**Análise de redes neurais a carros autônomos e ao mercado financeiro**

**Gabriel Duarte Batista de Nazaré¹, Samuel Moraes Batistela².**

1Universidade do Vale do Paraíba/Instituto de Pesquisa e Desenvolvimento, Avenida Shishima Hifumi, 2911, Urbanova - 12244-000 - São José dos Campos-SP, Brasil, [gabrielduartebn@gmail.com](mailto:gabrielduartebn@gmail.com).

2Universidade do Vale do Paraíba/Instituto de Pesquisa e Desenvolvimento, Avenida Shishima Hifumi, 2911, Urbanova - 12244-000 - São José dos Campos-SP, Brasil, samuel.batistela12@gmail.com.

**Resumo**

O resumo deve descrever adequadamente os objetivos claros e definidos, além da metodologia, resultados e conclusão do trabalho. O resumo deve conter de 7-12 linhas. Não use recuo de parágrafo antes da palavra Resumo e escreva todo o texto em um só parágrafo. Deve ser conciso e resumir todo o artigo. Serão aceitos trabalhos originais, com natureza de trabalho divididos nas categorias Resultado de Projeto, Revisão de Literatura ou Trabalho de Disciplina, sendo no máximo 2 trabalhos por autor remetente, limite de mais 4 (quatro) coautores, e de 1 a 2 orientadores por trabalho enviado. **A SUBMISSÃO NÃO DEVERÁ SER REALIZADA PELOS ORIENTADORES.**

**Palavras-chave**: Usar no mínimo 3 e no máximo 5 palavras-chave, separadas entre si por ponto final e finalizadas por ponto.

**Área do Conhecimento:** Engenharias – Engenharia e Ciência da Computação

**Introdução**

Nos últimos anos, o campo da inteligência artificial (IA) tem experimentado avanços significativos, principalmente devido ao desenvolvimento e aplicação de redes neurais artificiais. As redes neurais, inspiradas na estrutura e no funcionamento do cérebro humano, são sistemas computacionais projetados para reconhecer padrões e resolver problemas complexos de forma eficiente.

O começo do estudo sobre redes neurais se deu em 1943 por Warren McCulloch, um neurofisiologista e psiquiatra americano que estudou na Universidade de Yale, e Walter Pitts, um lógico autodidata. Juntos, no trabalho publicado com o título "*A Logical Calculus of the Ideas Immanent in Nervous Activity*", propuseram um modelo matemático do neurônio biológico baseado em lógica limiar, onde poderia realizar operações lógicas básicas [1]. Este modelo, conhecido como neurônio MCP (McCulloch-Pitts), estabeleceu a base para as futuras implementações computacionais das redes neurais​​ [2].

No ano de 1949 um psicólogo canadense chamado Donald Hebb publicou um estudo chamado “*The Organization of Behavior*”. Este artigo explora o funcionamento do cérebro humano, introduzindo o conceito de que os caminhos neurais são fortalecidos a cada vez que são utilizados, fundamental para o processor de aprendizado e adaptação [3]. Posteriormente o mundo de estudos de inteligências artificiais adotou esse pensamento, a chamando de regra de Hebb.

Chegando em 1958, Frank Rosenblatt, psicólogo americano considerado por alguns como o pai do *deep learnig,* começou a desenvolver e implementar o primeiro modelo de rede neural, chamado de percéptron. Este modelo foi constituído por neurônios MPL (Percéptrons de Múltiplas Camadas), formando uma rede neural simples de apenas uma camada de entrada e uma camada de saída [3].

Isto possibilitou que inúmeros modelos pudessem ser criados desde a sua concepção, essas redes têm se mostrado ferramentas poderosas para uma variedade de aplicações, incluindo reconhecimento de fala, visão computacional, tradução automática, diagnósticos médicos e muitas outras áreas.

Neste trabalho iremos abordar alguns dos conceitos-chave, como a arquitetura, os paradigmas de aprendizagem e as funções de ativação. Além disso, serão analisadas duas aplicações de uma rede neural no mundo real, uma sendo na confecção de carros autônomos e quando aplicado ao mercado financeiro.

**Conceitos básicos**

**1. Arquitetura**

As redes neurais possuem diferentes arquiteturas, e a definição dessa arquitetura é essencial para lidar com diferentes tipos de dados e resolver problemas. A arquitetura de uma rede neural define o tipo de alimentação da informação e como esses dados são processados pelo sistema.

Uma rede de propagação para frente, ou rede de alimentação direta, é caracterizada pelo fluxo unidirecional de informações entre as suas camadas. Também chamadas de nós, as camadas em uma rede de propagação para frente normalmente são dispostas em camada de entrada, camadas intermediárias ou ocultas, e camada de saída.

Figura 1 - Rede de propagação para frente

Diagrama

Descrição gerada automaticamente

Fonte: Site Deep Learning Book

A Figura 1 demonstra o funcionamento de uma rede feedforward. A entrada de uma informação é feita pelos nós de entrada, depois é alimentada para os nós ocultos e, em seguida, os dados são enviados para os nós de saída, mostrando os resultados obtidos pelo processamento da rede neural.

Outro tipo de arquitetura é a que possibilita a realimentação da informação: a rede recorrente. Esta rede permite que os dados fluam em duas direções e que exista uma retroalimentação, permitindo a manutenção de uma 'memória' de informações.

Figura 2 - Rede realimentada

Diagrama

Descrição gerada automaticamente

Fonte: Site Deep Learning Book

A Figura 2 mostra como o processo de realimentação funciona em uma rede recorrente, onde existe uma realimentação de dados nos nós ocultos e uma realimentação dos dados da camada de saída.

A diferença nas tratativas dos dados nessas arquiteturas molda o modelo de rede neural para ser mais adequado ou otimizado para certos tipos de problemas, quando comparado a outros diversos tipos de arquiteturas. Um exemplo disso é a classificação e o reconhecimento de padrões, em que uma rede *feedforward* é mais utilizada para resolver esse problema em comparação a uma rede recorrente. Por sua vez, a rede recorrente é mais utilizada em sequências temporais e contextos de longo prazo, como tradução automática.

**2. Paradigmas de aprendizagem**

Os paradigmas de aprendizagem se referem a diferentes abordagens e metodologias utilizadas para treinar redes neurais. Dois dos paradigmas mais relevantes são a aprendizagem supervisionada e a aprendizagem não supervisionada. Cada um deles oferece abordagens distintas para resolver problemas com base nos dados disponíveis e nos objetivos do treinamento.

A aprendizagem supervisionada é um modelo no qual a rede neural é treinada utilizando dados rotulados, sendo cada exemplo no conjunto de treinamento composto por uma entrada e a saída desejada correspondente. Neste tipo de modelo, a importância está em fazer o mapeamento correto das saídas para as respectivas entradas, priorizando a precisão nas previsões futuras quando testado com novos dados. Isto é feito ao comparar as saídas de dados da rede com as saídas desejadas, é possível realizar um ajuste nos pesos para direcionar a saída para um valor desejado para a resolução do problema.

Já a aprendizagem não supervisionada, diferentemente da supervisionada, não atua com dados rotulados. O objetivo dessa rede não é a predição de uma saída específica, mas identificar atributos e padrões nos dados, isso permite que a análise de uma alta quantidade de dados seja realizada de uma forma mais exploratória para possibilitar uma aglomeração de padrões.

**3. Função de ativação**

As funções são um elemento importante para a elaboração de redes neurais, definindo se um nó deve ser ativado ou não, com base na relevância da informação para o problema em questão. Isso possibilita que certas informações possam ser descartadas, introduzindo uma não-linearidade nos modelos e permitindo que a rede aprenda a fazer representações complexas de dados. A seguir serão comentados alguns dos modelos mais utilizados de funções de ativação.

**3.1 Função de escada – Função binária**

Figura 3 - Gráfico de função binária

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

Fonte: Site v7labs

Este é um modelo simples, restringindo as saídas das redes neurais em valores binários [0, 1]. Esta saída tem como influência a superação de um valor predeterminado, definido como um limiar. Ao receber uma informação que atinja ou passar esse limiar, será retornado 1 e 0 caso contrário. Essa função é normalmente utilizada em perceptrons simples.

**3.2 Função Linear**

Figura 4 - Gráfico da função linear

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Fonte: Site v7labs

Este modelo também é um dos modelos mais simples de funções de ativação, onde a saída é proporcional a sua entrada. Ao falar em termos matemáticos, pode-se dizer que ao utilizar funções lineares nas camadas de uma rede neural, a rede inteira seria resumida em uma combinação linear de entradas. Assim como o modelo passado, esse modelo limita a capacidade de aprendizado de padrões complexos presentes nos dados.

**3.3. Função sigmoidal**

Figura 5 - Gráfico da função sigmoidal

**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

Fonte: Site v7labs

Esta função opera de forma semelhante a uma função binária, mas com a diferença na maneira como ocorre a mudança de estados. Em vez de uma transição abrupta de 0 para 1, a transição é suavizada. A aplicação desse modelo resulta em uma saída não linear, o que permite à rede aprender limites de decisão mais complexos e assumir valores intermediários entre 0 e 1.

**Aplicações**

**1. Carros autônomos**

Em carros autônomos, as redes neurais são treinadas para reconhecer padrões e tomar decisões com base em dados sensoriais recebidos de câmeras, radares, LiDARs e outros sensores.

Nessa etapa, iremos analisar a utilização de redes neurais do artigo "Learning a Driving Simulator" realizado pela empresa Comma.ai [4]. O produto dessa empresa é a transformação de carros comuns em carros autônomos, fazendo a integração de carros tradicionais a câmeras, sensores e um software embarcado que contém a inteligência artificial que irá fazer a automatização.

O artigo explora uma abordagem para criar um simulador de direção que possa prever sequências de vídeo realistas de estradas usando técnicas de aprendizado de máquina. O objetivo é desenvolver um agente que possa aprender a simular eventos futuros na estrada e ajudar no planejamento de manobras para carros autônomos. A abordagem se baseia em variações de autoencoders e redes neurais recorrentes condicionadas por ações.

Autoencoder é um tipo de rede neural usada principalmente para aprendizado não supervisionado, com o objetivo de aprender a codificar dados de entrada em uma representação de menor dimensão e depois reconstruí-los de volta para algo semelhante ao original.

O primeiro passo é usar um autoencoder para reduzir a dimensionalidade das imagens de vídeo capturadas pela câmera do carro. Isso envolve converter as imagens em uma representação compacta em um espaço latente, resultando em imagens que mantêm detalhes importantes, como marcas na estrada e bordas de carros, sendo esses detalhes utilizados no processamento da rede neural.

Com as informações fornecidas do treinamento do autoencoder, a rede neural irá atuar no espaço latente criado utilizando uma rede neural recorrente para responder a ações específicas do carro. Isto visa a previsão de estados futuros no espaço latente para conseguir prever as ruas e as aproximações de carros.

**Discussão**

A discussão deve estar bem embasada e considera os resultados do autor e artigos da literatura atuais.

**Conclusão**

Submeta seu artigo respeitando a data limite para tal, que será rigorosamente respeitada.

A próxima seção ilustra o formato a ser seguido para referências de livros, teses e obras completas; capítulos de livros; periódicos; anais de congressos e publicações eletrônicas.

**Referências**

[3] CARDON, André; MÜLLER, Daniel Nehme; NAVAUX, Philippe. **Introdução às redes neurais artificiais**. Porto Alegre, 1994.

[2] FLECK, Leandro et al. Redes neurais artificiais: Princípios básicos. **Revista Eletrônica Científica Inovação e Tecnologia**, v. 1, n. 13, p. 47-57, 2016.

[1] PINHEIRO, A. Redes neurais artificiais. Disponível em: <https://medium.com/@augusto_Pinheiro/redes-neurais-artificiais-133de77c7240>. Acesso em: 31 maio 2024.

[4] SANTANA, E.; HOTZ, G. **Learning a Driving Simulator**. 2016. Disponível em: <http://arxiv.org/abs/1608.01230>. Acesso em: 6 jun. 2024.