CAPÍTULO 6

INTELIGÊNCIAS ARTIFICIAIS GENERATIVAS COM PYTHON: FUNDAMENTOS, APRENDIZADO DE MÁQUINA, REDES NEURAIS CLÁSSICAS E PROFUNDAS, TRANSFORMERS E ENGENHARIA DE PROMPT PARA GERAÇÃO DE IMAGENS

Data de submissão: 18/12/2024 *Data de aceite: 02/01/2025*

Márcio Mendonça
Universidade Tecnológica Federal do Paraná
PPGEM-CP - Programa de Pós-Graduação em Engenharia Mecânica PP/CP
Cornélio Procópio-PR
http://lattes.cnpq.br/5415046018018708

Guilherme Cyrino Geromel
Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - IFSP Piracicaba
Piracicaba-SP
http://lattes.cnpq.br/7535398878830738

Fabio Rodrigo Milanez
UniSENAI PR Campus Londrina
Londrina-PR
http://lattes.cnpq.br/3808981195212391

Angelo Feracin Neto
Universidade Tecnológica Federal do Paraná
Departamento Acadêmico de Engenharia Elétrica (DAELE)
Cornélio Procópio–PR
http://lattes.cnpq.br/0580089660443472

Marcos Antônio de Matos Laia
Universidade Federal de São Joao Del Rei
Departamento De Ciência Da Computação – UFSJ
Minas Gerais–MG
http://lattes.cnpq.br/7114274011978868

Marcos Banheti Rabello Vallim
Universidade Tecnológica Federal do Paraná
Departamento Acadêmico de Engenharia Elétrica (DAELE)
Cornélio Procópio–PR
http://lattes.cnpq.br/2326190172340055

Vitor Blanc Milani
Universidade Tecnológica Federal do Paraná - Mestrando - PPGEM-CP - Programa de Pós-Graduação em Engenharia Mecânica PP/CP
Cornélio Procópio-PR
http://lattes.cnpq.br/4504374098250296

Marta Rúbia Pereira dos Santos
Centro Estadual de Educação Tecnológica Paula Souza
Etec Jacinto Ferreira de Sá – Ourinhos
Ourinhos-SP
http://lattes.cnpq.br/3003910168580444

Vicente de Lima Gongora
UniSENAI PR Campus Londrina
Londrina-PR
http://lattes.cnpq.br/6784595388183195

Henrique Cavalieri Agonilha
Universidade Filadélfia (Unifil)
Londrina - PR
http://lattes.cnpq.br/9845468923141329

Pedro Henrique Calegari
Engenheiro Mecânico I Engenheiro de Segurança do Trabalho I Gerente de Projetos
Unopar Universidade Norte do Paraná – Gerente projetos Bosch Car Service
Jacarezinho-PR
http://lattes.cnpq.br/1239023712415204

Andressa Haiduk
Dimension Engenharia
Ponta Grossa - PR
http://lattes.cnpq.br/2786786167224165

Kazuyochi Ota Junior
Universidade Tecnológica Federal do Paraná
Mestre PPGEM-CP - Programa de Pós-Graduação em Engenharia Mecânica PP/CP
Cornélio Procópio - PR
http://lattes.cnpq.br/3845751794448092

Gabriel Henrique Oliveira Uliam
Egresso Universidade Tecnológica Federal do Paraná Departamento Acadêmico de Engenharia Elétrica (DAELE)
Cornélio Procópio - Pr
http://lattes.cnpq.br/9917773125320806

Fabio Nogueira de Queiroz
Centro Paula Souza
Departamento Computação-FATEC Ourinhos
Ourinhos – SP
http://lattes.cnpq.br/4466493001956276

Edinei Aparecido Furquim dos Santos
Governo do Paraná Secretaria de estado da Fazenda
Maringá – PR
http://lattes.cnpq.br/8706436030621473

Francisco de Assis Scannavino Junior
Universidade Tecnológica Federal do Paraná Departamento Acadêmico de Engenharia Elétrica (DAELE) – Cornélio Procópio - Pr
http://lattes.cnpq.br/4513330681918118

RESUMO: O texto apresenta uma visão geral do campo de inteligências artificiais generativas, com ênfase no uso de Python e na evolução de métodos e arquiteturas. Destaca conceitos fundamentais do aprendizado de máquina, passando das abordagens clássicas de redes neurais para modelos mais complexos de aprendizado profundo. Enfatiza o papel dos Transformers, originalmente voltados à linguagem, em tarefas visuais, bem como a importância do "*prompt engineering*" para direcionar e controlar a qualidade e o estilo das criações. Além disso, menciona exemplos práticos, códigos em Python, estudos de caso e aplicações concretas na geração de imagens. Por fim, o artigo conclui sugerindo futuros

estudos e aprofundamentos no tema.
PALAVRAS-CHAVE: Redes Neurais Artificiais*, Machine Learning,* Arquiteturas *e* Topologias de Redes Neurais.

GENERATIVE ARTIFICIAL INTELLIGENCES WITH PYTHON: FUNDAMENTALS, MACHINE LEARNING, CLASSICAL AND DEEP NEURAL NETWORKS, TRANSFORMERS AND PROMPT ENGINEERING FOR IMAGE GENERATION

ABSTRACT: The text provides an overview of the field of generative artificial intelligence, emphasizing the use of Python and the evolution of methods and architectures. It highlights fundamental machine learning concepts, moving from classical neural network approaches to more complex deep learning models. It underscores the role of Transformers—originally geared towards language—in visual tasks and the importance of "prompt engineering" to guide and control the quality and style of generated content. Additionally, it references practical examples, Python code, case studies, and concrete applications in image generation. In conclusion, the article suggests avenues for future research and further topic exploration.
KEYWORDS: Artificial Neural Networks, Machine Learning, Neural Network Architectures and Topologies

1 | INTRODUÇÃO

A inteligência artificial está cada vez mais presente no cotidiano de pessoas e empresas, notadamente em *chatbots* (DUTT; SASUBILLI; YERRAPATI, 2020), mas suas aplicações vão além (SICILIANO; KHATIB, 2008). Ela pode ser dividida em quatro áreas principais: Aprendizado de Máquina (HAYKIN, 2009), Lógica Fuzzy — cujo aumento de complexidade de sistemas reduz a precisão e o significado das declarações, tornando-as quase mutuamente exclusivas (ZADEH, 1968) —, sistemas evolutivos e agentes inteligentes, incluindo a robótica em grupo (MENDONÇA, et al, 2019). No aprendizado de máquina, as redes neurais artificiais, inspiradas em neurônios biológicos, desempenham papel central, refletindo um campo extenso e em contínua expansão.

Não será escopo dessa pesquisa fornecer maiores detalhes matemáticos e códigos de todos as Redes Neurais Artificiais analisadas.

A figura 1 mostra uma estrutura básica de um neurônio artificial (HAYKIN, 2009), no qual tem uma somatória de entradas com pesos sinápticos uma função de ativação e um sinal de Bias

O código para execução de uma porta lógica com 3 entradas pode ser conferido no link https://colab.research.google.com/drive/1sSUaY44x6EHR-dGBa45X8wgeRCPbqnl3#scrollTo=IXTnlg7l-D2V&line=63&uniqifier=1

Os resultados encontrados pelo treinamento efetuado pelo código do em *Phyton* originou os seguintes resultados, valor dos pesos sinápticos de cada entrada, o valor do Bias e o vetor de saída que só ocorre o valor 1 na ultima linha, quando todas as entradas forem 1.

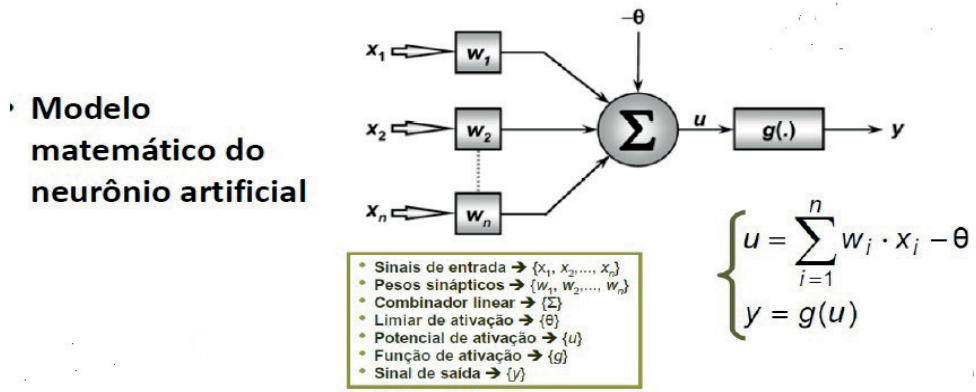

Figura 1 – Neurônio Artificial

Fonte: Haykin (2009)

Pesos sinápticos finais: [0.18351649 0.58862023 0.28512041]

Bias final: -0.509764542962082

Saídas calculadas: [0, 0, 0, 0, 0, 0, 0, 1]

Quanto ao aprendizado de máquinas, foco desse artigo. A evolução das redes neurais clássicas até os modelos baseados em *transformers* reflete décadas de avanços em técnicas de aprendizado de máquina e computação. Essa trajetória inclui o desenvolvimento de arquiteturas cada vez mais sofisticadas e especializadas para resolver problemas em diferentes domínios, como visão computacional, processamento de linguagem natural (PLN), jogos, entre outros.

A linguagem natural é base para os *chatbots*, que precisam de estímulo para interação com eles. Neste contexto, surge a chamada engenharia de prompt.

Criar um bom *prompt* exige clareza, contexto e especificidade para direcionar a resposta desejada. É fundamental evitar ambiguidades, estruturando o pedido de forma lógica e direta. Especifique o objetivo esperado, como texto explicativo, lista ou código, e forneça contexto relevante para que a solicitação seja compreendida.

Detalhe informações importantes para limitar o escopo e inclua o formato ou estilo desejado, como parágrafos, listas ou linguagem formal. O uso de exemplos concretos ajuda a ilustrar expectativas e evita vaguidade, garantindo respostas mais focadas. Por fim, revise o *prompt* com base nos resultados, ajustando-o para refinar os pedidos e obter respostas mais alinhadas às suas necessidades. Posto isso, em alguns casos prompts negativos se fazem necessário para questões de ética, bom senso, segurança.

Para exemplificar duas instanciações empregando inteligência artificial (HAZIRBAS, et al., 2022) em imagens serão apresentadas na figura 2 e 3 consequentemente, ressalta-se que a segunda usa o conceito de prompt negativo. A figura 2 cujo prompt “desenhe uma

floresta ao amanhecer com raios de sol dourados entre as arvores".

Figura 2 – Floresta ao amanhecer.

Fonte: Imagem gerada por IA. Plataforma ChatGPT (2024).

Já a figura 3 "Desenhe uma imagem de crianças brincando em um parquinho CHATGPT. Desenvolvida no *Image Generator*. https://chatgpt.com/g/g-pmuQfob8d-image-generator. Nessa imagem não deve aparecer elementos como por exemplo bebidas alcoólicas ou material adulto" emprega o conceito de prompt negativo, na segunda parte da descrição do *prompt*, proibindo qualquer referência a elementos não desejados.

Figura 3 – Crianças brincando em um parquinho.

Fonte: Imagem gerada por IA. Plataforma ChatGPT (2024).

Os avanços no campo da inteligência artificial têm sido marcados pela evolução contínua de métodos e ferramentas capazes de lidar com desafios cada vez mais complexos. Dentro desse panorama, as redes neurais ocupam uma posição central, servindo como a base para o desenvolvimento de soluções inovadoras em diversas áreas, como visão computacional, processamento de linguagem natural e sistemas generativos.

A evolução das redes neurais, desde os modelos clássicos até as arquiteturas

modernas, reflete o amadurecimento das técnicas de aprendizado de máquina e das capacidades computacionais. Essa progressão será abordada nos capítulos subsequentes, com uma análise detalhada dos principais tipos de redes neurais e suas aplicações práticas, estabelecendo as bases para a compreensão das tecnologias que estão moldando o futuro da inteligência artificial.

2 | REDES NEURAIS ARTIFICIAIS

Dada a relevância das redes neurais no contexto do aprendizado de máquina e suas diversas aplicações, é necessário analisar suas principais arquiteturas e como elas se integram ao desenvolvimento de sistemas inteligentes. A evolução dessas redes reflete avanços teóricos e computacionais, resultando em modelos que vão desde os clássicos até os mais sofisticados, como os *Transformers*.

As diferentes arquiteturas são projetadas para abordar problemas específicos e explorar padrões complexos nos dados, desempenhando um papel central em áreas como visão computacional, processamento de linguagem natural e sistemas generativos. A seguir, serão apresentados os principais tipos de redes neurais, com ênfase em suas características estruturais, métodos de aprendizado e aplicações práticas.

2.1 Redes neurais clássicas

As primeiras redes neurais eram baseadas na arquitetura ***Perceptron***, um modelo matemático simples introduzido por Frank Rosenblatt nos anos 1950. Essas redes consistiam em um único neurônio ou em uma camada simples de neurônios, capazes de resolver problemas linearmente separáveis. Contudo, eram limitadas devido à incapacidade de resolver problemas não lineares, como demonstrado no livro ***Perceptrons*** de Marvin Minsky e Seymour Papert citado em (Haykin, 2009).

Para o aprendizado das redes neurais clássicas, e as demais técnicas de aprendizado de máquina é necessário entender como funciona um neurônio artificial, inspirado no biológico

Para superar essas limitações:

- Surgiram os **Perceptrons Multicamadas (MLPs)**, que introduziram camadas ocultas.
- O algoritmo de **retropropagação do erro** (*backpropagation*), popularizado nos anos 1980 por Rumelhart, Hinton e Williams, permitiu o treinamento eficiente dessas redes, ajustando os pesos com base nos gradientes do erro.

2.2 Redes neurais convolucionais (CNNS)

Anteriormente a classificação de imagens era feita por análise de pixels, entretanto computacionalmente muitos casos se tornavam enviáveis. A melhor solução para tratamento de imagens está na extração de características.

Nos anos 1990, as redes convolucionais ganharam destaque com a introdução do ***LeNet*** por Yann LeCun. As CNNs são especializadas para **visão computacional** e foram projetadas para capturar padrões espaciais em dados visuais (Li, 2011). Suas principais características incluem:

- **Camadas convolucionais**, que aprendem filtros para detectar bordas, texturas e objetos.
- ***Pooling***, para redução da dimensionalidade e maior eficiência computacional.

Nos anos 2010, arquiteturas como ***AlexNet*** (2012), **VGG** e ***ResNet*** revolucionaram o campo, aproveitando avanços no hardware (GPUs) e grandes conjuntos de dados como o *ImageNet*.

Anteriormente a classificação de imagens era feita por análise de pixels, entretanto computacionalmente muitos casos se tornavam enviáveis. A melhor solução para tratamento de imagens está na extração de características (TAN, 2019).

Detalhado as etapas para reconhecimento de um gato em uma rede convolucional pré-treinada, observando que existe a possibilidade de treinamento da mesma, entretanto devido a restrição de tamanho do texto, essa opção não será escopo.

De modo resumido Imagem do gato → Pré-processamento → Convolução → *Pooling* → Extração de características profundas → Classificação em Fully *Connected* → Probabilidade da classe “gato” (PARKHI, et al 2012).

A imagem ilustra o fluxo típico de uma Rede Neural Convolucional (CNN) aplicada a um problema de classificação de imagens. Vamos detalhar cuidadosamente cada um dos blocos, descrevendo o papel e as transformações que ocorrem em cada etapa:

1. Entrada (Input):

A camada de entrada recebe a imagem original em formato bruto, por exemplo, uma imagem colorida de tamanho fixo (como 32x32x3, 64x64x3 etc. dependendo da aplicação). Cada pixel da imagem é convertido em um conjunto de valores numéricos — normalmente, intensidade dos canais de cor (vermelho, verde e azul).

- A partir dessa imagem, a rede vai extrair padrões cada vez mais complexos ao longo das camadas seguintes.

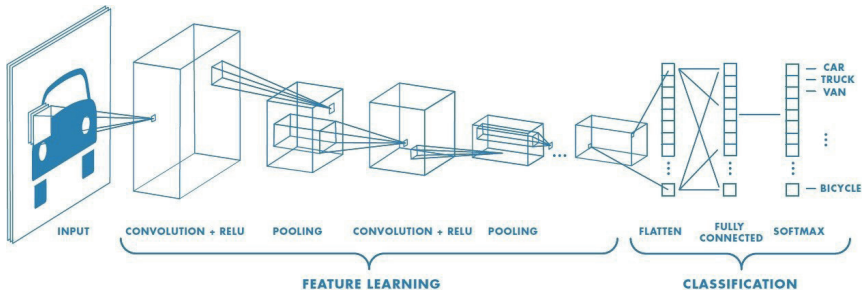

Figura 3 – Camadas de uma rede convolucional clássica.

Fonte: YAIMING, 2023

2. Camada de Convolução (*Convolution*) + Função de Ativação (ReLU):

A camada de convolução aplica um conjunto de filtros (também chamados de kernels) sobre a imagem de entrada. Cada filtro é um pequeno bloco de pesos aprendíveis (por exemplo, 3x3 ou 5x5) que é deslizado pela imagem. Ao aplicar o filtro, produz-se um mapa de características (*feature map).*

- Cada filtro "foca" em um tipo específico de padrão local, como bordas, texturas, curvas ou cantos. Quanto mais avançamos nas camadas, mais complexos esses padrões se tornam.
- Após a convolução, é comum aplicar uma função de ativação não linear, como a ReLU (*Rectified Linear Unit*), que transforma valores negativos em zero e mantém os positivos. Isso torna o modelo mais potente ao introduzir não-linearidade.

3. Camada de Pooling:

Depois da convolução, normalmente aplica-se uma camada de *pooling*, como *Max Pooling*. O objetivo do *pooling* é reduzir a dimensionalidade espacial (altura e largura) dos mapas de características, mantendo as informações mais relevantes.

- Por exemplo, o *Max Pooling* de tamanho 2x2 pega blocos 2x2 do mapa de características e seleciona o valor máximo desses 4 pixels. Isso reduz a resolução espacial do mapa de características pela metade, mas mantém características fortes.
- A redução de dimensionalidade ajuda a diminuir a quantidade de parâmetros e a complexidade da rede, além de conferir invariança a pequenas variações na posição dos padrões da imagem.

4. Repetição dos Blocos Convolucionais + *Pooling*:

A rede pode ter várias camadas convolucionais seguidas por camadas de *pooling.* A cada etapa, os filtros convolucionais extraem características mais complexas:

- Nas primeiras camadas, os filtros detectam bordas, contornos simples e textu-

ras básicas.

- Em camadas mais profundas, os filtros podem reconhecer partes de objetos ou padrões mais específicos.
- A combinação dessas operações gera mapas de características mais ricos e informativos, porém com dimensão espacial reduzida.

***5. Flatten* (Achatamento)**:

Depois que passamos por várias camadas convolucionais e de *pooling*, a saída final dessa etapa ainda é um conjunto de mapas de características bidimensionais. A operação de *flatten* transforma esses mapas em um vetor unidimensional.

- Esse vetor agrega todos os valores presentes nos mapas de características finais em uma única estrutura linear, que servirá como entrada para as camadas totalmente conectadas.

6. Camadas Totalmente Conectadas (*Fully Connected Layers*):

As camadas totalmente conectadas (também chamadas de densas ou *feed-forward*) funcionam de forma similar a uma rede neural tradicional, onde cada neurônio está ligado a todos os neurônios da camada anterior.

- Essas camadas combinam as características extraídas pelas camadas convolucionais para produzir uma representação de alto nível que ajudará a decidir a classe da imagem.
- Geralmente, nessas camadas, também são aplicadas funções de ativação (como *ReLU*) e, às vezes, camadas de regularização, como *Dropout*, para evitar *overfitting*.

7. Camada de Saída (*Softmax*):

A última camada do modelo é normalmente uma camada totalmente conectada seguida de uma função de ativação *softmax* (no caso de classificação multiclasse).

- O *softmax* pega o vetor de pontuações produzidas pela última camada densa e converte em probabilidades (valores entre 0 e 1 que somam 1).
- Cada saída do *softmax* corresponde à probabilidade de a imagem pertencer a uma determinada classe (carro, caminhão, van, bicicleta etc.).

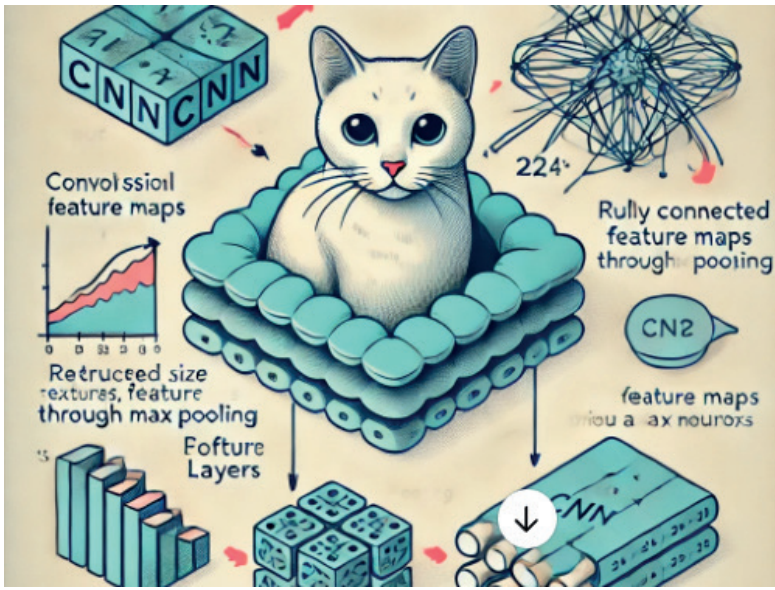

Figura 4 – Representação da classificação de imagem através de uma CNN.

Fonte: Imagem gerada por IA. Plataforma ChatGPT (2024).

Embora as CNNs sejam altamente eficientes, suas limitações em capturar dependências de longo alcance em dados sequenciais motivaram o desenvolvimento de outras abordagens, como redes baseadas em atenção e *Transformers*. Uma das CNNs de bastante empregada na literatura é a YOLO.

o YOLO (*You Only Look Once*) é um modelo de detecção de objetos baseado em uma arquitetura de rede neural convolucional. A partir de versões iniciais como YOLOv1 até as mais recentes (YOLOv7, YOLOv8 etc.), a idéia central é usar uma única rede para prever *bounding* boxes e classes de objetos simultaneamente. Diferentemente de abordagens anteriores, em que o processo de detecção era dividido em etapas – por exemplo, um modelo gera propostas de regiões e outro classifica cada região –, o YOLO realiza a detecção "de uma só vez" (daí o nome), o que o torna extremamente rápido.

Resumidamente, o YOLO funciona da seguinte forma:

1. Entrada: Uma imagem é fornecida à rede.

2. Extração de Características: Por meio de camadas convolucionais, a rede extrai mapas de características relevantes da imagem.

3. Predição de Bounding Boxes e Classes: As camadas finais da rede geram previsões simultâneas sobre a localização (coordenadas) e a probabilidade de cada classe para vários "*anchor boxes*".

4. Filtragem de Resultados: As detecções redundantes ou com baixa confiança são removidas por meio de técnicas como *Non-Maximum Suppression* (NMS).

Por ser uma CNN otimizada para detecção, o YOLO é conhecido por seu bom balanceamento entre velocidade e precisão, o que o torna ideal para aplicações em tempo real, como sistemas de vigilância, robótica e visão computacional embarcada em dispositivos móveis (ZHAO; QIAO,2023).

2.3 Redes baseadas em atenção

As redes baseadas em atenção introduziram uma nova forma de lidar com dados sequenciais, permitindo que o modelo destaque elementos mais relevantes da entrada para a tarefa. Esse mecanismo, inicialmente proposto por Bahdanau et al. (2014) no contexto de modelos Seq2Seq com Atenção, aprimorou o alinhamento dinâmico entre entrada e saída, superando dificuldades das RNNs em capturar dependências de longo alcance.

A atenção expandiu-se para diversas áreas, desde PLN (tradução, sumarização, respostas a perguntas) até visão computacional (segmentação, detecção de objetos) e domínios como bioinformática e finanças. A limitação das RNNs em termos de eficiência e paralelismo levou ao surgimento de arquiteturas exclusivamente baseadas em atenção, culminando nos Transformers (Vaswani et al., 2017). Esses modelos estabeleceram um novo patamar, eliminando a recorrência e melhorando significativamente a performance em tarefas sequenciais.

2.4 Redes *transformers*

Os *Transformers*, introduzidos no artigo *"Attention is All You Need"* (Vaswani et al., 2017), revolucionaram o campo da inteligência artificial ao estabelecer um novo paradigma no processamento de dados sequenciais. Essa arquitetura eliminou a necessidade de estruturas recorrentes, como nas redes neurais recorrentes (RNNs), ao utilizar exclusivamente mecanismos de atenção. Essa abordagem trouxe melhorias significativas em desempenho e eficiência computacional, redefinindo o estado da arte em várias aplicações.

O principal avanço dos *Transformers* está no uso da Atenção Multi-Cabeças (*Multi-Head Attention*), que permite ao modelo processar diferentes partes de uma sequência simultaneamente, capturando dependências complexas entre os elementos. Além disso, a arquitetura é composta por componentes modulares como Codificadores e Decodificadores (*Encoders e Decoders)*, projetados para transformar a entrada em representações intermediárias e, posteriormente, gerar a saída correspondente. Um elemento essencial é o *Positional Encoding*, que incorpora informações sobre a ordem sequencial dos dados, algo crítico em tarefas como tradução automática e modelagem de linguagem.

Entre as principais vantagens dos *Transformers*, destacam-se:

- **Paralelização durante o treinamento**: A ausência de recorrência possibilita

maior aproveitamento do poder computacional de *GPUs* e *TPUs*, reduzindo o tempo necessário para treinar modelos grandes.

- **Captura de dependências de longo alcance:** O mecanismo de atenção facilita a identificação de relações entre elementos distantes na sequência, superando limitações das RNNs e LSTMs.

A arquitetura do *Transformer* é altamente flexível e extensível, permitindo adaptações para diferentes domínios. Alguns dos modelos derivados mais relevantes incluem:

- **BERT (2018):** Projetado para tarefas de compreensão de texto, como análise de sentimentos e perguntas e respostas.
- **GPT (2018-2023):** Focado na geração de texto e modelos de linguagem geral.
- **Vision Transformers (ViT):** Adaptado para processamento de imagens, utilizando atenção em substituição a convoluções tradicionais.

O funcionamento do *Transformer* pode ser resumido nas seguintes etapas:

1. Entrada e Representação Inicial: A sequência de entrada é convertida em vetores contínuos (embeddings), aos quais são adicionados os embeddings posicionais, que preservam a ordem dos elementos na sequência.

2. Codificador (Encoder): É composto por múltiplas camadas idênticas, cada uma contendo:

3. Atenção Multi-Cabeças: Calcula a relevância entre todos os tokens na sequência de entrada.

4. Rede *Feedforward*: Processa individualmente os tokens após a atenção, aplicando transformações não lineares.

5. Camadas de Normalização e *Dropout*: Estabilizam o treinamento e reduzem o risco de *overfitting*.

6. Decodificador (*Decoder*): Similar ao codificador, mas com adaptações específicas:

7. Atenção Mascarada: Garante que os tokens futuros não sejam considerados durante a geração, preservando a causalidade.

8. Atenção ao Codificador: Integra o contexto da sequência de entrada ao processamento da saída parcial.

9. Rede Feedforward e Normalização: Operam como no codificador.

10. Saída e Geração: A camada final do decodificador aplica uma função *softmax* para converter as representações em probabilidades, prevendo o próximo token ou elemento na sequência.

11. Treinamento: O modelo é otimizado utilizando funções de perda, como a entropia cruzada, para minimizar os erros na previsão dos *tokens*.

A evolução das redes neurais reflete um ciclo contínuo de inovação. Os *transformers*

representam o auge atual dessa evolução, sendo aplicados a uma ampla gama de domínios, mas a pesquisa continua avançando em busca de arquiteturas mais eficientes e capazes.

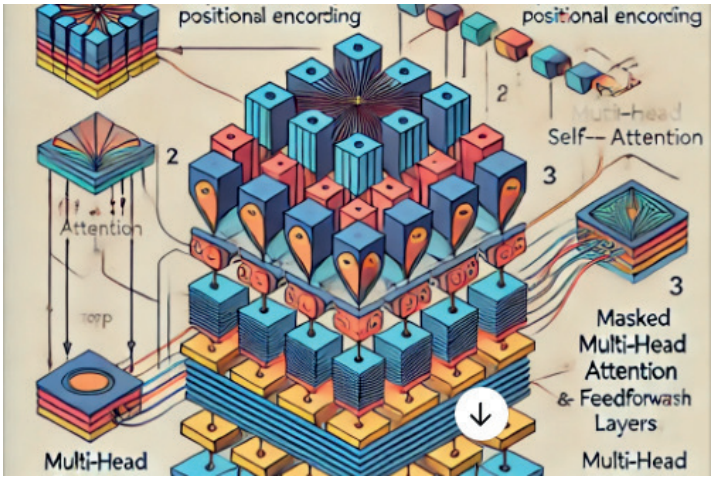

Figura 5 – Representação do sequenciamento de dados através de uma rede transformers.

Fonte: Imagem gerada por IA. Plataforma ChatGPT (2024).

3 I CONCLUSÃO

O estudo analisou fundamentos e práticas da inteligência artificial com foco em arquiteturas de redes neurais, cobrindo desde modelos clássicos (perceptrons multicamadas, redes convolucionais) até abordagens mais recentes (redes baseadas em atenção, Transformers). Utilizando Python para implementações, o trabalho demonstrou aplicações práticas, ressaltando a evolução das redes neurais, a influência do aprendizado profundo em visão computacional e processamento de linguagem natural, e a importância da engenharia de prompts para personalização e controle criativo de modelos gerativos.

Ao longo do texto, casos práticos e experimentos computacionais ilustraram a relevância das técnicas, bem como suas limitações (alto custo computacional, desafios de treinamento em larga escala). Como perspectivas futuras, destaca-se o desenvolvimento de frameworks para otimizar a engenharia de prompts e a integração de modelos baseados em atenção com arquiteturas híbridas, visando maior eficiência, acessibilidade e desempenho preditivo.

REFERÊNCIAS

BAHDANAU, Dzmitry; CHO, Kyunghyun; BENGIO, Yoshua. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2016. Disponível em: https://arxiv.org/abs/1409.0473. Acesso em: 8 dez. 2024.

CHATGPT. Image Generator. Disponível em: https://chatgpt.com/g/g-pmuQfob8d-image-generator. Acesso em: 13 dez. 2024.

DUTT, V.; SASUBILLI, S. M.; YERRAPATI, A. E. Dynamic Information Retrieval with Chatbots: A Review of Artificial Intelligence Methodology. In: **INTERNATIONAL CONFERENCE ON ELECTRONICS, COMMUNICATION AND AEROSPACE TECHNOLOGY (ICECA)**, 2022, Coimbatore, India. Anais… Coimbatore: IEEE, 2022. p. 1299–1303.

HAYKIN, Simon. ***Neural Networks and Learning Machines.*** **3. ed**. Upper Saddle River, NJ: Prentice Hall, 2009.

HAZIRBAS, C.; BITTON, J.; DOLHANSKY, B.; PAN, J.; GORDO, A.; FERRER, C. C. **Towards measuring fairness in AI: the casual conversations dataset**. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, v. 4, n. 3, p. 324-332, jul. 2022. DOI: 10.1109/TBIOM.2021.3132237.

LI, H. Computer network connection enhancement optimization algorithm based on convolutional neural network. In: **INTERNATIONAL CONFERENCE ON NETWORKING, COMMUNICATIONS AND INFORMATION TECHNOLOGY (NETCIT)**, 2021, Manchester, United Kingdom. Anais… Manchester: IEEE, 2021. p. 281-284. DOI: 10.1109/NetCIT54147.2021.00063.

MENDONÇA, M.; KONDO, H. S.; BOTONI DE SOUZA, L.; PALÁCIOS, R. H. C.; SILVA DE ALMEIDA, J. P. L. Semi-Unknown Environments Exploration Inspired by Swarm Robotics using Fuzzy Cognitive Maps. In: **IEEE INTERNATIONAL CONFERENCE ON FUZZY SYSTEMS (FUZZ-IEEE)**, 2019, New Orleans, LA, USA. Anais… [S. l.: s. n.], 2019. p. 1-8.

MENDONÇA, Márcio et al. Inteligência artificial aplicada: engenharia de prompt para otimização de trabalhos com ChatGPT. In: *Ciência e tecnologia: catalisadores da inovação.* Ponta Grossa: Atena Editora, 2024. Cap. 2, p. 13-26.

PARKHI, O. M.; VEDALDI, A.; ZISSERMAN, A.; JAWAHAR, C. V. Cats and Dogs. In: **IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION**, 2012. Anais… [S. l.: s. n.], 2012.

PASSINO, M. K.; YOURKOVICH, S. **Fuzzy Control**. Menlo Park: Addison-Wesley, 1997.

SICILIANO, B.; KHATIB, O. (Eds.). ***Springer Handbook of Robotics.*** **2. ed**. Heidelberg: Springer-Verlag Berlin Heidelberg, 2016.

TAN, M.; LE, Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In: **INTERNATIONAL CONFERENCE ON MACHINE LEARNING**, 2019. Anais… [S. l.: s. n.], 2019.

VASWANI, Ashish et al. *Attention Is All You Need.* 2023. Disponível em: https://arxiv.org/abs/1706.03762. Acesso em: 8 dez. 2024.

YAIMING, Yao. Research on facial recognition system based on deep learning. In: **INTERNATIONAL CONFERENCE ON MACHINE LEARNING AND AUTOMATION**, 2023. Disponível em: http://dx.doi.org/10.54254/2755-2721/34/20230332. Acesso em: 13 dez. 2024.

ZHAO, T.; QIAO, N**. Research on Target Detection Technology of Aircraft Satellite Images Based on Improved YOLOv5** Model. In: INTERNATIONAL CONFERENCE ON BIG DATA & ARTIFICIAL INTELLIGENCE & SOFTWARE ENGINEERING (ICBASE), 4., 2023, Nanjing, China. Nanjing, 2023. p. 89-94. DOI: 10.1109/ICBASE59196.2023.1030317