

PROJETO 4 - APLICAÇÃO DE UMA CNN PARA RECONHECIMENTO DE DÍGITOS MANUSCRITOS

Fernanda Macedo de Sousa - 17/0010058

Mariana Alencar do Vale - 16/0014522

Resumo: Este relatório apresenta a implementação e análise do resultado da implementação de uma Rede Neural Convolutacional (CNN) para reconhecimento de dígitos manuscritos, como quarto projeto da disciplina Introdução à Inteligência Artificial.

Palavras-chave: AI; Machine Learning; Python; CNN; Keras; MNIST;

1 Introdução

Uma Rede Neural Convolutacional, ou CNN (Convolutional Neural Network) é um tipo específico de rede neural artificial normalmente utilizada no processamento e análise de imagens digitais. Uma convolução é um filtro que passa por uma imagem, a processa e extrai os recursos importantes.

Diante disso, o quarto projeto da disciplina consiste em usar e testar variações de uma rede CNN implementada no *framework* Keras, para reconhecimento de dígitos manuscritos, a partir da base do MNIST. Com a finalidade de responder algumas questões relacionadas à comparação de resultados de acordo com diferentes cenários e como essas diferenças afetam o resultado final de classificação. O modelo explicativo utilizado encontra-se no seguinte [link](#).

2 Materiais e métodos

O projeto foi realizado através do [Google Colab](#) com a linguagem de programação Python, de forma que os métodos não foram separados tão rigorosamente, mas foi seguido o processo comumente usado na implementação de um modelo de *Machine Learning* através do uso de um *notebook*. O arquivo “Como abrir o trabalho pelo Google Colab.pdf” apresenta instruções para a execução do *notebook* por essa plataforma.

O projeto foi dividido em três partes. Na primeira, realiza-se uma criação de uma rede neural normalizada e uma rede neural não-normalizada, ambas com 10% dos dados da “train_set” como validação e 90% para treinamento, ambas com 3 camadas convolucionais. O fator retirado para não realizar a normalização das imagens é a re-escala da imagem para valores entre (0.0,1.0).

Na segunda parte, compara-se a já criada rede neural normalizada com “train_set” dividido entre 90% para treinamento e 10% validação com uma nova rede neural normalizada implementada, com “train_set” dividido entre 70% treinamento 30% validação, ainda com 3 camadas convolucionais.

Na última parte, foram criadas mais duas redes neurais normalizadas, com divisão de 90% para treinamento e 10% para a validação, porém uma delas possui apenas uma camada convolucional e a outra possui duas.

3 Resultados quadro, gráficos e figuras

Test loss: 0.0345 accuracy: 0.9925

Imagem 1 – Resultado da CNN normalizada com 10% dos dados usados para validação e 90% para treino

Test loss: 0.0448 accuracy: 0.9902

Imagem 2 – Resultado da CNN não normalizada com 10% dos dados usados para validação e 90% para treino

Test loss: 0.0366 accuracy: 0.9915

Imagem 3 – Resultado da CNN normalizada com 30% dos dados usados para validação e 70% para treino

Test loss: 0.0683 accuracy: 0.9850

Imagem 4 – Resultado da CNN normalizada com apenas uma camada convolucional, utilizando 10% dos dados usados para validação e 90% para treino

Test loss: 0.0372 accuracy: 0.9917

Imagem 5 – Resultado da CNN normalizada com duas camadas convolucionais, utilizando 10% dos dados usados para validação e 90% para treino

4 Análise de Resultados

As variações propostas pelo projeto apresentaram, como o esperado, resultados diferenciados.

A primeira se trata da diferença entre uma rede neural normalizada e uma não normalizada. Como pode ser notado pela **Imagem 1** e **Imagem 2**, o desempenho da rede neural normalizada é superior daquela não normalizada. Isso se dá pois a não normalização acrescenta um efeito adicional sobre o neurônio, o que leva a decisões erradas. Normalização tende a fazer o processo de treinamento ter um comportamento melhor ao melhorar as condições numéricas dos dados, neste caso, torna-las todas da mesma proporção.

O treinamento do modelo foi realizado em 2 modos, como descrito anteriormente, sendo: um com 70% dos dados para treinamento e 30% dos dados para validação; e outro com 90% dos dados usados para treinamento e 10% validação. Conforme pode ser observado na comparação entre os resultados apresentados na **Imagem 1** com os resultados apresentados na **Imagem 3**, uma CNN normalizada com a separação de 10% dos dados para validação e 90% para treinamento possui uma melhor acurácia do que a mesma implementação divergindo apenas na utilização dos dados, sendo 70% para treinamento e 30% para validação. Embora a diferença nesse caso seja pequena, é evidente que quanto maior o número de dados usados

para treinamento do modelo, maior será a acurácia. Sendo necessário, portanto, um equilíbrio da quantidade possível de dados a serem disponibilizados para treinamento, validação e teste.

Por fim, nota-se a diferença entre quantidade de camadas convolucionais aplicadas pela rede neural. Pode notar que quanto mais camadas convolucionais temos, maior a acurácia. As **Imagens 1, 4 e 5** representam a diferença dos resultados das camadas. Isso se deve a função das camadas convolucionais no contexto de redes neurais. Elas servem como um filtro para destacar detalhes da imagem, então quanto mais camadas tiver, mais detalhes serão identificados, ou seja, maior a profundidade da imagem.

5 Considerações Finais/Conclusões

Como podemos ver, das situações consideradas no projeto, a rede neural normalizada com três camadas convolucionais que utiliza a proporção 90 a 10 para o “train_set” é a que apresenta a melhor acurácia.

Referências Bibliográficas

[1] Base de dados utilizada: <https://keras.io/api/datasets/mnist/>

[2] O que é o Colaboratory? <https://colab.research.google.com/notebooks/intro.ipynb#>

[3] Medium (rede neural):
<https://medium.com/@mjbhobe/mnist-digits-classification-with-keras-ed6c2374bd0e>

[4] Redes Neurais: <https://www.mql5.com/pt/articles/497>

[5] <http://www.faqs.org/faqs/ai-faq/neural-nets/part2/>

[6] Medium (Processamento de imagens):

<https://medium.com/neuronio-br/entendendo-redes-convolucionais-cnns-d10359f21184#:~:text=O%20reconhecimento%20de%20imagem%20%C3%A9,entre%20convolu%C3%A7%C3%B5es%20e%20fully%20connected.>