## **Resume Classification – CNN**

* Barath Vignesh. S

## **Introduction**

I have developed a Convolutional Neural Network (CNN) for binary image classification. The objective is to distinguish between two classes using a dataset comprising PNG images extracted from PDF resumes obtained from Kaggle. This one-page documentation summarizes the key aspects of the approach, including dataset details, model architecture, training strategy, and evaluation metrics.

## **Dataset Details**

The dataset consists of PNG images extracted from PDF resumes sourced from Kaggle. The images were preprocessed and used for binary image classification.

## **Model Architecture**

The CNN model follows a sequential architecture with the following key components:

* **Convolutional Layers (Conv2D)**: Three convolutional layers with the 'valid' padding scheme, utilizing L2 regularization to prevent overfitting.
* **Max Pooling Layers (MaxPooling2D):** Max pooling layers applied after each convolutional layer to reduce spatial dimensions.
* **Dropout Layers:** Employed for regularization to enhance model generalization.
* **Flatten Layer:** Used to flatten the output for input into the dense layers.
* **Dense Layers**: Two dense layers follow the flattened output, with the final layer utilizing the sigmoid activation function for binary classification.

## **Training Strategy**

### **Data Augmentation**

Data augmentation techniques, including rescaling, zooming (zoom\_range=0.2), and horizontal and vertical shifting (width\_shift\_range=0.2, height\_shift\_range=0.2), were applied to enhance model robustness.

### **Loss Function and Optimization**

Binary cross-entropy loss function was employed, suitable for binary classification tasks. The Adam optimizer was chosen for efficient weight updates during training.

### **Regularization**

L2 regularization was incorporated into the convolutional layers to mitigate overfitting.

1. **Early Stopping**

Callbacks for early stopping were implemented to halt training when the model's performance on the validation set ceased to improve, preventing overfitting and reducing training time.

## **Evaluation Metrics**

The model achieved a test accuracy of **73%** and a train accuracy of **89%** after approximately **4 hours of training**. The confusion matrix, classification report were employed for detailed performance analysis.

## **Conclusion**

The documented approach reflects a thoughtful consideration of dataset properties, model design, and training strategy, leading to a model capable of effectively classifying binary images. The achieved accuracy metrics indicate a balance between model performance and generalization.