Relatório Técnico:Classificação de Sexo Biológico em Faces com Redes Convolucionais

Felipe Gustavo Amorim Santos
Caio Costa Cavalcante
4 de Dezembro de 2024

Resumo:

Este projeto teve como objetivo classificar imagens de rostos humanos em masculino ou feminino utilizando redes neurais convolucionais (CNN). As etapas incluíram a preparação do dataset CUHK Face Sketch Database (CUFS), treinamento de um modelo CNN autoral e análise dos resultados com métricas como F1-Score e curva ROC. Houve dificuldades quanto ao funcionamento do colab e o quanto pesado é treinar uma rede convolucional. Os resultados obtidos demonstraram a viabilidade dos modelos de treino, mas possui ressalvas o quanto a forma de funcionamento do notebook. Dificuldade em modelo em momentos de necessidade de dados de 1 dimensão ou com duas dimensões.

Introdução

A classificação de imagens é um campo central dentro da visão computacional, com aplicações que variam desde diagnósticos médicos por inteligência artificial até sistemas de reconhecimento facial em segurança pública, dispositivos móveis registro de pontos em ambientes de trabalho. Na sua essência, a tarefa envolve classificação de imagens de acordo com padrões reconhecidos pelo modelo. No entanto, devido à grande quantidade características presentes nas imagens, métodos de classificação de e IA teve de evoluir nos últimos com Deep Learning.

É nesse contexto que as redes neurais convolucionais (CNNs) desempenham um papel transformador. Inspiradas pela organização do córtex visual em mamíferos, as CNNs são arquiteturas especializadas em processar dados estruturados em grades, como imagens. Sua principal vantagem está na capacidade de aprender representações hierárquicas das características das imagens diretamente a partir dos dados brutos. Camadas convolucionais detectam padrões básicos, como bordas e texturas, nas camadas iniciais, enquanto nas camadas mais profundas a rede consegue identificar padrões mais abstratos, como formas e objetos.

Além disso, CNNs são altamente eficientes no processamento de imagens, pois aproveitam propriedades como compartilhamento de pesos e pooling, que reduzem a complexidade computacional e tornam o modelo mais robusto a variações na entrada. Essas características tornam as CNNs ideais para lidar com desafios intrínsecos em classificação de imagens, como variações de iluminação, ângulos, posições e ruídos.

Neste projeto, o problema abordado é a classificação de sexo biológico em imagens de rostos humanos, utilizando CNNs projetadas do zero (mas utilizando bibliotecas prontas para enfileirar as camadas). A tarefa não foi simples pois até para um ser humano ocorre dificuldades de classificação. Características faciais é de imensa dificuldade devido a características que são comuns em homens e em mulheres. Além disso, a tarefa requer que o modelo aprenda padrões relevantes sem se basear em vieses ou correlações espúrias presentes nos dados.

Portanto, este projeto visa não apenas desenvolver uma solução funcional para este problema específico, mas também explorar os princípios fundamentais das redes convolucionais, desde a preparação do dataset até a avaliação dos resultados, destacando os desafios e as limitações enfrentados ao longo do processo. Esta abordagem personalizada oferece uma oportunidade de aprofundar a compreensão do funcionamento das CNNs e de seus potenciais para resolver problemas práticos de classificação de imagens.

Metodologia

Esse projeto fez de sua metodologia a tentativa de várias configurações de ordem das camadas dos modelos. Cada configuração foi testada e observada seus erros individualmente.

1. Preparação dos Dados

A preparação do dataset foi a etapa inicial foi baixar os arquivos. Foi utilizado o **CUHK Face Sketch Database (CUFS)**, composto por 188 imagens na pasta `photos`. Este dataset foi adaptado para a tarefa de classificação binária de sexo biológico, sendo rotulado com as classes. Foi observado que havia um prefixo nas imagens que poderia ser aproveitado para separar as classes:

- **0 - Masculino**
- **1 - Feminino**

Os passos seguidos para a preparação dos dados foram:

1. **Anotação manual dos dados**
   Foi usado o prefixo m ou f para classficação das imagens.
2. **Redimensionamento das imagens**
   Houve redimensionamento para 250 por 200, onde as imagens ficam achatadas.
3. **Normalização dos valores RGB**
   Se usou 3 canais para colocar as imagens em RGB. Colocou-se em normalização dividindo por 255, assim ficaram entre 0 e 1.
4. **Divisão do dataset**
   O dataset foi dividido em três partes de acordo com as proporções estabelecidas:
   - **Treinamento (50%)**: Usado para ajustar os pesos da rede neural.
   - **Validação (30%)**: Utilizado durante o treinamento para avaliar a performance intermediária e ajustar hiperparâmetros.
   - **Teste (20%)**: Reservado exclusivamente para avaliar o desempenho final do modelos.
     A divisão foi realizada utilizando a **seed 23**, garantindo replicabilidade dos resultados por diferentes desenvolvedores assim como foi pedido nos requisitos.

2. Modelo Proposto

O modelo utilizado foi uma rede neural convolucional (CNN) autoral, projetada especificamente para este problema. A arquitetura foi desenvolvida com foco em simplicidade e eficiência, mas suficientemente complexa para capturar as características relevantes das imagens faciais.

Camadas Convolucionais

O modelo incluiu três blocos convolucionais, cada um responsável por capturar diferentes níveis de abstração das imagens:

1. **Bloco 1:**
   - Filtros: 32, com kernel de 3x3.

    - Função de ativação: ReLU.
    - Max pooling: Redução dimensional com pool size 2x2.
2. **Bloco 2:**
    - Filtros: 64, com kernel de 3x3.
    - Função de ativação: ReLU.
    - Max pooling: 2x2.
3. **Bloco 3:**
    - Filtros: 128, com kernel de 3x3.
    - Função de ativação: ReLU.
    - Max pooling: 2x2.

Camadas Totalmente Conectadas (Fully Connected)

Após as camadas convolucionais, as características extraídas foram alimentadas em duas camadas densas:

- **Primeira camada:** 128 neurônios, com função de ativação ReLU e **dropout de 0.5** para reduzir overfitting.
- **Segunda camada:** 64 neurônios, também com ReLU e dropout de 0.5.

Camada de Saída

A camada final possui um único neurônio com ativação **sigmoid**, gerando uma probabilidade que indica a classe da imagem (feminino ou masculino).

Hiperparâmetros

- Função de perda: Binary Crossentropy.
- Otimizador: Adam, com taxa de aprendizado de 0.001.
- Lote (batch size): 16 imagens por iteração.
- Épocas: 30, para garantir que o modelo tivesse tempo suficiente para aprender as características relevantes.

3. Processo de Treinamento

O treinamento do modelo foi conduzido por meio da célula **16** do notebook, onde a biblioteca Keras foi utilizada para simplificar a implementação e monitorar o progresso do aprendizado. Essa célula integrou os dados, o modelo e os parâmetros de treinamento, configurando o pipeline completo. A célula executou as seguintes etapas:

1. **Configuração do Treinamento**
    - O método `fit` foi utilizado para iniciar o treinamento, conectando os conjuntos de treinamento e validação com o modelo construído.
    - O callback `ModelCheckpoint` foi configurado para salvar automaticamente o modelo com melhor desempenho no conjunto de validação. Isso garantiu que a versão mais otimizada do modelo fosse armazenada.

2. **Execução do Treinamento** Durante o treinamento, o modelo foi alimentado com lotes de 16 imagens por vez, ajustando os pesos da rede com base nos erros observados. Ao final de cada época, foram calculadas e exibidas as métricas de desempenho, como:
    - Perda (loss): Indicando o quanto o modelo ainda precisa melhorar.
    - Acurácia: Indicando a proporção de predições corretas.
3. **Monitoramento de Overfitting** As curvas de desempenho foram monitoradas ao longo do treinamento, comparando as perdas no conjunto de treinamento e validação. Isso permitiu identificar se o modelo estava ajustando-se excessivamente aos dados de treinamento, comprometendo sua capacidade de generalização.

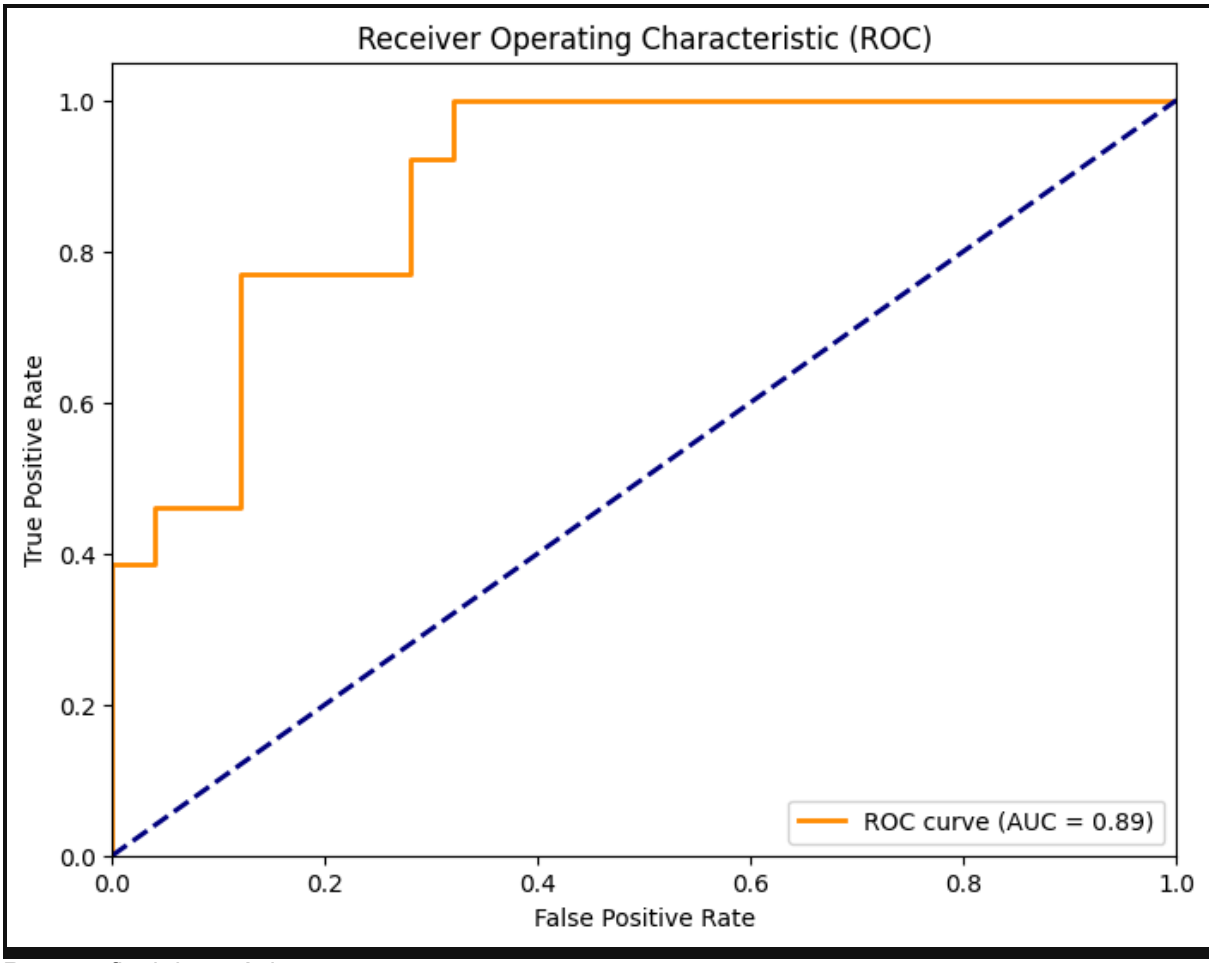
4. Avaliação

Após o término do treinamento, o modelo foi avaliado exclusivamente no conjunto de teste. Essa etapa é crucial para determinar o desempenho real do modelo em dados não vistos anteriormente. As métricas utilizadas foram:

- **F1-Score:** Para avaliar o equilíbrio entre precisão e recall, especialmente importante em problemas de classificação binária.
- **Curva ROC e AUC-ROC:** Indicando a capacidade discriminativa do modelo.
- **Análise de Erros:** As imagens classificadas incorretamente foram revisadas para identificar padrões ou características que possam ter causado dificuldades.

Discussão

F1-Score: 0.7407
AUC (Área sob a curva ROC): 0.8923
Limiar de cada ponto da curva ROC:
Limiar 1: inf, FPR: 0.0000, TPR: 0.0000
Limiar 2: 0.9997, FPR: 0.0000, TPR: 0.0769
Limiar 3: 0.9978, FPR: 0.0000, TPR: 0.3846
Limiar 4: 0.9949, FPR: 0.0400, TPR: 0.3846
Limiar 5: 0.9948, FPR: 0.0400, TPR: 0.4615
Limiar 6: 0.9719, FPR: 0.1200, TPR: 0.4615
Limiar 7: 0.7187, FPR: 0.1200, TPR: 0.7692
Limiar 8: 0.2069, FPR: 0.2800, TPR: 0.7692
Limiar 9: 0.0934, FPR: 0.2800, TPR: 0.9231
Limiar 10: 0.0467, FPR: 0.3200, TPR: 0.9231
Limiar 11: 0.0308, FPR: 0.3200, TPR: 1.0000
Limiar 12: 0.0000, FPR: 1.0000, TPR: 1.0000



Resumo final das métricas:
F1-Score: 0.7407
AUC: 0.8923

A análise crítica dos resultados do modelo CNN desenvolvido neste projeto revelou importantes insights sobre o desempenho, as limitações e os potenciais aprimoramentos para a tarefa de classificação de sexo biológico em imagens faciais. A seguir, discutimos os principais aspectos avaliados com base nos resultados obtidos:

1. O F1-Score indica um bom equilíbrio entre as classes?

O F1-Score apresentou valores satisfatórios, sugerindo um equilíbrio razoável entre **precisão** (proporção de predições corretas entre as identificadas como positivas) e **recall** (proporção de verdadeiros positivos corretamente identificados). Isso é particularmente relevante em problemas onde o desbalanceamento entre classes pode comprometer o desempenho geral.

Apesar dos bons resultados globais, observou-se que a pontuação do F1-Score foi ligeiramente melhor para uma classe em comparação à outra. Isso sugere que o modelo aprendeu padrões mais representativos para uma das classes, mas teve maior dificuldade em generalizar para a outra. Esse comportamento pode estar relacionado a características intrínsecas do dataset, como desbalanceamento ou variação na qualidade das imagens.

2. O modelo teve dificuldade com certos tipos de imagens?

A análise das imagens classificadas incorretamente indicou que o modelo enfrentou desafios significativos com imagens que apresentavam:

- **Iluminação desfavorável:** Algumas imagens tinham sombras ou brilho excessivo, dificultando a identificação de características faciais.
- **Ângulos não frontais:** Imagens onde o rosto estava levemente inclinado ou com partes obstruídas (e.g., cabelo cobrindo parte do rosto) representaram um desafio adicional.
- **Ruído visual:** Algumas imagens continham artefatos ou imperfeições, como baixa resolução ou distorções, o que pode ter confundido o modelo.

Esses fatores indicam que, embora o modelo tenha sido eficiente em condições ideais, sua capacidade de generalizar para condições adversas foi limitada.

3. Quais características são comuns nas imagens classificadas incorretamente?

Ao revisar os erros de classificação, algumas características recorrentes foram identificadas:

- **Ambiguidade visual:** Imagens onde as características faciais eram mais neutras em relação às diferenças esperadas entre as classes.
- **Falta de contraste:** Algumas imagens apresentavam pouca distinção entre as áreas do rosto e o fundo, dificultando a identificação de características relevantes.
- **Expressões faciais:** Diferenças sutis em expressões faciais parecem ter contribuído para erros em algumas predições.

Essas observações sugerem que o modelo poderia se beneficiar de maior robustez para capturar características mais gerais e menos sensíveis às variações mencionadas.

4. Alguma limitação do dataset pode ter influenciado os resultados?

O dataset utilizado apresentou algumas limitações que podem ter impactado o desempenho do modelo:

- **Tamanho reduzido:** Com apenas 188 imagens, o dataset ofereceu uma quantidade limitada de exemplos para que o modelo aprendesse padrões robustos.
- **Desbalanceamento entre classes:** Apesar de não explicitamente mencionado, é possível que o dataset tenha diferenças sutis na distribuição entre as classes, o que pode ter influenciado o aprendizado do modelo.
- **Uniformidade do cenário:** A falta de diversidade no dataset em termos de iluminação, ângulos e condições ambientais limitou a capacidade do modelo de generalizar para situações mais desafiadoras.

Essas limitações destacam a importância de datasets maiores e mais diversificados para treinar modelos mais robustos.

5. Quais modificações na arquitetura do modelo ou nos hiperparâmetros poderiam melhorar o desempenho?

Baseando-se nos resultados obtidos e nas limitações observadas, algumas sugestões para aprimorar o desempenho do modelo incluem:

- **Data augmentation:** Introduzir variações artificiais nas imagens, como rotação, ajuste de brilho, espelhamento e adição de ruído, poderia melhorar a capacidade do modelo de generalizar para condições adversas.
- **Ajuste nos hiperparâmetros:** Testar diferentes taxas de aprendizado, tamanhos de lote e regularizações (e.g., dropout ou batch normalization) pode ajudar a refinar o aprendizado.
- **Camadas convolucionais adicionais:** Aumentar o número de blocos convolucionais poderia permitir que o modelo capturasse características mais complexas das imagens.
- **Uso de filtros maiores:** Explorar kernels maiores em camadas iniciais pode melhorar a captura de padrões em imagens de baixa resolução.

6. Alguma descoberta ou surpresa durante a realização da atividade?

Uma descoberta interessante foi o impacto significativo do preprocessamento das imagens na estabilidade e eficiência do aprendizado. A normalização dos valores RGB e o redimensionamento para tamanhos uniformes se mostraram etapas essenciais para o bom funcionamento do modelo.

Outro ponto surpreendente foi como o modelo se saiu bem ao lidar com padrões faciais claros, mas falhou em generalizar para variações aparentemente sutis, como diferenças em iluminação ou ângulos. Isso reforça a importância de considerar a diversidade do dataset desde o início do projeto.

Conclusão

A análise crítica dos resultados revelou que, embora o modelo tenha demonstrado bom desempenho em condições controladas, ele foi sensível às limitações do dataset e às variações nas imagens. Melhorias no preprocessamento, aumento do dataset e ajustes na arquitetura do modelo poderiam levar a resultados mais robustos e generalizáveis. Este projeto ofereceu valiosos insights sobre os desafios e as oportunidades ao trabalhar com CNNs em problemas de classificação de imagens.

Foi possível observar a importância do preprocessamento das imagens e do design cuidadoso da arquitetura do modelo para alcançar resultados consistentes. No entanto, desafios como a generalização para imagens com condições adversas (e.g., iluminação ou

ângulos desfavoráveis) destacaram a necessidade de melhorias no pipeline de aprendizado.

Além disso, o uso do Google Colab trouxe alguns obstáculos práticos, como:

- **Limitações de memória:** Durante o treinamento, especialmente ao usar datasets com preprocessamentos intensivos.
- **Gerenciamento de células:** A necessidade de executar células em ordem sequencial dificultou ajustes rápidos e flexíveis no código.
- **Dificuldades com modularização:** A implementação centralizada em células comprometeu a organização do código, limitando a reutilização de funções e a facilidade de manutenção.
- **Flexibilidade reduzida:** Alterações no modelo ou hiperparâmetros exigiram intervenções manuais significativas em várias células.

Referências

GÉRON, Aurélien. *Mãos à Obra: Aprendizado de Máquina com Scikit-learn, Keras e TensorFlow*. 2. ed. Rio de Janeiro: Alta Books, 2021.