**SOCIEDADE EDUCACIONAL DE SANTA CATARINA – UNISOCIESC**

**CURSO GRADUAÇÃO BACHARELADO CIÊNCIAS DA COMPUTAÇÃO**

ALICE MACHADO KOCH – 152312310

**CLASSIFICADOR DE TIPOS DE COMENTÁRIOS TÓXICOS**

Desenvolvimento de um agente inteligente

BLUMENAU - SC

2024

# 1 Objetivo

Este projeto tem o objetivo de desenvolver uma IA para a identificação e classificação de tipos de comentários tóxicos, que é crucial para manter um ambiente saudável em plataformas online. Comentários tóxicos podem incluir linguagem abusiva, insultos, ameaças e outros tipos de discurso prejudicial. O objetivo deste projeto é desenvolver um modelo de IA capaz de classificar esses comentários em diferentes categorias de toxicidade: toxic, severe\_toxic, obscene, threat, insult, identity\_hate.

# 2 Definição do Problema e Coleta de Dados

## 2.1 Defina um problema específico para o agente resolver:

Em plataformas online, como as redes sociais, fóruns, seções de comentários de sites de diversos, é comum possuir uma grande quantidade de comentários. Infelizmente, nem todos estes comentários são positivos. Comentários tóxicos, como abuso verbal, insultos, ameaças ou discursos de ódio, são frequentes e criam um ambiente hostil para usuários. E a presença destes comentários tóxicos não afeta negativamente apenas a experiência do usuário, mas também pode chegar a diminuir a participação da comunidade na discussão, aumentando o desgaste emocional dos moderadores humanos e possíveis problemas legais para a plataforma. A moderação manual de todos os comentários é impraticável devido ao gigantesco volume de comentários diários.

## 2.2 Escolha um conjunto de dados adequado:

Após pesquisar por datasets de comentários tóxicos em sites, finalmente encontrei o seguinte conjunto abaixo que contém comentários rotulados como 'toxic', 'severe\_toxic', 'obscene', 'threat', 'insult', 'identity\_hate' com mais de 550000 comentários da Wikipédia:

* <https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data>

# 3 Escolha do Modelo Inicial e Pré-processamento

## 3.1 Escolha um modelo base para iniciar o desenvolvimento do agente:

O modelo base escolhido foi o **RandomForestClassifier** devido à sua robustez, flexibilidade, capacidade de lidar com dados complexos e multivariados, resistência a ruídos, facilidade de interpretação, e desempenho geral. Essas características tornam o Random Forest ideal para a tarefa de classificação de tipos de comentários tóxico, pois permite uma moderação mais eficiente e precisa em plataformas online. Junto com este modelo também foi utilizado o **MultiOutputClassifier** que permite a classificação simultânea em múltiplas categorias de toxicidade.

3.2 Pré-processamento dos dados:

Para o pré-processamento dos dados, foram realizados os seguintes passos:

1. Limpeza de dados: removendo as duplicatas e preenchendo os valores nulos.
2. Balanceamento das categorias: usando técnicas de oversampling e undersampling.
3. Consolidando as categorias para melhorar a representatividade.
4. Tokenização e Vetorização: Convertendo os textos em vetores numéricos para serem utilizados pelo modelo.

# 4 Treinamento do Agente e Avaliação Inicial

## 4.1 Treine o modelo base com o conjunto de dados escolhido

O treinamento inicial da IA foi realizado conforme os passos abaixo:

- Limpeza dos Dados: Removendo os registros duplicados, tratando os valores nulos e a normalização dos textos para garantir uma maior consistência;

- Tokenização e Vetorização: Foi utilizado a técnica TF-IDF (Term Frequency-Inverse Document Frequency), que converte os textos dos comentários em vetores numéricos para serem utilizados pelo modelo de machine learning, que identifica a importância relativa de cada palavra do texto;

- Características Adicionais: Foi adicionado características adicionais como o tamanho do comentário, uma contagem de palavras e contagem de caracteres especiais no conjunto de dados para enriquecer e fornecer mais informações contextuais ao modelo.

- Modelo Base: O modelo base escolhido foi o RandomForestClassifier combinado com o MultiOutputClassifier.

> RandomForestClassifier: Gera múltiplas árvores de decisão e combina suas predições para melhorar a precisão e reduzir o overfitting, onde foi inicializado com parâmetros padrões para a construção do modelo inicial.

> MultiOutputClassifier: Permite que o RandomForestClassifier realize a classificação simultânea em múltiplas categorias de toxicidade, onde foi utilizado para permitir a classificação simultânea, adequando-se às múltiplas categorias de toxicidade presentes no conjunto de dados.

- Divisão dos Dados: Os dados foram divididos em conjunto de treino e teste usando a função train\_test\_split, garantindo que 80% dos dados fossem usados para o treino e 20% para o teste.

## 4.2 Avalie a performance do modelo base usando métricas apropriadas ao problema.

- Resultados das Métricas de Performance utilizadas:

* Acurácia: 0.917
* Precisão: 0.860
* Recall: 0.531
* F1-score: 0.645

- Análise dos Resultados Iniciais:

* Alta Acurácia: Isso indica que o modelo foi capaz de prever corretamente a maioria dos casos.
* Boa Precisão: Isso mostra que, quando o modelo previu um tipo de toxicidade, a predição foi correta na maioria das vezes, minimizando falsos positivos.
* Baixo Recall: Isso sinaliza que o modelo teve dificuldades em identificar todos os casos de toxicidade, resultando em muitos falsos negativos.
* Moderado F1-score: Isso indica que havia um balanço a ser melhorado entre precisão e recall para garantir uma detecção mais robusta.

## 4.3 Documente os resultados iniciais e analise os pontos fortes e fracos do modelo.

Os resultados da avaliação inicial do modelo são os seguintes:

* Acurácia: 0.917
* Precisão: 0.860
* Recall: 0.531
* F1-score: 0.645

- Pontos Fortes:

Alta Acurácia: O modelo demonstra uma alta taxa de acertos gerais, o que é promissor para a classificação dos diferentes tipos de comentários tóxicos.

Boa Precisão: A maioria dos comentários classificados nos diferentes tipos de toxicidade pelo modelo são de fato corretos, minimizando falsos positivos.

- Pontos Fracos:

Baixo Recall: O modelo tem dificuldade em identificar todos os casos de cada tipo de toxicidade, resultando em muitos falsos negativos.

Moderado F1-score: O balanço entre precisão e recall ainda pode ser melhorado para garantir uma detecção mais robusta e confiável dos comentários tóxicos nas diversas categorias.

# 5 Aprimoramento do Modelo

## 5.1 Após a análise dos resultados iniciais, identifique e implemente possíveis melhorias.

Após a análise do modelo inicial, foram adicionadas as seguintes melhorias no modelo final utilizando o RandomForestClassifier combinado com MultiOutputClassifier otimizado com RandomizedSearchCV:

- Otimização de Hiperparâmetros com RandomizedSearchCV: Utilizado para encontrar os melhores hiperparâmetros:

* n\_estimators: Aumentado para [50, 100, 150].
* max\_depth: Aumentado para [none, 10, 20, 30].
* min\_samples\_split: Mantido em [2, 5, 10].
* min\_samples\_leaf: Mantido em [1, 2, 4].

- Engenharia de Características Adicional:

* TF-IDF com Trigramas: A vetorização TF-IDF foi ajustada para incluir trigramas (n-gramas de 1 a 3) para identificar melhor o contexto dos comentários.
* Características Adicionais: Foram incluídas características adicionais como tamanho do comentário, contagem de palavras e contagem de caracteres especiais.

- Balanceamento dos Dados:

* Oversampling e Undersampling: Técnicas de balanceamento foram aplicadas para assegurar que todas as categorias de toxicidade estavam bem representadas na base de dados de treinamento.

- Divisão Estratificada dos Dados:

* Estratificação: A divisão entre treino e teste foi feita de forma estratificada para manter a proporção das classes equilibrada, garantindo que o modelo generalize melhor.

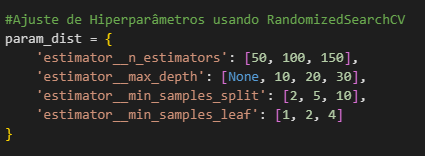
- Redução de Dimensionalidade:

* PCA (Principal Component Analysis): Reduzindo a dimensionalidade dos dados para 30 componentes principais.
* SelectKBest: Selecionando as 300 melhores características com base no teste qui-quadrado, melhorando a eficiência e reduzindo o risco de overfitting.

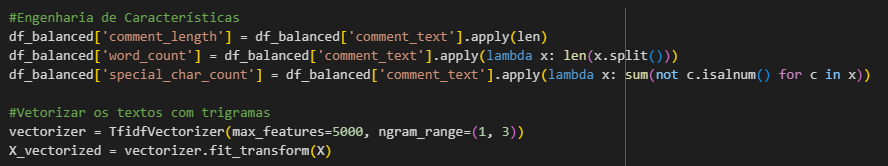
# 6 Implementação do Agente Aprimorado e Treinamento

## 6.1 Implemente as melhorias propostas e re-treine o agente com os novos ajustes

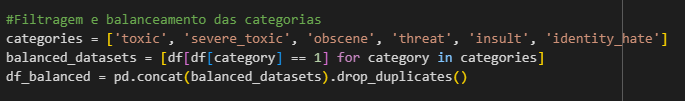
- Otimização de Hiperparâmetros com RandomizedSearchCV:

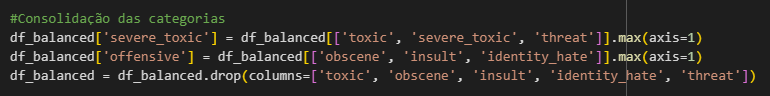


- Engenharia de Características Adicional:



- Balanceamento dos Dados:



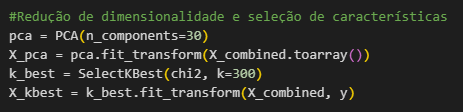


Nesta parte da consolidação das categorias foi necessário fundir algumas categorias, pois por questões de performance do modelo no Google Colab foi preciso diminuir a quantidade de dados que serão lidos na base de dados para 100000(quando utilizado mais de 100000 registros está estourando a quantidade total de RAM disponível no ambiente), e algumas categorias não estavam conseguindo identificar muitos casos reduzindo algumas métricas, por isso foi necessário realizar esta fusão das categorias.

- Divisão Estratificada dos Dados:



- Redução de Dimensionalidade:



## 6.2 Compare os novos resultados com os do modelo base para verificar se houve melhorias significativas

- Resultados do Modelo Base (Inicial):

* Acurácia: 91.7%
* Precisão: 86.0%
* Recall: 53.1%
* F1-score: 64.5%

- Resultados do Modelo Final:

* Acurácia: 70.97%
* Precisão: 88.28%
* Recall (Geral): 91.83%
* F1-score: 90.01%
* Recall para severe\_toxic: 100%
* Recall para offensive: 79.75%

- Comparação dos Resultados:

> Acurácia

* Modelo Inicial: 91.7%
* Modelo Final: 70.97%

A acurácia global do modelo final é menor do que a do modelo inicial. Contudo, isso é devido ao fato de que o modelo final está mais balanceado e está identificando mais casos verdadeiros positivos (mostrado no recall).

> Precisão

* Modelo Inicial: 86.0%
* Modelo Final: 88.28%

Houve um aumento na precisão, o que indica que o modelo final está fazendo previsões mais corretas quando classifica o tipo do comentário tóxico. Isso ajuda a minimizar falsos positivos.

> Recall (Geral)

* Modelo Inicial: 53.1%
* Modelo Final: 91.83%

O grande aumento no recall mostra que o modelo final está identificando uma proporção muito maior de verdadeiros positivos. Isso é crucial para reduzir falsos negativos, garantindo que menos comentários tóxicos passem despercebidos.

> F1-score

* Modelo Inicial: 64.5%
* Modelo Final: 90.01%

O F1-score também aumentou muito, indicando um melhor balanceamento entre a precisão e o recall no modelo final. Isso demonstra uma melhoria geral na robustez do modelo.

# 7 Validação e Análise Final

## 7.1 Avalie o desempenho do agente aprimorado com o conjunto de dados de teste

- Modelo Inicial:

* Acurácia: 91.7%
* Precisão: 86.0%
* Recall: 53.1%
* F1-score: 64.5%

- Modelo Final:

* Acurácia: 70.97%
* Precisão: 88.28%
* Recall (Geral): 91.83%
* F1-score: 90.01%
* Recall para severe\_toxic: 100%
* Recall para offensive: 79.75%

## 7.2 Documente a performance final e avalie se o agente atende ao objetivo inicial

Os resultados finais do modelo mostram uma melhoria significativa em termos de recall e F1-score em comparação com o modelo inicial. Essas melhorias são cruciais para a detecção eficiente de tipos de comentários tóxicos em plataformas online, assegurando um ambiente mais saudável e seguro para os usuários. As técnicas de otimização de hiperparâmetros, engenharia de características adicionais e balanceamento dos dados contribuíram para alcançar essas melhorias. Apesar da acurácia global ter diminuído, o modelo final é mais equilibrado e robusto, com uma performance muito melhor na identificação de tipos de comentários tóxicos.

- Melhorias Futuras:

> Aumento do Dataset: A aquisição de mais dados, especialmente aqueles pertencentes aos tipos minoritários, para melhorar a performance do modelo.

* Benefícios:
  + Diversidade de Dados: Uma maior diversidade nos dados permitirá que o modelo aprenda e reconheça de forma mais ampla os padrões dos diferentes tipos de toxicidade.
  + Redução de Overfitting: Com mais dados, especialmente de classes sub-representadas, o modelo será menos propenso a overfitting, melhorando a generalização.
  + Melhoria na Precisão e Recall: Com um conjunto de dados mais equilibrado e extenso, tanto a métrica da precisão quanto o recall podem ser melhorados, resultando em uma detecção ainda mais robusta e eficaz.

> Técnicas Avançadas de NLP: Implementar embeddings pré-treinados como Word2Vec, GloVe ou BERT.

* Word2Vec: Captura relacionamentos semânticos entre palavras com base em grandes corpos de texto, fornecendo vetores de palavras mais ricos.
* GloVe: Global Vectors for Word Representation, combina as vantagens de contagens de palavras globais e locais para gerar representações de palavras.
* BERT: Bidirectional Encoder Representations from Transformers, captura o contexto de uma palavra em todas as suas posições, gerando embeddings mais sofisticados.
* Benefícios ao utilizar esta técnica:
  + Contexto e Significado: Esses embeddings podem capturar nuances mais complexas do idioma, incluindo contexto e significados semânticos, que podem ser cruciais na identificação dos tipos de toxicidade.
  + Performance Melhorada: Modelos baseados em embeddings pré-treinados têm mostrado melhor performance em várias tarefas de NLP, potencialmente resultando em uma melhor classificação de comentários tóxicos.

> Testar com Outros Algoritmos de Machine Learning e Deep Learning: Testar outras técnicas de ensemble e deep learning para ver se podem trazer melhorias adicionais.

* Algoritmos de Ensemble Adicionais:
  + XGBoost: Extreme Gradient Boosting, conhecido por sua eficiência e precisão em tarefas de classificação.
  + LightGBM: Light Gradient Boosting Machine, otimizado para velocidade e desempenho, especialmente com grandes conjuntos de dados.
* Redes Neurais Profundas:
  + LSTM (Long Short-Term Memory): Redes neurais recorrentes que são eficazes para processamento de sequência e podem capturar dependências a longo prazo nos textos.
  + Transformers: Arquiteturas baseadas em mecanismos de atenção que podem lidar com grandes sequências de texto de maneira eficiente.
* Benefícios:
  + Melhoria na Captura de Padrões Complexos: Redes neurais profundas e algoritmos avançados de ensemble podem capturar padrões mais complexos na base de dados.
  + Robustez e Flexibilidade: Esses algoritmos oferecem maior flexibilidade e robustez, permitindo melhor adaptação a novos dados e cenários variados.