Redes Neurais Convolucionais no Apoio ao Diagnóstico: Aplicações Práticas e Considerações Éticas

Disciplina: Tópicos Avançados em Informática Biomédica

Prof. Regente: Filipe Santana

Mestranda: Dieine Estela Bernieri Schiavon







Roteiro

Primeira Parte da Aula: conceitos teóricos

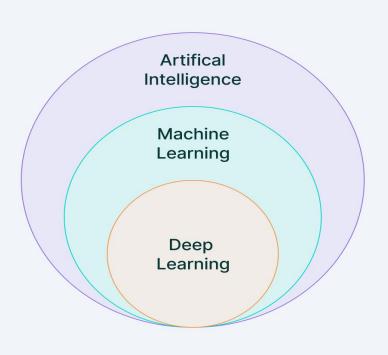
- Introdução: Inteligência Artificial, Machine Learning e Deep Learning
- Deep Learning no Apoio ao Diagnóstico
- Arquitetura de uma Rede Neural Convolucional
- Considerações Éticas
- Desafios

Roteiro

Segunda Parte da Aula: prática

- Construindo uma Rede Neural Convolucional multiclasse
- Classificando tumores cerebrais com uso de bibliotecas (Tensorflow e Keras)
- Verificando as predições do modelo

IA x Machine Learning x Deep Learning

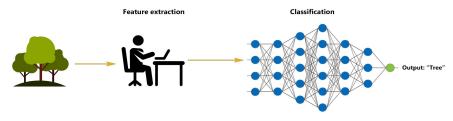


- IA: O esforço para automatizar tarefas intelectuais normalmente realizadas por humanos (Chollet, 2018)
- ML: Subcampo da IA, utiliza algoritmos para reconhecer padrões em conjuntos de dados e aprender com eles, melhorando a performance ao longo do tempo
- DL: Subcampo do ML baseados em Redes Neurais Artificiais

Ref: Adaptado de Chollet, 2018

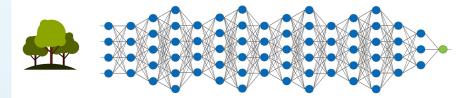
Machine Learning x Deep Learning

Machine Learning



Deep Learning

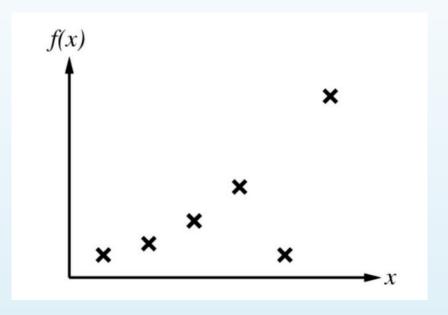
Feature extraction + Classification



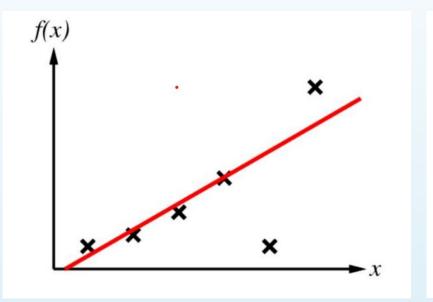
Diferença??

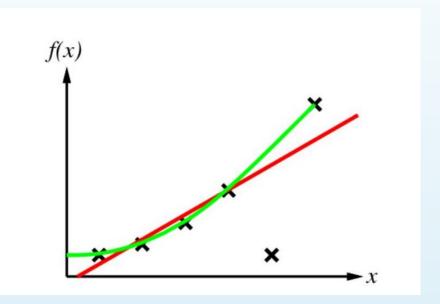
Fonte: https://primo.ai/index.php?title=Deep_Learning

Aprendizado como uma Busca por Hipóteses
 Procurar no espaço de hipóteses pela hipótese capaz de descrever as relações entre os objetos e que melhor se ajuste aos dados

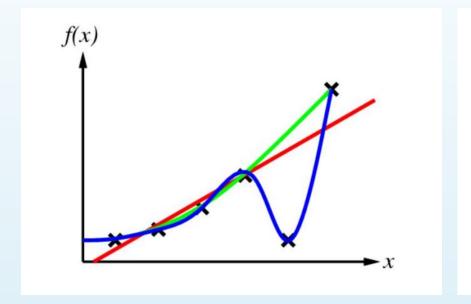


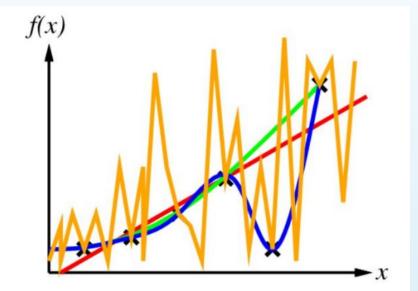
 Aprendizado como uma Busca por Hipóteses
 Procurar no espaço de hipóteses pela hipótese capaz de descrever as relações entre os objetos e que melhor se ajuste aos dados



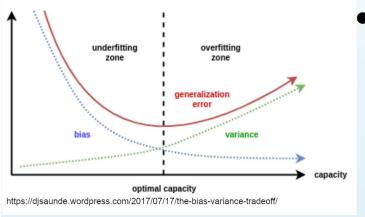


 Aprendizado como uma Busca por Hipóteses
 Procurar no espaço de hipóteses pela hipótese capaz de descrever as relações entre os objetos e que melhor se ajuste aos dados





- Viés (bias) hipótese indutiva: extrair regras gerais a partir de exemplos específicos. O modelo de IA aprende automaticamente a partir dos dados fornecidos.
- "O viés é necessário para que ocorra o aprendizado/generalização"



Generalização: extrapola padrões de exemplos conhecidos para fazer previsões sobre novos dado.

- O "conhecimento" de uma Rede Neural está todo contido nos parâmetros aprendidos, que são os pesos e o viés.
- Os pesos são multiplicados por cada sinal enviado pelos seus respectivos neurônios, e o viés é adicionado como:

$$y(x) = w \cdot x + b$$

Onde:

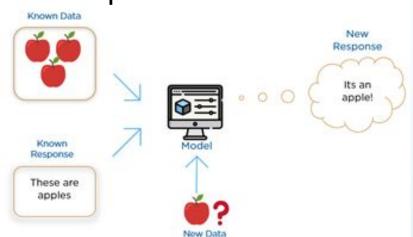
w é o peso

b é o bias (viés)

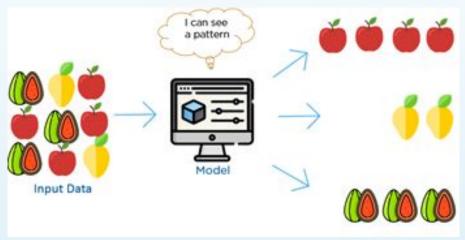
x é a entrada (inputs)

Tipos de Aprendizado de Máquina

Supervisionado

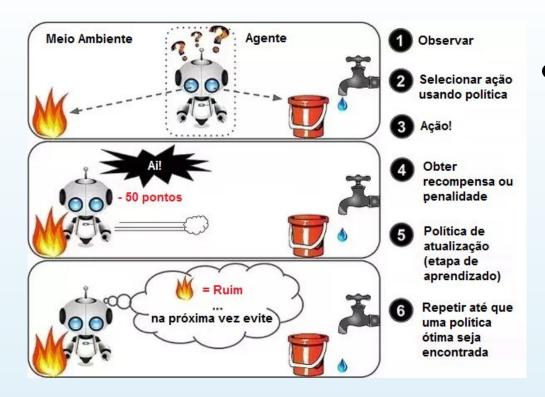


Não Supervisionado



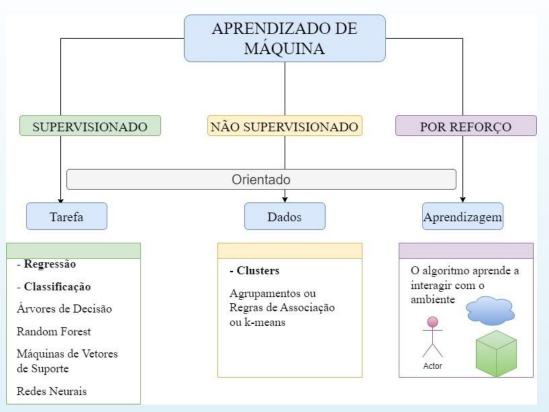
Fonte: https://www.redesdesaude.com.br/aprendizado-de-maquina-machine-learning/

Tipos de Aprendizado de Máquina



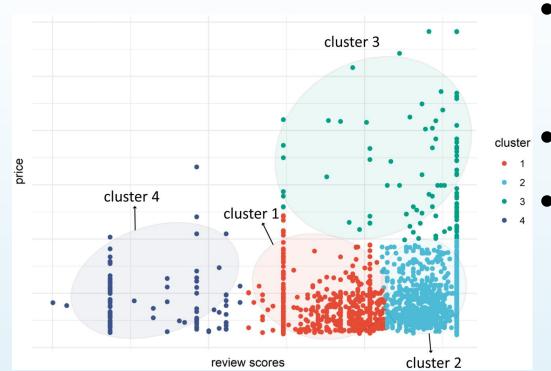
Aprendizado Por Reforço

Tipos de Algoritmos de Machine Learning



Fonte: Adaptado de https://www.elsevier.com/es-es/connect/ehealth/inteligencia-artificial-y-salud-tipos-de-algoritmos-de-machine-learning

Não Supervisionado: Clusters



- De uma forma geral, queremos encontrar uma representação + informativa dos dados
- Condensar a informação em pontos + relevantes
- Exemplos de uso na saúde: detecção de anomalias, previsão de epidemias, descoberta subtipos de doenças, etc...

Fonte: https://www.sciencedirect.com/science/article/pii/S026427512200364X

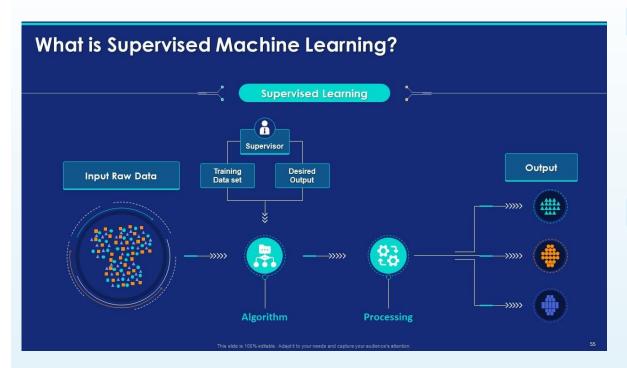
Não Supervisionado: Regras de Associação

Transaction 1	9 9 %
Transaction 2	(4) (9) (9)
Transaction 3	
Transaction 4	(4)
Transaction 5	Ø 🕑 🦠
Transaction 6	∅ 🐌 ⊜
Transaction 7	∅
Transaction 8	Ø 0

- Tanto X quanto Y podem estar na mesma prateleira
- Os descontos promocionais poderão ser aplicados a apenas 1 dos produtos
- Anúncios em X podem ser direcionados aos compradores que compram Y

Fonte: https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html

Supervisionado

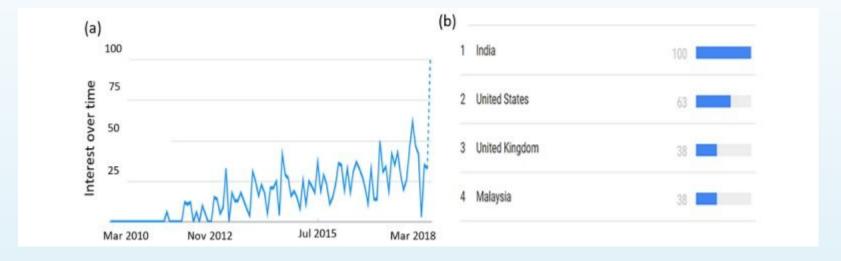


- O termo supervisionado vem da simulação da presença de um "supervisor externo", que conhece a saída (rótulo) desejada para cada exemplo.
- Com isso o supervisor externo pode avaliar a capacidade de hipótese induzida de predizer o valor de saída para novos exemplos.

Fonte: https://www.slideteam.net/artificial-intelligence-machine-learning-deep-learning-ppt-powerpoint-presentation-slide-templates.html

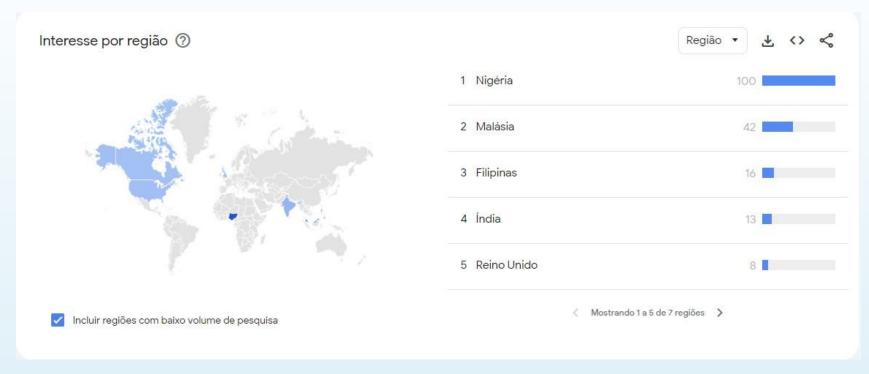
Deep Learning no Apoio ao Diagnóstico

O avanço contínuo no monitoramento da medicina, do genoma, da farmacêutica e da saúde é resultado do desenvolvimento e aplicação de dispositivos tecnológicos. Isso tornou possível capturar facilmente dados para análise e processamento. Da mesma forma, o aprimoramento da tecnologia também permite armazenar grandes quantidades de dados com informações úteis. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6696854/



Deep Learning no Apoio ao Diagnóstico

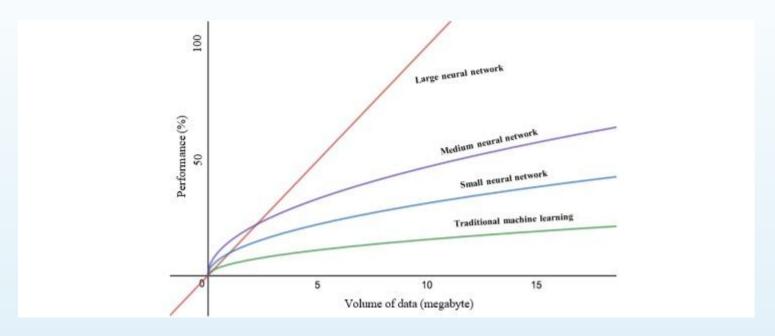
Pesquisa realizada no Google Trends: big data in healthcare em outubro/2023 (últimos 5 anos)



Fonte: https://trends.google.com/trends/

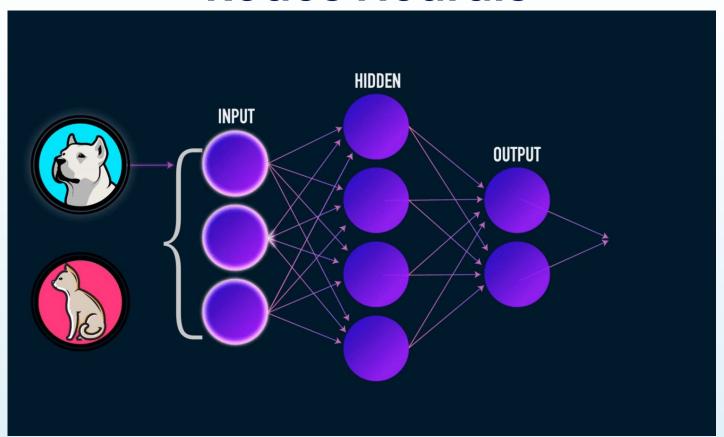
Deep Learning no Apoio ao Diagnóstico

Como as técnicas de aprendizado de máquina aumentam de acordo com a quantidade de dados



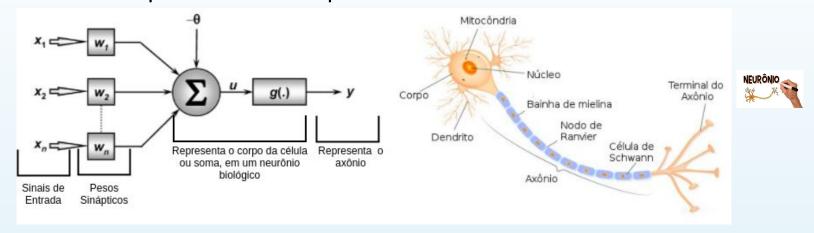
Fonte: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6696854

Redes Neurais



Redes Neurais

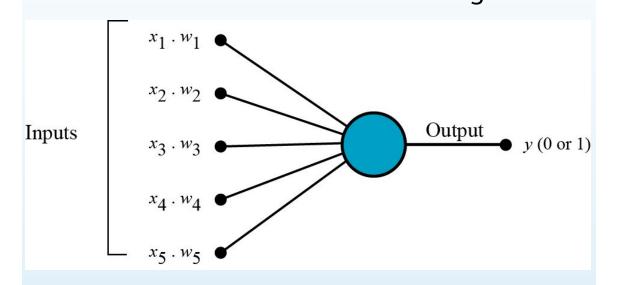
 As Redes Neurais são a unidade funcional do Aprendizado Profundo e são projetadas para imitar o comportamento do cérebro humano a fim de resolver problemas complexos baseados em dados

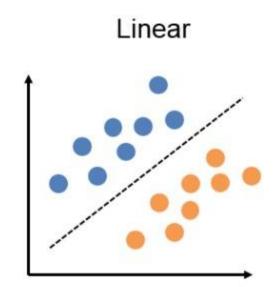


- Perceptron é uma rede neural de camada única
- Multi-Layer perceptron é um tipo de Rede Neural Artificial

Perceptron

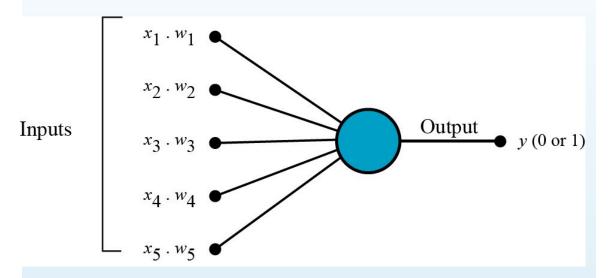
 Um perceptron é essencialmente um método para definir um limite linear para separar um espaço n-dimensional em 2 regiões: Classe Positiva ou Classe Negativa

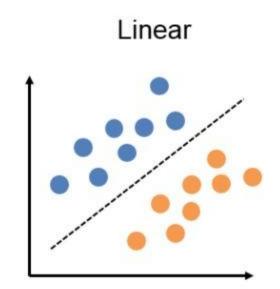




Perceptron

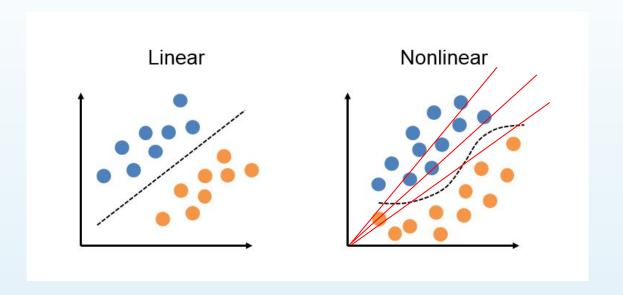
 Segue o modelo "feed-forward", que significa que as entradas são enviadas para o neurônio, processadas resultando em uma saída



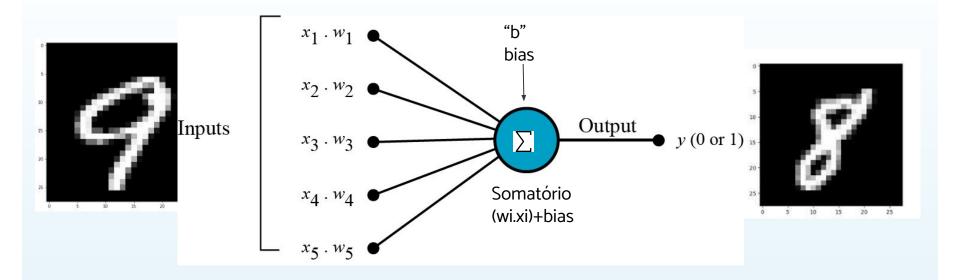


Neurônio / Perceptron

Mas nem todos os problemas têm soluções Lineares

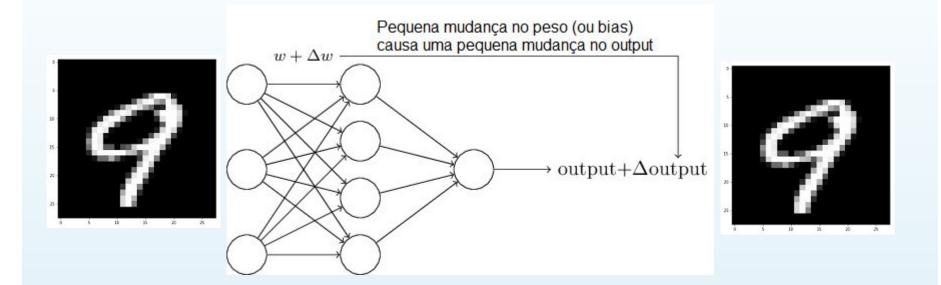


Neurônio / Perceptron



E se eu fizer uma pequena mudança nos pesos (ou bias)? Resolve o problema?

Neurônio / Perceptron

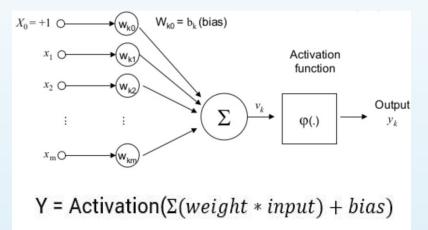


Função de Ativação

 Introduz não linearidade no funcionamento dos perceptrons para considerar a linearidade variável com as entradas

• É extremamente importante nas RNAs pois basicamente decidem se um neurônio vai ser ativado ou não. Ou seja, se a informação que o neurônio está recebendo é relevante ou se

deve ser ignorada

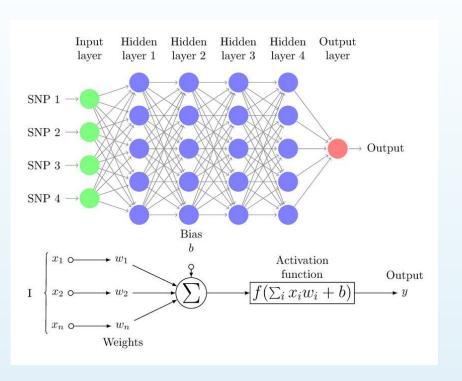


Função de Ativação

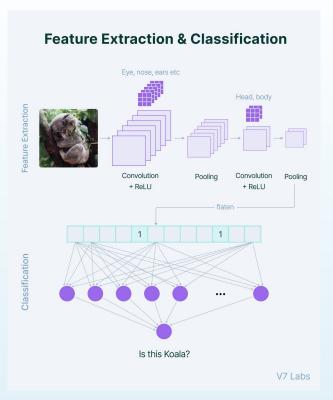
Commonly Used Activation Functions	Range
1. Step function: $f(z) = \begin{cases} 0 & z < 0 \\ 1 & z \ge 0 \end{cases}$	₹0,1}
2. Signum function: $f(z) = \begin{cases} -1 & z < 0 \\ 0 & z = 0 \end{cases}$	{-1,1}
3. Linear function: f(z) = x	(-∞,∞)
4. ReLU function: f(z) = \(\frac{5}{2} \) \(\frac{2}{2} \) \(\f	(0,0)
5. Sigmoid function: $f(z) = \frac{e^x}{1+e^x}$	(0,1)
6. Hyperbolic tan: $tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	(-1,1)
by Dr. Pankaj Kumar Porwal (BTech - IIT Mumbai, PhD - Cornell University) : Principal, Techno India NJR Institute of Technolog	gy, Udaipur

Redes Neurais Artificiais

- Multi-Layer Perceptron (MLP)
- Camada de entrada
- Camadas ocultas (Hidden layers)
- Camada de saída



Redes Neurais Convolucionais



- Utilizadas em problemas de Visão Computacional (imagens, vídeo) e áudios.
- Foram adaptadas com sucesso para tarefas de PLN (Transformers)

Fonte: https://www.v7labs.com/blog/neural-network-architectures-guide

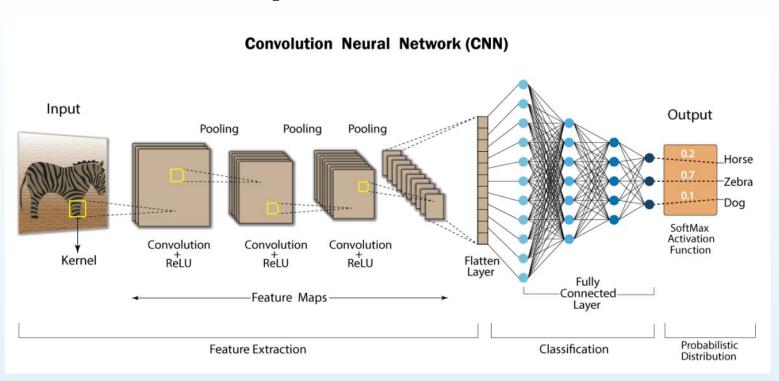
Arquitetura da RNC



- Camadas Convolucionais
- Função de Ativação
- Camadas de Pooling
- Flatten
- Dense (Totalmente conectadas)
- output

Fonte: https://www.v7labs.com/blog/neural-network-architectures-guide

Arquitetura RNC



https://developersbreach.com/convolution-neural-network-deep-learning/

Inputs - Visão Computacional

Imagens RGB e GrayScale



Fonte: https://cloudinary.com/cookbook/convert_image_to_grayscale

Vídeos



Fonte: https://support.apple.com/pt-br/HT207587

Input: Áudio





A Proposal for a Convolutional Neural Network for Classification of Alzheimer's Disease Through Features Extracted from Speech



Recordings

Dieine Estela Bernieri Schiavon¹, Carla Diniz Lopes Becker¹
⁽¹⁾ Federal University of Health Sciences of Porto Alegre, Brazil



INTRODUCTION

Background: Alzheimer's disease, a progressive neurodegenerative pathology, is among the most common forms of dementa in the elderly. Memory alterations are a frequent symptom, and speech and language alterations are also frequently mentioned, as can already be noticed in the initial phase of the disease. Deep machine learning is usual in extracting speech features to detect early conjuitive decline. This study aims to propose a Convolutional Neural Network model for classifying Alzheimer's disease through features extracted from speech recordings.

METHODS

- A Convolutional Neural Network model was created based on the architecture of EfficietNet-B5, but more simplified, using fewer layers.
- □ We tested the model of speech recordings containing pauses in the individuals' speech. For comparison purposes, the final model was tested by removing pauses.
- It was two databases, DementiaBank's Pitt Corpus with 478 samples, and ADReSS20, which is a subset of the first, with 4077 samples.
- □ Audios were converted into Mel Spectrograms, using the Librosa library (python package for music and audio analysis). After conversion, the spectrograms were saved in the png extension.
- For the final output of the binary classifier, the following labels were considered: with Alzheimer's disease or without Alzheimer's disease.
- ☐ The performance training uses a 5-fold stratified cross-validation (CV) method, with 30 epochs in each fold. The batch size used was 64, and the optimizer Adam whose learning rate was set to 0.00001 for all experiments.
- After training, the best fold, considering the learning curves, was identified, and save the weights. With the best fold weights, the model was retrained with the training data (80%) and validated (20%) using the holdout technique.
- Subsequently, the pre-trained EfficientNet-85 network for the transfer learning approach was used for the final result.

RESULTS

☐ **Proposed CNN:** model inspired by EfficientNet-B5 architecture. The details of the layers and their order in the proposed model are in Figure 1.

☐ After defining the architecture, crossvalidation was performed in several experiments until the model reached a satisfactory result. The best folder learning curves for this approach are in Figure 2.

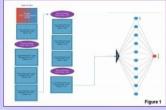
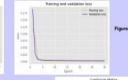


Figure 2a September 25 of the september 25 of

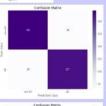


☐ Metrics to the Proposed Model

Metrics	With Pauses	Without Pauses
Accuracy	91.91%	90.26%
Sensitivity	98.03%	85.71%
ROC Curve	0.90	0.91

☐ Results of the pre-trained EfficientNet-B5 network

Metrics	With Pauses	Without Pauses
Accuracy	97.37%	95.39%
Sensitivity	97.04%	98.03%
Precision	97.04%	92.13%
Specificity	97.62%	93.27%
F-score	97.04%	94.09%
ROC Curve	0.97	0.96





CONCLUSIONS

- This study proposes a Convolutional Neural Network to support diagnosing patients with Alzheimer's disease.
- □The authors point out important limitations regarding the number of data available for analysis. The experience and knowledge acquired with Convolutional Neural Networks state that performance will increase, avoiding overfitting, with a considerable increase in the number of samples available for training. Alternatively, pretrained networks on large datasets helps to improve classification results when the database is small.
- □The authors encourage further research considering pauses in connected speech in Alzheimer's patients and the use of a Convolutional Neural Network for binary classification.
- ☐The initial results studied here proved promising and could be explored in future studies for lowcost, easy-to-perform, and non-invasive biomarkers detecting Alzheimer's disease.

REFERENCES

 [1] Tan M, Le QV. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks [Internet]. arXiv.org. 2019.
 Available from: https://arxiv.org/abs/1905.11946

[2] Becker JT. The Natural History of Alzheimer's Disease. Archives of Neurology. 1994 Jun 1;51(6):585.

[3] Luz S. Haider F, de la Fuente S, Fromm D, MacWhinney B. Alzheimer's Dementia Recognition through Spontaneous Speech: The ADReSS Challenge. arXiv:200406633 [cs, eess, stal] [Internet]. 2020 Aug S. Available from: https://arxiv.org/abs/2004.06833

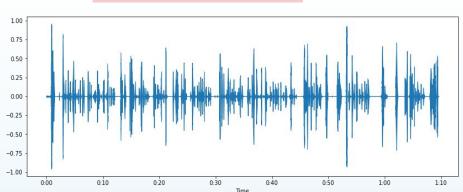
[4] McFee B, Metsai A, McVicar M, Balke S, Thome C, Raffel C, et al. librosa/librosa: 0.9.2 [Internet]. Zenodo. 2022 [cited 2022 Dec 26]. Available from:

https://zenodo.org/record/6759664#.Y6n8x-zMK00

Dieine Bernieri Schiavon e-mail: dieineb@ufcspa.edu.br Carla Diniz Lopes Becker, PhD e-mail: carladiniz@ufcspa.edu.br

Sons da fala gravados em áudio mp3

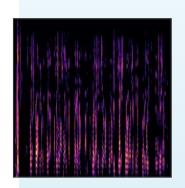
Waveform (Forma de onda)

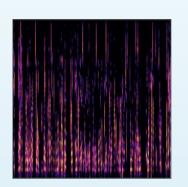


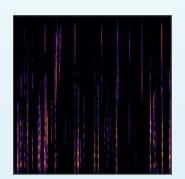
Fonte: https://www.dxomark.com/how-we-test-smartphone-audio-recording/

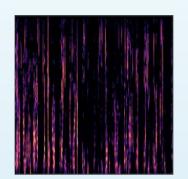
Fonte: própria

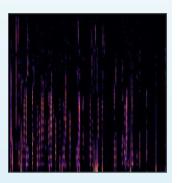
Espectrogramas Transformados em imagem.png para entrada na RNC











Fonte: própria

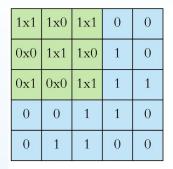
1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

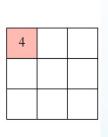
1	0	1
0	1	0
1	0	1

Input

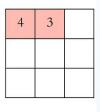
Filter / Kernel

- A operação de convolução associa dois sinais a um terceiro sinal
- A conv é aplicada aos dados de entrada usando um filtro para produzir um mapa de características (feature maps)





1	1x1	1x0	0x1	0
0	1x0	1x1	1x0	0
0	0x1	1x0	1x1	1
0	0	1	1	0
0	1	1	0	0



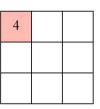
Input x Filter

Feature Map

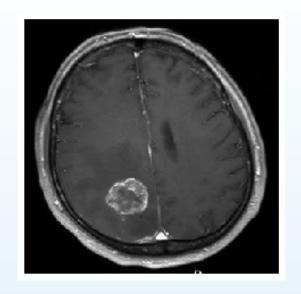
Input x Filter

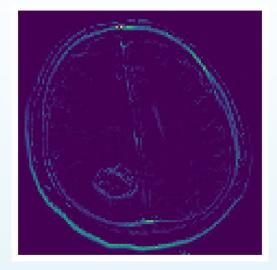
Feature Map

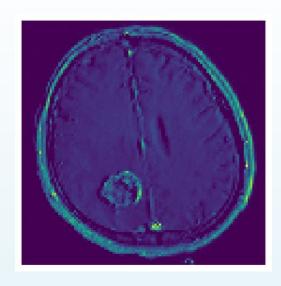
1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0



https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2

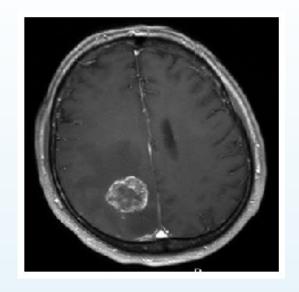


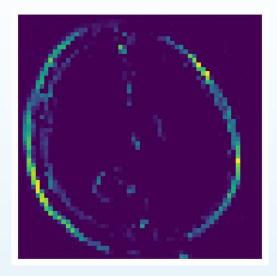


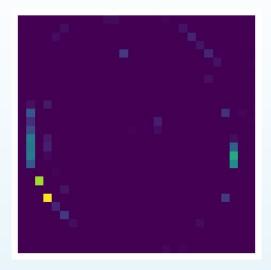


Original Image

First Conv. Layer Channel 3 First Conv. Layer Channel 7



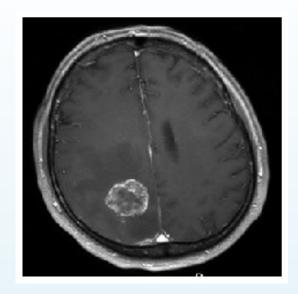


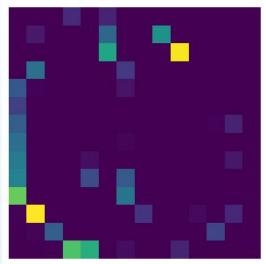


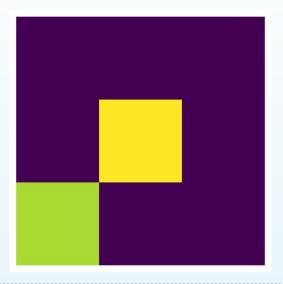
Original Image

Second Conv. Layer Channel 28

Third Conv. Layer Channel 63







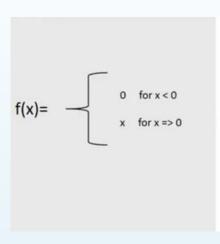
Original Image

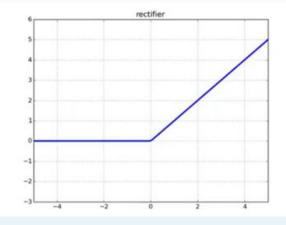
Fourth Conv. Layer Channel 6

Sixth Conv. Layer Channel 45

Função de Ativação

A função ReLU (Rectified Linear Unit)





As funções de ativação servem para trazer a não-linearidades ao sistema, para que a rede consiga aprender qualquer tipo de funcionalidade.

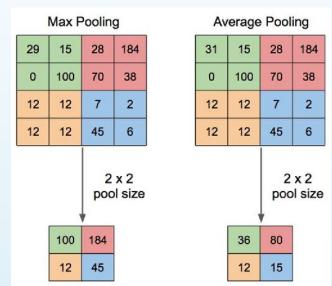
Essa função zera todos os valores negativos da saída da camada anterior.

Camada Pooling

Reduz a dimensionalidade da imagem

 O agrupamento de camadas reduz a resolução de cada mapa de características, reduzindo a altura e largura, mas mantendo a

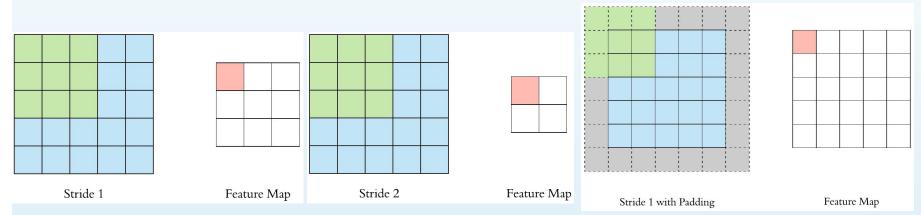
profundidade



Fonte Image,: https://www.researchgate.net/figure/Illustration-of-Max-Pooling-and-Average-Pooling-Figure-2-above-shows-an-example-of-max_fig2_333593451

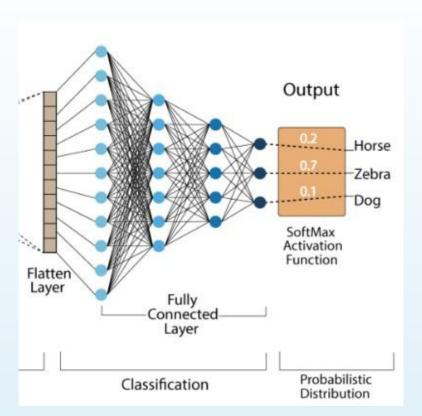
Stride e Padding

- Stride especifica quanto o filtro de convolução se move em cada etapa.
 Por padrão o valor é 1. Stride maior que 1 reduz a dimensionalidade da imagem resultante
- **Padding 'same'** mantem a mesma dimensionalidade, podemos usar preenchimento para cercar a entrada com zeros e 'valid' é sem preenchimento, reduzindo a dimensionalidade da imagem resultante



https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2

Camadas totalmente conectadas



- Após as camadas de convolução e pooling as características/features de alto nível são achatadas em um vetor unidimensional e passadas para as camadas totalmente conectadas (Dense) (neurônios)
- Essas camadas combinam as características de alto nível e fazem a classificação final (output)

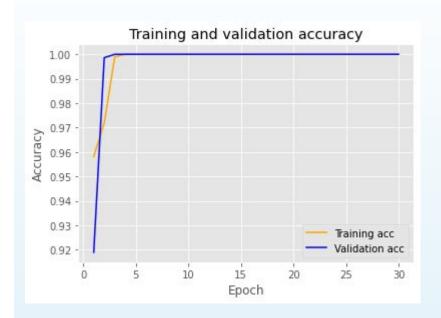
Implementação - Exemplo

Arquitetura e hiperparâmetros de uma Rede Neural Convolucional para classificação Binária

```
model = Sequential()
#First Layer /Primeira Camada = Imagem de entrada (224, 224, 3)
model.add(Conv2D(filters=32, kernel size=(5,5), padding='same', activation='relu', input shape=(224, 224, 3)))
model.add(MaxPool2D(pool size=(2,2)))
#Second Layer / Seaunda camada Convolucional
model.add(Conv2D(filters=64, kernel size=(3,3), padding='same', activation='relu'))
model.add(MaxPool2D(pool size=(2,2)))
#Third Layer / Terceira camada Convolucional
model.add(Conv2D(filters=128, kernel size=(3,3), padding='same', activation='relu'))
model.add(MaxPool2D(pool size=(2,2)))
#Flatten / Vetorização
model.add(Flatten())
model.add(Dropout(0.40))
# Fully Conected / Camada Totalmente Conectada
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.40))
#Output layer/ Camada de Saída
model.add(Dense(1, activation='sigmoid'))
#Adam Compiler/ Compilador Adam - Learning rate= 0.001
model.compile(optimizer = 'adam', loss = "binary crossentropy", metrics = ["accuracy"])
model.summary()
```

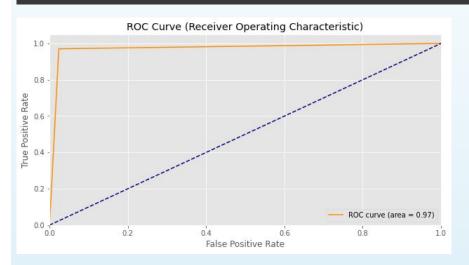
Layer (type)	Output	Shape	Param #
conv2d_3 (Conv2D)	(None,	224, 224, 32)	2432
max_pooling2d_3 (MaxPooling2	(None,	112, 112, 32)	0
conv2d_4 (Conv2D)	(None,	112, 112, 64)	18496
max_pooling2d_4 (MaxPooling2	(None,	56, 56, 64)	0
conv2d_5 (Conv2D)	(None,	56, 56, 128)	73856
max_pooling2d_5 (MaxPooling2	(None,	28, 28, 128)	0
flatten_1 (Flatten)	(None,	100352)	0
dropout_2 (Dropout)	(None,	100352)	0
dense_2 (Dense)	(None,	128)	12845184
dropout_3 (Dropout)	(None,	128)	0
dense_3 (Dense)	(None,	1)	129
Total params: 12,940,097			
Trainable params: 12,940,097 Non-trainable params: 0			

Curvas de Aprendizagem



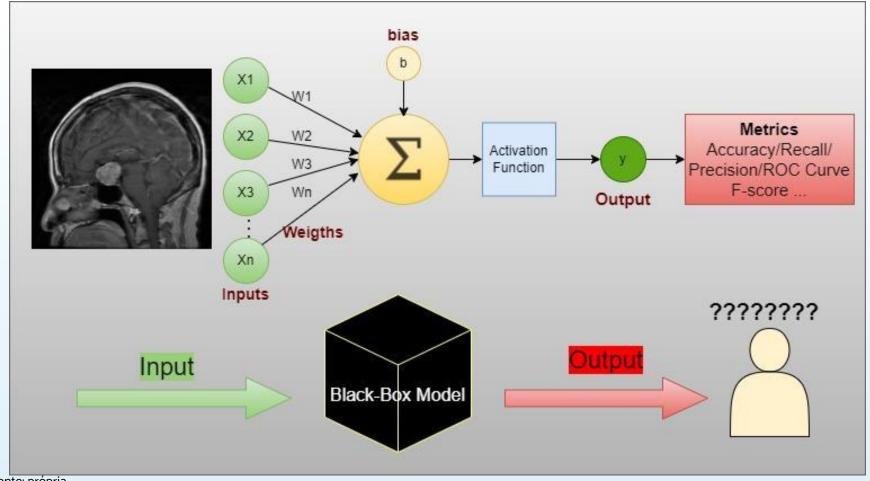


Métricas



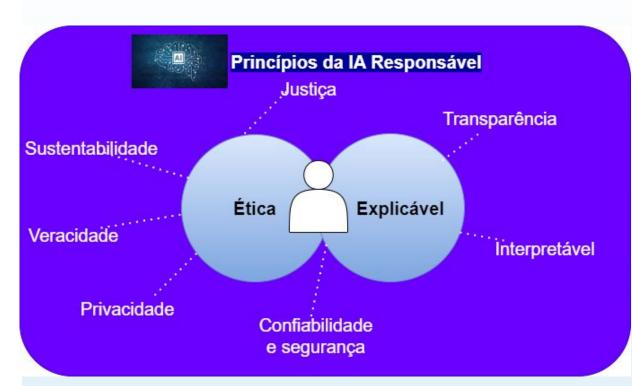


Fonte: própria



Fonte: própria

Ética na Inteligência Artificial



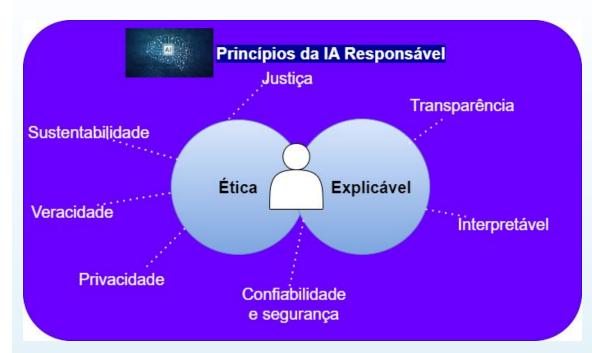
Justiça de IA é o tratamento igualitário de indivíduos em algoritmos.

Confiabilidade é a precisão de um sistema ou processo na produção de resultados confiáveis.

A **veracidade** na IA refere-se à precisão e confiabilidade das informações produzidas e processadas pelos sistemas de IA.

Fonte diagrama: adaptado de https://learn.microsoft.com/pt-br/azure/cloud-adoption-framework/innovate/best-practices/trusted-ai Fonte da imagem Al:

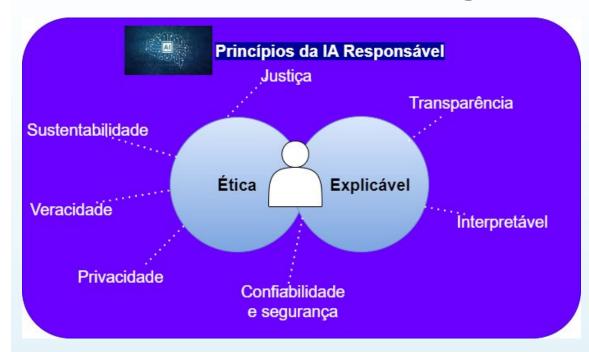
Ética na Inteligência Artificial



o aumento da **privacidade** serve como proteção contra crimes cibernéticos, como roubo de identidade ou fraude financeira. A privacidade incentiva a confiança do público nos sistemas de IA.

A **sustentabilidade** dos sistemas de IA refere-se à concepção de algoritmos de uma forma que minimize o seu impacto ambiental ao longo do seu ciclo de vida.

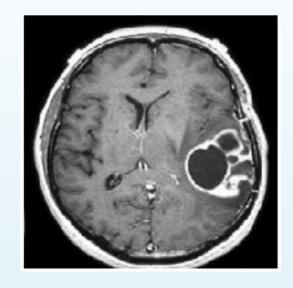
Ética na Inteligência Artificial

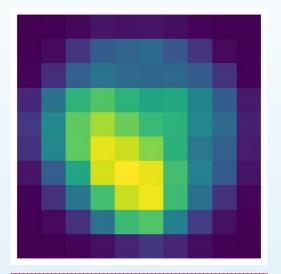


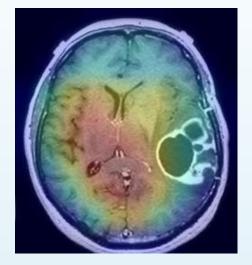
A **Transparência** apoia a ideia de que o uso e o desenvolvimento de tecnologias de IA devem ser transparentes para todas as partes interessadas. A transparência pode estar relacionada à 'transparência de uma organização' ou à 'transparência de um algoritmo'

Interpretabilidade do output do modelo

CAM (class activation visualization)





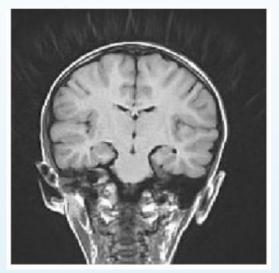


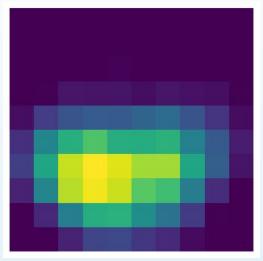
False Negative

Fonte: própria

Interpretabilidade do output do modelo

CAM (class activation visualization)





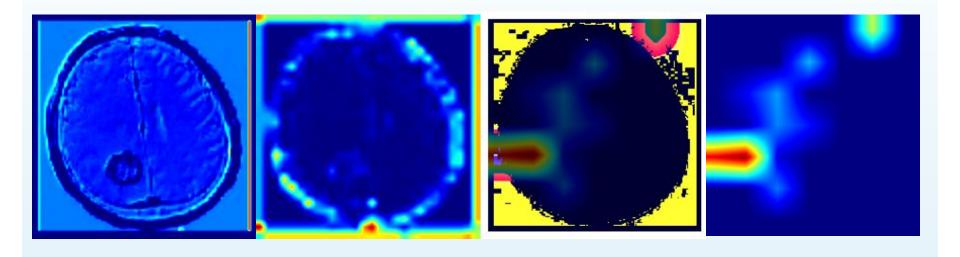


False Positive

Fonte: própria

Interpretabilidade do output do modelo

Grad-CAM (class activation visualization)



True Positive

Referências

- Deep Learning with Python, First Edition. François Chollet, 2017 Ed.
 Manning
- Deep Learning with Python, Second Edition. François Chollet, 2021 Ed.
 Manning
- <u>https://www.deeplearningbook.com.br/</u>
- Livro: Redes Neurais princípios e prática. Autor: Simon Haykin; tradução Paulo Martins Engel - 2ª edição, Porto Alegre: Bookman 2001
- Ética na IA implementações práticas:
 https://github.com/Nkluge-correa/teeny-tiny_castle
- Worldwide AI Ethics https://www.airespucrs.org/worldwide-ai-ethics

Informações e Contato

- Universidade Federal de Ciências da Saúde de Porto Alegre (UFCSPA)
 - PPG Tecnologias da Informação e Gestão em Saúde
- Contato
 - dieineb@ufcspa.edu.br
 - https://github.com/dieineb
 - https://br.linkedin.com/in/dieine-bernieri-schiavon-01b586120







Dilemas Éticos no Desenvolvimento de Software

Convidamos você a participar do "Dilemas Éticos no Desenvolvimento de Software", uma pesquisa para profissionais e estudantes de Tl.

À medida que a tecnologia continua a permear todas as facetas das nossas vidas, as considerações éticas que rodeiam o desenvolvimento tecnológico tornam-se cada vez mais significativas. Esta pesquisa tem como objetivo aprofundar a intrincada rede de desafios éticos que os desenvolvedores encontram em sua busca pela inovação.

Obrigado por se juntar a nós para desvendar os dilemas éticos no desenvolvimento de software.

Para participar clique no Link: https://nkluge-correa.github.io/AIRES-Ethical-Dilemmas/

Projeto RAIES (Rede de IA Ética e Segura) apoiado pela FAPERGS - (Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul), Brasil. https://www.raies.org/

Obrigada!





