

Evento: XXI Jornada de Extensão  
ODS: 9 - Indústria, Inovação e Infra-estrutura

# APLICAÇÃO DE INTELIGÊNCIA ARTIFICIAL NO RECONHECIMENTO DE EXPRESSÕES FACIAIS<sup>1</sup>

## APPLICATION OF ARTIFICIAL INTELLIGENCE IN RECOGNITION OF FACIAL EXPRESSIONS

Leandro Perius Heck<sup>2</sup>, Cristiano Alex Künas<sup>3</sup>, Edson Luiz Padoin<sup>4</sup>

<sup>1</sup> Projeto desenvolvido na Disciplina de Projeto de TCC do Curso de Ciência da Computação da UNIJUÍ

<sup>2</sup> Aluno do Curso de Graduação em Ciência da Computação da UNIJUÍ, leandro.h@sou.unijui.edu.br

<sup>3</sup> Aluno do Curso de Graduação em Ciência da Computação da UNIJUÍ, cristiano.kunas@sou.unijui.edu.br

<sup>4</sup> Professor Doutor do Departamento de Ciências Exatas e Engenharias da UNIJUÍ, Orientador, padoin@unijui.edu.br

### 1. Introdução

Atualmente existe um crescente interesse na melhoria da interação entre humanos e computadores. Para que uma efetiva interface humano-computador inteligente seja alcançada faz-se necessário que o computador relaciona-se naturalmente com os usuários, semelhante à maneira que os humanos interagem [Leao et al. 2012]. Um das maneiras de encontrar e promover esse tipo de integração é com auxílio da Computação Afetiva, Inteligencia Artificial, com *Machine Learning* e *Deep Learning*.

Fazer com que uma máquina reconheça, modele e expresse emoções não é uma tarefa simples. Quando seres humanos interagem entre si, boa parte dessa interação é baseada na linguagem verbal e na utilização da linguagem corporal por meio de gestos e expressões faciais que carregam e transmitem as emoções dos interlocutores [Leao et al. 2012].

Nas últimas décadas a comunidade científica vem tendo um crescente interesse no reconhecimento de emoções. Existem atualmente diversas maneiras de expressar as emoções humanas, estas vêm sendo estudadas e melhoradas ao longo dos anos. Muitas fontes de dados têm sido exploradas, dentre elas destacam-se textos, envio de *emoticons*, voz e as expressões faciais. Atualmente, trabalhos sobre esse tema investigam inúmeras técnicas e maneiras de extração de características e classificação de padrões em expressões faciais [do Amaral et al. 2013], como por exemplo, o uso das Redes Neurais.

Nesse sentido, o presente artigo apresenta uma proposta de utilização de uma Rede Neural Artificial (RNA) aplicada no reconhecimento de expressões faciais. Seu objetivo é diferenciar os diferentes sentimentos que os indivíduos expressam a partir dos padrões. Sendo uma área de grande relevância para consumidores e organizações em áreas de saúde, marketing, cinema dentre outras. No marketing, sendo possível entender o que os clientes sentem ao consumir um produto, no cinema analisar as reações das pessoas diante as cenas de um filme.

O restante do trabalho esta organizado da seguinte forma. Na Seção 2 é apresenta a metodologia que será utilizada na implementação e o ambiente de execução para realização dos testes. Na Seção 3 é apresentados os resultados obtidos na implementação da RNA, seguido das Conclusões e Trabalhos Futuros.

Evento: XXI Jornada de Extensão  
ODS: 9 - Indústria, Inovação e Infra-estrutura

## 2. Metodologia

A Rede Neural desenvolvida trata-se de uma rede neural convolucional, que vem sendo aplicada com sucesso no processamento e análise de imagens digitais. Por esta razão tem-se basicamente quatro etapas a serem seguidas para seu desenvolvimento, são elas:

**i) Etapa 1 Operador de Convolução:** Esta etapa funciona como filtros que enxergam pequenos quadros e vai percorrendo toda a imagem captando os seus traços mais relevantes. Por exemplo, em uma imagem 48x48 e um filtro que cobre uma área 3x3 da imagem com movimento de 1 salto, o filtro irá percorrer a imagem inteira. No filtro 3x3 é pegos ponto a ponto e realizada a multiplicação por um detector de características, este definido pela biblioteca que é utilizada, formando no final um mapa de características ou *feature map*.

**ii) Etapa 2 Pooling:** Nesta etapa tem a função de simplificar a informação da camada anterior, no caso, o mapa de características. Assim como na convolução, é definida uma unidade de área, geralmente se utiliza uma matriz 2x2, para transitar por toda a saída da camada anterior. A unidade é responsável por resumir a informação daquela área em um único valor. Para selecionar este elemento é preciso escolher a maneira que será feita a sumarização.

O método mais utilizado é o *MaxPooling*, que retorna o maior número da unidade, e passa este valor para a saída. Essa sumarização de dados serve para diminuir a quantidade de pesos a serem aprendidos pela Rede Neural, além de também evitar *overfitting*.

**iii) Etapa 3 Flattening:** Esta etapa recebe a matriz criada na etapa de *Pooling* como entrada, e basicamente opera uma transformação na matriz da imagem, alterando seu formato para um array, ou seja, converte a matriz para um vetor de características

**iv) Etapa 4 Rede Neural:** Esta etapa recebe a camada de *Flattening* como entrada, e submete o vetor de características como entrada para treinamento da Rede Neural, que é um modelo computacional baseado no sistema nervoso central humano. Dessa maneira, será capaz de reconhecer padrões em uma grande massa de dados de forma a classificá-los em alguma categoria.

Para a implementação da RNA utilizou-se o ambiente de desenvolvimento *Google Colab* e a linguagem de programação *Python*, versão 3.6.9. O *Google Colab* é um serviço de nuvem gratuito hospedado pelo *Google* para Aprendizado de Máquina e IA [COLABORATORY 2020], possuindo aceleradores de GPU grátis, bibliotecas já pré-instaladas, construído com base no *Jupyter Notebook*, suporta comandos *bash* além de armazenar os *notebooks* no próprio *Drive*.

As principais bibliotecas utilizadas na RNA, são, *TensorFlow 2.0* e o *Keras*, que são voltadas ao aprendizado de máquina profundo. O *Keras* é o *framework* mais utilizado na área pela sua facilidade de uso e rápida prototipagem, além de ser uma camada de alto-nível para outros *frameworks* de *Deep Learning*, como *TensorFlow*, *Theano* e *CNTK* [Keras 2020]. No *Tensorflow 2.0*, o *Keras* foi "incorporado" ao *TensorFlow* através do módulo *tf.keras*, ou seja, toda a API do *Keras* está presente na nova versão do *TensorFlow* [TENSORFLOW 2020].

A base de dados utilizada para este trabalho foi FER-2013 (*Facial Expression Recognition 2013*). Esta base de dados reúne um conjunto de dados *open-source* criado por Pierre- Luc Carrier e Aaron

Evento: XXI Jornada de Extensão

ODS: 9 - Indústria, Inovação e Infra-estrutura

Couville. Este conjunto de dados consiste em 35.887 imagens de faces em tom de cinza e 48 x 48 *pixels*, com um total de 7 emoções.

### 3. Resultados

Os resultados foram satisfatórios, a rede apresentou uma acurácia 65,67%. Na Figura 1 é apresentado o gráfico de treinamento da rede, realizado na plataforma do *Google Colab*, ilustrando a evolução da acurácia e a taxa de *loss*. Examinando o gráfico, nota-se que o aprendizado da rede se mantém constante, desde seu início, com poucas variações. Isto ocorre, pela utilização de um percentual menor na técnica de *Dropout*, sendo zerado apenas 20% da rede, enquanto ela está em treinamento. Nota-se que a partir da época 9, não se tem mais grandes melhoras no treinamento da rede, ela possui uma pequena melhora, mas não significativa, não sendo necessário que a rede treine todas as interações definida no início da compilação.

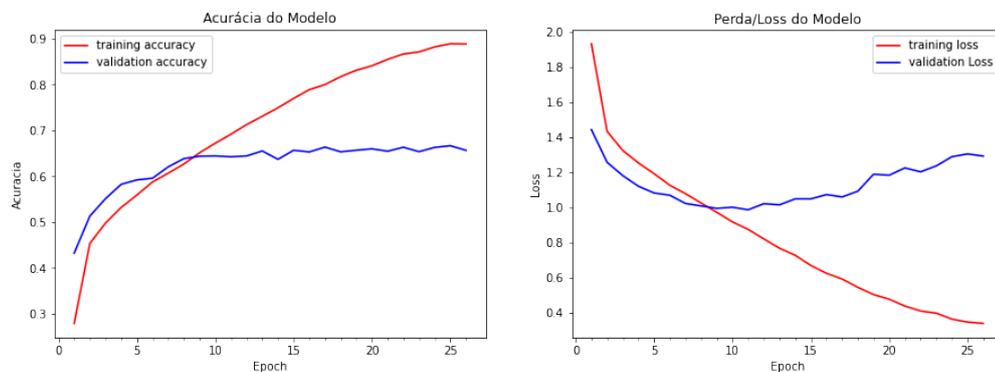


Figura 1 - Gráfico da taxa de acurácia e loss do treinamento.

O fato da RNA ter apresentado pouca variação no momento do aprendizado, é a baixa taxa de *Droupout*, de 20%. Pois durante o treinamento a rede neural já pode conter informações corretas sobre os dados de saída, e apenas uma pequena taxa da Rede Neural é zerada, ou seja, ela começa a se especializar mais rapidamente com os dados de saída, assim não apresentando muitas oscilações em seu treinamento. Mas quando a taxa de *Dropout* for maior as chances de eliminar informações que a rede neural já considera como corretas, são maiores. Dessa forma, a rede neural acaba precisando treinar mais, apresentando mais oscilações em seu treinamento.

Em relação a taxa de loss, ela começa alta e vai reduzindo, logo após a época 9, o percentual de perda começa a aumentar a cada nova época, se a rede fosse treinada até o final, no caso definiu-se 100 interações para a RNA, o valor da perda poderia ser muito maior, assim obteve-se uma perda de apenas 1,40.

### 4. Conclusões e Trabalhos Futuros

Este trabalho apresentou o desenvolvimento e a utilização de uma Rede Neural Artificial aplicada no reconhecimento de expressões faciais e extração dos padrões presentes em imagens.

A proposta apresentou desempenho razoável em sua precisão da acurácia, que com maiores testes e modificações em seus parâmetros pode chegar a uma taxa de acuracidade maior.

**Evento:** XXI Jornada de Extensão

**ODS:** 9 - Indústria, Inovação e Infra-estrutura

Como trabalhos futuros, pretende-se realizar a implementação do algoritmo sequencial e paralelo e iniciar o treinamento, teste, validação e análise de desempenho da aplicação da Rede Neural, pois como o *Google Colab* disponibiliza GPUs gratuitas, a intenção é reduzir o tempo de treinamento da rede Neural, pois atualmente a etapa de treinamento é a etapa que mais demanda tempo na elaboração de uma Rede Neural.

## Referências

COLABORATORY, G. (2020). Get started with google colab. Disponível em: <https://colab.research.google.com/notebooks>. Acesso em: 17 jul 2020

do Amaral, V., Giraldo, G. A., and Thomaz, C. E. (2013). Lbp estatístico aplicado ao reconhecimento de expressões faciais.

Keras (2020). Keras - simple. flexible. powerful. Disponível em: <https://keras.io/>. Acesso em: 17 jul 2020.

Leão, L. P., Bezerra, J. S., Matos, L. N., and Nunes, M. A. S. N. (2012). Detecção de expressões faciais: uma abordagem baseada em análise do fluxo óptico. *Revista GEINTEC-Gestão, Inovação e Tecnologias*, 2(5):472-489.

TENSORFLOW(2020). Tensorflowcore. Disponível em: <https://www.tensorflow.org/tutorials>. Acesso em: 11 jul 2020.

**Parecer CEUA:** 4338191018

**Parecer CEUA:** 3.104.922/2019