



Análise da variação de parâmetros em uma rede neural convolucional para reconhecimento de dígitos escritos à mão

(Kelvin Clovis Montoli de Souza – Bolsista PDA)

Kelvin Clovis Montoli de Souza, discente de graduação, Universidade Federal do Pampa, Campus Bagé

Fábio Luís Livi Ramos, docente, Universidade Federal do Pampa

kelvinsouza.aluno@unipampa.edu.br

O uso de redes neurais nos últimos anos tem aumentado, parte desse aumento se deve às demandas das novas soluções tecnológicas, que acabam por ter necessidade na utilização de serviços que sejam autômatos e capazes de tomar múltiplas decisões em um espaço limitado de tempo. Com isto, o trabalho tem por objetivo realizar a análise de redes neurais convolucionais, que são amplamente utilizadas para reconhecimento de objetos (ou padrões) em imagens. Para realizar esta análise foram desenvolvidos 5 modelos de redes neurais convolucionais, sendo que estas redes neurais foram desenvolvidas para serem capazes de reconhecer dígitos escritos à mão e em base decimal. Com relação aos modelos desenvolvidos, todos eles possuem alguns parâmetros fixos, sendo eles: Filtro convolucional 3x3, possuem 5 camadas, utilizam a função de ativação ReLU da 1º à 4º camada e Softmax na 5º camada. Com relação ao ambiente em que a pesquisa foi desenvolvida, todos os modelos foram desenvolvidos utilizando a linguagem de programação Python e com o uso da API “Keras”, o treinamento das redes foi conduzido no ambiente “Google Colab”, utilizando os servidores da Google para processar os dados e efetivamente treinar os modelos desenvolvidos, além disso, todos os modelos gerados foram treinados utilizando um dataset de 60.000(28x28x1) imagens para treino e 10.000 imagens para realizar o teste da rede, sendo que o é chamado de teste, é algo realizado de forma autônoma durante o processo de treino, em que, por exemplo, uma destas 10.000 imagens é inserida na rede e é avaliado se o resultado da última camada da rede desenvolvida realmente corresponde ao que se espera da imagem. É dado como acerto, por exemplo, se uma imagem que contenha o número 5 for apresentada a rede e ela efetivamente conseguir identificar o 5. Sobre os 5 modelos desenvolvidos, eles foram subdivididos em 2 grupos, sendo que o primeiro grupo(modelos de 1 a 3) tem por objetivo avaliar o impacto na variação de feature maps por camada para modelos que possuem a mesma quantia de stride e o segundo grupo(modelos 4 e 5) tem por objetivo avaliar o impacto na variação do parâmetro stride. O que será avaliado no resultado será o tempo de treinamento da rede e a acurácia do modelo. Tendo em mente o grupo um, em que o parâmetro “stride” se manteve constante, foi possível perceber que o tempo de treinamento da rede poderia ser reduzido severamente se a ordem da quantidade de feature maps fosse alterada. Também era possível notar um aumento no tempo de treinamento e perda na acurácia no caso de alteração. Com relação à estas afirmações, são dadas pois os modelos 1 e 3 possuem 753 feature maps, contudo, o tempo de treinamento do modelo 1 foi de 8m44s, enquanto do modelo 3 foi de 11m57s e acurácias de 99.34% e 99.05% . Outro fator que foi possível observar é que a redução no número de feature maps ocasionou em um modelo(2) com acurácia extremamente semelhante ou até mesmo melhor (de acordo com qual modelo estivesse sendo comparado(3)), atingindo esta acurácia com um tempo de treinamento menor quando em comparação aos outros dois modelos, tempo de treinamento do modelo 2 foi de 3m42s, usando apenas 362 feature maps, obtendo uma acurácia de 99.19% . Considerando agora o grupo 2, do qual lidava com 2 modelos com uma diferença no parâmetro “stride”, foi possível perceber que este parâmetro está severamente interligado a acurácia e ao tempo de treinamento do modelo, de modo que o parâmetro segue a seguinte relação: quanto menor ele for, maior será a acurácia do modelo, porém maior será o tempo de treinamento do mesmo, analogamente, quanto maior for o salto, menor será a acurácia e menos tempo de treinamento será necessário, por exemplo, modelo 4 precisou de 26m21s para o treinamento e teve uma acurácia de 99.63%. Os resultados obtidos neste grupo são dado como esperados, visto que stride é o parâmetro que se refere à quantidade de “saltos” que um filtro convolucional deve realizar ao percorrer a imagem, contudo, o que foi interessante observar é o quão

impactante este parâmetro consegue ser, dado que o modelo 4 foi o modelo de maior acurácia entre todos os modelos, porém de maior tempo de treinamento. Concluindo, foi possível analisar que ao desenvolver um modelo de rede neural para reconhecimento de dígitos escritos à mão, é necessário ter uma atenção especial para o parâmetro stride e levar em consideração se a acurácia que será ganha por um uso reduzido do stride realmente será ganha, visto que o tempo de treinamento aumenta severamente e além do tempo de treinamento maior, é necessário ressaltar que o tempo de execução deste modelo, está atrelado a quantidade de feature maps e a quantidade de strides utilizado, tendo este fator em mente e considerando os modelos apresentados, acredita-se que o modelo 2 seja o modelo mais eficiente, por usar um stride aceitável(2), ter uma estrutura pequena(362) e possuir pouca diferença de acurácia quando em comparação a modelos mais custosos.

Agradecimentos: UNIPAMPA – PDA, CNPq e UNIPAMPA

Palavras-chave: Deep Learning; Redes Neurais Convolucionais; IA; Detecção de imagens.