

# Classificação automatizada de denúncias no MPBA usando *machine learning*

**Flávio Pimentel Batista**  
Diretoria de Tecnologia da  
Informação  
Ministério Público do Estado  
da Bahia  
Salvador, Brasil  
[flavio.batista@mpba.mp.br](mailto:flavio.batista@mpba.mp.br)

**Milton Cerqueira da Silva  
Sobrinho**  
Diretoria de Tecnologia da  
Informação  
Ministério Público do Estado  
da Bahia Salvador, Brasil  
[milton.sobrinho@mpba.mp.br](mailto:milton.sobrinho@mpba.mp.br)

**Yuri Gonzalez Araujo**  
Diretoria de Tecnologia da  
Informação  
Ministério Público do Estado  
da Bahia  
Salvador, Brasil  
[yuri.araujo@mpba.mp.br](mailto:yuri.araujo@mpba.mp.br)

**Braian Varjão**  
SENAI/CIMATEC  
[braiangama@gmail.com](mailto:braiangama@gmail.com)

**Resumo** — Este artigo propõe a classificação automatizada da área de atuação das denúncias realizadas pelos cidadãos ao Ministério Público do Estado da Bahia (MPBA) por meio do site “Atendimento ao Cidadão”, utilizando algoritmos de aprendizagem de máquina (*machine learning*). Foi aplicado o processo de KDD nas bases textuais de denúncias visando a redução da taxa de erro da classificação efetuada pelo cidadão, atualmente estimada em 29,43%. Com isso, pretende-se melhorar a eficiência no atendimento à população, bem como, proporcionar economia de recursos institucionais. Foram utilizados nove algoritmos de classificação treinados com a base de dados de denúncias e os resultados foram tabelados considerando *FI Score* e acurácia como principais métricas de avaliação. Com base nos resultados, houve redução na taxa de erro para 25,00%, destacando a relevância deste estudo, além de apontar algumas sugestões de melhorias no site e de procedimentos administrativos internos do MPBA.

**Palavras-chave** — *Inteligência Artificial, Machine Learning, Deep Learning, KDD, NLP, Mineração de texto.*

## I. INTRODUÇÃO

A Constituição Federal de 1988 [1] estabelece, em seu Artigo 5º, o direito de petição dos cidadãos aos Poderes Públicos em defesa de direitos ou contra ilegalidades, assegurando que esses possam denunciar violações e exigir respostas do Estado. Nesse contexto, os canais de comunicação e atendimento ao público, sejam presenciais ou eletrônicos, tais como os site de denúncias, são essenciais para assegurar que os cidadãos possam exercer sua cidadania de maneira plena, relatando irregularidades, crimes ou qualquer situação que exija intervenção pública. Quando o poder público, através das Ouvidorias e Ministérios Públicos, realiza o atendimento célere às denúncias cadastradas pelos cidadãos, o Estado demonstra seu compromisso em proteger os direitos individuais e coletivos, promovendo a plena cidadania.

O Ministério Público do Estado da Bahia (MPBA) [2] é instituição permanente que desempenha papel crucial na defesa da ordem jurídica, do regime democrático e dos direitos sociais e individuais indisponíveis. Em setembro de 2021, a instituição lançou o site “Atendimento ao Cidadão” [3], uma plataforma digital projetada para melhorar a acessibilidade e agilidade dos serviços ofertados à população. O site permite que os cidadãos realizem denúncias, consultem

o andamento de processos e comuniquem-se diretamente com as áreas de atuação do MPBA, reforçando o compromisso do órgão com a transparência e eficiência no atendimento à população.

Ao realizar uma denúncia, o cidadão deve cadastrar algumas informações tais como “O que aconteceu?” e “Para onde deseja enviar essa Denúncia?” sendo que esta última representa uma das dez áreas de atuação do MPBA, a saber: cível, consumidor, criança e adolescente, criminal, direitos humanos, educação, meio ambiente, moralidade administrativa, saúde e segurança pública. Este campo também determina qual a unidade do MPBA, denominada, Centro de Apoio Operacional (CAO), será a destinatária da denúncia e realizará os devidos acompanhamentos. Atualmente existem dez CAOs na estrutura administrativa do MPBA: CAOCIFE (cível), CEACON (consumidor), CAOCA (criança e adolescente), CAOCRIM (criminal), CAODH (direitos humanos), CEDUC (educação), CEAMA (meio ambiente), CAOPAM (moralidade administrativa), CESAU (saúde) e (CEOSP) segurança pública.

Após recepcionada pelo Centro de Apoio Operacional correspondente à área de atuação informada pelo cidadão, a denúncia fica em uma fila para atendimento cuja primeira atividade é analisar detalhadamente as informações constantes na mesma. Quando o cidadão classifica corretamente a área de atuação de sua denúncia, o CAO dá prosseguimento aos trâmites administrativos e processuais internos.

Porém, quando a denúncia analisada não é atribuição do CAO em questão ou do MPBA, é redirecionada para o Centro de Apoio ou órgão externo competente. Estima-se que 29,43% das denúncias sejam cadastradas equivocadamente pelos cidadãos.

A classificação incorreta da área de atuação de uma denúncia feita pelo cidadão gera impactos negativos tanto para ele quanto para o MPBA. Quando a denúncia é competência do MPBA, mas é encaminhada ao CAO errado, ela precisa ser redirecionada para a área correta, voltando ao início da fila de atendimento. Isso gera retrabalho para os servidores e aumenta o tempo de tramitação. Como consequência, o tempo de resposta ao cidadão também se prolonga. O mesmo ocorre com denúncias que não são de responsabilidade do MPBA, exigindo correções que atrasam ainda mais o processo. Portanto, é fundamental reduzir ou até

mesmo eliminar os erros na classificação manual das denúncias feitas pelos cidadãos.

O objetivo deste trabalho é aplicar técnicas de *Natural Language Processing* (NLP) juntamente com algoritmos de classificação automatizada baseados em *machine learning* e *deep learning* para otimizar o processo de classificação das denúncias sugerindo a área de atuação correspondente ao teor da denúncia.

Foram comparados os resultados entre os algoritmos para verificar se o melhor desempenho reduz a taxa de erro na classificação das denúncias, em comparação com a classificação manual feita pelos cidadãos.

O presente artigo está dividido como segue. A Seção II aborda outros estudos similares na literatura. Na Seção III é apresentada a fundamentação teórica com as definições dos principais conceitos concernentes ao trabalho. Na Seção IV é demonstrado o método do projeto desenvolvido, detalhando os procedimentos e técnicas implementadas. Em seguida é apresentada a Seção V, na qual são discutidos os resultados obtidos por meio da execução dos algoritmos de *machine learning* na base de denúncias selecionada. Por fim, considerações finais, dificuldades e sugestões de trabalhos futuros são apresentados na Seção VI.

## II. REVISÃO BIBLIOGRÁFICA

A classificação automática de textos utilizando técnicas de processamento de linguagem natural (NLP) e aprendizado de máquina tem sido amplamente explorada em diversos contextos, incluindo a classificação de queixas e denúncias. Estudos como o de Q. Ain et al. [4] e V. Vinayak e J. Chandrasekharan [5] têm contribuído significativamente para o avanço nesse campo, proporcionando uma base sólida para a comparação e o desenvolvimento do presente trabalho.

O trabalho de Q. Ain et al. [4] propõe um modelo de classificação de queixas usando NLP e algoritmos de *machine learning* em um contexto de atendimento ao cidadão. Eles exploram algoritmos tradicionais, como *Naive Bayes* e *Support Vector Machine* (SVM), para categorizar as queixas recebidas por um portal de denúncias. O estudo também discute o impacto do desbalanceamento das classes e a necessidade de técnicas de pré-processamento robustas para melhorar a acurácia do modelo. Semelhante ao presente trabalho, o estudo de Q. Ain et al. [4] enfrenta o desafio da classificação multiclasse e emprega técnicas de NLP como tokenização e lematização para preparar os dados textuais.

Por outro lado, o trabalho de V. Vinayak e J. Chandrasekharan [5] foca no uso de *deep learning* para a classificação de queixas, aplicando modelos como *LSTM* e *CNN* em conjunto com *embeddings* como *Word2Vec* e *BERT*. O uso de técnicas avançadas de *deep learning* diferencia esse estudo daqueles que utilizam apenas algoritmos de *machine learning* tradicionais. V. Vinayak e J. Chandrasekharan [5] demonstram que os modelos baseados em *embeddings* são mais eficazes na captura de contextos complexos, algo que também foi explorado no presente trabalho, que utilizou o modelo de rede neural baseado em *transformer* para melhorar o desempenho da classificação.

Ao comparar esses estudos com o trabalho aqui apresentado, algumas semelhanças e diferenças se destacam. Tanto [4] quanto [5] focam na aplicação de algoritmos de classificação de texto em queixas e denúncias, utilizando técnicas de NLP para pré-processamento e vetorização dos dados. No entanto, o presente trabalho inova ao aplicar esses conceitos em um contexto institucional específico do Ministério Público do Estado da Bahia (MPBA), envolvendo 11 classes de diferentes áreas de atuação, o que adiciona uma camada de complexidade ao problema de classificação. Enquanto o estudo de [4] trata majoritariamente de queixas em um contexto mais geral, e o de [5] foca em consumidores, o presente trabalho se concentra na classificação de denúncias jurídicas, no qual os contextos similares entre as classes (como CEOSP e CAOCRIM) é um desafio adicional.

Além disso, este artigo vai além dos estudos anteriores ao comparar diretamente o desempenho de algoritmos de *machine learning* e *deep learning* no contexto de denúncias, enquanto [4] focam apenas em modelos tradicionais e [5] abordam somente técnicas de *deep learning*. Outro diferencial do presente trabalho está no uso combinado de técnicas de *Bag of Words* (TF-IDF) e *embeddings BERT* e propor melhorias no atendimento ao cidadão por meio da classificação automática das denúncias.

## III. FUNDAMENTAÇÃO TEÓRICA

### A. Aprendizagem de Máquina (Machine Learning)

O termo “Aprendizagem de Máquina” (*Machine Learning*) refere-se a uma disciplina da IA que estuda o desenvolvimento de algoritmos e modelos estatísticos que permitem aos computadores obterem conclusões e tomarem decisões baseados na análise autônoma de dados, ou seja, sem seguir regras previamente programadas. Esses algoritmos, através de treinamentos e iterações sobre os conjuntos de dados, “aprendem” a identificar padrões, perfis, correlações e tendências, podendo, inclusive, realizar previsões e diagnósticos, utilizando probabilidade, inferência, entre outras ferramentas da Estatística [6].

Segundo [6] existem vários tipos de aprendizado de máquina. Dentre estes, o aprendizado supervisionado (*supervised learning*) preconiza que os algoritmos sejam treinados com um conjunto de dados rotulados, ou seja, os dados de entrada são acompanhados por saídas desejadas (rótulos). Por sua vez, no aprendizado profundo (*deep learning*) são utilizadas redes neurais artificiais com múltiplas camadas (redes neurais profundas) para modelar dados complexos, particularmente eficazes em tarefas como reconhecimento de imagem, processamento de linguagem natural e jogos complexos.

### B. Redes Neurais

Redes Neurais são modelos estatísticos inspirados no funcionamento dos neurônios biológicos, compostos de múltiplas camadas de unidades de processamento interconectadas. Cada unidade (“neurônio”) recebe uma entrada, processa-a aplicando uma função de ativação, e transmite a saída para as próximas unidades na camada

subsequente. A relação entre as entradas e a saída da rede é regida através de pesos atribuídos a cada ligação entre os neurônios. O aprendizado da rede em si consiste em encontrar, baseado no trato com o conjunto de dados, os valores mais adequados para estes pesos, a fim de que se estabeleça essa relação, com menor erro possível [7].

Sua arquitetura consiste em uma camada de neurônios de entrada (que receberão as variáveis independentes do conjunto de dados), seguida (ou não) por várias camadas ocultas intermediárias e, por fim, a camada de saída (normalmente ligadas à variável dependente).

Redes neurais normalmente são empregadas na solução de demandas mais complexas, envolvendo conjunto de dados maiores. Algumas aplicações mais comuns são:

- Reconhecimento de imagens;
- Processamento de linguagem natural;
- Reconhecimento de fala e áudio;
- Visão computacional;
- Automação; e
- Simulações complexas.

De acordo com [8], algumas das arquiteturas de rede neural mais comuns são:

- **Redes Neurais Recorrentes (RNNs):** Têm conexões cíclicas e são adequadas para dados sequenciais, como séries temporais e texto;
- **Redes de Memória de Longo Curto Prazo (LSTMs):** Tipo especial de RNN projetadas para aprender dependências de longo prazo em sequências de dados. Adequadas para NLP;
- **Transformers:** uma nova arquitetura de rede neural baseada exclusivamente em mecanismos de atenção, sem recorrência ou convoluções;

As arquiteturas de redes neurais são especificamente configuradas de acordo com o tipo de problema a ser resolvido. Por exemplo, a camada de *embeddings* de um modelo *transformer* como o BERT pode ser utilizada como entrada para um modelo de classificação que emprega uma rede LSTM.

### C. Knowledge Discovery in Databases (KDD)

A descoberta de conhecimento em banco de dados (*Knowledge Discovery in Databases - KDD*) é um processo iterativo e interativo, cujo objetivo é apoiar, de forma automatizada, o ser humano na extração de conhecimentos úteis em grandes volumes de dados digitais [9].

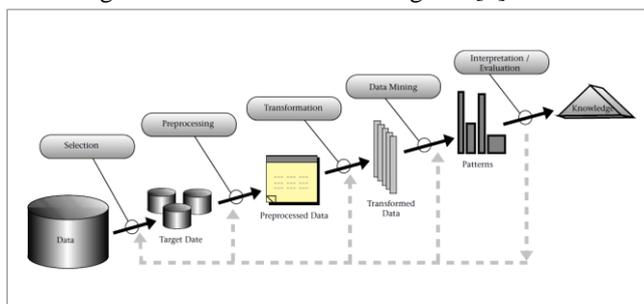


Fig. 1. Etapas do KDD de acordo com [9]

A Figura 1 ilustra as principais etapas do processo de KDD: seleção, pré-processamento, transformação, mineração dos dados e interpretação/avaliação dos resultados.

A etapa de seleção de dados é responsável pela identificação e coleta dos dados relevantes a serem analisados, integrando todo ou um subconjunto de variáveis ou amostras de dados, formando um conjunto de dados (*dataset*). A relevância e descobertas dos dados ocorre através de um processo iterativo, em cada ciclo mais se conhece sobre os dados e a importância deles, assim nessa etapa ocorre o desenvolvimento no entendimento do domínio da aplicação, bem como a definição das metas do KDD sob ponto de vista de negócio. Esta etapa é crucial para o bom desempenho das etapas posteriores, principalmente para o desempenho dos modelos na mineração. Algum atributo faltante pode ter um grande impacto no resultado. Assim como, é custoso e complexo manter um grande conjunto de dados, prezando por um ponto de equilíbrio na busca e manutenção dos dados a serem utilizados pelo processo.

O pré-processamento engloba a limpeza e preparação dos dados, o que inclui o preenchimento ou remoção de dados vazios, ruídos e o tratamento de inconsistências. O foco está na melhoria da confiabilidade dos dados. Dentre as técnicas podem ser empregadas métodos estatísticos complexos para valores discrepantes, ou mesmo, algoritmos de mineração específicos para preenchimentos de informações.

A transformação de dados tem por objetivo transformar ou consolidar os dados em formatos apropriados para mineração, incluindo a redução e projeção de características. O objetivo é encontrar características ou atributos úteis de acordo com a meta definida inicialmente. Assim, aplicam-se técnicas como redução de dimensionalidade (seleção e extração de atributos) e transformação de atributos (discretização ou funções de transformação).

Já a etapa de mineração dos dados consiste em realizar análise exploratória para selecionar modelos e hipóteses, o que envolve escolher algoritmos de mineração de dados e métodos para buscar padrões pertinentes aos objetivos definidos inicialmente. Isso pode incluir métodos como classificação, regressão, *clustering* e associação. Bem como decidir quais parâmetros e ajustes finos são apropriados ao problema. A correta execução dos passos anteriores ajuda significativamente o desempenho da etapa de mineração.

Por fim, a etapa de interpretação e avaliação dos resultados analisa os padrões descobertos para garantir que sejam válidos, novos e úteis, definindo a compreensibilidade e a utilidade do modelo construído. Essa etapa pode incluir a visualização dos padrões extraídos e dos dados associados. Além disso, envolve a utilização e documentação do conhecimento descoberto para apoiar a tomada de decisões ou integrá-lo em sistemas de informação. O sucesso dessa etapa é crucial para determinar a eficácia de todo o processo de KDD.

### D. Natural Language Processing (NLP)

Processamento de Linguagem Natural (*Natural Language Processing*) é o campo da Inteligência Artificial que estuda algoritmos capazes de analisar, reconhecer, compreender e gerar textos em linguagens humanas. Para isso, vale-se tanto

de ferramentas e métodos estatísticos quanto de abordagens baseadas em regras [10].

Como uma forma de revolucionar a interação entre os humanos e os computadores através da comunicação, o NLP tem aplicação em diversas áreas do cotidiano, sendo algumas delas:

- Análise de sentimentos em textos;
- Assistentes Virtuais;
- Tradução em tempo real de línguas;
- Classificação de textos;
- Reconhecimento de entidades nomeadas; e
- Análise de documentos técnico/científicos.

No contexto de processamento de textos, algumas técnicas aplicadas são:

- **Tokenização:** dividir textos em unidades menores, como palavras ou frases. A tokenização é o primeiro passo para estruturar o texto bruto, permitindo que algoritmos de *machine learning* trabalhem com os dados de forma mais eficiente;
- **Remoção de Stop Words:** *stop words* são palavras comuns que geralmente não contribuem para o significado do texto, como “e”, “o”, “em”, etc. Removê-las pode ajudar a reduzir o ruído e melhorar a performance dos modelos de *machine learning*;
- **Stemming e lematização:** Ambos os processos buscam reduzir as palavras à sua forma base. O *stemming* remove e substitui sufixos para chegar à raiz da palavra, enquanto a lematização leva em conta o contexto linguístico para converter a palavra em sua forma canônica ou lema. Essas técnicas ajudam a consolidar variações da mesma palavra, facilitando a classificação e a análise;
- **Vetorização:** é representação algébrica de elementos textuais comumente organizados em vetores ou matrizes. *Bag of Words (BoW)* é uma técnica que transforma texto em um conjunto de frequências de palavras, desconsiderando a ordem das palavras, mas mantendo a informação sobre a frequência de cada termo. Já o *TF-IDF (Term Frequency-Inverse Document Frequency)* refina o modelo BoW ao considerar não só a frequência de uma palavra em um documento, mas também a importância da palavra em toda a coleção de documentos, ajudando a destacar palavras importantes que são menos comuns nos documentos. O *Word Embeddings* são representações vetoriais de palavras que capturam contextos e semânticas complexas. Modelos como *Word2Vec*, *GloVe* ou *FastText* aprendem embeddings que refletem relações semânticas e sintáticas entre as palavras, permitindo uma compreensão mais profunda e eficaz do texto;
- **Seleção de atributos (*feature selection*):** processo de identificar e escolher os atributos (*tokens*) mais relevantes ou significativos de um conjunto de dados para utilização em modelos de *machine learning*;

## IV. MÉTODO

Os seguintes passos do processo de KDD foram aplicados no projeto: seleção, pré-processamento, transformação, mineração dos dados e interpretação e avaliação dos resultados. A Figura 2 ilustra o fluxo das etapas aplicadas à metodologia empregada.

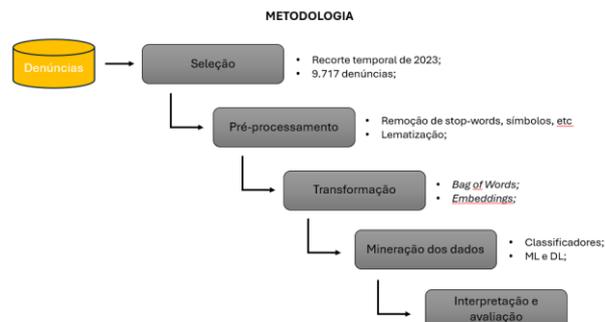


Fig. 2. Fluxo das etapas

Importante destacar que, considerando a sensibilidade dos dados das denúncias, todas as etapas foram executadas nos equipamentos locais dos autores sem utilização de serviços em nuvem. A linguagem de programação utilizada foi *Python* (v3.9.19) em conjunto com as principais bibliotecas para *Machine Learning: Pandas, Sklearn, Spacy, Nltk, TensorFlow, Torch, PyTorch, Transformers (Hugging Face)*, dentre outras para geração de imagem, funções básicas de carga de dados, cálculo e configuração para uso de *GPU*.

### A. Seleção dos dados

As denúncias utilizadas no presente trabalho foram extraídas a partir do banco de dados do site “Atendimento ao Cidadão”, após concessão da autorização dos gestores dos dados.

Um recorte temporal do ano de 2023 foi realizado, selecionando as denúncias identificadas e anônimas cadastradas nesse ano, totalizando 9.717 registros não sigilosos. Os campos código identificador, texto da ocorrência, tipo (identificada ou anônima), área de atuação (destino da denúncia) e data de cadastro foram coletados.

Considerando que existia a percepção de que o cidadão não efetuara corretamente a classificação das denúncias, foi solicitado auxílio a cada CAO para reclassificá-las. Após a revisão da rotulação foi identificado um percentual de 29,43% de erros na escolha do cidadão, considerando inclusive as denúncias que estão fora do escopo de atuação do MPBA.

Também foram executadas as seguintes adequações:

- Denúncias com textos vagos (por exemplo: “Conforme anexo”; “segue anexo”) foram classificadas como “DESCARTAR”;
- Classificação de algumas denúncias repetidas como “DESCARTAR”; e
- Denúncias que não são atribuição do MPBA foram classificadas como “OUTROS ORGAOS” a qual se tornou uma nova classe para o presente estudo.

### B. Pré-processamento dos dados

Os dados foram carregados e visualizados para uma compreensão inicial, o que incluiu a análise de estrutura de dados e valores faltantes. Em seguida foi realizada a remoção de 695 denúncias classificadas como “DESCARTAR”, a limpeza de dados textuais, o que englobou a remoção de pontuação, caracteres especiais, conversão para minúsculas, remoção de *stopwords* e nomes próprios. Por fim, foram aplicadas técnicas de NLP [10], como normalização, tokenização e *lematização* de maneira a padronizar as formas das palavras, assim como unigramas e bigramas, permitindo um *dataset* mais conciso, preservando ao máximo o contexto das denúncias.

Com a conclusão da execução das rotinas de pré-processamento, outras 422 denúncias foram excluídas via código por também estarem em duplicidade não apontadas pelos usuários, restando 8.600 registros no conjunto de dados final, distribuídos em 11 classes (CAOCIFE, CEACON, CAOCRIM, CAODH, CAOCA, CEDUC, CEAMA, CAOPAM, CESAU, CEOSP e OUTROS ÓRGÃOS) conforme Figura 3.

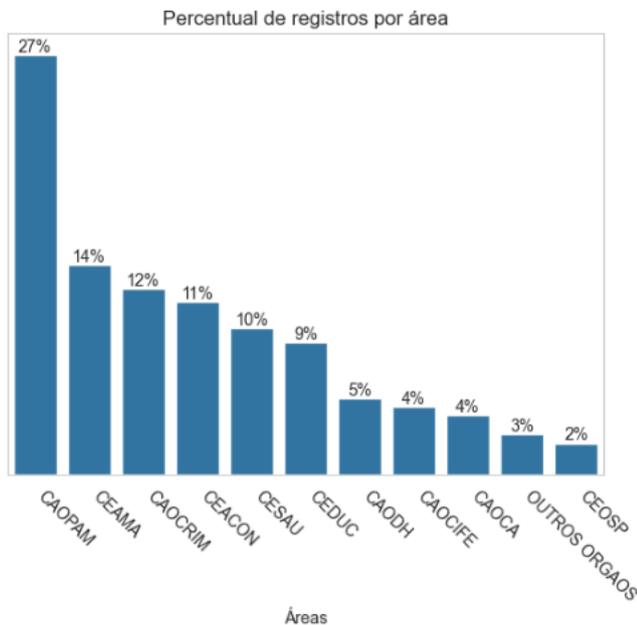


Fig. 3. Distribuição das denúncias por área.

Percebe-se o desbalanceamento dos dados utilizados no trabalho, mostrando que há uma concentração significativa de registros em determinadas áreas como CAOPAM. Considerando este cenário, no qual há um desbalanceamento claro entre as diferentes áreas de classificação de denúncias, foi utilizada a métrica F1 SCORE para medir o desempenho dos algoritmos de classificação.

### C. Transformação dos dados

Nesta etapa foi aplicada transformação de dados textuais, o que também engloba técnicas de NLP [10]. Foram utilizadas duas técnicas para representação do *corpus* em uma matriz numérica: *Bag of Words* (BoW) e *Embeddings*.

A técnica *Bag of Words* foi utilizada com o cálculo TF-IDF para fornecer a entrada para os algoritmos de *machine*

*learning*, ao passo que, para o *Embeddings*, foi utilizado o modelo BERT [11], o qual alimenta tanto os algoritmos de *machine learning* quanto de *deep learning*, com exceção do modelo baseado em *DistilBERT*, o qual utilizou um modelo de *embeddings* próprio [12].

Para a seleção de atributos mais relevantes foi utilizado o *Chi-Square* (Chi<sup>2</sup>) em conjunto com o *k-best*, a fim de filtrar atributos que têm menos impacto no resultado previsto, reduzindo a complexidade do modelo e o tempo de treinamento e aumentando a capacidade de generalização.

Para testar diferentes valores de “k”, foi executada um *pipeline* composto pelo *k-best* (Chi<sup>2</sup>) e classificador *Random Forest* com validação cruzada de cinco partições, utilizando o F1 SCORE como métrica de avaliação. Após teste de diversos valores de “k” (de 500 a 10.000), identificou-se o valor de 4.000 que produziu o melhor desempenho médio, ou seja, o maior F1 SCORE.

### D. Mineração dos dados

A divisão dos dados foi feita de forma estratificada para todos os algoritmos, garantindo que a distribuição das classes fosse mantida em cada conjunto. Para os de *machine learning* o *dataset* foi dividido em 90% para treinamento e 10% para teste. Enquanto para os de *deep learning* a divisão foi de 70% para treino, 15% para teste e 15% para validação.

Nesta etapa do processo, foram utilizados seis algoritmos de classificação baseados em *machine learning*: *Naive Bayes*, *Random Forest*, *Support Vector Machine (SVM)*, *Logistic Regression*, *XGBoost* e *k-NN* e três algoritmos baseados em redes neurais: *BertForSequenceClassification* [13], *DistilBERTForSequenceClassification* [14] e *Bert Embeddings* com *LSTM* [15].

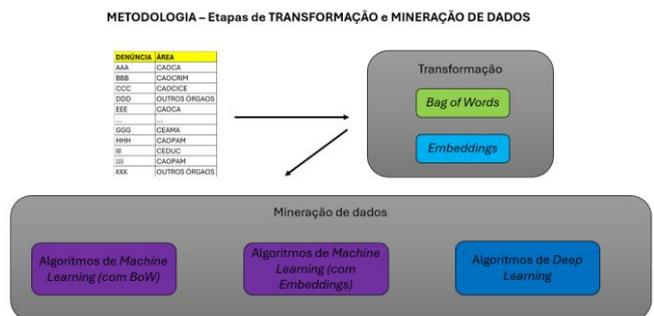


Fig. 4. Detalhamento da etapa “Mineração de dados”.

Os algoritmos de *machine learning* passaram por uma otimização de hiperparâmetros via *GridSearch* com validação cruzada de 5 partições (*folds*), utilizando o F1 Score como métrica principal.

Os hiperparâmetros utilizados em cada algoritmo foram:

- *Naive Bayes*: *fit\_prior* e *alpha*;
- *Random Forest*: *bootstrap*, *max\_depth*, *max\_features*, *min\_samples\_leaf*, *min\_samples\_split* e *n\_estimators*;
- *Support Vector Machine (SVM)*: *degree*, *shrinking*, *kernel*, *C* e *gamma*;
- *Logistic Regression*: *penalty*, *C* e *solver*;

- *XGBoost*: *learning\_rate*, *n\_estimators*, *max\_depth*, *min\_child\_weight*, *gamma*, *subsample*, *colsample\_bytree*, *reg\_alpha* e *reg\_lambda*;
- *k-Nearest Neighbors (kNN)*: *n\_neighbors*, *weights*, *algorithm* e *leaf\_size*;

Eles foram executados tanto com *Bag of Words* quanto com *embeddings* para vetorização do *corpus*, conforme Figura 4. Já os algoritmos de *deep learning* utilizaram exclusivamente *embeddings*.

A classificação multiclasse englobou onze categorias: dez áreas de atuação do MPBA e uma classe representando denúncias que não são atribuição do MPBA. Todos os nove algoritmos listados foram executados nesta estratégia.

### E. Interpretação e avaliação dos resultados

O resultado da execução dos nove algoritmos foi disposto em uma planilha juntamente com as métricas F1 SCORE e acurácia. Considerando o desbalanceamento da base de dados, o F1 SCORE foi a métrica escolhida para determinação do melhor modelo.

Após selecionado o melhor modelo, sua acurácia foi comparada com a acurácia da classificação realizada pelos usuários e revisada pelos CAOs a fim de melhorar a taxa de erro.

## V. RESULTADOS

Após a execução dos nove algoritmos de classificação, foram obtidos os resultados apresentados nesta seção.

### A. Resultados dos algoritmos de Machine Learning

Os algoritmos de *machine learning* utilizados no experimento foram: *Naive Bayes*, *Random Forest*, *Support Vector Machine (SVM)*, *Logistic Regression*, *XGBoost* e *k-NN*. Esses algoritmos foram avaliados utilizando duas técnicas de vetorização: *Bag of Words (BoW)* com TF-IDF e *embeddings*. Os resultados são apresentados nas Tabelas I e II.

TABELA I  
Métricas de desempenho dos algoritmos de *machine learning* com *BoW*

SN	Algoritmo	F1 Score (%)	Acurácia (%)
1	<i>Naive Bayes</i>	72,40	73,10
2	<i>Random Forest</i>	67,50	69,98
3	<i>Support Vector Machine</i>	71,30	71,50
4	<i>Logistic Regression</i>	72,90	74,00
5	<i>XGboost</i>	69,90	74,00
6	<i>k-Nearest Neighbors</i>	61,11	62,24

O algoritmo *Logistic Regression* obteve o melhor desempenho, com as duas métricas apuradas: F1 SCORE (72,90%) e acurácia (74,00%), o que sugere que ele foi o mais eficaz em equilibrar *precisão* e *recall* ao classificar as denúncias. Esse resultado indica que o modelo foi capaz de identificar corretamente a maior parte das classes,

minimizando tanto falsos positivos quanto falsos negativos. O algoritmo *Naive Bayes* apresentou um desempenho bastante competitivo, com F1 SCORE de 72,40% e acurácia de 73,10%, sendo uma boa opção quando se considera a simplicidade de sua implementação e baixa exigência computacional. Já o *Random Forest* e *XGBoost* tiveram desempenhos similares em acurácia (69,98% e 74,00%, respectivamente), mas diferiram no F1 SCORE (67,50% e 69,90%). O SVM também teve um desempenho sólido, com F1-Score de 71,30% e acurácia de 71,50% porém não superou a regressão logística. Por fim, o k-NN apresentou o pior desempenho, com F1 SCORE de 61,11% e acurácia de 62,24%. Isso pode ser atribuído ao fato de que o algoritmo k-NN não lida bem com grandes quantidades de variáveis ou quando os dados não possuem uma clara separação entre classes, como é o caso das denúncias neste contexto.

TABELA II  
Métricas de desempenho dos algoritmos de *machine learning* com *Embeddings*

SN	Algoritmo	F1 Score (%)	Acurácia (%)
1	<i>Naive Bayes</i>	49,80	54,80
2	<i>Random Forest</i>	66,50	68,40
3	<i>Support Vector Machine</i>	70,50	71,20
4	<i>Logistic Regression</i>	68,60	69,10
5	<i>XGboost</i>	68,00	69,10
6	<i>k-Nearest Neighbors</i>	66,10	67,40

Com a vetorização por *embeddings*, o algoritmo *Support Vector Machine* apresentou o melhor desempenho com F1 SCORE de 70,50% e acurácia de 71,20%, seguido pela *Logistic Regression* com F1 SCORE de 68,60% e acurácia de 69,10%. Os algoritmos *Random Forest* e *k-NN* tiveram performances semelhantes, mas inferiores ao SVM, evidenciando limitações ao lidar com *embeddings*. A técnica de *embeddings* evidenciou a dificuldade do *Naive Bayes*, que obteve o pior desempenho com F1 SCORE de 49,80% e acurácia de 54,80%, uma queda significativa em comparação ao uso de *Bag of Words*.

### B. Resultados de Deep Learning

Os algoritmos de *deep learning* aplicados foram: *BertForSequenceClassification*, *DistilBERTForSequenceClassification* e *Bert Embeddings* com *LSTM*. Todos utilizaram *embeddings* como técnica de vetorização. Os resultados são apresentados na Tabela III.

TABELA III  
Métricas de desempenho dos algoritmos de *deep learning*

SN	Algoritmo	F1 Score (%)	Acurácia (%)
1	<i>BertForSequenceClassification</i>	74,80	75,00

2	<i>DistilBERTForSequenc eClassification</i>	70,80	71,30
3	<i>Bert Embeddings com LSTM</i>	73,60	74,00

Dentre os algoritmos de *deep learning*, o *BertForSequenceClassification* obteve o melhor desempenho, com *F1 Score* de 74,80% e acurácia de 75,00%. O *DistilBERTForSequenceClassification* apresentou um desempenho ligeiramente inferior, com *F1 Score* de 70,80% e acurácia de 71,30%. O *Bert Embeddings* com *LSTM* também obteve um bom desempenho, com *F1 Score* de 73,60% e acurácia de 74,00%, ficando próximo ao *BertForSequenceClassification*.

### C. Matriz de confusão

A Figura 5 apresenta a matriz de confusão gerada pelo conjunto de dados de testes para o modelo *BertForSequenceClassification*. A partir dela é possível ter uma visão detalhada do desempenho de classificação, identificando as classes no qual o modelo é mais preciso e aquelas na qual ele tende a cometer erros.

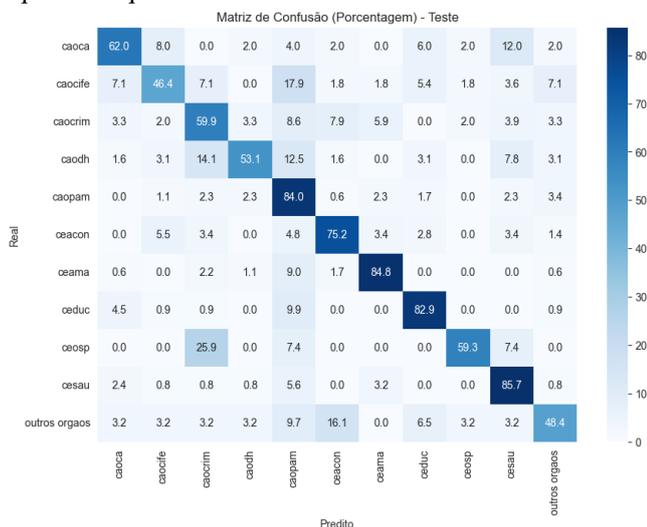


Fig. 5. Matriz de confusão do modelo *BertForSequenceClassification*.

O modelo apresentou uma boa performance para as classes CAOPAM, CEACON, CEAMA, CEDUC e CESAU indicando que ele é bastante eficiente ao classificá-las.

Por outro lado, para as classes CAOCIFE e OUTROS ÓRGÃOS, o modelo apresentou valores abaixo de 50%, indicando dificuldade em identificar corretamente essas classes. Isso pode ser devido ao menor número de exemplos disponíveis e à complexidade em diferenciá-las de outras classes. Além disso, o modelo parece ter dificuldade em capturar detalhes contextuais entre diferentes áreas de atuação. Por exemplo, uma denúncia sobre "meu pai que está esperando regulação" seria tratada pelo CAODH (Direitos Humanos) se o paciente tiver mais de 65 anos, e pelo CESAU (Saúde) caso contrário.

Em especial, identificou-se um percentual de 25,90% denúncias que seriam atribuição do CEOSP, mas que foram classificadas para o CAOCRIM. Em contato com as áreas de negócio, foi explicado que um dos possíveis motivos seria o

fato de que ambas apuram a ocorrência de crimes sendo que o CEOSP trata os crimes que foram cometidos por agentes de segurança, o que pode não estar claro nos textos informados pelos cidadãos.

### D. Análise

Ao comparar os resultados entre *machine learning* e *deep learning*, observa-se que os modelos de *deep learning*, especialmente o *BertForSequenceClassification*, superaram os algoritmos de *machine learning* em termos de *F1 SCORE* e acurácia, tanto com as técnicas de vetorização BoW quanto *embeddings*. Isso ocorre devido à capacidade dos modelos baseados em *BERT* de capturar relações mais complexas e sutis nos dados textuais, o que se reflete na maior precisão nas previsões. No entanto, é importante destacar que o desempenho de modelos como *Support Vector Machine* e *Logistic Regression*, especialmente com BoW, foi bastante competitivo.

A análise dos resultados evidencia o impacto do desbalanceamento de dados, especialmente em cenários como o do MPBA, no qual o volume de denúncias por classe varia significativamente. Algoritmos como *Naive Bayes*, que dependem fortemente da distribuição de probabilidades, foram mais impactados pelo desbalanceamento, apresentando desempenho inferior. Modelos mais robustos, como o *BertForSequenceClassification*, foram capazes de mitigar esses problemas, mas ainda apresentaram dificuldades em classes com poucos exemplos.

## VI. CONCLUSÃO

Este trabalho demonstrou que algoritmos de classificação baseados em inteligência artificial são promissores na tarefa de rotular textos, como denúncias, em diferentes classes.

Foram analisados os desempenhos dos classificadores *Naive Bayes*, *Random Forest*, *Support Vector Machine*, *Logistic Regression*, *XGBoost*, *k-Nearest Neighbors*, *BertForSequenceClassification*, *DistilBERTForSequenceClassification* e *Bert embeddings com LSTM*, com o objetivo de otimizar o processo de análise das denúncias efetuado pelos Centros de Apoio Operacionais do MPBA. A redução da taxa de erro na classificação visa melhorar a eficiência no atendimento à população e gerar economia de recursos institucionais.

Utilizando a métrica de avaliação F1-SCORE, o algoritmo *BertForSequenceClassification* obteve o melhor desempenho (74,80%) entre todos os avaliados. Além disso, este modelo reduziu a taxa de erro da classificação realizada pelos cidadãos e revisada pelos CAOs, de 29,43% para 25,00%. Em números absolutos, considerando apenas o ano de 2023, implicaria que aproximadamente 380 denúncias teriam sido encaminhadas corretamente.

Dentre as principais contribuições deste trabalho, destacam-se vários aspectos que merecem ser aprofundados. Com a implementação dos algoritmos de classificação automatizada, o tempo necessário para a triagem das denúncias classificadas equivocadamente pelos cidadãos seria reduzido. Essa melhoria não só aumentaria a satisfação dos cidadãos, mas também fortaleceria a confiança na capacidade de resposta da instituição.

Em segundo lugar, promoveria economia de recursos institucionais já que a automatização do processo de

classificação das denúncias reduziria a carga de trabalho manual, diminuindo a necessidade de alocação de recursos humanos para essa tarefa. Isso permitiria que os servidores se concentrassem em atividades de maior valor estratégico, resultando em uma utilização mais eficiente dos recursos disponíveis e, conseqüentemente, em uma redução de custos operacionais.

Este trabalho promove o fomento à inclusão de soluções de inteligência artificial na instituição. A aplicação prática de algoritmos de *machine learning* e *deep learning* demonstra o potencial da IA para resolver problemas complexos e repetitivos, incentivando a instituição a adotar mais tecnologias avançadas em outras áreas de atuação. Essa integração não só moderniza os processos internos, mas também posiciona a instituição na trilha da inovação tecnológica no setor público.

A principal dificuldade deste trabalho foi a correta rotulagem das denúncias feitas pelos cidadãos, devido à limitada disponibilidade dos Centros de Apoio Operacionais para colaborar na classificação focada neste estudo. Isso também resultou na redução dos dados disponíveis, pois só foi possível analisar denúncias cadastradas em 2023. Além disso, o tempo excessivo de processamento local dos dados (cerca de 20 horas para executar todos os modelos) devido à necessidade de sigilo dos dados limitou a execução de variações nos modelos propostos.

Para trabalhos futuros, sugere-se uma nova abordagem de classificação utilizando Modelos de Linguagem de Grande Escala (*Large Language Models – LLM*), embora a quantidade limitada de denúncias corretamente classificadas represente um desafio. Recomenda-se também a reformulação do site "Atendimento ao Cidadão" para integrar a classificação automatizada, permitindo que o cidadão digite sua denúncia e esta seja automaticamente encaminhada ao CAO apropriado pelo algoritmo. Por fim, sugere-se a padronização do cadastramento das denúncias no Sistema IDEA [16] para evitar tramitação paralela por e-mail ou duplicação de cadastros.

#### AGRADECIMENTOS

Os autores agradecem à Procuradoria-Geral de Justiça, ao Centro de Estudos e Aperfeiçoamento Funcional (CEAF), Coordenadoria de Segurança Institucional e Inteligência (CSI) do Ministério Público do Estado da Bahia pelo apoio institucional para realização desta especialização. Agradecem também aos dez Centros de Apoio Operacional

deste *Parquet* pelo auxílio na rotulagem das denúncias e explanação das rotinas de trabalho.

#### REFERÊNCIAS

- [1] República Federativa do Brasil, Constituição da República Federativa do Brasil de 1988, Brasília, DF: Presidência da República, 2023. [Online]. Disponível em: [https://www.planalto.gov.br/ccivil\\_03/Constituicao/Constituicao.htm](https://www.planalto.gov.br/ccivil_03/Constituicao/Constituicao.htm), acessado em: 11 de setembro de 2024.
- [2] MPBA, disponível em <https://www.mpba.mp.br/>, acessado em: 03 de junho de 2024.
- [3] Atendimento ao Cidadão, disponível em <https://atendimento.mpba.mp.br/>, acessado em: 03 de junho de 2024.
- [4] Q. Ain, A. Shaukat e U. Saif, "NLP based Model for Classification of Complaints: Autonomous and Intelligent System," 2022 International Conference on Digital Transformation (ICoDT), Islamabad, Pakistan, 2022, pp. 1-6. doi: 10.1109/ICoDT255437.2022.9787456.
- [5] V. Vinayak e J. Chandrasekharan, "Consumer Complaints Classification using Deep Learning & Word Embedding Models," 2023 13th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2023, pp. 1-5. doi: 10.1109/ICCCNT56998.2023.10307286.
- [6] L. Rokach, O. Maimon, and E. Shmueli, Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook, 3rd ed. Springer International Publishing, 2023.
- [7] M. T. Hagan, H. B. Demuth, M. H. Beale, O. Jesús, Neural Network Design. Poland: Martin Hagan, 2014.
- [8] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach. N.p.: CreateSpace Independent Publishing Platform, 2016.
- [9] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery: an overview," in Advances in knowledge discovery and data mining, American Association for Artificial Intelligence, 1996, USA, pp. 1-34.
- [10] N. Indurkha and F. J. Damerau, Handbook of Natural Language Processing, 2nd ed. Chapman & Hall/CRC, 2010.
- [11] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," North American Chapter of the Association for Computational Linguistics, 2019.
- [12] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," ArXiv, 2019, abs/1910.01108.
- [13] BertForSequenceClassification, disponível em [https://huggingface.co/docs/transformers/en/model\\_doc/bert#transformers.BertForSequenceClassification](https://huggingface.co/docs/transformers/en/model_doc/bert#transformers.BertForSequenceClassification), acessado em: 27 de junho de 2024.
- [14] DistilBERTForSequenceClassification, disponível em [https://huggingface.co/docs/transformers/en/model\\_doc/distilbert#transformers.DistilBertForSequenceClassification](https://huggingface.co/docs/transformers/en/model_doc/distilbert#transformers.DistilBertForSequenceClassification), acessado em: 27 de junho de 2024.
- [15] LSTM, disponível em <https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html>, acessado em: 27 de junho de 2024.
- [16] IDEA, disponível em <https://idea.sistemas.mpba.mp.br/idea/>, acessado em: 03 de junho de 2024.

**CENTRO UNIVERSITÁRIO SENAI CIMATEC  
ESPECIALIZAÇÃO EM DATA SCIENCE & ANALYTICS**

**ATA DE APRESENTAÇÃO DE PROJETO FINAL DE CURSO**

Ata de apresentação do Projeto Final de Curso, “**Classificação automatizada de denúncias no MPBA usando machine learning**”, submetido pelo aluno Milton Cerqueira da Silva Sobrinho, como parte dos requisitos para obtenção do Certificado de **Especialista em Data Science & Analytics** pelo Centro Universitário SENAI CIMATEC, às 17h30 do dia 27 de Agosto de 2024. Reuniu-se remotamente pela plataforma Microsoft Teams, a Banca Examinadora designada pelo Prof MSc Braian Varjão Gama Bispo – Orientador, constituída pelo Prof MSc Braian Varjão Gama Bispo e Prof Dr. Éldman de Oliveira Nunes. O Orientador deu início aos trabalhos com as devidas orientações, e a exposição foi realizada pelo estudante dentro do prazo de tempo estabelecido. Ao final da apresentação a banca reuniu-se atribuindo a seguinte nota: **9,3** (nove pontos e três décimos).

**A banca de avaliadores decidiu pela:**

**( X ) Aprovação do trabalho**

Caberá ao aluno apresentar em no máximo em 30 (trinta) dias a contar da data de assinatura desta Ata, uma cópia do trabalho em PDF, constando as considerações pontuadas pela banca. A Ata de Apresentação do Projeto Final de Curso deve ser digitalizada e inserida na terceira página do TCC ou como anexo do artigo.

**( ) Reprovação do trabalho**

O aluno terá que se matricular novamente no TCC – Trabalho de Conclusão de Curso e ser submetido a uma banca avaliadora no semestre seguinte.

As ações consequentes ao status de Aprovação deverão obedecer ao prazo proposto acima sob pena do parecer final ser modificado para o status de Reprovado automaticamente e sem possibilidade de recurso.

Para constar, lavrou-se a presente ata que vai assinada por todos os membros da Banca. Por estarem cientes de suas obrigações estão de acordo com os termos desse documento:

Salvador, 27 de Agosto de 2024.

---

**Braian Varjão Gama Bispo**  
Professor Orientador

---

**Éldman de Oliveira Nunes**  
Membro da banca