Inteligência Computacional
Rede Neural Convolucional: Classificação de emoções

Gabriel Tambara Rabelo, Marcos Eduardo M. Junqueira
Departamento de engenharia elétrica
Faculdade de Tecnologia - UnB
Brasília, Brasil
180017021@aluno.unb.br, 180023691@aluno.unb.br

***Abstract*—In recent times, advances in neural networks have invaded the popular imagination, especially with the progress of language models and computer vision applications. This is due to the great potential that this technology presents, with the ability to tackle complex problems without the need for theoretical models and with a high degree of adaptability.**

The following work aims to design and analyze the performance of a convolutional neural network for an emotion recognition application in images. For this purpose, a set of techniques was employed to improve the network's performance in terms of accuracy. Finally, the obtained results were analyzed, and possible future improvements for the presented model were identified.

Keywords: Convolutional Neural Network; Emotion Recognition; Data Augmentation; Computer Vision; Image Classification.

***Resumo*—Em tempos recentes, avanços em redes neurais invadiram o imaginário popular, sobretudo com o avanço de modelos de linguagem e aplicações de visão computacional. Isso se dá ao grande potencial que esta tecnologia apresenta, com a capacidade de atuar em problemas complexos, sem a necessidade de modelos teóricos e com um alto grau de adaptabilidade.**

O trabalho a seguir tem como objetivo projetar e analisar a performance de uma rede neural convolucional para uma aplicação de reconhecimento de emoções em imagens. Para isso foi realizado um conjunto de técnicas para melhora da performance da rede em relação a sua acurácia. Por fim, os resultados obtidos foram analisados e levantou-se possíveis melhorias futuras para o modelo apresentado.

Palavras-chave: Rede Neural Convolucional; Reconhecimento de Emoções; Aumento de Dados; Visão Computacional; Classificação de Imagens.

I. Introdução

II. Fundamentação teórica

A. Convolutional Neural Network

1) Estrutura Geral: Uma *Convolutional Neural Network*, ou CNN, é uma arquitetura de rede neural artificial profunda, portanto, parte das arquiteturas de *Deep learning*, ou redes profundas. As redes profundas são redes que se diferenciam de outras topologias por possuírem um maior número de camadas em sua estrutura, permitindo e objetivando alcançar um maior número de detalhes em sua identificação. Nesse modelo, cada camada é responsável por identificar pequenos padrões, e conforme o número de camadas aumenta, a percepção da rede se torna cada vez mais generalista a ponto de conseguir observar padrões maiores e mais complexos. A maior parte de suas aplicações se encontra em identificação de padrões visuais, o que, aplicando diretamente seus conceitos, consegue identificar pequenos detalhes em uma imagem a ponto de eventualmente conseguir identificar características de interesse para um operador da rede. Dentre as redes profundas, a que se especializa em identificação espacial é a CNN, ou Convoluted Neural Network. Enquanto outras redes neurais totalmente conectadas exigem que os dados de entrada sejam representados como um vetor unidimensional, perdendo a informações posicionais, as CNNs preservam essa estrutura com o uso de camadas convolucionais.

O maior mecanismo de sua funcionalidade é a convolução, em que um filtro, também chamado de kernel, percorre a matriz de entrada e realiza operações multiplicativas para extrair características. Como detalhe, esses filtros são aprendidos durante o treinamento da rede e muitas vezes um padrão inicial não é necessário. Através deste mecanismo, há o compartilhamento de pesos, onde à medida que o kernel desliza pela entrada, atuando em várias localizações, permite-se que as mesmas características sejam detectadas em diferentes partes da imagem. Isso leva a um compartilhamento de informações e uma redução significativa no número de parâmetros a serem aprendidos, tornando as CNNs adequadas quanto ao consumo de memória e processamento. Através do uso de camadas convolucionais, camadas de pooling, e camadas de compartilhamento de pesos, além outros componentes específicos das CNNs, a arquitetura é capaz de capturar informações espaciais e hierárquicas nas imagens, resultando em um desempenho geralmente superior em problemas de classificação de imagens em comparação com outras arquiteturas de redes neurais.

2) Camada de Convolução: A camada de convolução receberá a matriz de entrada, geralmente um vetor 3D, com duas camadas para formar uma imagem plana e mais uma para dividir os três canais de cores em RGB (*red*, *green*, e *blue*). Com isso, ela gerará uma nova matriz tridimensional, compondo os filtros usados para identificar elementos, através dos pesos de cada pixel e seus adjacentes. Comumente são

implementadas as funções de ativação como a RELU já nesta camada.

3) Camada de Pooling: As camadas de pooling são utilizadas para reduzir o tamanho das camadas de imagem da matriz na rede, e portanto, dos filtros obtidos, o equivalente à reduzir a resolução das imagens, o que reduziria o espaço para armazenar informações e as resumiria, de forma a facilitar para as próximas camadas abstrairem suas informações.

4) Camada de Dropout: As camadas de Dropout funcionam desativando aleatoriamente neurônios da rede durante o treinamento, impedindo sua contribuição para a propagação do gradiente e pesos. Essa desativação aleatória força a rede a aprender recursos redundantes e a torna mais robusta, reduzindo efeitos como o de overfitting, onde classifica-se melhor os dados de treinamento do que dados genéricos e diferentes apresentados à rede.

5) Camada Densa: A última camada geralmente é uma camada densa (totalmente conectada) com um número de neurônios igual ao número de classes. Essa camada é seguida por uma função de ativação, como a função softmax ou sigmoid, para produzir as probabilidades de cada classe no processo de identificação.

B. Data Augmentation

Data augmentation, ou aumento de dados, é uma técnica utilizada no treinamento de redes neurais que consiste em criar novas amostras de dados artificiais a partir das amostras originais. O objetivo é aumentar a quantidade e a diversidade dos dados disponíveis para o treinamento, melhorando assim a capacidade do modelo de classificar informações diferentes das apresentadas no seu treinamento. Para que isto ocorra, são realizadas transformações controladas que incluem operações como rotação, espelhamento, zoom etc. Essas variações nos dados de treinamento ajudam a evitar o *overfitting* e a aumentar a robustez do modelo.

C. k-Fold Cross Validation

O processo de *k-Fold Cross Validation* consiste em um método pra avaliar a performance de um determinado modelo, com o objetivo de ter uma avaliação menos enviesada e mais próxima da realidade. Para isso, o conjunto de dados é em sua completude é unificado e randomizado e,em seguida, este é dividido em *k* parcelas, chamadas de *folds*. Após isso, para cada *fold*, um *fold* é selecionado como conjunto de teste e o modelo é treinado em cima dos *folds* restantes como o conjunto de treinamento. Por fim, a performance do modelo pode ser inferida baseada na amostra de métricas obtidas para cada *fold*.

D. Model Tuning

O *tuning* do modelo se refere ao processo de escolha dos hiper-parâmetros do modelo, de forma a se obter o desempenhado desejado. Geralmente, são treinadas e testadas múltiplas configurações de hiper-parâmetros, de uma lista definida de possíveis topologias e valores, e são observados os resultados das métricas a fim de decidir qual será utilizada. Para isso existem diversas abordagens, como busca exaustiva utilizando todas as permutações possíveis de hiper-parâmetros (*Grid Search*); utilizando configurações aleatórias de hiper-parâmetros para um número finito de tentativas (*Random Search*) entre outros algoritmos de otimização.

III. Situação Problema

A. Identificação de Emoções

Diante dos mais diversos cenários na atualidade, principalmente na internet, percebe-se um grande contato com informações dos mais diversos tipos, incluindo imagens. Essas informações por vezes são apresentadas em tão grande número que a sua validação, para quaisquer meios que sejam visados, acaba sendo muito demorada ou até inviabilizada. Com o uso de ferramentas como a CNN é possível extrair informações iniciais, ou ao menos boas estimativas, sobre algumas das propriedades desses dados na rede, podendo auxiliar os mais diversos profissionais em pesquisas acadêmicas ou mercadológicas. Uma das aplicações diretas desse cenário é o uso da identificação de sentimentos em imagens, podendo ser utilizada para realizar uma compreensão emocional inicial em humanos, facilitando sua compreensão ou até auxiliando na identificação de intenções; também pode-se a utilizar para pesquisas acadêmicas na área de psicologia, visando analisar influências das mais diversas no humor de um indivíduo; além de também ser possível seu uso para a facilitação de interação entre o indivíduo e o computador, auxiliando, com respostas expressivas, pessoas com dificuldades motoras ou até usuários mais convencionais que se interessam por explorar novas formas de navegar pela rede.

Portanto, diante deste cenário, foi proposto o uso de redes convolucionais para classificação de emoções como forma de ilustrar a capacidade das redes, bem como demonstrar o conhecimento adquirido pelos escritores deste artigo no tópico, aplicando as mais diversas tecnologias envolvidas em um problema real de importância para a atualidade.

IV. Laboratório

A. Dados

Para a classificação, foram obtidas imagens do banco de dados FER-2013, composto por imagens de 48×48 pixeis que se encaixavam em sete categorias de emoções humanas, sendo elas: raiva, desgosto, medo, felicidade, neutro, tristeza e surpresa. Alguns exemplos do conjunto de dados pode ser observado pela Fig.[1]. É necessário observar que os conjuntos de dados utilizados foram separados em treinamento, validação e testes, visando validar a rede enquanto a mesma é treinada e também a realização do seu teste ao fim do treinamento.

Figura 1: Exemplos aleatórios de imagens obtidas do conjunto de dados de treinamento com suas respectivas classificações

O Conjunto de dados obtido para a realização deste projeto possui certas características que precisaram ser tratadas antes de seu uso imediato, visando o melhor aproveitamento do mesmo para a identificação e classificação. Para ilustrar o problema de desbalanceamento do conjunto de dados, a figura representada por Fig.[2] demonstra as diferenças de forma visual.

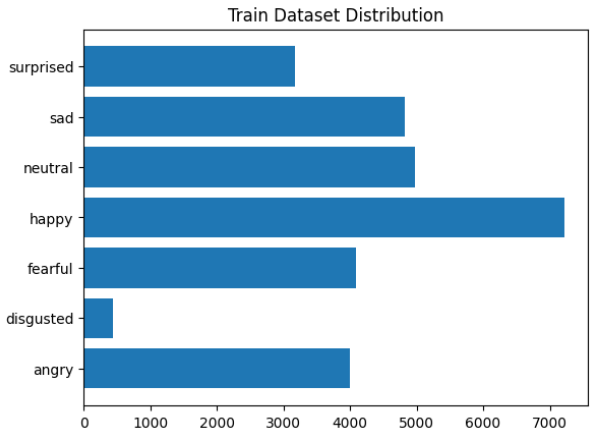


Figura 2: Distribuição de dados obtido pelo conjunto inicial

Percebe-se uma grande quantidade de imagens dentro do conjunto *happy*, ou 'feliz', e uma baixíssima quantidade de imagens do conjunto *disgusted*, ou com 'desgosto'. O resto das imagens possui uma boa variância entre suas amostras bem como uma quantidade interessante para se realizar uma classificação. A disparidade de classificação da classe 'feliz' não precisa necessariamente ser resolvida, contudo, a de 'desgosto' precisa ser tratada, caso contrário, haverão problemas de classificação como *overfitting* ou simplesmente má classificação dos detalhes, já que há 436 amostras, o que não chega à casa dos milhares, que seria de interesse para o trabalho a ser realizado.

Para resolver esta problemática de desbalanceamento podem ser realizadas algumas tarefas, dentre elas a re-amostragem, que envolve a redução ou aumento dos conjuntos de dados desproporcionais. Nesse cenário, a redução da classe 'feliz' envolveria uma perda na sua capacidade de classificação, bem como de generalização da rede neste aspecto, o que unicamente pioraria a rede; Já o aumento de dados, poderia ser realizado em todas as classes para que todas obtivessem o maior tamanho de classe disponível, neste caso de 1775. Esta solução certamente seria ideal caso fosse possível a obtenção de novas imagens orgânicas. Contudo, para o presente projeto, foi necessário o uso de aumento de dados, e diante disso, um aumento via *data augmentation* levaria a rede à obtenção de características indesejadas como uma grande dificuldade de generalização, por possuir um conjunto de dados, principalmente nas classes com menores tamanhos, pouco variantes, mesmo com uso de melhores técnicas de aumento de dados. Portanto, a solução escolhida foi a da realização do aumento de dados apenas na classe de 'desgosto', sendo a solução que gerou os melhores resultados e que serão apresentados.

Para realizar o aumento de dados, foram utilizadas algumas propriedades, sendo elas descritas a seguir. O tamanho final escolhido foi de seis vezes o tamanho inicial da classe, totalizando 2616 imagens. A nova distribuição de dados pode ser observada pela Fig.[3].

- Rotação aleatória de -10 a 10 graus.
- Inversão horizontal aleatória de 10% da largura
- Inversão vertical aleatória de 10% da altura
- Torção na imagem de intensidade de 20%
- Zoom aleatório entre 80% e 120% da imagem
- Inversão horizontal aleatória
- Preenchimento de pixeis novos com base nos pixeis das proximidades

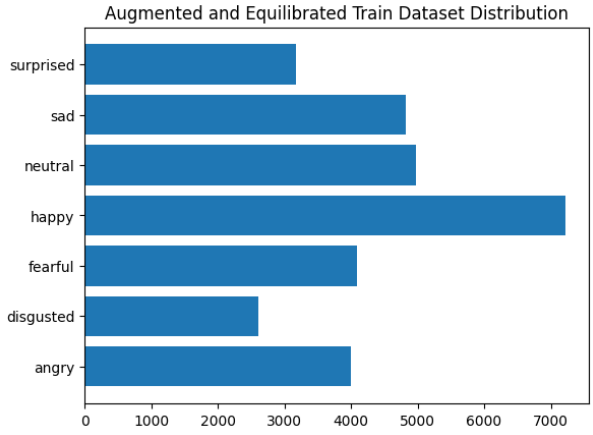


Figura 3: Distribuição de dados obtido pelo conjunto aumentado

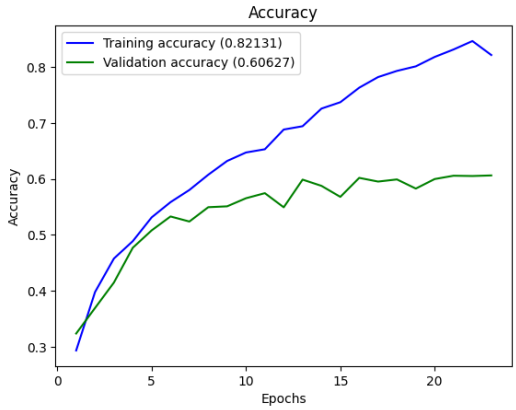
B. Modelo

Utilizando o *tuning* eventualmente chegou-se a uma topologia de rede cujos resultados fossem interessantes. Nesta rede, foi realizado um processo iterativo de correções de pequenos detalhes como inserções de novas camadas de *pooling* e *dropout*, visando um aumento da acurácia do conjunto de validação, a principal métrica utilizada para a análise da rede.

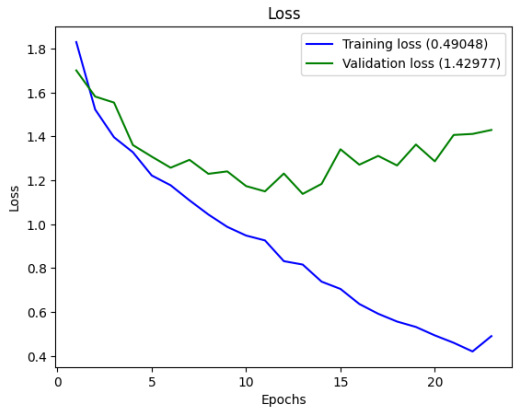
Quanto alguns dos hiperparâmetros tratados, foi definido, como resultado mais performático pelo *tuning*, valores de tamanho 6 para o kernel, 40 filtros, *pool size* de 3, e 500 neurônios. O otimizador utilizado foi o Adam, e duas camadas densas foram utilizadas, uma relu e outra softmax, sepadaradas por uma camada de *dropout*.

C. Análises finais e melhorias

Para o treinamento, foi selecionado um *batch size* de 64 e um máximo de 200 épocas, com paciência de 10 épocas para o parâmetro de *loss* do conjunto de validação, conjunto este cujo tamanho se dá como uma porção de 20% do conjunto de treinamento. Após 23 épocas, obtêve-se uma acurácia de 0.8213 para o conjunto de treinamento e uma acurácia de 0.6063 para o conjunto de validação. Os resultados ao longo do treinamento podem ser observados pelas imagens na Fig.[4].



(a) Acurácia por épocas de treinamento



(b) Perda por épocas de treinamento

Figura 4: Performance da terceira rede

Para o conjunto de teste, visando realizar uma validação final da topologia, obteve-se um *loss* de 1.3846, e uma acurácia de 0.6100.

Pode-se visualizar a capacidade de classificação da rede através da matriz de confusão apresentada pela Fig.[5]. Pode-se perceber o evidente destaque da diagonal principal, denotando uma boa capacidade de classificação correta das redes em comparação com uma classificação de falsos positivos. Percebe-se também um evidente melhor resultado para a categoria de feliz, que pode se dar pelo maior conjunto de dados orgânicos apresentados, além de evidenciar um bom resultado para a categoria de desgosto.

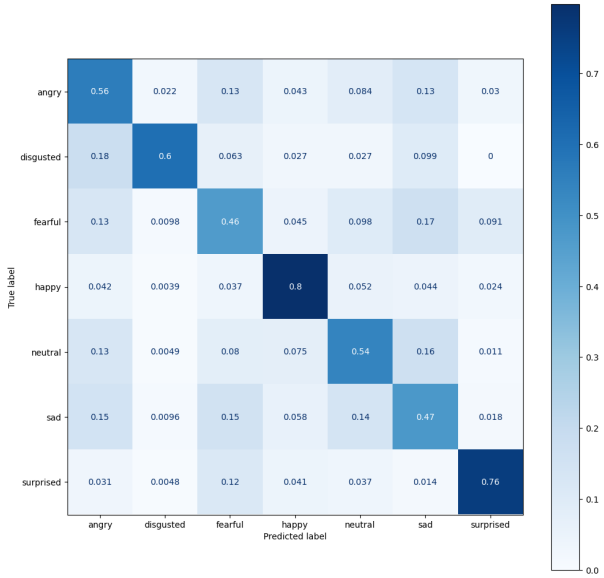


Figura 5: Matriz de confusão da rede

Fold	Acurácia	*Loss*
1	0.7723	0.9058
2	0.8348	0.6468
3	0.8692	0.5453
4	0.8917	0.4951

Tabela I: Resultados da Cross Validation da Rede

Por fim, para fins de verificação, foi realizada a *Cross Validation* da rede, com os resulto sendo observados na Tabela [I]. Dela, pode-se notar que as acurácias obtidas pela rede foram superiores à acurácia obtidas anteriormente sobre o conjunto de teste, além de ter um discrepância considerável entre o primeiro fold e os demais. Isso provavelmente deve-se ao fato de que o conjunto de treino para o qual a rede foi treinada anteriormente, no qual a rede apresentou níves de acurácias elevados, compõe grande parte dos *folds* criados no processo de *Cross Validation*. Dessa forma, pode-se observar que apesar da medidas utilizadas, a rede projetada ainda tem sinais de *overfitting*, possivelmente derivados da distribuição vastamente irregular de fotos entre as classes apresentadas.

V. Conclusão

Neste trabalho, foi possível projetar e avaliar o desempenho de uma CNN para classificação de emoções por meio de imagens. Ao longo do processo de construção da topologia da rede foram empregadas técnicas como *data augmentation* e tuning de hiper-parâmetros de forma a tentar obter um melhor desempenho na acurácia da classificação realizada.

Além disso, utilizando ferramentas como histórico de acurácia e perda do treinamento, em adição a matriz de confusão, foi possível obter o desempenho da rede para o conjunto de teste, que foi relativamente satisfatório.

Por fim, a técnica de *cross validation* teve o papel de explicitar o problema de *overfitting* que a rede apresentou, possivelmente derivado da distribuição desigual das classes presentes no banco de dados do FER-2013, de forma que mesmo as técnicas utilizadas terem resultado em um melhor desempenho da rede, ainda não forma suficientes para remediar completamente a situação.

Referências

[1] Keras official documentation, https://keras.io/

[2] Facial Emotion Recognition Using Deep Convolutional Neural Network, https://ieeexplore.ieee.org/document/9074302

[3] Facial Emotion Recognition of Students using Convolutional Neural Network, https://ieeexplore.ieee.org/document/8942386