


ENGENHARIA DE SOFTWARE E REDES NEURAIAS**SOFTWARE ENGINEERING AND NEURAL NETWORKS** <https://doi.org/10.63330/aurumpub.005-012>**Kaleb Thomé Ribeiro Oliveira**

Graduando, Engenharia de software

UniFatecie

E-mail: rosangelaengbio@gmail.com

RESUMO

Este trabalho tem como tema central a integração entre engenharia de software e redes neurais artificiais, abordando como a combinação dessas duas áreas pode resultar em soluções computacionais mais eficientes, inteligentes e adaptativas. O principal objetivo da pesquisa foi investigar de que forma as redes neurais, inspiradas no funcionamento do cérebro humano, podem ser aplicadas no contexto do desenvolvimento de software para melhorar processos como a estimativa de custos, identificação de falhas e otimização de desempenho. Para isso, utilizou-se uma metodologia de pesquisa bibliográfica, com base em autores como Haykin (2001), Pressman (2010), Pfleeger (1998), Silva, Spatti e Flauzino (2010), entre outros, além da análise de estudos de caso e aplicações práticas já desenvolvidas na área. A engenharia de software foi analisada desde seus fundamentos, incluindo suas fases essenciais como levantamento de requisitos, modelagem, codificação, testes e manutenção. Em paralelo, foram exploradas as características das redes neurais artificiais, como sua capacidade de aprendizado, adaptação e generalização, com destaque para aplicações concretas na estimativa de esforço de desenvolvimento de software. Os resultados da análise indicaram que a incorporação de redes neurais no ciclo de vida do software pode tornar os processos mais precisos, ágeis e responsivos a mudanças. Constatou-se ainda que o uso de RNAs promove a automação de tarefas que exigem julgamento subjetivo ou conhecimento especializado, contribuindo para a redução de falhas humanas e para a melhoria da qualidade final do produto de software. Conclui-se, portanto, que a convergência entre engenharia de software e redes neurais artificiais representa uma tendência promissora, capaz de transformar a forma como sistemas computacionais são projetados, desenvolvidos e mantidos, fortalecendo a inovação tecnológica no campo da inteligência computacional.

Palavras-chave: Redes Neurais Artificiais; Engenharia de Software; Inteligência Computacional.

ABSTRACT

The central theme of this work is the integration of software engineering and artificial neural networks, addressing how the combination of these two areas can result in more efficient, intelligent and adaptive computing solutions. The main objective of the research was to investigate how neural networks, inspired by the functioning of the human brain, can be applied in the context of software development to improve processes such as cost estimation, fault identification and performance optimization. To this end, a bibliographic research methodology was used, based on authors such as Haykin (2001), Pressman (2010), Pfleeger (1998), Silva, Spatti and Flauzino (2010), among others, as well as the analysis of case studies and practical applications already developed in the area. Software engineering was analyzed from its foundations, including its essential phases such as requirements gathering, modeling, coding, testing and maintenance. In parallel, the characteristics of artificial neural networks were explored, such as their ability to learn, adapt and generalize, with emphasis on concrete applications in estimating software development effort. The results of the analysis indicated that incorporating neural networks into the software life cycle can make processes more precise, agile and responsive to change. It was also found that the use of ANNs



promotes the automation of tasks that require subjective judgment or specialized knowledge, contributing to the reduction of human failures and improving the final quality of the software product. It is therefore concluded that the convergence between software engineering and artificial neural networks represents a promising trend, capable of transforming the way computer systems are designed, developed and maintained, strengthening technological innovation in the field of computational intelligence.

Keywords: Artificial Neural Networks; Software Engineering; Computational Intelligence.



1 INTRODUÇÃO

A integração entre a engenharia de software e as redes neurais artificiais configura o tema central deste trabalho, cuja relevância cresce na medida em que as demandas por sistemas computacionais inteligentes, eficientes e adaptativos se intensificam. A engenharia de software, enquanto disciplina voltada ao desenvolvimento sistemático de programas computacionais, apresenta métodos, técnicas e boas práticas para garantir a qualidade, a confiabilidade e a manutenção dos sistemas. Por sua vez, as redes neurais artificiais, inspiradas na estrutura e no funcionamento do cérebro humano, contribuem com capacidades de aprendizado, generalização e predição, atuando como ferramentas poderosas na automação e na análise de dados complexos. O objetivo deste trabalho é analisar como essas duas áreas se complementam e de que forma a aplicação de redes neurais pode melhorar processos dentro do ciclo de vida do software, especialmente no que diz respeito à estimativa de custos, à predição de esforço e à detecção de falhas. Parte-se da hipótese de que a incorporação de modelos neurais à engenharia de software representa um avanço significativo, tanto do ponto de vista técnico quanto gerencial, promovendo maior precisão nos processos de desenvolvimento e mais eficiência nos resultados obtidos. Justifica-se esta pesquisa pela crescente necessidade do mercado de software por soluções automatizadas e inteligentes, capazes de lidar com incertezas e adaptar-se a contextos variados. A metodologia adotada foi a pesquisa bibliográfica, com base em autores como Haykin (2001), Pressman (2010), Pfleeger (1998), Silva, Spatti e Flauzino (2010), entre outros, além da análise de estudos de caso que demonstram aplicações reais das RNAs em projetos de engenharia de software.

O trabalho está estruturado em quatro capítulos principais, além desta introdução. No primeiro capítulo, aborda-se a engenharia de software em sua totalidade, incluindo sua definição, ciclo de vida, fases do processo de desenvolvimento e metodologias utilizadas. No segundo capítulo, exploram-se os fundamentos das redes neurais artificiais, suas estruturas, funcionamento, tipos de neurônios e funções de ativação.

Em seguida, no terceiro capítulo, é feita uma análise histórica e técnica mais aprofundada das RNAs, destacando suas aplicações na área de tecnologia da informação, especialmente em sistemas computacionais. Por fim, no quarto capítulo, discute-se a interligação entre engenharia de software e redes neurais, destacando estudos de caso, experimentos, vantagens e limitações dessa integração. A conclusão do trabalho reafirma a importância da convergência entre essas duas áreas como forma de promover a inovação tecnológica, sugerindo que a união entre metodologias estruturadas e inteligência computacional representa um caminho promissor para o futuro do desenvolvimento de software.



2 DESENVOLVIMENTO

2.1 ENGENHARIA DE SOFTWARE

A engenharia de software, considerada um ramo da engenharia industrial, dedica-se à análise das metodologias de trabalho e das práticas recomendadas utilizadas por profissionais na criação de sistemas de software (Pfleeger, 1998). O principal objetivo dessa área está na utilização de abordagens sistemáticas que garantam que sistemas complexos atendam às necessidades dos usuários, sejam confiáveis, de fácil manutenção, com bom desempenho e que possam ser desenvolvidos dentro dos prazos e orçamentos estabelecidos (Pressman, 2010).

Também conhecida como uma forma de engenharia aplicada ao desenvolvimento de programas computacionais, essa disciplina compreende as etapas de especificação, construção e implantação do código do software (Rezende, 2002). De acordo com Sommerville (2007), a engenharia de software cobre todo o ciclo de vida do sistema, abrangendo desde o levantamento de requisitos, elaboração das especificações, definição da arquitetura interna, escolha das estratégias de programação, desenvolvimento propriamente dito, realização de testes e, finalmente, manutenção do software (Magela, 2006).

O processo de Engenharia de Software abrange várias etapas, incluindo a análise de requisitos, o design do sistema, a implementação, os testes e a manutenção. A análise de requisitos é crucial, pois é nesse estágio que as expectativas dos usuários são levantadas e documentadas. Um bom entendimento dos requisitos é essencial para evitar retrabalhos e garantir que o produto final atenda às necessidades do mercado.

O design do sistema envolve a criação da arquitetura do software, definindo como os diferentes componentes interagem entre si. Essa fase é seguida pela implementação, onde os desenvolvedores escrevem o código-fonte. A qualidade do código é um fator determinante, e práticas como revisão de código e programação em pares são frequentemente utilizadas para garantir sua robustez.

Os testes são outra fase crítica, pois asseguram que o software funcione corretamente e esteja livre de falhas. Testes unitários, de integração e funcionais são realizados para validar o comportamento do sistema em diferentes cenários. Após o lançamento, a manutenção se torna uma parte contínua do ciclo de vida do software, onde melhorias e correções são implementadas conforme necessário.

Além das técnicas e metodologias, a Engenharia de Software também abrange aspectos de gestão de projetos, como o gerenciamento de equipes e o controle de prazos e orçamentos. Metodologias ágeis, como Scrum e Kanban, têm ganhado popularidade, permitindo que as equipes respondam rapidamente a mudanças e entreguem incrementos de software de forma contínua.

Em resumo, a Engenharia de Software é uma área dinâmica e em constante evolução, que combina conhecimentos técnicos e habilidades de gestão para produzir software de alta qualidade, atendendo às demandas de um mercado cada vez mais exigente.



2.2 REDES NEURAIIS

As redes neurais artificiais são projetadas com base no funcionamento do cérebro humano, sendo compostas por diversos neurônios artificiais. Essas redes são implementadas por meio de programação computacional ou por meio de circuitos eletrônicos (HAYKIN, 2001). Os neurônios artificiais, também conhecidos como nós, são as unidades fundamentais dessas redes. Eles recebem múltiplos valores de entrada, cada um associado a um peso que define a força e a direção da conexão. Cada neurônio é equipado com uma função de ativação, que realiza o processamento da soma ponderada das entradas e gera um valor de saída correspondente (RUSSELL; NORVIG, 2004).

Um exemplo prático da aplicação de redes neurais pode ser encontrado no trabalho de Moura, Pinto e Lustosa Filho (2011), que desenvolveram uma ferramenta destinada à estimativa de custos de projetos de software, simulando o processo de análise realizado por engenheiros experientes. Para isso, utilizaram o modelo de pontos por função, atribuindo pesos a diversos fatores relevantes para o cálculo. Esses pesos compõem os dados de entrada da rede neural, que então estima o custo do projeto. O treinamento dessa rede foi realizado com informações de 70 projetos reais, enquanto outros 30 projetos serviram como base para validação dos resultados obtidos pela ferramenta, de acordo com o aprendizado da rede.

Em outra pesquisa, também baseada em redes neurais, foi desenvolvida uma ferramenta voltada à estimativa de custos de software, porém com abordagem distinta. Nesse caso, as entradas da rede incluem os requisitos do sistema e o tempo padrão estimado para sua implementação. A rede é treinada com base em tempos padrão fornecidos por especialistas. Ao submeter a rede a diferentes conjuntos de dados e comparar os resultados obtidos, a ferramenta ajusta seus parâmetros até alcançar uma estimativa próxima àquela feita por um profissional experiente (BORSOI et al., 2011).

Outra aplicação importante do uso de redes neurais artificiais está no trabalho de Garcia et al. (2011), cujo objetivo foi prever o esforço total necessário para o desenvolvimento de um software, com base em características comuns entre os projetos analisados. Para isso, foi implementada uma rede com três camadas, sendo que os valores de entrada correspondiam às variáveis mais relevantes para a estimativa, enquanto a saída era o custo final previsto. O treinamento da rede envolveu dados de 214 projetos, dos quais uma parte foi utilizada para testar e validar o desempenho do modelo.

2.3 REDES NEURAIIS ARTIFICIAIS

As Redes Neurais Artificiais (RNAs) são estruturas computacionais inspiradas no funcionamento do sistema nervoso dos seres vivos. Seu principal diferencial está na habilidade de adquirir e preservar conhecimentos, sendo constituídas por um conjunto de unidades de processamento (SILVA; SPATTI; FLAUZINO, 2010). Essas redes são compostas por Neurônios Artificiais interligados por inúmeras conexões, denominadas sinapses artificiais, que podem ser representadas por vetores ou matrizes. O modelo



matemático das RNAs baseia-se na arquitetura neural de organismos inteligentes, permitindo que elas aprendam por meio da experiência, o que as torna comparáveis, em certa medida, ao cérebro humano (HAYKIN, 2001). Essa semelhança decorre da capacidade de aprendizagem e das conexões entre os neurônios artificiais, chamadas de pesos sinápticos, onde o conhecimento é efetivamente armazenado (HAYKIN, 2001). As RNAs podem ser implementadas por meio de componentes eletrônicos ou simuladas em computadores digitais, destacando-se por características como aprendizado, adaptação, generalização e facilidade na prototipagem (SILVA; SPATTI; FLAUZINO, 2010).

O estudo pioneiro sobre RNAs foi conduzido em 1943 por McCulloch e Pitts, cujo objetivo era modelar matematicamente o neurônio biológico. Eles demonstraram que o comportamento do neurônio seguia o princípio do "tudo ou nada", sendo essencial que suas funções fossem descritas de maneira linear (CARDON; MÜLLER, 1994). Apesar dos avanços obtidos até os anos 1950, o primeiro modelo prático de RNA surgiu com o Perceptron, desenvolvido por Frank Rosenblatt em 1958. O Perceptron representa uma RNA simples, formada por uma camada de entrada e outra de saída, onde cada entrada é associada a um peso, sendo a saída calculada a partir da soma dos produtos entre entradas e seus respectivos pesos (SILVA; SPATTI; FLAUZINO, 2010).

2.4 REDE NEURAL PERCEPTRON

A Rede Neural Perceptron é um modelo fundamental em aprendizado de máquina, especialmente na área de redes neurais. Desenvolvido por Frank Rosenblatt na década de 1950, o perceptron é um tipo de rede neural de camada única que simula o funcionamento de um neurônio biológico. Ele é projetado para realizar tarefas de classificação binária, onde a saída é uma decisão entre duas classes. (MOLETTA, 2015)

O funcionamento do perceptron se baseia em uma combinação linear das entradas, que são ponderadas por pesos ajustáveis. Cada entrada é multiplicada por um peso correspondente, e o resultado é somado. Essa soma é então passada por uma função de ativação, geralmente uma função degrau, que determina a saída do perceptron. Se a soma ponderada exceder um determinado limiar, o perceptron ativa uma saída positiva; caso contrário, a saída é negativa. (MOLETTA, 2015)

O treinamento de um perceptron envolve a atualização dos pesos com base na diferença entre a saída prevista e a saída real, utilizando um algoritmo de aprendizado conhecido como regra de atualização do perceptron. Este processo é repetido por várias iterações até que o modelo atinja um nível satisfatório de precisão. (MOLETTA, 2015)

Ainda, Hebb escreve que, os neurônios que tem sua ação em conjunto, ou seja, são estimulados ou exercem estímulo, fortalecem a ligação sináptica, no caso das RNAs, os pesos sinápticos. Observou-se no estudo da programação destes sistemas especialistas que o acréscimo dos pesos ocorre de maneira proporcional, e simétrica (CARDON; MÜLLER, 1994).



Embora o perceptron tenha limitações, como a incapacidade de resolver problemas linearmente inseparáveis (por exemplo, o problema XOR), ele é a base para o desenvolvimento de redes neurais mais complexas, como as redes multicamadas, que utilizam múltiplas camadas de neurônios e técnicas de retropropagação. A simplicidade e a elegância do perceptron o tornam um ponto de partida essencial para quem deseja entender os fundamentos das redes neurais e do aprendizado de máquina. (MOLETTA,2015)

2.5 ENGENHARIA DE SOFTWARE E REDES NEURAIIS

A engenharia de software e as redes neurais artificiais representam dois campos interdependentes que, quando integrados, potencializam significativamente o desenvolvimento de soluções computacionais inteligentes e adaptativas. A engenharia de software, por sua vez, oferece os fundamentos metodológicos e técnicos necessários para a criação sistemática e eficiente de sistemas de software, enquanto as redes neurais — inspiradas na estrutura e funcionamento do cérebro humano — introduzem capacidades de aprendizado, generalização e predição ao processo de desenvolvimento. (SILVA; SPATTI; FLAUZINO, 2010).

Ao incorporar redes neurais artificiais no contexto da engenharia de software, torna-se possível otimizar atividades críticas como estimativas de esforço e custo, identificação de falhas, testes automatizados e análise de requisitos. Essas tarefas, tradicionalmente executadas com base em métodos estatísticos ou heurísticos, ganham maior precisão e adaptabilidade com o uso de modelos neurais, que aprendem a partir de dados históricos e comportamentos anteriores. Assim, as redes neurais passam a atuar como componentes inteligentes dentro das etapas do ciclo de vida do software, aumentando a confiabilidade dos sistemas desenvolvidos. (SILVA; SPATTI; FLAUZINO, 2010).

Além disso, a engenharia de software contribui para a estruturação e manutenção de aplicações baseadas em redes neurais, oferecendo frameworks e boas práticas para o design modular, testes, reuso de componentes e gerenciamento de versões. Essa colaboração entre as áreas é essencial para que os sistemas neurais, muitas vezes complexos e sensíveis a variações nos dados, sejam implementados de forma robusta e escalável. (SILVA; SPATTI; FLAUZINO, 2010).

A sinergia entre engenharia de software e redes neurais resulta em soluções computacionais mais eficientes, capazes de se adaptar a ambientes dinâmicos e de aprender continuamente com novas informações. Esse entrelaçamento não apenas aprimora a qualidade do software, mas também impulsiona a inovação tecnológica em áreas como sistemas preditivos, reconhecimento de padrões, automação inteligente e tomada de decisão autônoma. Com isso, a convergência entre essas duas disciplinas se consolida como um dos pilares do avanço da inteligência computacional e da construção de sistemas mais inteligentes, responsivos e personalizados.



3 CONCLUSÃO

A conclusão deste trabalho reafirma a relevância da integração entre a engenharia de software e as redes neurais artificiais como um caminho estratégico e inovador para o desenvolvimento de sistemas computacionais mais eficientes, flexíveis e inteligentes. Ao longo desta pesquisa, ficou evidente que ambas as áreas, embora distintas em suas origens e fundamentos, apresentam complementaridades técnicas e metodológicas que, quando exploradas em conjunto, oferecem soluções avançadas para os desafios contemporâneos da tecnologia da informação. A engenharia de software, com seu rigor metodológico e foco na qualidade de processos e produtos, fornece uma base sólida para o planejamento, implementação, teste e manutenção de sistemas. Já as redes neurais artificiais, inspiradas no funcionamento do cérebro humano, agregam ao processo de desenvolvimento a capacidade de aprender com dados, identificar padrões complexos, fazer previsões e adaptar-se a novas informações sem a necessidade de reprogramação explícita.

A análise dos estudos de caso apresentados demonstrou que a aplicação de redes neurais em processos como a estimativa de esforço e custo, a avaliação de requisitos, a previsão de defeitos e a automação de testes contribui significativamente para a otimização do ciclo de vida do software. Esses modelos computacionais, quando corretamente treinados e ajustados, superam limitações de abordagens tradicionais e reduzem a dependência de julgamento humano em tarefas subjetivas, promovendo maior consistência e produtividade. Além disso, o uso das RNAs permite que sistemas se tornem progressivamente mais autônomos, com capacidade de resposta e adaptação a contextos dinâmicos e incertos.

A pesquisa bibliográfica, baseada em autores renomados como Haykin, Pfleeger, Pressman, Sommerville, Silva, Spatti e Flauzino, entre outros, forneceu um arcabouço teórico robusto que sustentou a compreensão das características, potencialidades e desafios envolvidos na aplicação de redes neurais em projetos de software. A escolha por esse tipo de metodologia mostrou-se apropriada, uma vez que permitiu o levantamento de conceitos fundamentais, bem como a análise crítica de experiências práticas e evidências empíricas sobre o uso integrado das duas áreas.

Conclui-se, portanto, que a convergência entre engenharia de software e redes neurais artificiais representa mais do que uma tendência passageira: trata-se de uma necessidade frente às exigências de um mercado que demanda soluções cada vez mais inteligentes, personalizadas, rápidas e confiáveis. A consolidação dessa integração exige investimentos em capacitação profissional, pesquisa multidisciplinar e desenvolvimento de ferramentas que favoreçam a aplicabilidade desses conceitos no cotidiano das organizações. Espera-se que este trabalho sirva como ponto de partida para aprofundamentos futuros, fomentando discussões acadêmicas e práticas sobre o papel das redes neurais no contexto da engenharia de software, bem como sobre as transformações que essa interação pode promover na forma como os sistemas computacionais são concebidos, implementados e evoluem. Em um cenário de constante inovação



tecnológica, o diálogo entre essas áreas se configura como elemento-chave para o avanço da inteligência computacional e para a construção de um futuro digital mais eficiente, autônomo e inteligente.



REFERÊNCIAS

- AGUIAR, F. G. Utilização de redes neurais artificiais para detecção de padrões de vazamento indutos. São Carlos: Universidade de São Paulo, 2010.
- ANJOS, J. M. S.; VILLANI, E. Modelagem e verificação de uma proposta para arquitetura de controle de um efetuator robótico baseada em LabView™. 2010.
- AZEVEDO, F. M. de; BRASIL, L. M.; OLIVEIRA, R. C. L. de. Redes neurais com aplicação em controles e em sistemas especialistas. Florianópolis: Visual Books Editora, 2000.
- BARROS, L. C.; BASSANEZI, R. C. Tópicos de lógica fuzzy e biomatemática. Campinas: UNICAMP/IMECC, 2006.
- BENINI, L. C. Estimação da densidade de solos utilizando sistemas de inferência fuzzy. 2007. Tese (Doutorado) – Universidade Estadual Paulista, Botucatu, 2007.
- BOENTE, A. N. P. Um modelo fuzzy para avaliação da qualidade de produtos de software e da satisfação dos gerentes de projetos numa fundação pública estadual. 2009. Dissertação (Mestrado) – Universidade Estácio de Sá, Rio de Janeiro, 2009. Disponível em: <http://boente.eti.br/fuzzy/tese-fuzzy-boente.pdf>. Acesso em: 26 Jun. 2025.
- BORSOI, Beatriz et al. Redes neurais aplicadas na estimativa de prazos de projetos de software. 2011. Disponível em: http://www.inf.unioeste.br/epac/anais2011/artigos_epac/A15.pdf. Acesso em: 22 Jun. 2025.
- CARDON, A.; MÜLLER, D. N. Introdução às redes neurais artificiais. Porto Alegre: Universidade Federal do Rio Grande do Sul, 1994.
- CASTRO, Raphael D. Avaliação de riscos em projetos de software a partir do uso de técnicas de inteligência computacional. 2009. Monografia – Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, 2009. Disponível em: <http://tcc.ecomp.poli.br/20091/TCC%20Jose%20DCastro-TCC.pdf>. Acesso em: 26 Jun. 2025.
- CAVALCANTE, M. M. et al. A plataforma Arduino para fins didáticos: estudo de caso com recolhimento de dados a partir do PLX-DAQ. XXXIV Congresso da Sociedade Brasileira de Computação – CSBC, 2014.
- COZMAN, F. G. Generalizing variable elimination in bayesian networks. Workshop on Probabilistic Reasoning in Artificial Intelligence, 2000.
- CUNHA, F. L. et al. O uso de redes neurais artificiais para o reconhecimento de padrões em uma prótese mioelétrica de mão. Laboratório de Biocibernética e Engenharia de Reabilitação – D.E.E. – E.E.S.C./ USP, 2007.
- GARCIA, A. et al. Methodology for software development estimation optimization based on neural networks. 2011. Disponível em: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5893788. Acesso em: 26 Jun. 2025.
- GERHARDT, T. E.; SILVEIRA, D. T. Métodos de pesquisa. 1. ed. Porto Alegre: Editora da UFRGS, 2009.



GOMIDE, F. A. Redes neurais artificiais para engenharia e ciências aplicadas: curso prático. Controle & Automação, v. 23, n. 5, 2012.

GUYTON, A. C.; HALL, J. E. Tratado de fisiologia médica. 12. ed. São Paulo: Elsevier Editora Ltda., 2011.

HAYKIN, Simon. Redes neurais: princípios e práticas. 2. ed. Tradução: Paulo Martins Engel. Porto Alegre: Bookman, 2001.

HRUSCHKA JR, E. R. Imputação bayesiana no contexto da mineração de dados. 2003. Tese (Doutorado) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2003.

KLIR, George; YUAN, Bo. Fuzzy sets, fuzzy logic and fuzzy system: selected papers by Lotfi A. Zadeh. Tradução do autor. World Scientific Publishing, 1996.

MAGELA, R. Engenharia de software aplicada: princípios. Rio de Janeiro: Alta Books, 2006.

MENDES, Emilia. Construindo um modelo especialista de estimativa de esforço de aplicações web usando redes bayesianas. 2010. Disponível em: <https://www.unochapeco.edu.br/static/data/portal/downloads/1024.pdf#page=64>. Acesso em: 20 jun. 2025.

MOLETTA, Eduardo. Redes neurais com estados de eco aplicadas em controle dependente dos estados. Ponta Grossa: Universidade Tecnológica Federal do Paraná, 2015.

MONTANAR, T. Histologia: texto, atlas e roteiro de aulas práticas. 3. ed. Porto Alegre: Edição do Autor, 2016.

MOURA, Íthalo Bruno de; PINTO, João Phellipe; LUSTOSA FILHO, José. Neural Estimate: uma ferramenta para a estimativa de custo de software. Escola Regional de Computação dos Estados do Ceará, Maranhão e Piauí, Teresina, 2011. Disponível em: http://www.die.ufpi.br/ercemapi2011/artigos/ST2_11.pdf. Acesso em: 20 jun. 2025.

NATIONAL INSTRUMENTS. Líder global em soluções de teste, medição e controle. 2006. Disponível em: <http://www.ni.com/pt-br.html>. Acesso em: 22 Jun.2025.

OPENAI. ChatGPT-4. Quais são os diferentes tipos de inteligência artificial? Acesso em: 26 mai. 2025.

OPENAI. ChatGPT-4. Quais são as diferentes atividades cognitivas que a inteligência artificial é capaz de realizar? Acesso em: 26 Jun. 2025.

OPENAI. ChatGPT-4. Enumere as principais atividades cognitivas que a inteligência artificial é capaz de realizar. Acesso em: 26 Jun. 2025.

OPENAI. ChatGPT-4. Enumere as principais atividades práticas que a inteligência artificial é capaz de realizar. Acesso em: 26 Jun. 2025.

PAGANI, D. H. Estudo comparativo entre redes neurais artificiais e redes neurais pulsadas usando MATLAB®. Cascavel: Universidade Estadual do Oeste do Paraná, 2012.



PAULINO, C. D.; TURKMAN, M. A.; MURTEIRA, B. Estatística bayesiana. Lisboa: Fundação Calouste Gulbenkian, 2003.

PFLIEGER, S. L. Software engineering: theory and practice. Nova Jersey: Prentice-Hall, 1998.

PRESSMAN, R. S. Engenharia de software. 6. ed. Porto Alegre: AMGH Editora, 2010.

REZENDE, D. A. Engenharia de software e sistemas de informação. 2. ed. Rio de Janeiro: Brasport, 2002.

SALLE, CTP et al. Uso de redes neurais artificiais para estimar parâmetros de produção de galinhas reprodutoras pesadas em recría. CDPA – Faculdade de Veterinária / UFRGS, 2012.

SILVA, I. N. da; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais para engenharia e ciências aplicadas. São Paulo: Artliber Editora Ltda., 2010.

SOMMERVILLE, I. Engenharia de software. 8. ed. Boston: Addison Wesley, 2007.