UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
CURSO DE GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Samuel Moreira Ransolin

Detecção de Documentos Acadêmicos Falsificados: Uma Solução Baseada em Aprendizado de Máquina

Florianópolis
2025

Samuel Moreira Ransolin

Detecção de Documentos Acadêmicos Falsificados: Uma Solução Baseada em Aprendizado de Máquina

Relatório de Trabalho de Conclusão de Curso 1 do Curso de Graduação em Ciências da Computação do Centro Tecnológico da Universidade Federal de Santa Catarina para a obtenção do título de Bacharel em Ciências da Computação.
Orientadora: Giovana Nunes Inocêncio, M.a.

Florianópolis
2025

Samuel Moreira Ransolin

Detecção de Documentos Acadêmicos Falsificados: Uma Solução Baseada em Aprendizado de Máquina

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de "Bacharel em Ciências da Computação" e aprovado em sua forma final pelo Curso de Graduação em Ciências da Computação.

Florianópolis, 11 de dezembro de 2025.

Banca Examinadora:

Giovana Nunes Inocêncio, M.a.
Universidade Federal de Santa Catarina

Prof. Jean Everson Martina, Dr.
Universidade Federal de Santa Catarina

Lucas Machado da Palma, Dr.
Universidade Federal de Santa Catarina

Gabriel Estevam de Oliveira, M.e.
Universidade Federal de Santa Catarina

RESUMO

Nos últimos anos, no Brasil, o expressivo aumento do número de ingressantes, formandos e de instituições de ensino superior, trouxe consigo desafios relacionados à validação da autenticidade de certificados acadêmicos. Atualmente, o processo consiste na verificação de forma predominantemente manual, sujeita a erros e falhas, como a aceitação de documentos falsificados. Nesse contexto, surge a Jornada do Estudante, um sistema disponibilizado pelo Ministério da Educação (MEC), que oferece, entre outras funcionalidades, o acompanhamento de registros acadêmicos de estudantes por meio de uma rede distribuída, de forma a garantir que somente instituições de ensino digitalmente certificadas possam registrar créditos e certificações. O presente trabalho de conclusão de curso revisita o estado-da-arte em detecção via aprendizado de máquina de documentos falsificados, além de propor um protótipo de solução híbrida, que combina análise multimodal, *clustering*, detecção de anomalias e classificação de documentos de acordo com seu grau de legitimidade. Ao integrar esse sistema à Jornada do Estudante, é possível validar automaticamente os documentos antes de seu registro em *blockchain*, aumentando significativamente a segurança e a confiabilidade do processo de credenciamento.

Palavras-chave: segurança da informação, detecção de fraude, *machine learning*, *clustering*, detecção de anomalias, extração multimodal

ABSTRACT

In recent years, the significant increase in the number of higher education institutions, incoming students, and graduates in Brazil has brought challenges related to the validation of academic certificates' authenticity. This process is currently performed predominantly manually and prone to errors and failures, such as the acceptance of fraudulent documents. In this context, the Jornada do Estudante, a system provided by the Ministry of Education (MEC), was introduced; among other features, it enables the tracking of students' academic records through a distributed network, ensuring that only digitally certified institutions can register credits and certifications. This undergraduate thesis revisits the state of the art in machine learning-based forgery detection for academic documents and proposes a hybrid prototype solution that combines multimodal analysis, clustering, anomaly detection, and document classification according to their degree of legitimacy. By integrating this system with the Jornada do Estudante, documents can be automatically validated before being recorded on blockchain, significantly enhancing the security and reliability of the credentialing process.

Keywords: information security, fraud detection, machine learning, clustering, anomaly detection, multimodal feature extraction

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

No Brasil, entre 2013 e 2023, o número de matrículas de alunos na educação superior aumentou 36,2%, com uma média de crescimento anual de 3,2%. O número de concluintes acompanhou essa mesma tendência de crescimento, sendo que o ano de 2013 registrou cerca de 992 mil graduandos, enquanto 2023 terminou com mais de 1,3 milhões. Para acomodar essa demanda, existem 2580 entidades de ensino superior no país (BRASIL, 2024), das quais 87,8% são privadas. Essas estatísticas revelam um saldo extremamente positivo, mas também trazem à tona desafios que precisam ser superados, como a melhoria nos processos de regulação, supervisão e avaliação destas entidades por parte do Ministério da Educação do Brasil (MEC), temática que pretende ser explorada nesse trabalho.

Atualmente, a gerência, armazenamento e cuidado de documentos acadêmicos, como diplomas e históricos escolares, é responsabilidade da instituição de ensino que os emitem (MEC, 1978). Além disso o próprio processo para a emissão desses documentos é burocrático, não computadorizado e sujeito a erros ou até mesmo fraudes, já que suas validações não possuem transparência ou redundância (Palma et al., 2019). Assim, a falta de modernização desses procedimentos deixam brechas que são conhecidas e utilizadas por agentes mal intencionados, possibilitando a criação de falsas instituições de ensino, especializadas na venda de pacotes que incluem históricos escolares falsificados, atas de colação de grau inexistentes e outros certificados contrafeitos, amparados em documentos oficiais adulterados, de forma a conferir aparência de legalidade a diplomas que, efetivamente, não têm qualquer base acadêmica real (Dias; Leal, 2022).

O comércio clandestino de diplomas falsos oferece certificados em diversas áreas e níveis, desde medicina até direito; desde a graduação até o doutorado, por valores que podem chegar a R$100.000, tornando essa prática altamente lucrativa e atraente para fraudadores (Palma et al., 2019). Investigações recentes demonstram que quadrilhas estruturadas conseguem emitir dezenas de milhares de documentos forjados, comercializados em sites especializados, com suposta publicação em diários oficiais (Fantástico, 2025). Para além da corrupção, esse tipo de fraude compromete a confiança pública nas instituições de ensino e no mercado de trabalho: indivíduos sem qualificação adequada podem assumir funções críticas, enquanto diplomas legítimos perdem valor diante da insegurança sobre sua autenticidade (Mohammed; Nwobodo; Ekene, 2024).

Neste cenário, o MEC, em parceria com o Ministério da Economia, disponibiliza e desenvolve o sistema da Jornada do Estudante junto a Universidade Federal de Santa Catarina (por meio do Laboratório Bridge e do Laboratório de Segurança em Computação), a Universidade Tecnológica Federal do Paraná e a Universidade Federal de Mato Grosso do Sul (MEC, 2022). Este sistema permite que alunos acompanhem seus dados estudantis durante toda a trajetória educacional, além da disponibilização de documentos acadêmicos pertinentes. Em conjunto a isso, existe a iniciativa para que se torne uma plataforma

conjunta para a emissão e registro destes certificados, unificando diplomas, históricos escolares, currículos e até mesmo dados regulatórios das instituições de ensino superior (RNP, 2023).

Para armazenar estes dados, o sistema da Jornada do Estudante utiliza uma *blockchain Hyperledger Fabric*, que aproveita características como descentralização da posse e imutabilidade dos registros, além da rastreabilidade às emissões. O projeto realiza o processamento dos dados em rede através de *smart contracts* e baseia-se em gestão de identidade forte, com certificados digitais que servem como base da identidade, de forma que somente entidades reconhecidas pelo projeto possam efetuar transações (Palma et al., 2019; RNP, 2023).

Ainda assim, hoje, a Jornada do Estudante não elimina o risco do registro de documentos falsificados, mas seu arcabouço permite o desenvolvimento de uma solução para este desafio. O presente Trabalho de Conclusão de Curso (TCC) trata da implementação e validação de um protótipo de *software* de inteligência artificial, que combina diferentes técnicas de aprendizado de máquina, capaz de identificar certificados falsos antes de sua inserção neste ambiente. Dessa forma, ao integrar essa tecnologia à Jornada do Estudante, espera-se aprimorar o processo de registro e emissão de documentos acadêmicos no Brasil, de forma que o emprego dessa análise automatizada, em tempo real, possa garantir maior segurança e confiabilidade a estes procedimentos.

1.1 OBJETIVOS

Esta seção apresenta o objetivo geral e objetivos específicos deste trabalho.

1.1.1 Objetivo Geral

Desenvolver um protótipo de software capaz de classificar documentos acadêmicos por grau de probabilidade de fraude, com base na integração de diferentes técnicas de aprendizado de máquina e análise multimodal.

1.1.2 Objetivos Específicos

Para atingir o objetivo geral, os seguintes objetivos específicos devem ser cumpridos:

- Construir uma base de documentos acadêmicos para o treinamento, teste e validação do protótipo.
- Desenvolver sistema para pré-processamento dos documentos, aplicando reconhecimento óptico de caracteres e normalização dos documentos digitalizados;
- Projetar e implantar sistema para extração de características visuais, textuais e estruturais dos documentos processados;

- Desenvolver um modelo de agrupamento dos documentos com base nas *features* extraídas e formar *clusters* de referência para comportamento padrão;
- Criar detectores de anomalias baseados nos *clusters* formados, capazes de calcular escores de desvio e classificá-los em categorias discretas de suspeita de fraude;

2 FUNDAMENTAÇÃO TEÓRICA

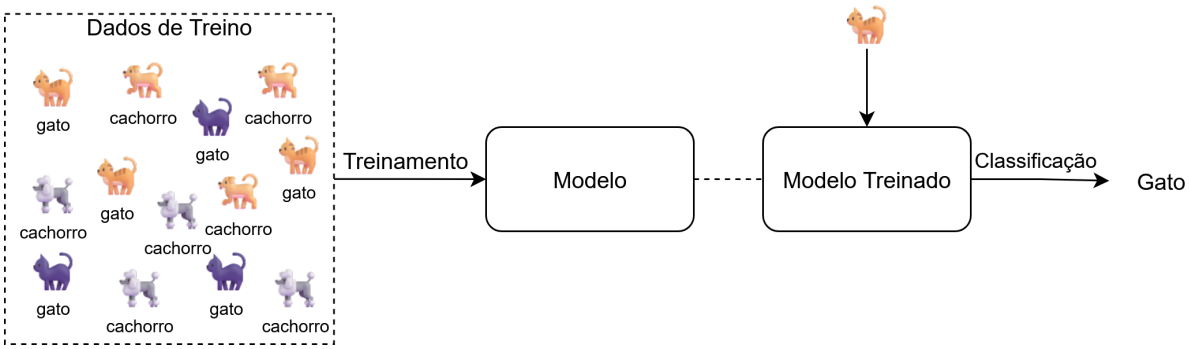
Este capítulo aborda os conceitos teóricos necessários para a compreensão do presente trabalho.

2.1 APRENDIZADO DE MÁQUINA

O aprendizado de máquina é um subcampo da inteligência artificial que tem por objetivo o desenvolvimento de algoritmos capazes de aprender e tomar decisões a partir de um conjunto de dados, sem que seja necessária a programação explícita para essas tarefas específicas (Dietterich, 2003). Fundamentalmente, esses sistemas buscam aprender padrões em coleções de dados para, a partir da generalização desse conhecimento, realizar inferências sobre novas informações. Esse processo de aprendizado utiliza modelos matemáticos, principalmente estatísticos, que capturam relações complexas entre variáveis de entrada e saída através do ajuste de parâmetros internos, permitindo que a aplicação melhore seu desempenho conforme é exposta a mais dados (Sarker, 2021; Dietterich, 2003). Em geral, as técnicas de *machine learning* são categorizadas em três paradigmas principais: aprendizagem supervisionada, não supervisionada e por reforço (Sarker, 2021).

O aprendizado supervisionado caracteriza-se pela utilização de conjuntos de dados rotulados, onde tanto as entradas quanto as saídas desejadas são conhecidas durante o treinamento. Nesse paradigma, o algoritmo aprende através de exemplos, de forma a possibilitar tarefas como classificação – a atribuição de classes discretas aos dados, como ilustrado pela Figura 1 – e regressão – a predição de valores contínuos (Dietterich, 2003). Algoritmos clássicos dessa categoria incluem máquinas de vetores de suporte, redes neurais artificiais e métodos *ensemble* (Sarker, 2021).

Figura 1 – Representação de Aprendizado Supervisionado

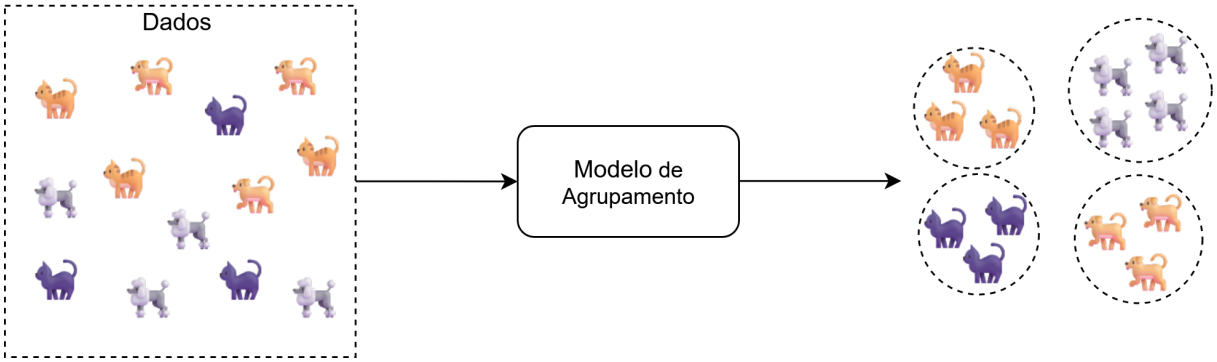


Fonte: o autor

O aprendizado não-supervisionado, por sua vez, opera sobre dados não rotulados, sem o conhecimento das saídas desejadas, e busca compreender a organização natural

de um dado conjunto a partir da identificação de padrões intrínsecos. Essa abordagem engloba técnicas como agrupamento (*clustering*), ilustrado pela Figura 2, e detecção de anomalias (Sarker, 2021). Ainda, diferentes estratégias de aprendizado podem ser incorporadas, como algoritmos semi-supervisionados, utilizados quando um *dataset* tem poucos dados classificados, de forma a aproveitar a estrutura implícita do conjunto não categorizado para melhorar o desempenho do modelo (Sarker, 2021).

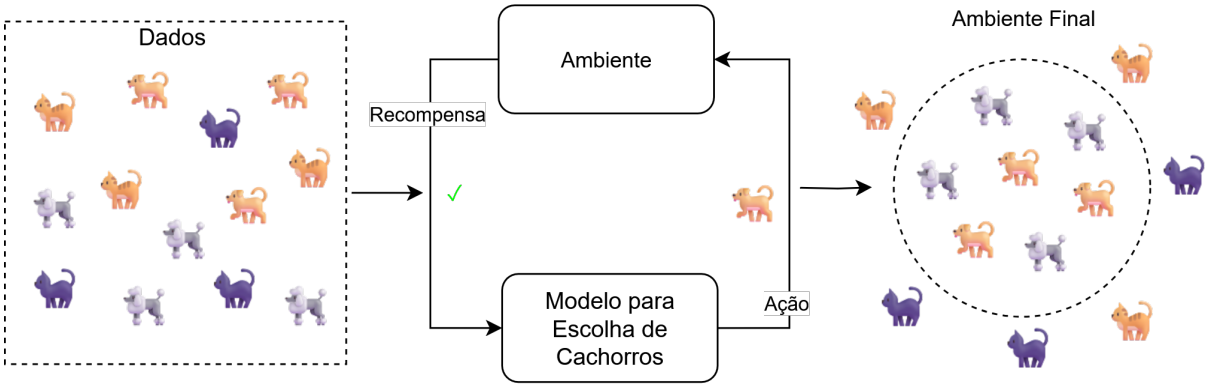
Figura 2 – Representação de Aprendizado Não Supervisionado



Fonte: o autor

Adicionalmente, o aprendizado por reforço representa um paradigma distinto onde um modelo aprende através de interações com um ambiente, sendo recompensado ou penalizado com base em suas ações, de forma que gradualmente desenvolva estratégias ótimas (Sarker, 2021). A Figura 3 ilustra o exemplo de um modelo com recompensas positivas para a identificação de cães.

Figura 3 – Representação de Aprendizado por Reforço



Fonte: o autor

Embora os conceitos fundamentais de aprendizado de máquina tenham sido estabelecidos há quase um século, com contribuições embrionárias nas décadas de 1950 e 1960

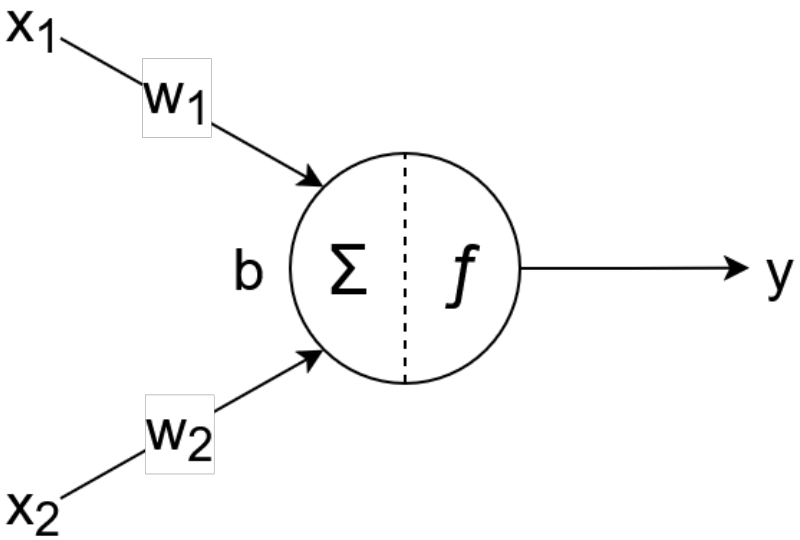
(Dietterich, 2003), essa área de estudo tem recebido grande destaque nas últimas décadas. Esse ressurgimento deve-se principalmente ao aumento exponencial da capacidade computacional e à disponibilidade massiva de dados digitais. Também, a evolução do *hardware* – particularmente o advento de unidades de processamento gráfico de alto desempenho (GPU) – possibilitou o treinamento de modelos complexos, antes computacionalmente intratáveis, transformando o aprendizado de máquina em uma tecnologia fundamental para aplicações modernas em diversas áreas (Sarker, 2021).

2.2 REDES NEURAIS PROFUNDAS

Redes neurais profundas, comumente utilizadas no paradigma de aprendizado supervisionado, são uma especialização de redes neurais artificiais. Diferentemente das técnicas tradicionais de *machine learning*, que requerem a engenharia manual de características, as abordagens baseadas em aprendizado profundo aprendem autonomamente representações complexas de dados brutos. Isso é possível devido à sua estrutura multicamada, que permite a extração progressiva de características de baixo nível – em uma imagem, por exemplo, bordas e linhas – até padrões de alto nível – como objetos e faces, no mesmo exemplo (Menghani, 2023).

A unidade mais básica de uma rede neural artificial é um neurônio artificial (referido neste estudo apenas como neurônio ou nó).

Figura 4 – Representação de uma Neurônio Artificial



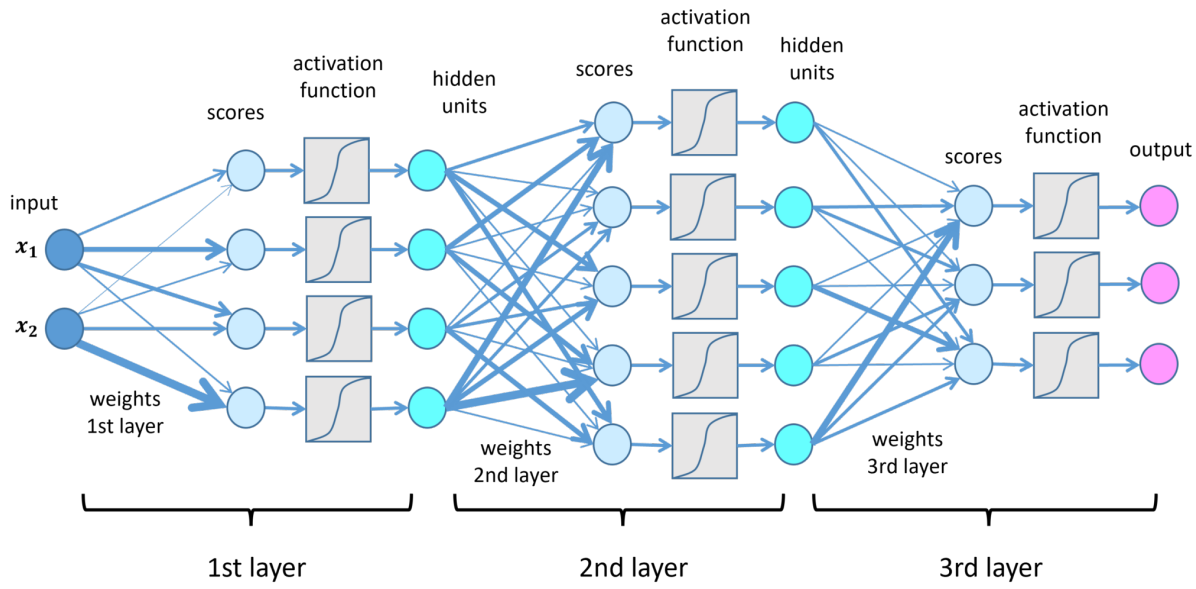
Fonte: o autor

No exemplo representado na Figura 4, um neurônio recebe dados de entrada, x_1 e x_2, e produz uma saída y. Para isso, cada entrada é multiplicada por seu respectivo peso,

w_1 e w_2, e somada junto a um termo de viés b – assim, a equação resultante é igual a $\sum = x_1 w_1 + x_2 w_2 + b$. Finalmente, aplica-se uma função de ativação f sobre a soma para converter esse valor em um intervalo desejado – a tangente hiperbólica produziria um número dentro do intervalo $(-1, 1)$, enquanto a sigmoide produziria um intervalo entre $(0, 1)$, por exemplo – resultando na saída y (Goodfellow; Bengio; Courville, 2016). Em outras palavras, os pesos indicam a importância, ou força, da conexão entre a entrada e o neurônio; o viés atua como um limiar de ativação que independe das entradas; e a função de ativação transforma uma entrada linear em uma saída não linear, o que permite um mapeamento complexo entre entradas e saídas.

Uma rede neural artificial é formada pela interligação de neurônios, assim, uma camada da rede é denominada a partir de um grupo de nós interligados, que processam dados de uma maneira específica. Combinando uma camada de entrada, camadas intermediárias e uma camada de saída, obtém-se uma rede neural profunda, ilustrada na Figura 5. Dessa forma, cada camada recebe entradas ponderadas a partir das camadas anteriores, aplicam uma função de ativação e propagam o resultado para as camadas subsequentes (Goodfellow; Bengio; Courville, 2016). Diferentes configurações destas, como a variação das conexões entre neurônios ou o emprego de funções de ativação distintas, tem por efeito especializações, ou habilidades de aprendizado específicas, assim, a utilização de múltiplas camadas permite a assimilação de representações hierárquicas complexas. (Alzubaidi et al., 2021).

Figura 5 – Representação de uma Rede Profunda

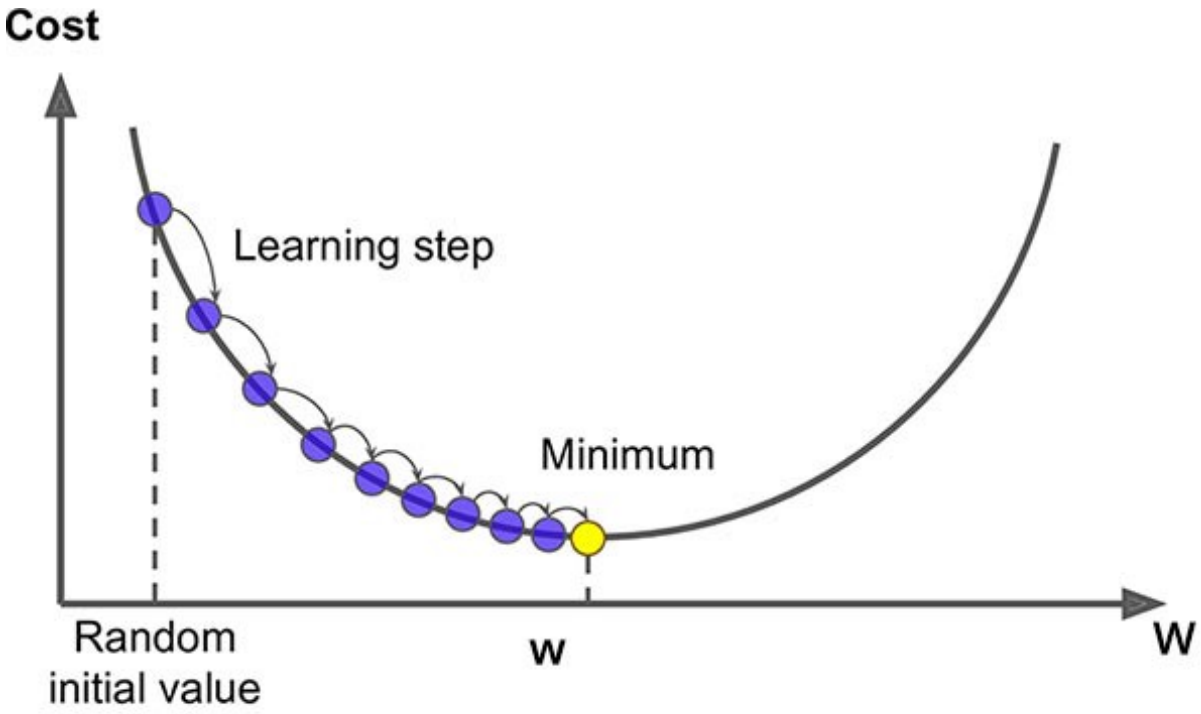


Fonte: `https://lamarr-institute.org/blog/deep-neural-networks/`. Acesso em: 25 junho 2025

O processo de aprendizado da rede é denominado treinamento e consiste na estima-

ção e ajuste dos parâmetros através de um algoritmo de retropropagação. Seu objetivo é calcular o gradiente de uma função que mede o erro entre o valor de saída computado e o esperado, além de ajustar os pesos e vieses dos neurônios na direção oposta ao gradiente, para minimizar o erro. Esse processo é executado em cada camada, propagando o erro desde a camada de saída até a de entrada, de forma iterativa, por múltiplas épocas, até que a rede converja para uma solução otimizada (Goodfellow; Bengio; Courville, 2016), como ilustrado na Figura 6, em que o eixo y representa valores de erro e o eixo x valores de peso.

Figura 6 – Visualização do Gradiente Descendente



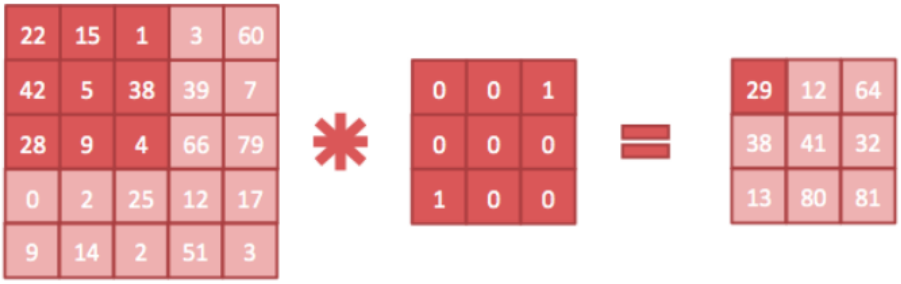
Fonte: `https://mlpills.dev/machine-learning/gradient-descent/`. Acesso em: 25 junho 2025

2.2.1 Redes Neurais Convolucionais

Redes neurais convolucionais são tipos de redes neurais profundas especialmente úteis na área de visão computacional, especializadas na computação de dados estruturados em topologia de malha, representados matricialmente. O que propicia essa propriedade é o emprego de operações de convolução em pelo menos um módulo da rede, o que consiste em uma mudança no processamento de entrada dos neurônios: ao invés da simples soma ponderada pelos pesos, um cálculo é efetuado a partir da aplicação de um filtro sobre um *input* (Goodfellow; Bengio; Courville, 2016). Dessa forma, a estrutura de um modelo básico combina camadas convolucionais com camadas de subamostragem, todas esparsamente conectadas (Alzubaidi et al., 2021).

Convolução é uma operação matemática sobre duas funções para a criação de uma terceira, que representa, em termos simplórios, a sobreposição delas. No contexto de redes convolucionais, consiste na multiplicação de duas matrizes seguida de uma soma (Wang; Raj, 2017), como demonstrado na Figura 7.

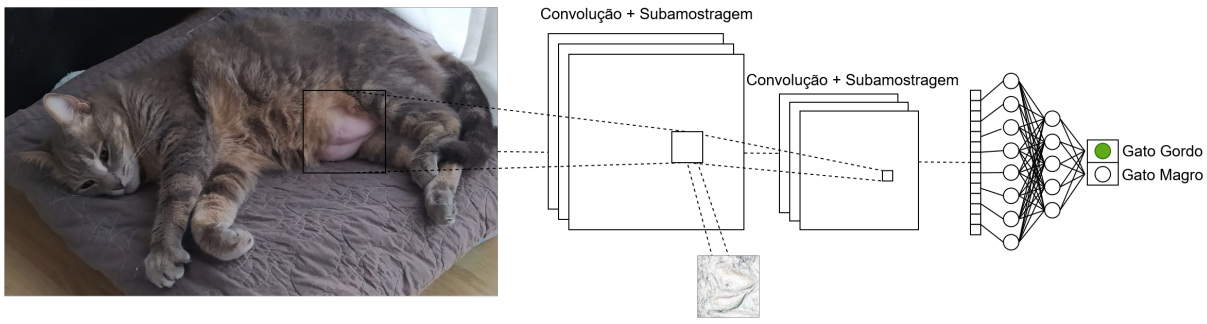
Figura 7 – Visualização de uma Operação de Convolução



Fonte: (Wang; Raj, 2017)

Um filtro – ou *kernel*, que é uma matriz de parâmetros aprendíveis – desliza sobre os dados de entrada, a um passo definido, computando o produto escalar entre seus pesos e os valores correspondentes na região coberta, de forma a criar um mapa de características que representa a presença de padrões específicos detectados pelo filtro (Goodfellow; Bengio; Courville, 2016). Dessa forma, certas configurações, como o tamanho e os pesos do *kernel*, ou o número de passos de deslizamento, controlam a resolução espacial e a capacidade de modelar relações de vizinhança, como figurativamente representado na Figura 8, em que a escolha de um filtro inicial para a detecção de bordas permite o treinamento da rede para a distinção entre gatos magros e gordos.

Figura 8 – Representação da Classificação de uma Imagem por uma Rede Convolucional



Fonte: o autor

Outro importante aspecto das redes convolucionais são as camadas de subamostragem, que reduzem a dimensão espacial dos dados de entrada ao passo em que preservam suas características mais relevantes, processo conhecido como *pooling*. Atingem isso com

a obtenção de apenas uma amostra para cada região analisada, para isso, empregam estratégias como *max-pooling* (extração do maior valor de entrada) e *average-pooling* (extração da média dos valores de entrada). A combinação de camadas convolucionais com camadas de subamostragem conferem às essas redes três propriedades fundamentais: a invariância à pequenas transformações, distorções e translações da entrada, que permite que características sejam detectadas independentemente de sua localização; a capacidade de extrair hierarquias espaciais através da redução progressiva de dimensionalidade; e a redução do custo computacional das camadas subsequentes (Wang; Raj, 2017).

Dessa forma, a extração de *features* acontece conforme a entrada percorre a rede. Utilizando como exemplo uma imagem para a entrada, as camadas iniciais capturam informações de baixo nível, como bordas, cantos e texturas. Em camadas intermediárias, essas informações passam a compor padrões semânticos locais, como delimitações de objetos, até que, em camadas mais profundas, transformam-se em caracterizações globais de alto nível, como formas completas. Isso permite que as camadas finais possam prever, ou extrair, representações complexas e abstratas (Alzubaidi et al., 2021), como o exemplo utilizado anteriormente – a classificação entre gatos gordos e magros.

2.2.1.1 Reconhecimento Óptico de Caracteres (OCR)

O reconhecimento óptico de caracteres (OCR), no contexto de computação, descreve um tipo de software que tem por objetivo a conversão de texto (impressos ou presentes em imagens) a um formato digital, que possa ser processável por máquina (Islam; Islam; Noor, 2016). Originalmente, os métodos de OCR baseavam-se em técnicas de segmentação da visão computacional clássica junto a heurísticas, como a segmentação de linhas e caracteres por projeções de histograma, seguida pela aplicação de regras heurísticas para identificar cada símbolo (Shi; Bai; Yao, 2015). No entanto, as abordagens que utilizam o aprendizado de máquina – em especial, redes convolucionais – têm se mostrado mais eficientes e precisas (Shi; Bai; Yao, 2015).

De acordo com Islam; Islam; Noor (2016), um processo de reconhecimento de caracteres moderno tipicamente passa pelo seguinte *pipeline*:

1. Aquisição da imagem: obtém-se a imagem com o texto a partir de uma fonte externa, como *scanner* ou câmera;

2. Pré-processamento: aplicam-se técnicas como remoção de ruído, operações morfológicas e limiarização para melhorar a qualidade da imagem;

3. Segmentação de caracteres: separam-se os caracteres através da análise de componentes conectados, perfis de projeção ou métodos avançados para tratar textos sobrepostos ou fragmentados;

4. Extração de *features* e classificação de caracteres: com os caracteres separados, utiliza-se uma rede convolucional para a extração e classificação de seus padrões;

5. Pós-processamento: para refinar os resultados e aumentar a acurácia, combinam-se técnicas de processamento de linguagem natural, como corretores ortográficos e dicionários com modelos probabilísticos, como cadeias de Markov e N-gramas.

2.2.2 Processamento de Linguagem Natural

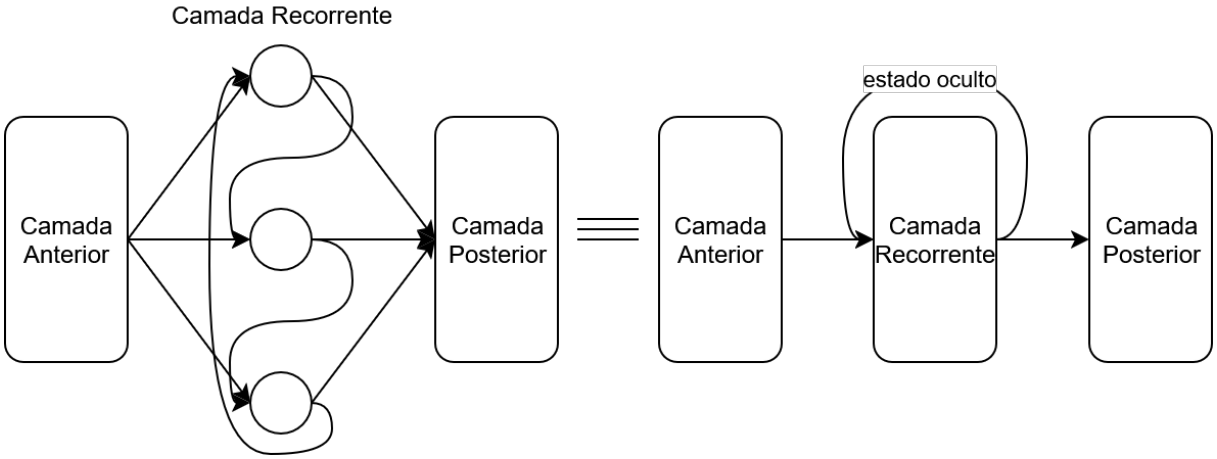
O processamento de linguagem natural é um subcampo da inteligência artificial que se dedica ao desenvolvimento de algoritmos capazes de compreender, interpretar e processar linguagem humana de forma computacional. Tradicionalmente, as abordagens baseavam-se em métodos estatísticos, modelos de N-gramas, campos aleatórios condicionais e máquinas de vetores de suporte. Contudo, as últimas décadas têm visto a adoção de técnicas baseadas em redes neurais profundas, que demonstraram capacidade superior de capturar relações complexas entre palavras e compreender contextos linguísticos mais amplos (Otter; Medina; Kalita, 2020).

Fundamentalmente, as técnicas modernas de processamento de linguagem consistem na transformação de texto em representações numéricas que preservam informações sintáticas e semânticas. *Embeddings* de palavras constituem o passo base dessa transformação, que consiste em um mapeamento destas para vetores densos, onde termos semanticamente similares ocupam posições próximas (Otter; Medina; Kalita, 2020). Isso possibilita o aprendizado de estruturas gramaticais e padrões linguísticos, que por sua vez permitem a predição de palavras com base em contextos previamente vistos.

As abordagens modernas de processamento de linguagem dividem-se, majoritariamente, em duas arquiteturas de redes profundas: especializações de redes neurais recorrentes, como LSTM (*long short-term memory*), desenvolvidas para capturar dependências de longo prazo em sequências; ou arquiteturas *Transformers*, capazes do processamento de sequências de forma paralela (Zhang; Shafiq, 2024).

Diferentemente dos modelos de rede apresentados anteriormente, em que os nós processam entradas de maneira independente e passam o resultado adiante, as redes neurais recorrentes utilizam conexões recorrentes, isto é, a saída de um neurônio é retroalimentada à sua camada no próximo passo de processamento, como ilustrado na Figura 9. Essa retroalimentação permite a conservação de um estado oculto que pode ser atualizado a cada passo temporal. Isso permite a captura de padrões e relações temporais dentro de uma sequência, tornando-as boas candidatas ao processamento de texto e outros dados onde a ordem dos elementos é importante (Tealab, 2018).

Figura 9 – Representação de uma Rede Neural Recorrente



Fonte: o autor

Em teoria, essas redes são capazes de capturar dependências de longo prazo, no entanto, modelos básicos acabam gradualmente perdendo informações distantes conforme as sequências de entrada ficam longas (Otter; Medina; Kalita, 2020). Isso ocorre por conta do problema de dissipação de gradiente, fenômeno inerente a redes profundas treinadas por *backpropagation*, onde os gradientes utilizados para atualizar os pesos das camadas anteriores diminuem exponencialmente à medida que se retropropagam (Goodfellow; Bengio; Courville, 2016). Isso significa que as camadas iniciais recebem gradientes muito pequenos, o que resulta em uma atualização insignificante dos pesos e, consequentemente, aprendizado lento ou até mesmo nulo.

Para resolver essa limitação, as redes LSTM utilizam células que controlam seletivamente o fluxo de informações. Essas estruturas guardam um estado por intervalos de tempo e são compostas, de forma geral, por três mecanismos principais: *forget gates* determinam quais informações devem ser descartadas; *input gates* determinam quais novas informações serão armazenadas no estado da célula; *output gates* determinam quais partes do estado interno influenciarão a saída. Essas decisões são tomadas a partir do cálculo de funções – tipicamente sigmoide, que resulta em um intervalo entre 0 e 1 – sobre parâmetros aprendíveis (Hochreiter; Schmidhuber, 1997). Dessa forma, a rede passa a aprender a lembrar dependências de longo prazo, ou esquecer informações contextuais desnecessárias.

Embora os modelos LSTM sejam eficazes no processamento de texto, possuem uma limitação principal: precisam processar a entrada sequência a sequência – ou palavra a palavra –, porque a computação dos estados seguintes dependem da computação dos estados anteriores. Isso faz com que o treinamento e a inferência sejam muito lentos e pouco escaláveis (Zhang; Shafiq, 2024).

As arquiteturas *Transformer* superam essa limitação através do processamento paralelo de todos os termos da sequência de entrada. Isso é possibilitado pelas camadas de

atenção, que calculam pesos que interconectam diferentes posições do *input*, permitindo que o modelo foque nas palavras mais relevantes às que estão em processamento. Esse processo acontece da seguinte forma: nas camadas iniciais da rede, a sequência de *embeddings* passa por três projeções lineares independentes, de forma a criar, para cada termo, vetores de consultas, chaves e valores; as camadas de atenção então computam similaridades entre as coleções de consulta e de chaves, normalizam esses escores em coeficientes para a obtenção de uma matriz de pesos, e combinam coleções de valores correspondentes de forma ponderada por esses pesos. Assim, operações aplicadas sobre consultas e chaves resultam na obtenção de valores que capturam, em cada posição, as dependências contextuais mais relevantes de toda a sequência – tudo isso de modo paralelo (Vaswani et al., 2017).

Dessa forma, os modelos *Transformer* estruturam-se em duas partes, a primeira consiste na utilização de blocos de codificação, que aplicam sucessivamente camadas de atenção e de propagação comum, assim, esses blocos "prestam atenção"somente na sequência de entrada. A segunda emprega blocos de decodificação que, em relação aos blocos de codificação, adicionam camadas intermediárias de atenção codificador-decodificador. Essas camadas recebem em sua entrada, além da saída das camadas de atenção anteriores, as saídas diretas dos blocos de codificação e, dessa forma, "prestam atenção"tanto na sequência de saída quanto a de entrada (Vaswani et al., 2017). A consequência dessa arquitetura é que o processo de codificação transforma as sequências de entrada em representações abstratas, que incorporam informações de todo o contexto, enquanto a decodificação utiliza essas representações para gerar novos termos de saída, condicionando cada novo elemento tanto às informações abstraídas da entrada quanto ao contexto já produzido pela própria sequência de saída (Vaswani et al., 2017).

2.3 ANÁLISE MULTIMODAL

A análise multimodal refere-se à integração de informações provenientes de diferentes modalidades de dados, como texto, imagem, áudio e vídeo. Esse tipo de análise reconhece que informações relevantes podem estar distribuídas entre diferentes categorias de dados, sendo que cada uma contribui com aspectos únicos e complementares para a compreensão completa do conteúdo (Baltrusaitis; Ahuja; Morency, 2017). Utilizando o caso do presente estudo como exemplo, em um documento acadêmico, o texto isoladamente pode parecer legítimo, no entanto, quando combinado com inconsistências visuais, pode revelar sinais de falsificação, e vice-versa.

No contexto de aprendizado de máquina, o processo conhecido como fusão multimodal consiste em combinar as diferentes representações de tipos de dados em uma unificada, possibilitando que os modelos consigam aprender as características mais relevantes em diferentes domínios de forma prática e eficiente, o que resulta em representações mais ricas e discriminativas (Baltrusaitis; Ahuja; Morency, 2017; Ahmad; Nurtanio; Zainuddin, 2024). Dessa forma, as estratégias de fusão são normalmente classificadas em quatro categorias

principais, dependendo do momento em que ocorrem: fusão precoce, onde as representações brutas são combinadas e servem de entrada para o modelo de aprendizado; fusão intermediária, onde as representações são incorporadas durante as etapas de aprendizado, como em camadas intermediárias de uma rede neural; fusão tardia, onde as decisões de classificadores de modalidades distintas são combinadas; e fusão híbrida, que combina diferentes estratégias de fusão (Baltrusaitis; Ahuja; Morency, 2017).

A metodologia deste trabalho utilizará a fusão precoce para combinar os dados extraídos, representados em vetores, a partir das modalidades visuais e textuais. Para isso, Yang et al. (2019) e Ahmad; Nurtanio; Zainuddin (2024) apresentam diferentes técnicas:

- Concatenação básica: abordagem mais simples e direta. Consiste na concatenação simples dos vetores de dados;
- Adição elemento a elemento: consiste na soma direta entre os respectivos elementos de cada modalidade;
- Transformação bilinear: consiste no produto matricial entre os vetores;
- Soma ponderada (*gated summation*): utiliza uma rede neural com funções de ativação específicas, como *weighted sigmoid gate units*, para somar os elementos de forma ponderada, como se o modelo passasse a entender a importância de cada fonte de informação;
- Produto ponderado: semelhante à soma ponderada utiliza uma rede neural, mas emprega mecanismos de atenção para o aprendizado, dessa forma, o resultado é o produto matricial entre os vetores, ponderado por uma matriz de pesos.

Não existe uma técnica ótima para todos os casos, visto que são subjetivas às características específicas dos domínios ao qual se deseja a união e, assim, a escolha pela melhor abordagem é geralmente determinada pela análise empírica dos resultados (Yang et al., 2019).

Além disso, o emprego da fusão multimodal é geralmente associado a outras estratégias auxiliares, que podem tratar tanto os vetores de representação antes da fusão quanto o resultante. Por exemplo, diferentes modalidades possuem características estatísticas e distribuições distintas, dessa forma, pode ser necessário pré-processamento para normalização das escalas dos dados. Da mesma forma, o vetor resultante, por juntar dados muitas vezes situados em espaços de alta dimensionalidade, precisa da aplicação de técnicas de redução dimensional para evitar problemas relacionados à *maldição da dimensionalidade*, termo cunhado por Bellman (1957), que explica que ao lidar com dados em espaços de alta dimensão, sua amostragem e modelagem torna-se progressivamente mais esparsa e difícil dado que o volume do espaço cresce exponencialmente.

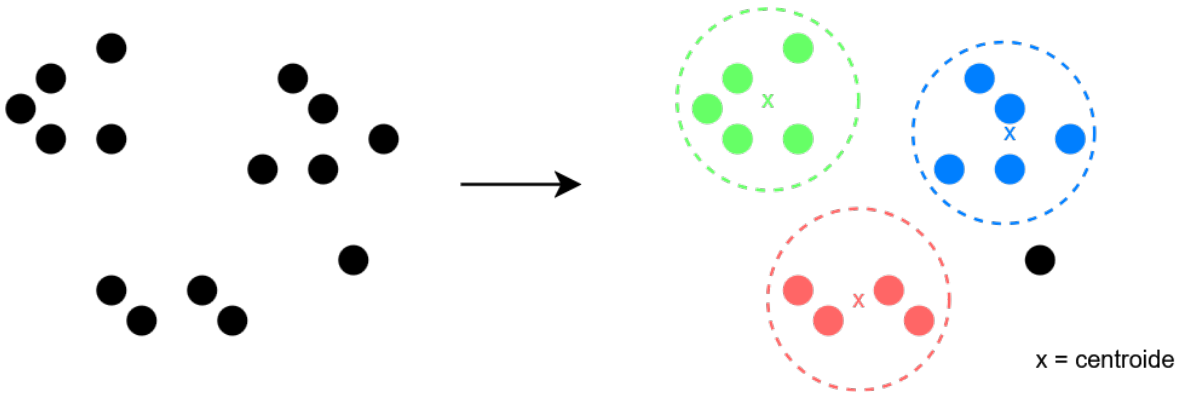
2.4 ALGORITMOS DE AGRUPAMENTO

No contexto de aprendizado de máquina, as técnicas de agrupamento, ou *clustering*, têm por objetivo a organização de uma coleção de dados em grupos, ou *clusters*, de forma que os elementos dentro de um mesmo grupo tenham muito mais similaridades entre si do que entre elementos em outros grupos. Essas técnicas geralmente são utilizadas quando não se tem conhecimento sobre como os dados podem ser categorizados, dessa forma, tradicionalmente são associadas à aprendizagem não supervisionada (Grira; Crucianu; Boujemaa, 2005). Assim, têm como propósito descobrir estruturações, relações, associações ou hierarquias ocultas dentro da coleção, de maneira a proporcionar melhor compreensão sobre os dados e seus processos subjacentes de criação (Fullér; Károly; Galambos, 2018).

Como trata-se de uma larga área de estudo, os algoritmos de *clustering* variam significativamente em sua compreensão do que constitui um *cluster* e de como este pode ser encontrado de forma eficiente, dessa forma, noções populares incluem grupos com pequenas distâncias entre membros, áreas densas do espaço de dados, intervalos ou distribuições estatísticas específicas (Fullér; Károly; Galambos, 2018). Por fins de brevidade, aqui explicam-se os métodos de agrupamento particional – que buscam criar grupos a partir do particionamento único da coleção de dados, de forma que um elemento não pertença a mais de um grupo (Grira; Crucianu; Boujemaa, 2005) –, em especial, algoritmos baseados em distância e em densidade.

Para os algoritmos baseados em distância, cada *cluster* é representado por um protótipo e a atribuição de elementos a estes é guiada pela minimização de alguma medida de dissimilaridade. Em termos mais concretos, tanto os *clusters* quanto os dados são tipicamente traduzidos como pontos em um espaço. Busca-se então a minimização de uma função de custo, como a soma das distâncias quadráticas entre cada elemento e um *cluster*. Isso acontece através do calculo iterativo da função de custo seguido do ajuste posicional do *cluster*, processo repetido alternadamente até que a valor de custo fique abaixo de um limiar determinado, garantindo a convergência do algoritmo, ou seja, a posição ótima do agrupamento (Grira; Crucianu; Boujemaa, 2005; Fullér; Károly; Galambos, 2018).

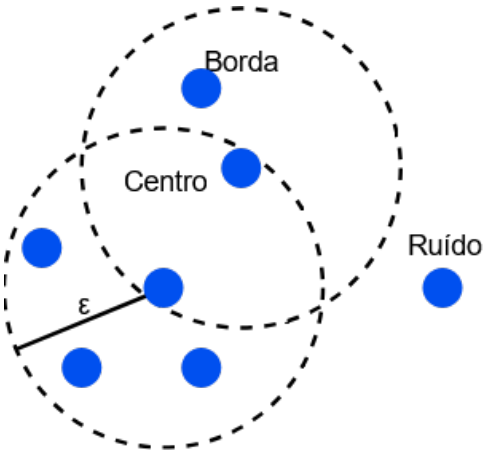
Figura 10 – Representação de um Algoritmo de *Clustering* por Distância



Fonte: o autor

Os algoritmos baseados em densidade consideram que *clusters* são regiões de alta densidade de dados separadas por áreas de baixa densidade, assim, tanto o formato dos agrupamentos quanto a distribuição dos elementos podem ser feitos de forma arbitrária. O algoritmo DBSCAN pode ser utilizado para exemplificar esse conceito: definem-se os parâmetros ϵ (raio de vizinhança) e $MinPts$ (número mínimo de pontos ou elementos), assim, um ponto considera outro como vizinho quando está dentro de ϵ. Então, conta-se o número de vizinhos para cada ponto: aqueles com vizinhança densa – isto é, número de vizinhos maior que $MinPts$ – tornam-se pontos centrais e formam núcleos de *clusters*; aqueles atingíveis por vizinhança dentro de um *cluster* são absorvidos e chamados de pontos de fronteira; e aqueles pouco densos e não atingíveis são classificados como ruído. (Grira; Crucianu; Boujemaa, 2005).

Figura 11 – Representação do Algoritmo DBSCAN



Fonte: o autor

2.4.1 Detecção de Anomalias

A detecção de anomalias é definida como o processo de identificar padrões em um conjunto de dados cujo comportamento difere significativamente do esperado, assim, pode ser tratado como um problema de identificação de *outliers* ou ruídos em algoritmos de agrupamento. Quando identificadas, embora nem sempre representem atividades maliciosas, frequentemente indicam eventos de interesse que merecem investigação (Agrawal; Agrawal, 2015; Fullér; Károly; Galambos, 2018).

A metodologia típica segue três etapas principais: parametrização e pré-processamento, onde os dados são normalizados em formatos comuns; treinamento, onde são construídos modelos de comportamento normal; e finalmente, detecção, em que uma nova observação é comparada aos modelos construídos (Agrawal; Agrawal, 2015). Dessa forma, a detecção de anomalias baseada em *clustering* funciona estabelecendo modelos de normalidade através do agrupamento de dados e, posteriormente identificando novos elementos que apresentam desvios expressivos desses modelos de referência.

3 TRABALHOS CORRELATOS

As subseções seguintes apresentam os mais recentes trabalhos com temática ou abordagem semelhante.

3.1 VISÃO GERAL

Durante a pesquisa por bibliografia que trata especificamente do problema da identificação de documentos falsificados, foram encontrados poucos resultados, sobretudo no âmbito acadêmico, já que a maioria dos trabalhos com temática similar aborda a classificação de fraudes – dificuldade ilustrada pela Tabela 1, que apresenta a bibliografia obtida que trata desse tema. É importante distinguir que, a detecção de fraudes foca em adulterações de arquivos originais, como a mudança de notas, datas ou nomes, enquanto a de documentos falsificados busca identificar aqueles completamente forjados desde sua criação, sem terem sido emitidos por instituições oficiais, por exemplo. Isso não significa que as ideias e técnicas não possam ser aproveitadas e adaptadas entre um contexto e outro, pelo contrário, este trabalho de conclusão de curso tem como referência métodos nos dois domínios.

Tabela 1 – Bibliografia Pesquisada com Temas Abordados

Artigos de Detecção de Fraude	Aborda Detecção de Falsificação
Kim (2022)	✓
Mohammed; Nwobodo; Ekene (2024)	
Jaiswal; Sharma; Yadav (2022)	
Boonkrong (2024)	
Ahmad; Nurtanio; Zainuddin (2024)	✓
Moolchandani; Pakshwa; Singh (2024)	
James; Gupta; Raviv (2020)	
Alameri et al. (2023)	✓

Nesta área, é predominante o emprego de estratégias de visão computacional, como o artigo de Jaiswal; Sharma; Yadav (2022), que utiliza *autoencoders* convolucionais para a extração de características em imagens hiperespectrais, focando em identificar incompatibilidade entre tintas. A análise de imagens é frequentemente combinada com técnicas complementares para melhorar a robustez da detecção: Alameri et al. (2023) propõem uma abordagem não supervisionada que utiliza correlações entre espectros de materiais dos documentos para gerar redes ponderadas, aplicando algoritmos de *clustering* para identificar padrões anômalos; James; Gupta; Raviv (2020) introduziram outra perspectiva ao reformular o problema como comparação de grafos, em que obtém, via OCR, caixas deli-

mitadoras de tamanho entre caracteres, utilizando-as para o treinamento de classificadores que detectam a manipulação de *pixels*.

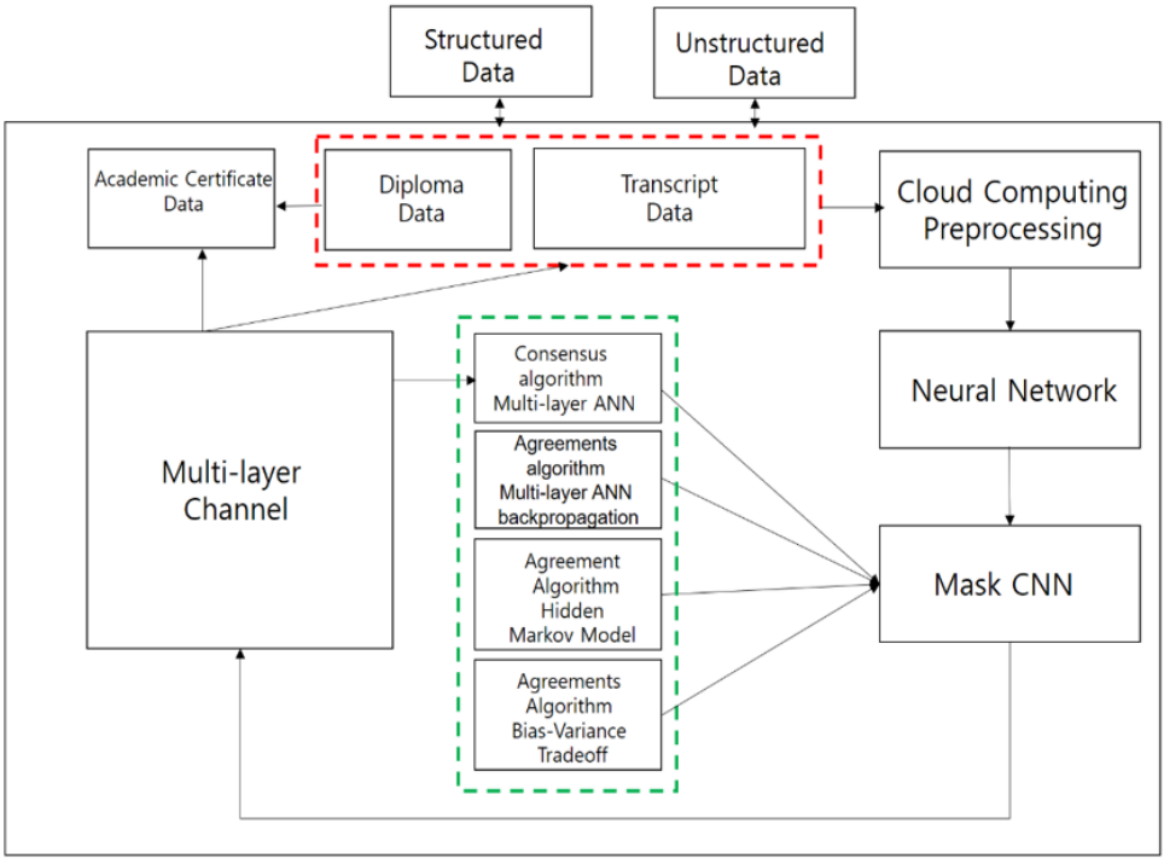
Alternativamente, também existem propostas que abordam a prevenção de fraudes através de outras tecnologias, como Boonkrong (2024), que propõe o emprego de funções criptográficas para detectar modificações em documentos previamente submetidos, em que são armazenados os valores de *hash* dos arquivos originais e legítimos, de forma que validações posteriores possam ser comparadas com o certificado primário. Contudo, essas abordagens preventivas não lidam com a classificação de documentos falsificados em sua concepção, representando uma lacuna pouco explorada, que o presente estudo visa preencher. Em sequência seguem os trabalhos que guiaram a concepção da estratégia da presente pesquisa.

3.2 INTEGRANDO APRENDIZADO DE MÁQUINA À *BLOCKCHAIN* PARA PREVENÇÃO E DETECÇÃO DE FRAUDES EM DIPLOMAS

Para o problema de prevenção de fraudes de diplomas, o artigo de Kim (2022) propõe uma *blockchain* que incorpora algoritmos de aprendizado de máquina em diversas partes do processo de verificação de documentos e de consenso da rede. Além disso, de forma semelhante ao trabalho de Boonkrong (2024), quando um certificado é aceito, seu *hash* é calculado e integrado ao seu registro na rede, permitindo sua verificabilidade, de forma que qualquer adulteração seja facilmente detectada.

O autor apresenta o fluxo para submissão de um diploma na *blockchain* em quatro etapas principais, conforme Figura 12.

Figura 12 – Representação da Arquitetura de Validação de Diplomas por Kim



Fonte: (Kim, 2022)

A primeira etapa, "*Cloud Computing Preprocessing*", consiste na entrada e pré-processamento do documento digitalizado e, através de um serviço em nuvem, normaliza os dados brutos da imagem, extrai metadados – como resolução, dimensões e contraste –, aplica correções de perspectiva e cria uma versão virtualizada semi-persistente do arquivo, preparada para ser processada pelas etapas conseguintes, rejeitando entradas com formato e qualidade inconsistentes.

Em seguida, as imagens passam pela etapa "*Neural Network*", que utiliza *Faster R-CNN* para rapidamente filtrar e detectar artefatos visuais suspeitos. Através de sua rede de propostas regionais (*Region Proposal Network*), o algoritmo captura regiões de interesse – como selos da universidade, assinaturas, marcas d'água e outros padrões – e cria escores de confiança para cada uma, identificando áreas possivelmente adulteradas.

Os documentos passam então para a etapa "*Mask CNN*", que através de um modelo *Mask R-CNN* e das regiões de interesse previamente identificadas, segmenta a imagem, criando máscaras binárias a nível de *pixel*, ou seja, para cada região, o algoritmo estima quais *pixels* pertencem ao document legítimo e quais foram forjados. Essas máscaras são então encapsuladas como prova imutável dentro do bloco que será registrado na blockchain,

além de servir como outra medida de escore de confiança do documento.

Por fim, ambas as pontuações de confiança são combinadas e passadas ao mecanismo de consenso dessa *blockchain*, representado pela etapa "*Multi-layer Channel*"que, ao invés de utilizar estratégias tradicionais como prova de trabalho ou prova de participação, combina múltiplos algoritmos de aprendizado de máquina. De forma geral, essa arquitetura é composta por quatro componentes principais:

- Um algoritmo baseado em uma rede neural multicamadas, que processa os escores de confiança fornecidos pelos processamentos anteriores e, a partir de certo limiar de confidência, imediatamente aprova o documento;
- Um algoritmo de aprendizado complementar à rede neural multicamadas, que cruza referências com padrões aprendidos de decisões anteriores – por exemplo, quando um diploma inicialmente validado como autêntico posteriormente se prova fraudulento – e ajusta os pesos da rede;
- Um algoritmo que utiliza um modelo oculto de Markov para, a partir de uma cadeia de regras, examinar padrões temporais e fornecer avaliações probabilísticas da autenticidade do documento;
- Um algoritmo que equilibra a complexidade (*overfitting*) com a generalização (*underfitting*) do modelo de rede neural, ou seja, procura encontrar *trade-off* ótimo entre viés e variância para decisões de consenso confiáveis.

Assim, cada nó da *blockchain* executa esses algoritmos e vota, com base na ponderação dos resultados, se devem incluir ou rejeitar o bloco com o diploma. Para qualquer decisão, exige-se quórum de pelo menos 2/3 de votos. Caso não seja atingido, seja por discordâncias das máscaras ou pela recusa de validadores de alta reputação, uma nova rodada de votação é iniciada com a reexecução das etapas "*Mask CNN*"e "*Multi-layer Channel*"com parâmetros ajustados. Esse ciclo se repete até obter consenso ou direcionar o diploma a uma auditoria humana. Dessa forma, quando um documento é aprovado na rede, é classificado como autêntico no *ledger*; quando reprovado, é sinalizado como fraudulento.

Por fim, quando um terceiro – como empresa, universidade ou empregador – deseja verificar a validade de um diploma já registrado, basta a verificação do *hash* do documento já submetido.

3.3 APLICANDO ANÁLISE MULTIMODAL A DOCUMENTOS

O trabalho de Jain; Wigington (2019) não lida diretamente com a identificação de documentos falsificados ou fraudados, mas sim do problema geral de classificação de imagens. No entanto, a abordagem utilizada pelos autores é altamente relevante, pois

mostra a eficácia da análise multimodal e pode ser aproveitada por este trabalho de conclusão de curso.

O *paper* propõe uma abordagem multimodal para a classificação de imagens de documentos diversos em dezesseis categorias utilizando o *dataset* RVL-CDIP. A proposta combina a fusão de características visuais e textuais para a rotulagem entre imagens e classes. Para isso, segue o *pipeline*:

1. Pré-processamento: normaliza e redimensiona as imagens para utilizações posteriores;
2. Extração de texto: utiliza OCR para extrair texto das páginas;
3. Extração multimodal, em paralelo:

   Modalidade textual: utiliza um modelos de linguagem para capturar informações semânticas dos textos extraídos;

   Modalidade visual: extrai características das imagens a partir de uma rede convolucional;
4. Fusão multimodal: combina as extrações textuais e visuais;
5. Classificação final: utiliza uma rede convolucional, que tem como entrada a fusão multimodal, para classificar os documentos.

Para a modalidade textual, os autores trazem à tona o problema de que texto extraído por OCR pode ser muito ruidoso, contendo erros a nível de caracteres ou até palavras. Por isso, capturam representações do conteúdo em três diferentes granularidades.

A nível de sequência, empregam ULMFiT (*Universal Language Model Fine-tuning*), um modelo baseado em redes LSTM que, como abordado durante a fundamentação teórica, processa o documento como uma sequência de palavras e mantém uma "memória interna", que armazena informações contextuais conforme processa cada palavra sequencialmente. Isso permite que a rede neural capture sequências lógicas, dependências de longo prazo e contexto semântico entre palavras distantes. Como saída, o algoritmo produz uma representação vetorial do texto que leva em consideração as características citadas.

A nível de palavra, para representar cada uma, empregam FastText *embeddings*. Uma técnica de *embedding* de termos, também previamente exposto, que consiste em transformar os termos em vetores numéricos, de forma que palavras com significados similares fiquem próximas no espaço matemático. O vetor final do documento é calculado como a média dos *embeddings* de todas as palavras presentes.

A nível de caractere, para capturar padrões ortográficos, aplicam N-gramas de caracteres – sequências contínuas de n caracteres abduzidos de uma palavra – e criam um vetor numérico normalizado das ocorrências dos padrões obtidos.

Em resumo, as características de sequência preservam o contexto semântico geral do documento, as representações de palavra mantêm similaridades semânticas locais, e os

N-gramas de caracteres oferecem robustez contra erros de OCR e palavras desconhecidas. Essas três representações são combinadas através de um método *ensemble*, que produz um vetor unificado que comporta essas *features* textuais.

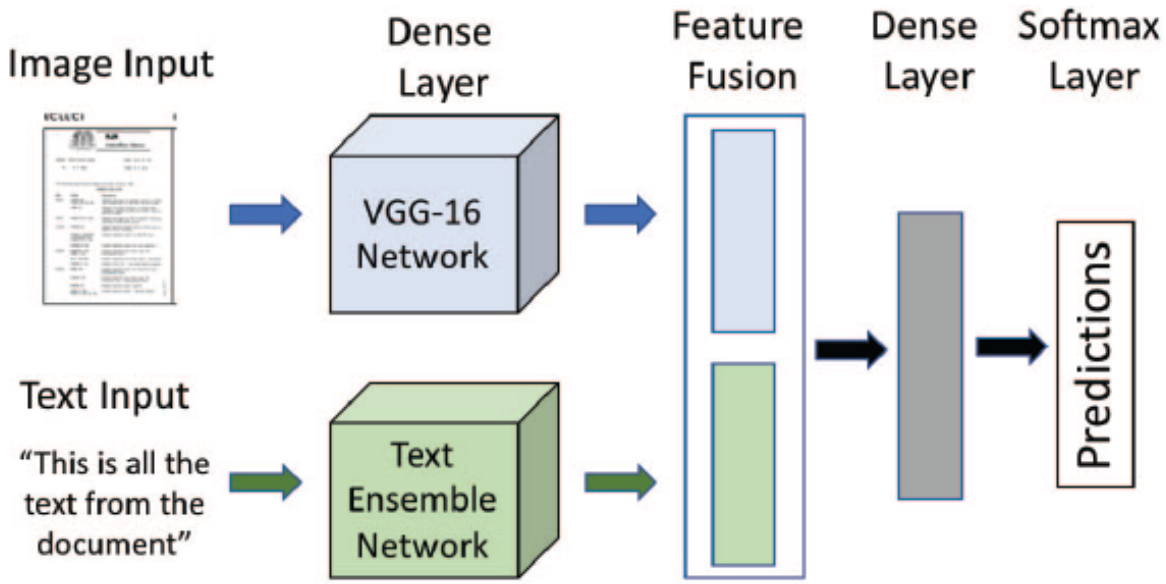
Para a modalidade visual, o trabalho emprega a arquitetura de rede VGG-16 (*Visual Geometry Group*) para extrair características hierárquicas através de suas camadas convolucionais, capturando padrões de layout, tipografia, elementos gráficos e padrões de formatação presentes nos documentos. Como saída, a rede produz um vetor multidimensional que representa uma codificação densa e compacta de todas as informações visuais relevantes do documento.

O principal interesse desse trabalho é a fusão das informações textuais e visuais, que tem por objetivo criar uma representação unificada, que preserve e potencialize as informações complementares de ambas as modalidades, e que permita que um modelo de classificação final explore sinergias entre essas diferentes características. Os autores propõem duas estratégias principais para combinar essas representações, das quais destaca-se a segunda, que combina os vetores anteriormente extraídos.

Essa abordagem explora quatro métodos distintos de junção. De forma geral, o primeiro é a concatenação simples, onde os vetores de características textuais e visuais são diretamente concatenados para formar um vetor unificado. O segundo método utiliza adição elemento a elemento, somando diretamente as representações de ambas as modalidades. O terceiro emprega *compact bilinear pooling*, uma técnica mais sofisticada que calcula o produto externo entre os vetores de características para capturar interações complexas entre as modalidades, permitindo que o modelo de classificação posterior aprenda correlações não-lineares entre informações visuais e textuais. Por fim, o quarto método implementa *multimodal gated units*, que utilizam mecanismos de atenção para aprender uma função de controle que determina automaticamente como ponderar e combinar as características de cada modalidade, o que permite que o modelo posterior adapte dinamicamente a importância relativa de informações visuais ou textuais dependendo do contexto específico do documento – para documentos altamente textuais como contratos ou relatórios científicos, as características semânticas podem ser mais discriminativas, enquanto para documentos com layouts visuais distintivos como formulários ou apresentações, as características visuais podem ser mais relevantes.

Em sequência, para realizar a classificação final, os autores utilizam uma camada densa e uma camada final *softmax* para a predição, como ilustra a Figura 13.

Figura 13 – Representação da Arquitetura de Fusão Multimodal por Jain; Wigington



Fonte: (Jain; Wigington, 2019)

Finalmente, o artigo compartilha as conclusões finais, onde explicita como essa abordagem superou consistentemente os métodos que utilizam apenas uma modalidade, com o método de adição elemento a elemento curiosamente alcançando os melhores resultados para a fusão multimodal, atingindo uma acurácia de 93,6% no *dataset* RVL-CDIP.

3.4 TRATANDO A AUTENTICAÇÃO DE DOCUMENTOS ACADÊMICOS COMO UM PROBLEMA DE AGRUPAMENTO

O trabalho de Mohammed; Nwobodo; Ekene (2024) lida diretamente com o problema de verificação da autenticidade de certificados acadêmicos. Para isso, propõe uma abordagem baseada em *clustering*, fundamentado na premissa de que documentos legítimos apresentam padrões consistentes de características que podem ser identificados através desses agrupamentos.

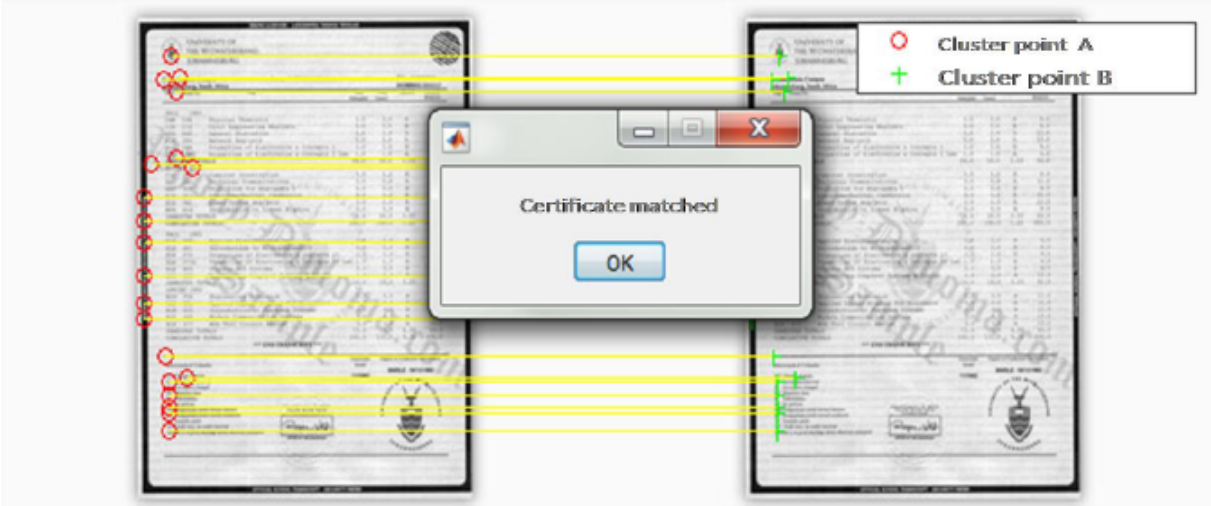
O *dataset* utilizado pelos autores consiste em mais de vinte e quatro mil amostras de documentos não rotulados, oficialmente emitidos por duas universidades: Enugu State University of Science and Technology e University of California Irvine.

A metodologia dos autores utiliza o algoritmo K-means como técnica principal para descobrir os padrões dominantes em certificados acadêmicos. Dessa forma, o processo de treinamento consiste na aplicação desse algoritmo sobre as características extraídas dos documentos do *dataset*, com o objetivo de agrupá-los em dezesseis grupos. O algoritmo inicializa centroides e atribui iterativamente cada arquivo ao centroide mais próximo usando um modelo de equidistância. Os centroides são atualizados através do cálculo das médias dos pontos pertencentes a cada *cluster* até atingir convergência. Para a verificação

de documentos, o sistema extrai suas características e calcula distâncias em relação aos centroides estabelecidos, classificando-o como legítimo, quando próximo de algum padrão conhecido, ou suspeito, quando distante de todos os *clusters*.

Embora os autores não especifiquem os processos de extração de características, as figuras apresentadas sugerem fortemente o uso de *features* visuais. As imagens, como exemplificado na Figura 14, mostram marcações circulares em pontos específicos dos certificados, destacando elementos como logos institucionais e aspectos estruturais dos documentos. Assume-se, portanto, que o sistema captura e converte essas informações em vetores numéricos compatíveis com o K-means.

Figura 14 – Resultado da Verificação de um Documento Autêntico



Fonte: (Mohammed; Nwobodo; Ekene, 2024)

O modelo foi validado experimentalmente com certificados reais e demonstrou capacidade de distinguir documentos autênticos de falsificados, já que os resultados reportaram uma acurácia geral de 86,53%. Os autores destacam como principal vantagem a capacidade do sistema de operar efetivamente com *datasets* limitados, característica importante em cenários reais onde a disponibilidade de dados rotulados é restrita por questões éticas e de privacidade.

4 METODOLOGIA

Este capítulo define, de forma preliminar, a metodologia que será seguida durante o desenvolvimento do trabalho e está sujeita a mudanças conforme coleta e análise dos documentos acadêmicos.

4.1 VISÃO GERAL DA METODOLOGIA

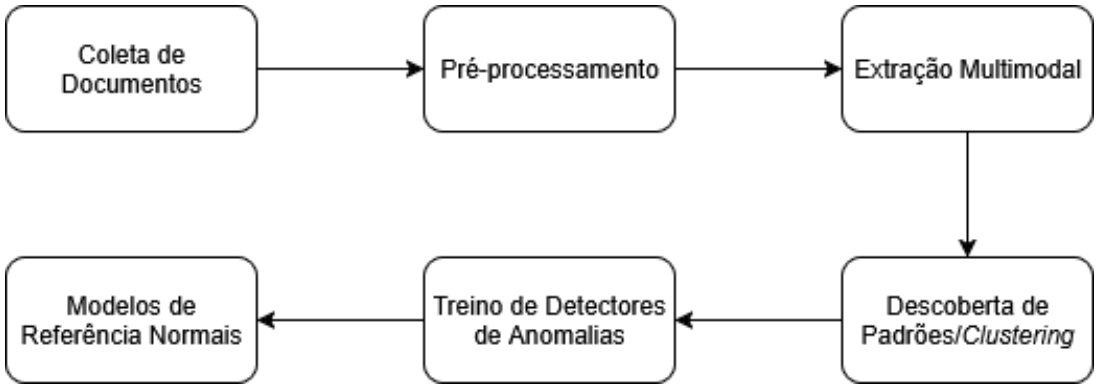
A metodologia proposta tem como base uma abordagem de aprendizado não-supervisionado e detecção de anomalias através da análise e extração multimodal. Busca-se rotular documentos com base em um nível de probabilidade de fraudulência, para isso, utilizam-se extrações de características visuais (como textura, fonte, espaçamento, selos e assinaturas), textuais (como padrões linguísticos, formatação de números e distribuição de termos) e estruturais (como posição de campos, margens e tabelas), que serão refinadas conforme realização do TCC. Combinando essas *features* multidimensionais, é possível realizar o agrupamento dos documentos em *clusters* que representam padrões dominantes normais. Em sequência, modelos de detecção de anomalias são utilizados para a criação de detectores de referência a partir dos *clusters*, possibilitando a classificação de um novo documento submetido, em tempo real, através da avaliação do grau de desvio em relação aos padrões aprendidos – quanto maior o desvio e escore de anomalia, maior a probabilidade de que o documento seja falsificado. Finalmente, essa pontuação é mapeada para categorias discretas de suspeita, fornecendo um nível de probabilidade de fraude para cada inserção. O processo completo consiste em duas etapas: treinamento dos modelos de referência e classificação de novos documentos.

A escolha dessa abordagem tem por base a premissa de que documentos falsificados apresentam inconsistências sutis, tornando-os atípicos em relação aos padrões estabelecidos por documentos legítimos, sendo detectáveis através da análise multimodal das características extraídas de diversos contextos.

4.1.1 Treinamento dos Modelos de Referência

Representada na Figura 15, a fase de treinamento inicia com a coleta de certificações acadêmicas diversas, seguida do pré-processamento através de técnicas de normalização de imagens e aplicação de OCR. Com o *dataset* formado, é realizada a extração e processamento multimodal de características visuais, textuais e estruturais dos documentos. Em sequência, com base nos dados obtidos na etapa anterior, é realizada a identificação de padrões utilizando algoritmos de *clustering* para identificar grupos de documentos com comportamentos similares, estabelecendo padrões dominantes de normalidade. Por fim, detectores de anomalias são treinados para cada padrão descoberto, gerando modelos de referência normais.

Figura 15 – Representação do Fluxo de Treino

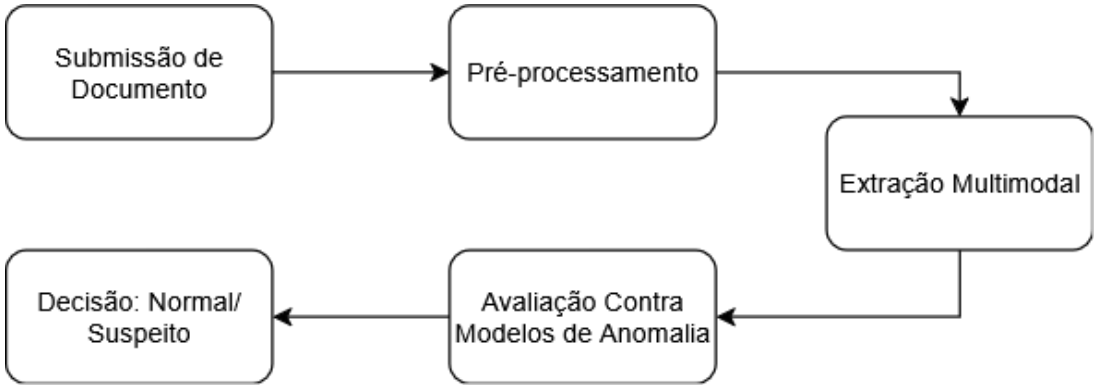


Fonte: o autor

4.1.2 Classificação de Novo Documento

Representada na Figura 16, o processo de classificação de novo documento utiliza os modelos de referência estabelecidos na fase de treinamento para determinar a probabilidade de falsificação.

Figura 16 – Representação do Fluxo de Análise de Novo Documento



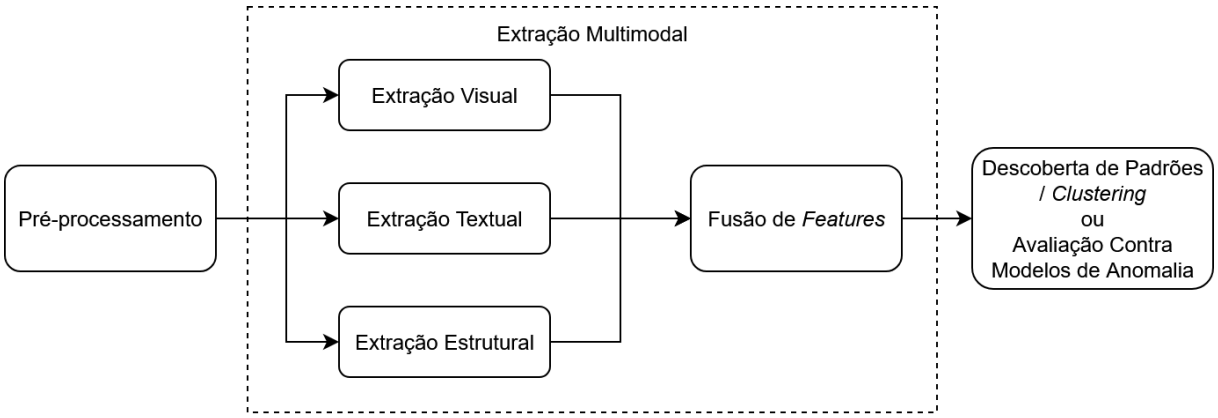
Fonte: o autor

O processo inicia com a submissão de um novo certificado e, para garantir consistência na representação das características, passa pelo mesmo *pipeline* de pré-processamento e extração multimodal utilizado na fase de treinamento. Em seguida, os dados de representação do documento, obtidos na etapa anterior, são comparados contra todos os modelos de referência normal. Cada modelo calcula um escore de anomalia baseado na distância, ou similaridade, em relação ao padrão estabelecido pelo modelo. Essas pontuações representam, por fim, a probabilidade de falsificação do registro. Finalmente, utilizam-se métricas de consenso para categorizar o arquivo, isto é, classificá-lo como normal ou suspeito a partir de determinado limiar de pontos.

4.1.3 Extração Multimodal de Características

O módulo de extração multimodal, utilizado tanto no fluxo de treino quanto no fluxo de classificação de uma submissão, é responsável por capturar diferentes aspectos dos documentos. Essa abordagem permite aproveitar o mesmo *pipeline* de processamento combinando características independentes e, no contexto deste trabalho, complementares. Busca-se poder detectar tanto falsificações grosseiras quanto sofisticadas, uma vez que mesmo contrafações bem-feitas tendem a apresentar inconsistências sutis.

Figura 17 – Representação do Fluxo de Extração Multimodal de Características



Fonte: o autor

Ao invés de focar em características de domínios específicos, como nos trabalhos correlatos, essa abordagem combina três diferentes subprocessos de extração de *features* em paralelo, como representado na Figura 17:

- Extração visual: extrai características ligadas ao layout, qualidade e consistência visual dos documentos. Inclui análise de textura, propriedades de fonte (espessura, tamanho, espaçamento), qualidade de assinaturas e selos, resolução de imagem, e padrões de cores e contrastes;
- Extração textual: utiliza modelos de processamento de linguagem natural para extrair características linguísticas e de formatação. Analisa padrões textuais, distribuição de termos, consistência na formatação de números, datas e códigos, além de verificar a coerência semântica do conteúdo;
- Extração estrutural: extrai características ligadas à organização espacial e estrutural dos documentos. Examina posicionamento de campos, formatação de tabelas, alinhamentos, margens, espaçamentos e a disposição geral dos elementos no documento.

Por fim, é realizada a fusão das características extraídas em todos os subprocessos. Para isso, os dados são normalizados e são aplicadas técnicas de redução dimensional,

evitando a *maldição da dimensionalidade*, o que resulta em uma representação completa, unificada e compacta de cada documento.

5 PRÓXIMOS PASSOS

5.1 CRONOGRAMA

A seguir, segue o cronograma planejado para a próxima etapa do projeto e a disciplina de Trabalho de Conclusão de Curso 2.

Tabela 2 – Cronograma para TCC2.

Etapas	2025				
	Ago	Set	Out	Nov	Dez
Obtenção e análise dos documentos acadêmicos	X	X			
Desenvolvimento do *software*		X	X	X	
Escrita da monografia				X	X
Entrega TCC II					X
Defesa pública					X

REFERÊNCIAS

AGRAWAL, Shikha; AGRAWAL, Jitendra. Survey on Anomaly Detection using Data Mining Techniques. **Procedia Computer Science**, v. 60, p. 708–713, 2015. ISSN 1877-0509. DOI: 10.1016/j.procs.2015.08.220.

AHMAD, Muh; NURTANIO, Ingrid; ZAINUDDIN, Zahir. Diploma Verification Through Multi-Modal Image Processing: Face Detection, Perforation Text Recognition, and System Architecture Evaluation. In: p. 73–78. DOI: 10.1109/ICOIACT64819.2024.10913411.

ALAMERI, Mohammed et al. Unsupervised Forgery Detection of Documents: A Network-Inspired Approach. **Electronics**, v. 12, abr. 2023. DOI: 10.3390/electronics12071682.

ALZUBAIDI, Laith et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. **Journal of Big Data**, v. 8, n. 1, p. 53, mar. 2021. ISSN 2196-1115. DOI: 10.1186/s40537-021-00444-8. Acesso em: 27 jun. 2025.

BALTRUSAITIS, Tadas; AHUJA, Chaitanya; MORENCY, Louis-Philippe. Multimodal Machine Learning: A Survey and Taxonomy. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, PP, maio 2017. DOI: 10.1109/TPAMI.2018.2798607.

BELLMAN, Richard E. **Dynamic Programming**. Princeton, NJ, USA: Princeton University Press, 1957. (Rand Corporation research study). ISBN 9780691079516.

BOONKRONG, Sirapat. Design of an academic document forgery detection system. **International Journal of Information Technology**, p. 1–13, jun. 2024. DOI: 10.1007/s41870-024-02006-6.

BRASIL. **Censo da Educação Superior 2023: notas estatísticas**. Brasília, DF: Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), 2024.

DIAS, Pâmela; LEAL, Arthur. **Sites vendem diploma de curso superior para quem sequer pisou em sala de aula: 'Documentação 100% original, emitida de dentro da universidade', diz atendente**. 2022. Disponível em: https://oglobo.globo.com/brasil/noticia/2022/11/sites-vendem-diploma-de-curso-superior-para-pessoas-que-nao-concluiram-ou-sequer-pisaram-em-uma-universidade.ghtml. Acesso em: 5 abr. 2025.

DIETTERICH, Thomas G. Machine learning. In: ENCYCLOPEDIA of Computer Science. GBR: John Wiley e Sons Ltd., 2003. p. 1056–1059. ISBN 0470864125.

FANTÁSTICO. **Quatro pessoas são presas pela venda de 50 mil diplomas falsos e milhares carteirinhas de estudante**. 2025. Disponível em: https://oglobo.globo.com/brasil/noticia/2022/11/sites-vendem-diploma-de-curso-superior-para-pessoas-que-nao-concluiram-ou-sequer-pisaram-em-uma-universidade.ghtml. Acesso em: 5 abr. 2025.

FULLÉR, Róbert; KÁROLY, Artúr István; GALAMBOS, Péter. Unsupervised Clustering for Deep Learning: A tutorial survey. **Acta Polytechnica Hungarica**, v. 15, p. 29–53, dez. 2018. DOI: 10.12700/APH.15.8.2018.8.2.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep Learning**. [S.l.]: MIT Press, 2016. http://www.deeplearningbook.org. Acesso em: 16 jun. 2025.

GRIRA, Nizar; CRUCIANU, Michel; BOUJEMAA, Nozha. Unsupervised and Semi-supervised Clustering: a brief survey. **A Review of Machine Learning Techniques for Processing Multimedia Content**, set. 2005.

HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long Short-Term Memory. **Neural Computation**, v. 9, p. 1735–1780, nov. 1997. DOI: 10.1162/neco.1997.9.8.1735.

ISLAM, Noman; ISLAM, Zeeshan; NOOR, Nazia. A Survey on Optical Character Recognition System. **ITB Journal of Information and Communication Technology**, dez. 2016. DOI: 10.48550/arXiv.1710.05703.

JAIN, Rajiv; WIGINGTON, Curtis. Multimodal Document Image Classification. **2019 International Conference on Document Analysis and Recognition (ICDAR)**, p. 71–77, 2019. DOI: 10.1109/ICDAR.2019.00021.

JAISWAL, Garima; SHARMA, Arun; YADAV, Sumit. Deep feature extraction for document forgery detection with convolutional autoencoders. **Computers & Electrical Engineering**, v. 99, p. 107770, abr. 2022. DOI: 10.1016/j.compeleceng.2022.107770.

JAMES, Hailey; GUPTA, Otkrist; RAVIV, Dan. OCR Graph Features for Manipulation Detection in Documents, set. 2020. DOI: 10.48550/arXiv.2009.05158.

KIM, Seong-Kyu. Blockchain Smart Contract to Prevent Forgery of Degree Certificates: Artificial Intelligence Consensus Algorithm. **Electronics**, v. 11, p. 2112, jul. 2022. DOI: 10.3390/electronics11142112.

MEC. **Aplicativo do MEC ganha prêmio de reconhecimento nacional**. 2022. Disponível em: https://www.gov.br/mec/pt-br/assuntos/noticias/2022/aplicativo-do-mec-ganha-premio-de-reconhecimento-nacional. Acesso em: 13 maio 2025.

MEC. **Portaria MEC/DAU n 33 de 2 de agosto de 1978**: Estabelece a sistemática para o registro de diplomas de curso superior. Brasília, DF: Ministério da Educação do Brasil, 1978.

MENGHANI, Gaurav. Efficient Deep Learning: A Survey on Making Deep Learning Models Smaller, Faster, and Better. **ACM Computing Surveys**, Association for Computing Machinery (ACM), v. 55, n. 12, p. 1–37, mar. 2023. ISSN 1557-7341. DOI: 10.1145/3578938.

MOHAMMED, Shamsudeen; NWOBODO, Lois; EKENE, Njoku. Certificate Fraud Verification Model Using Clustered-Based Classification Approach. **Exploremetics**

Journal of Innovative Engineering and Technology, v. 5, n. 1, p. 60–72, jun. 2024. ISSN 2636-590. Disponível em: https://www.researchgate.net/publication/381164278_CERTIFICATE_FRAUD_VERIFICATION_MODEL_USING_CLUSTERED-BASED_CLASSIFICATION_APPROACH. Acesso em: 5 jul. 2025.

MOOLCHANDANI, Jhankar; PAKSHWA, Rinki; SINGH, Kulvinder. Machine Learning for Identifying and Validating Document Authenticity. In: 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT). [S.l.: s.n.], 2024. p. 1–9. DOI: 10.1109/ICCCNT61001.2024.10725983.

OTTER, Daniel; MEDINA, Julian; KALITA, Jugal. A Survey of the Usages of Deep Learning for Natural Language Processing. **IEEE Transactions on Neural Networks and Learning Systems**, PP, p. 1–21, abr. 2020. DOI: 10.1109/TNNLS.2020.2979670.

PALMA, Lucas M. et al. Blockchain and Smart Contracts for Higher Education Registry in Brazil. **International Journal of Network Management**, v. 29, 2019. DOI: https://doi.org/10.1002/nem.2061.

RNP, Rede. **Blockchain da jornada acadêmica**. 2023. Disponível em: https://www.youtube.com/watch?v=xqezMbjCeTM. Acesso em: 13 maio 2025.

SARKER, Iqbal. Machine Learning: Algorithms, Real-World Applications and Research Directions. **SN Computer Science**, v. 2, mar. 2021. DOI: 10.1007/s42979-021-00592-x.

SHI, Baoguang; BAI, Xiang; YAO, Cong. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, PP, jul. 2015. DOI: 10.1109/TPAMI.2016.2646371.

TEALAB, Ahmed. Time series forecasting using artificial neural networks methodologies: A systematic review. **Future Computing and Informatics Journal**, v. 3, n. 2, p. 334–340, 2018. ISSN 2314-7288. DOI: https://doi.org/10.1016/j.fcij.2018.10.003. Disponível em: https://www.sciencedirect.com/science/article/pii/S2314728817300715.

VASWANI, Ashish et al. Attention Is All You Need, jun. 2017. DOI: 10.48550/arXiv.1706.03762.

WANG, Haohan; RAJ, Bhiksha. On the Origin of Deep Learning, fev. 2017. DOI: 10.48550/arXiv.1702.07800.

YANG, Fan et al. Exploring Deep Multimodal Fusion of Text and Photo for Hate Speech Classification. In: p. 11–18. DOI: 10.18653/v1/W19-3502.

ZHANG, Hongzhi; SHAFIQ, M. Survey of transformers and towards ensemble learning using transformers for natural language processing. **Journal of Big Data**, v. 11, fev. 2024. DOI: 10.1186/s40537-023-00842-0.