Redes convolucionais em classificadores de galáxias

Miguel Boing
Engenharia Eletrônica
19105113
miguel-boing@hotmail.com

***Resumo*—A classificação de corpos celestes como estrelas e galáxias é uma tarefa da astronomia tradicionalmente realizada apenas por astrônomos, porém com o avanço da tecnologia cada vez mais fotografias são tiradas com maior resolução, gerando bancos de dados gigantescos, tornando a tarefa de classificação tremendamente trabalhosa. Diversos esforços como o Galaxy Zoo foram realizados junto a comunidade para a classificação porém esta é uma solução imediata e como alternativa surgem modelos de deep learning que consigam realizar a tarefa. Este trabalho visa discutir o uso de arquiteturas famosas como Resnet e VGG para a classificação de galáxias usando os bancos de dados obtidos durante o Galazy Zoo 2.**

***Palavras Chave*—Redes convolucionais, Deep Learning, Galáxias, Galaxy Zoo.**

I. Introdução

No mundo da astronomia, uma tarefa recorrente é a classificação de objetos astronômicos, tais como estrelas e galáxias. Para tal classificação desenvolveu-se o modelo Hubble Tunning Fork (figura 1).

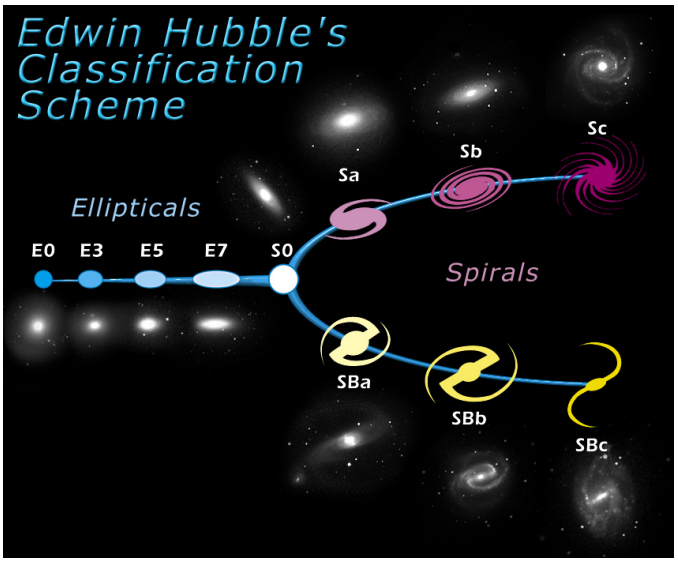

Figura 1. Esquemático de Classificação Morfológica em Garfo de Hubble.

Nesta classificação as galáxias são dividas em dois grandes grupos:[1]

- Galáxias em espiral: São subdividas em espirais barradas que contam com uma barra cruzando seu centro e espirais não barradas que não contam com esta característica.
- Galáxias elípticas: Apresentam formato elipsoidal.

Esta classificação é feita tradicionalmente por astrônomos para milhares de galáxias, porém com o avanço dos telescópios os bancos de dados tem crescido exponencialmente tornando esta tarefa impossível de ser feita apenas por cientistas. Inicialmente foram feitos modelos computacionais que buscavam automatizar etapas desta classificação, uma técnica muito utilizada relacionava a cor da galáxia com a sua forma, já que era mais fácil analisar o espectro de ondas eletromagnéticas do que o formato em si, de forma que galáxias azuis representam galáxias novas e a cor azulada se deve a berços de estrelas e estrelas jovens portanto as galáxias novas são geralmente espirais, já galáxias de cor vermelha tendem a ser mais velhas com estrelas no fim da idade e costumam ser elípticas. Esta classificação funciona até certo ponto, pois podem ser encontrados casos diferentes, onde a cor de uma galáxia não depende apenas de suas estrelas, mas também de gases e poeira estelar, estima-se que esse é o caso de aproximadamente 20% das galáxias observadas.[2]

A. Galaxy Zoo 1

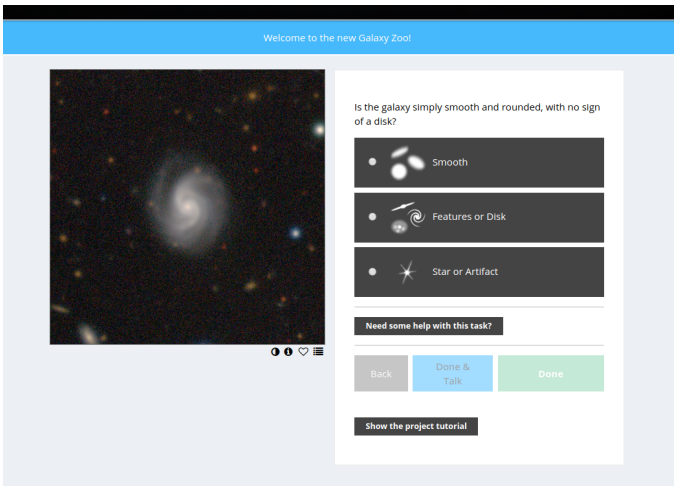

Figura 2. Exemplo de uma etapa de classificação de uma estrela.

Como solução surgiu a iniciativa Galaxy Zoo que busca usar a comunidade isto é, cidadãos comuns através de trabalho voluntário, para classificar as galáxias, como exposto na figura 2. A iniciativa foi um sucesso, tendo uma média de 60000 classificações por hora e tendo aproximadamente 8 milhões de classificações em 10 dias. Cada galáxia teve em média

38 avaliações e o método de decisão foi por maioria de votos, porém com diferentes usuários tendo diferentes pesos de voto, usuários mais assertivos (classificações iguais a maioria) recebiam pesos maiores.

B. Galaxy Zoo 2

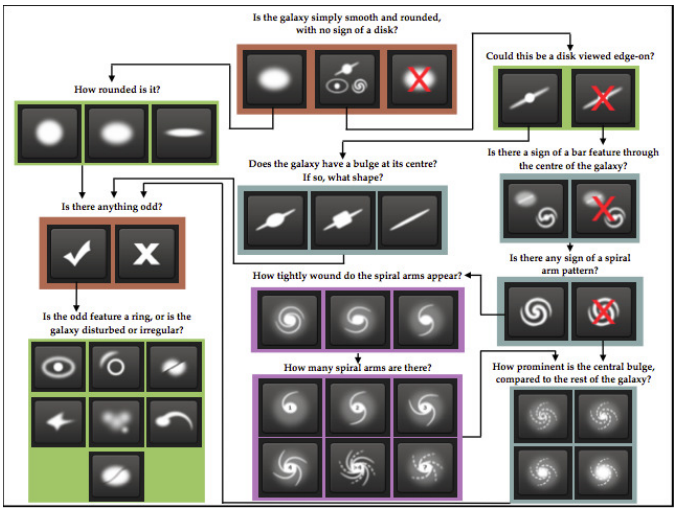

Figura 3. Árvore de classificação usada pelos voluntários durante o Galaxy Zoo 2.

Com o sucesso do primeiro projeto, surgiu Galaxy Zoo 2, com mais classificação para os voluntários e adição de imagens ao dataset. Deste projeto surgiram diversos artigos científicos possibilitados pelas classificações de voluntários, foram criados diversos fóruns pela comunidade para listar certos objetos incomuns, como galaxias interagindo, colorações verdes e outros objetos raros.

Apesar das enormes vantagens que projetos abertos para a comunidade como este trazem, o número de fotos nos bancos de dados continua a aumentar imensamente (hoje se encontra na casa de petabytes), e por tanto até mesmo a ajuda colaborativa da comunidade não será suficiente e o uso de modelos de machine learning serão necessários transformando o esforço humano em criar bancos de dados para treinamento destes modelos.

Este trabalho busca discutir a eficiência das redes convolucionais na classificação de galáxias através do dataset Galaxy Zoo. Utilizando modelos populares, discutir sua eficácia e testar diferentes estratégias, tais como comparações de desempenho e custo computacional.

II. Galaxy Zoo - The Galaxy Zoo Challenge

Para o desenvolvimento deste trabalho foram utilizados os datasets, as métricas e o modelo vencedor da competição The Galaxy Zoo Challenge para benchmarking. A competição foi realizada em 2013 pela empresa de ciência de dados Winton Capital com prêmio de 10 mil dólares para o primeiro lugar.[3]

A. Dataset

O dataset utilizado contém 61578 amostras divididas em 37 classes, obtidas da classificação feita pelos voluntários do Galaxy Zoo 2. Cada objeto é classificado por probabilidades de pertencer a cada uma das classes. Na figura 5 pode-se obter uma noção de como são distribuídas as probabilidades para algumas das classes de galáxias do dataset. Na figura 4 também é possível ver algumas das imagens contidas no dataset, observa-se também que algumas das imagens contém mais de um ponto luminoso, por isso, o objeto em questão será sempre o ponto centralizado da imagem.

Figura 4. Densidades de probabilidades usando estimativa de densidade de Kernel não paramétrica.

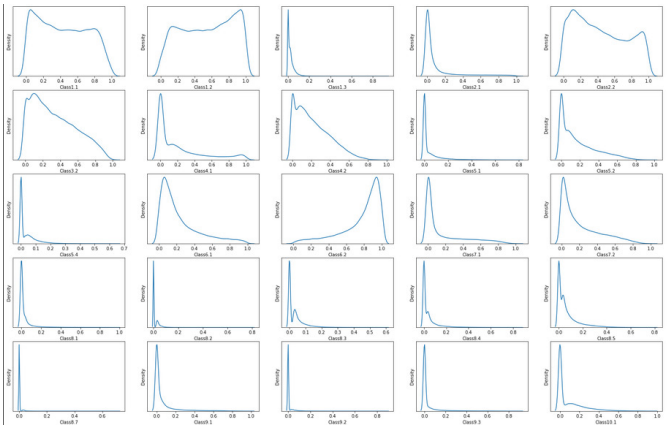

Figura 5. Densidades de probabilidades usando estimativa de densidade de Kernel não paramétrica.

B. Métricas de desempenho

A métrica utilizada na competição foi a RMSE (Root Mean Squared Error) e a mesma será mantida por motivos de comparação. Além disso os modelos desenvolvidos serão comparados entre sí em tempo de treinamento.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - a_i)^2}$$

C. Modelo de Benchmark

O modelo de benchmarking escolhido teve o maior score da competição RMSE de 0.09223 sobre o conjunto não rotulado, e sobre o conjunto de validação de 0.09234. Sua separação do dataset rotulado é feita apenas entre 80% treinamento e 20% validação. O batch size escolhido foi 64 e a resolução foi de 160 por 160 pixeis.

1) Estrutura do modelo de Benchmark: O modelo de benchmark conta com aproximadamente 3.4 milhões de parâmetros. Usa data augmentation com aplicações de zoom, rotações, inversões e mudanças de constraste.

2) Treinamento do modelo de Benchmark: O treinamento foi feito com learning rate constante de 0.001 durante 50 épocas, sem o uso de técnicas de callback e durou 7.4 horas usando GPU's do Kaggle.

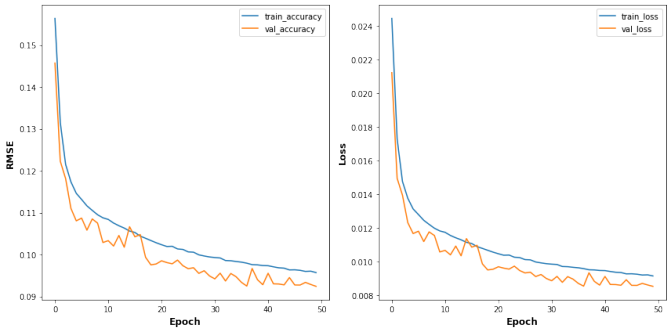

Figura 6. Gráfico demonstrando o RMSE e as perdas respectivamente dos conjuntos de treinamento e validação em relação a épocas treinadas no modelo de Benchmark.

III. Desenvolvimento dos modelos

Os modelos escolhidos para o desenvolvimento deste trabalho foram: VGG16, Resnet50 e Xception. O motivo desta escolha vem do fato de que as arquiteturas tem uma diferença de número de parâmetros - na casa das unidades de milhões - entre si, o que torna possível comparar a diferença entre treinamento e desempenho em relação ao custo de uma maior complexidade.

A. Subconjuntos

O dataset com 61578 amostras foi subdividido em datasets de treinamento com 36946 amostras, validação e testes ambos com 12316 amostras cada.

B. Data Augmentation

O uso de Data Augmentation foi utilizado para aumentar o desempenho dos modelos, mas foram tomados alguns cuidados pois como apresentado nas seções anteriores algumas características devem permanecer constantes como o formato, a cor, e a centralização das galaxias na foto. Por tanto, adotaram-se apenas camadas de rotação, zoom, contraste e espelhamento.

C. Output Layer

A camada de saída utilizada está ilustrada na figura 7 e conta com ativações do tipo sigmoid, pois apesar de ser um problema de classificação multiclasse, cada neurônio retornará uma classificação binária com uma probabilidade para cada uma das classes. As camadas intermediárias usam ativação relu.

D. Treinamento

O treinamento foi feito sob as mesmas condições do Benchmark: 50 épocas, batch size de 64, resolução 160 por 160 pixeis mas foi escolhido empiricamente um learning rate de 0.0001 onde nenhum dos modelos divergia. Visando reduzir overfitting e reduzir tempo de treinamento foram aplicados. callbacks:

- ReduceLROnPlateau: Usado com paciência de 4 épocas, reduzirá o learning rate, caso o modelo não obtenha melhor desempenho.[4]
- EarlyStopping: Para o treinamento caso não haja mais melhorias de desempenho durante 7 épocas.[4]

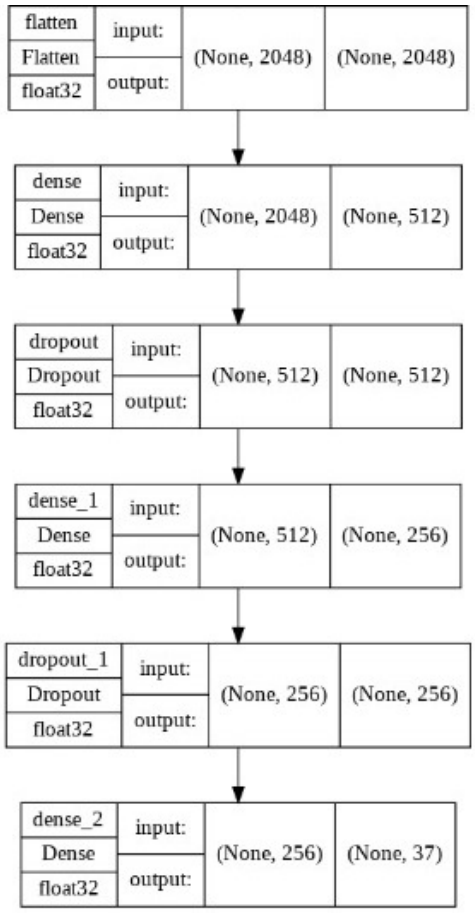

Figura 7. Camada de saída utilizada nos três modelos.

IV. Resultados

Os resultados estão apresentados na figura 11 onde é possível ver que o modelo VGG 16 teve desempenho de

0.0848 no conjunto de testes, Resnet teve desempenho de 0.0818 e Xception 0.0796. VGG 16 teve um total de 15 mihões de parâmetros e treinou por 2.14 horas. O modelo Resnet teve 25 milhões de parâmetros e treinou por 1.37 horas, já o modelo Xception treinou por 3.69 horas com 22 milhões de parâmetros. O histórico de treinamento dos modelos VGG 16, Resnet e Xception estão demonstrados nas figuras 8, 9 e 10 respectivamente.

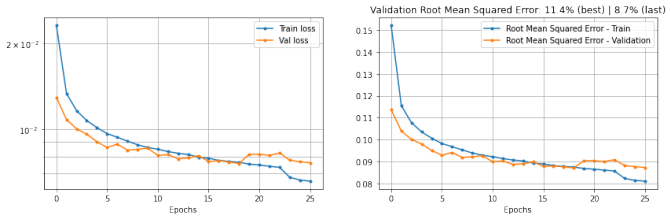

Figura 8. Treinamento do Modelo VGG 16.

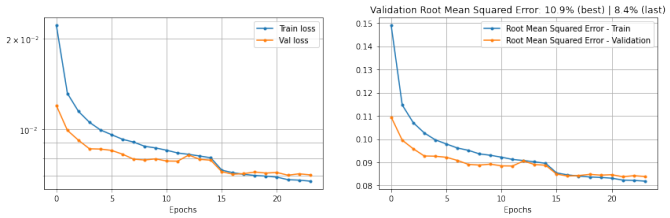

Figura 9. Treinamento do Modelo Resnet.

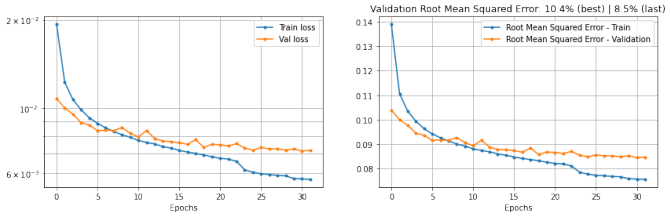

Figura 10. Treinamento do Modelo Xception

RMSE	Conjunto Treinamento	Conjunto de Validação	Conjunto de Teste	Parâmetros
Benchmark	0,0954	0,0924	-	3.460.501
VGG 16	0,0848	0,0849	0,0848	15.118.181
Resnet	0,0819	0,0821	0,0818	24.754.725
Xception	0,0797	0,0798	0,0796	22.051.405

Figura 11. Comparação entre resultados relacionando desempenho com número de parâmetros.

V. Conclusão

Percebe-se que ambos os três modelos tiveram um resultado significativamente superior ao modelo de Benchmark, porém deve-se levar em conta a quantidade de parâmetros utilizadas pois modelos mais complexos costumam levar mais tempo para serem treinados e tem custo computacional superior. Por conta da memória RAM limitada disponível para os treinamentos foi-se necessário utilizar diferentes seções de treinamento para cada um dos modelos e técnicas de transfer learning foram aplicadas para salvá-los.

Foram apresentados dados de duração de treinamento em horas, porém está informação deve ser bastante relativizada pois o treinamento dos modelos foram feitos usando a plataforma Google Colab e a cada inicialização de uma nova seção uma diferente placa de vídeo pode ser atribuída, mais rápida ou mais devagar. Técnicas como EarlyStopping e ReduceLROnPlateau se mostraram essenciais para o treinamento em um ambiente que conta com poder computacional reduzido, evitando não apenas overfitting mas custo computacional através de treinamento desnecessário.

Dentre os 3 modelos treinados VGG 16 conseguiu ótimos resultados considerando seu número de parâmetros, já o modelo Resnet teve o maior número de parâmetros porém teve resultados inferiores que o Xception, mostrando que o aumento do número de parâmetros nem sempre resultará num desempenho superior.

Por tanto, conclui-se que o uso de arquiteturas famosas de redes convolucionais são alternativas possíveis e necessárias para a classificação morfológica de galáxias uma vez que os dados aumentam com o avanço das tecnologias de sondas espaciais e telescópios tornando inviável a classificação humana, sendo a mesma usada apenas quando à necessidade de criar datasets rotulados para treinamentos de modelos.

Referências

[1] GALáXIA. In: WIKIPEDIA, the free encyclopedia. Wikimedia, 2022. Disponível em: <https://pt.wikipedia.org/wiki/Gal%C3%A1xia>. Acesso em: 01 agosto. 2022.

[2] In: ADVANCES in Machine Learning and Data Mining for Astronomy. CRC Press, 2012. Disponível em: <https://books.google.com.br/books?id=YafMBQAAQBAJ&printsec=frontcover&hl=pt-BR#v=onepage&q&f=false>. Acesso em: 09 jul. 2022.

[3] GALAXY Zoo - The Galaxy Zoo Challenge: Classify the morphologies of distant galaxies in our universe. Winton Capital, 2013. Disponível em: <https://www.kaggle.com/competitions/galaxy-zoo-the-galaxy-challenge/>.

[4] ABADI, M. et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. Software available from tensorflow.org. Disponível em: <https://www.tensorflow.org/>.