
Aula 9 - Classificação de imagens com redes convolucionais

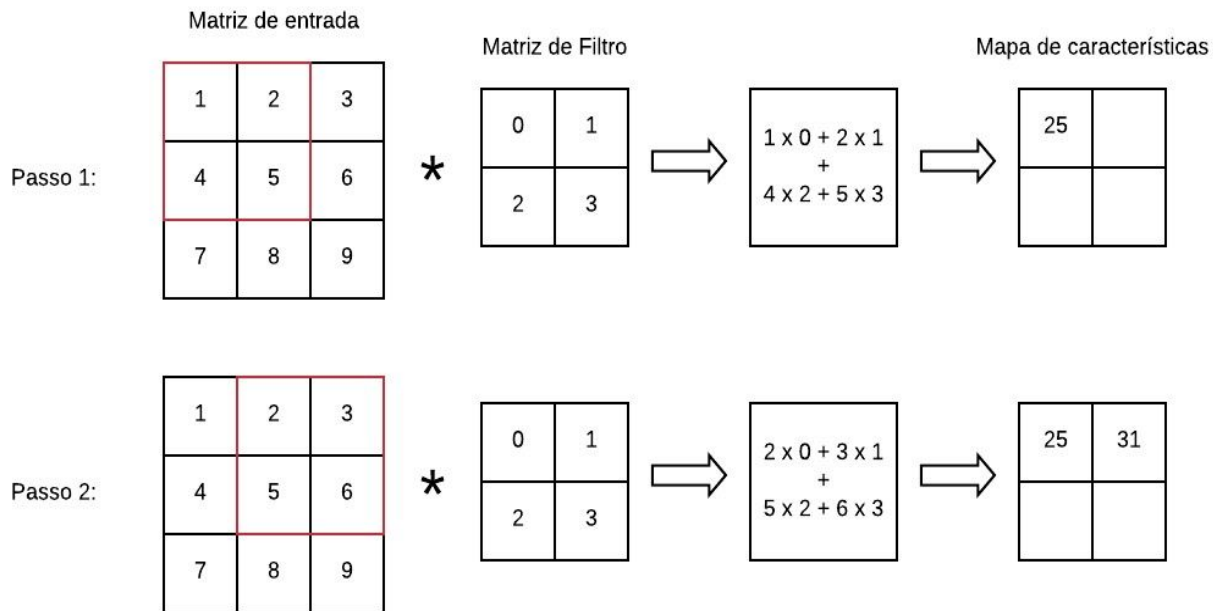
AGENDA

- Redes Neurais Convolucionais
- Redes Pré-treinadas
- Data Augmentation

Redes Neurais Convolucionais

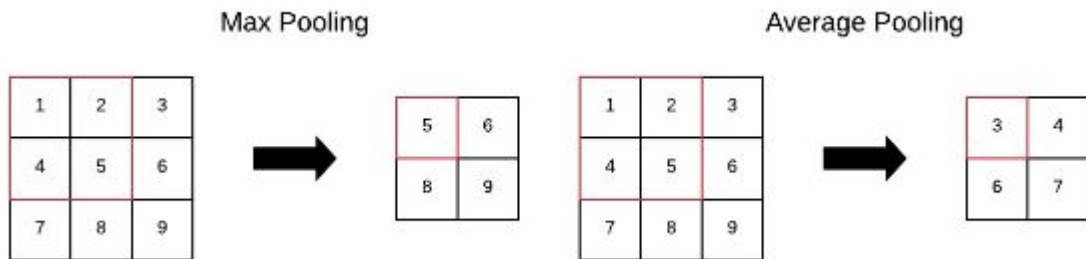
- Utilizadas com dados que possuem topologia de grade e dependência espacial como séries temporais (1D) e imagens (2D).
- Aplicação de um filtro sobre os dados de entrada para extração de características.
- A operação de convolução aplicada sobre a entrada ocorre de modo que uma matriz de filtro(kernel) é sobreposta à matriz de entrada em passos denominados strides.

Redes Neurais Convolucionais



Redução de dimensionalidade - Pooling

- Os mapas de características extraídos pelas CNN possuem alta dimensionalidade o que torna as operações matemáticas ao longo da rede custosas computacionalmente.
- Uma abordagem para contornar isso é a redução de dimensionalidade por meio da aplicação da operação de *pooling*.



Redes Pré-treinadas

- Para que se possa extrair as características das imagens de maneira eficaz é comum o empilhamento de uma alta quantidade de camadas de convolução nas redes.
- Durante o treinamento de uma rede convolucional os valores dos filtros aplicados são otimizados de modo que se possa obter extratores de características da vez mais acurados.
- Esses dois fatores fazem com que o treino de uma rede robusta demore um longo período de tempo e exija grande poder computacional.
- Nesse cenário que se encaixam as redes pré-treinadas.

Redes Pré-treinadas

- O uso de uma rede convolucional pré-treinada consiste em utilizar um modelo que já foi treinado e já tem filtros bastantes otimizados capazes de extrair com eficácia as características.
- Nesses casos toda a arquitetura da modelo é reutilizada de modo que apenas a última camada, responsável pela classificação, é alterada para classificar um novo conjunto de dados.
- Desta forma apenas os pesos da última camada são ajustados o que torna o treinamento bem mais rápido.

Redes Pré-treinadas

- Algumas arquiteturas de redes pré-treinadas podem ser utilizadas por meio da biblioteca Tensor Flow.
- Os modelos disponíveis foram treinados no dataset ImageNet que possui 1000 classes.
- Alguns dos modelos disponíveis são:
 - VGG16
 - VGG19
 - ResNET50
 - Xception

Redes Pré-treinadas

- A forma geral de se implementar esta solução é a seguinte:
 - Primeiramente é instanciado um modelo sequencial do Tensor Flow onde as camadas da nossa rede serão adicionadas.
 - Em seguida todas do modelo pré-treinado, no caso a VGG16, são adicionadas com exceção do topo, onde ocorre a classificação de fato. Neste passo é especificado que o modelo utilize os pesos do *dataset* ImageNet.
 - Por fim é adicionada a camada totalmente conectada com dois neurônios(duas classes) que será responsável pela classificação.

Redes Pré-treinadas

- Após a montagem do modelo necessário configurar alguns parâmetros através da função *compile*.
- Parâmetros:
 - optimizer: define o otimizador que será utilizado durante o processo de treino para aperfeiçoamento dos pesos.
 - loss: função de perda, utiliza-se a *binary_crossentropy* para classificação binária e a *categorical_crossentropy* para multiclasse. É essa função que será otimizada.
 - metrics: lista de métricas que serão avaliadas durante o treino.

Redes Pré-treinadas

- Após a definição dos parâmetros a rede será então treinada.
- Para o treinamento a função *fit()* do modelo é executada recebendo como parâmetros o conjunto de imagens de treino, o conjunto de imagens de validação e o número de épocas.
- O número de épocas define a quantidade de etapas que a rede será treinada, um número alto pode levar a rede a se sobreajustar aos dados e um número baixo pode levar a um aprendizado deficiente. O ideal é avaliar valores diferentes até que se consiga um número aceitável em termos de acurácia e tempo de treino.

Data Augmentation

- Quando se tem um conjunto de imagens pequeno ou com muita semelhança entre as instâncias pode acarretar na obtenção de um modelo deficiente.
- Para contornar esse problema existe uma técnica bastante utilizada para aumentar a variabilidade de imagens no conjunto com base nas imagens originais.
- Para isso são aplicadas um conjunto de transformações nas imagens originais de maneira aleatória, gerando assim "novas" imagens para compor o *dataset*.

Data Augmentation

- Dentre as transformações que podem ser utilizadas estão:
 - Rotação: a imagem sofre uma rotação baseado em valor aleatório em grau.
 - Zoom: é aplicado uma porcentagem aleatória de Zoom da imagem com base em um intervalo pré-definido.
 - Deslocamento Horizontal/Vertical: As imagens são deslocadas de forma aleatória em 20% nos eixos X e Y.
 - Giro Horizontal/Vertical: As imagens são giradas no eixo X e/ou eixo Y.

GRATIDÃO!

