https://www.icge.org.br/. Acesso em: 30 jun. 2025.
12. KIMBALL, Ralph; ROSS, Margy. **The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling**. 2. ed. New York: John Wiley & Sons, 2002.
13. LIU, S. et al. Comparison of evaluation metrics of deep learning for imbalanced imaging data in osteoarthritis studies. **Osteoarthritis and Cartilage**, v. 31, n. 9, p. 1242-1248, set. 2023. Disponível em: https://doi.org/10.1016/j.joca.2023.05.006. Acesso em: 6 set. 2025.
14. LOPES, Adriana. **Infopesquisa na Bibliografia Conscienciológica.** Verbete. In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Foz do Iguaçu, PR, 26 dez. 2024. Defendido no Tertuliarium do Centro de Altos Estudos da Conscienciologia (CEAEC). Disponível em: https://encyclossapiens.space/buscaverbete. Acesso em: 30 jun. 2025, 15h45.
15. MANNING, Christopher D.; SCHÜTZE, Hinrich. **Foundations of Statistical Natural Language Processing**. Cambridge, MA: MIT Press, 1999.
16. MCNAMEE, Paul; MAYFIELD, James C.; PIATKO, Christine D. **Processing Named Entities in Text. Johns Hopkins APL Technical Digest,** v. 30, n. 1, 2011.
17. MOTA, Cristina; SANTOS, Diana (eds.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM.** [S.l.: s.n.], dez. 2009. ISBN 978-989-20-1656-6.
18. NADER, Rosa (Org.). **MANUAL DE VERBETOGRAFIA**: da Enciclopédia da Conscienciologia. Foz do Iguaçu: Editares, 2012.
19. NLTK PROJECT. **NLTK: Natural Language Toolkit**. 2024. Disponível em: https://www.nltk.org/. Acesso em: 30 jun. 2025.
20. OLIVEIRA, Felipe. **Cultura da amizade evolutiva.** Verbete (n. 6.950, 13 fev. 2025). In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Foz do Iguaçu, PR: Centro de Altos Estudos da Conscienciologia (CEAEC). Defendido no Tertuliarium do CEAEC. Disponível em: https://encyclossapiens.space/buscaverbete. Acesso em: 30 jun. 2025, 15h42.
21. OSMO, Flavio. **Compaixão discernida** (N. 3.811; 11.07.2016). Verbete. In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Apres. Coordenação da ENCYCLOSSAPIENS; revisores Equipe de Revisores da ENCYCLOSSAPIENS. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica (ENCYCLOSSAPIENS); Associação Internacional Editares, 2023.

Vol. Digital Único (PDF). CCXL + 34.372 p. p. 9.333-9.337. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h39.

22. PRUSTY, Sashikanta; PATNAIK, Srikanta; DASH, Sujit Kumar. SKCV: Stratified K-fold cross-validation on ML classifiers for predicting cervical cancer. **Frontiers in Nanotechnology**, v. 4, 2022. Disponível em: https://doi.org/10.3389/fnano.2022.972421. Acesso em: 6 set. 2025.
23. PYTHON SOFTWARE FOUNDATION. **The re module**. In: PYTHON SOFTWARE FOUNDATION. *Python 3.12.4 documentation*. [S. l.]: Python Software Foundation, [2024?]. Disponível em: https://docs.python.org/3/library/re.html. Acesso em: 30 jun. 2025.
24. ROQUE, Marlene. **Biblioteca de Alexandria**. In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Apres. Coordenação da ENCYCLOSSAPIENS; revisores Equipe de Revisores da ENCYCLOSSAPIENS. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica; Associação Internacional Editares, 2023. v. único (digital, PDF). p. 7.376–7.383. Verbete n. 3.820, de 20 jul. 2016. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h41.
25. SCIKIT-LEARN. **CountVectorizer**: documentation. Version 1.7.0. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html. Acesso em: 30 jun. 2025.
26. SCIKIT-LEARN. **F1-score**. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html. Acesso em: 30 jun. 2025.
27. SCIKIT-LEARN. **GridSearchCV**: scikit-learn 1.7.0 documentation. [2025?]. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html. Acesso em: 30 jun. 2025.
28. SCIKIT-LEARN. **MLPClassifier**: scikit-learn 1.7.0 documentation. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html. Acesso em: 30 jun. 2025.
29. SCIKIT-LEARN. **StandardScaler**: User Guide. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html. Acesso em: 30 jun. 2025.

30. SCIKIT-LEARN. **TfidfVectorizer**. In: scikit-learn: Machine Learning in Python. [S. l.]: scikit-learn developers, [2025]. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html. Acesso em: 30 jun. 2025.
31. SKLEARN.MANIFOLD.TSNE. In: **Scikit-learn**. [S. l.], [s.d.]. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html. Acesso em: 15 jul. 2025.
32. SLANEY, John; THIÉBAUX, Sylvie. **Blocks world revisited. Artificial Intelligence,** Amsterdam, v. 125, n. 1–2, p. 119–153, 2001. Disponível em: https://doi.org/10.1016/S0004-3702(00)00079-5. Acesso em: 12 maio 2025.
33. SOLC, Tomaz. **Unidecode**. Versão 1.4.0. Python Package Index (PyPI), 2025. Disponível em: https://pypi.org/project/Unidecode/. Acesso em: 30 jun. 2025.
34. TURING, A. M. **Computing machinery and intelligence.** Mind, Oxford, v. 59, n. 236, p. 433–460, out. 1950. Disponível em: https://doi.org/10.1093/mind/LIX.236.433. Acesso em: 6 maio 2025.
35. SPACY. **spaCy · Industrial-strength Natural Language Processing in Python**. Disponível em: https://spacy.io/. Acesso em: 30 jun. 2025.
36. TELES, Mabel. **Zéfiro: a paraidentidade intermissiva de Waldo Vieira**. 2. ed. Foz do Iguaçu: Editares, 2019.
37. TERTULIARIUM. **Página inicial**. [**S.l.**]: [**s.n.**], [**2025**]. Disponível em: https://www.tertuliarium.org. Acesso em: 30 jun. 2025.
38. VAN DER MAATEN, Laurens; HINTON, Geoffrey. **Visualizing Data using t-SNE.** Journal of Machine Learning Research, v. 9, p. 2579-2605, 2008.
39. VIEIRA, Waldo. **Alcoolismo (N. 97; 04.12.2005).** Verbete. In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Apres. Coordenação da ENCYCLOSSAPIENS; revisores Equipe de Revisores da ENCYCLOSSAPIENS. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica (ENCYCLOSSAPIENS); Associação Internacional Editares, 2023. Vol. digital único (PDF), CCXL + 34.372 p. p. 865–868. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h36.
40. VIEIRA, Waldo. **Ansiedade (N. 67; 30.10.2005)**. In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Apres. Coordenação da ENCYCLOSSAPIENS; revisores Equipe de Revisores da ENCYCLOSSAPIENS. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica

(ENCYCLOSSAPIENS); & Associação Internacional Editares, 2023. v. digital único (PDF), p. 1416–1418. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h35.

41. VIEIRA, Waldo. **Conscienciologia (N. 212; 19.04.2006).** Verbete. In: VIEIRA, Waldo (Org.). *Enciclopédia da Conscienciologia*. Apresentação: Coordenação da ENCYCLOSSAPIENS. Revisão: Equipe de Revisores da ENCYCLOSSAPIENS. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica (ENCYCLOSSAPIENS); Associação Internacional Editares, 2023. Vol. digital único (PDF). p. 9976–9980. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h33.
42. VIEIRA, Waldo. In: **Wikipedia**. [S. l.]: Wikimedia Foundation, [2025]. Disponível em: https://pt.wikipedia.org/wiki/Waldo_Vieira. Acesso em: 30 jun. 2025.
43. VIEIRA, Waldo. **Rotina útil.** In: VIEIRA, Waldo (org.). *Enciclopédia da Conscienciologia*. Apres. Coordenação da ENCYCLOSSAPIENS; revisores Equipe de Revisores da ENCYCLOSSAPIENS. 10. ed. rev. e aum. Foz do Iguaçu, PR: Associação Internacional de Enciclopediologia Conscienciológica (ENCYCLOSSAPIENS); & Associação Internacional Editares, 2023. Verbete. Vol. digital único (PDF). p. 29.655-29.657. Disponível em: https://encyclossapiens.space/ec/ECDigital10.pdf. Acesso em: 30 jun. 2025, 15h34.
44. WINOGRAD, Terry. **SHRDLU.** [S.l.: s.n.], [s.d.]. Disponível em: https://hci.stanford.edu/winograd/shrdlu/. Acesso em: 30 jun. 2025.

APÊNDICE

APÊNDICE A — Hiperparâmetros utilizados para ajuste dos modelos e vetorizadores

1. **Vetorizadores:**
    a. min_df: [0.2, 1].
    b. max_df: [0.8, 1.0].
    c. binary: [False, True].
    d. smooth_idf: [False, True].
    e. sublinear_tf: [False, True].
    f. norm: ['l1', 'l2', None].
2. **Modelos:**
    a. alpha: [0.1, 0.5, 1.0, 2.0, 3.0].
    b. C: [0.5, 1.0, 1.5].
    c. class_weight: ['balanced', None].
    d. solver: ['lbfgs', 'newton-cg'].
    e. kernel: ['linear', 'poly', 'rbf', 'sigmoid'].
    f. max_depth: [None, 20].
    g. criterion: ['gini', 'entropy'].
    h. min_samples_leaf: [2, 4].
    i. max_features: [None, 'sqrt', 'log2'].
    j. hidden_layer_sizes: [(50,), (100,)], [(50, 25), (100, 50)], [(50, 25, 10), (100, 50, 25)].
    k. alpha: [0.0001, 0.001, 0.01].
    l. learning_rate: ['constant', 'adaptive'].
    m. activation: ['identity', 'logistic', 'tanh', 'relu'].