**Implementation and Reporting Tasks:**

**Simple CNN-Model.**

* **A simple CNN typically consists of a series of convolutional layers, followed by pooling layers and fully connected layers, with the aim of gradually reducing the spatial dimensions of the input image and learning higher-level features. The output of the final fully connected layer is then fed to a softmax function for classification.**
* **On the other hand, ResNets are a type of CNN that were introduced to address the problem of vanishing gradients. They use a special skip-connection or residual connection that allows the network to learn residual functions instead of directly trying to learn the mapping between the input and output. This approach enables the network to effectively learn from deeper layers without encountering the problem of vanishing gradients.**
* **In essence, ResNets are deeper than simple CNNs and they utilize a skip-connection to enable the network to learn more effectively from the deeper layers. This allows ResNets to achieve better performance on very deep networks, as compared to simple CNNs.**

**# Creating a Simple CNN Model.**

**Graphical user interface

Description automatically generated with medium confidence**

**Graphical user interface, text, application, website

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**# Building a Model:**

**Text

Description automatically generated with medium confidence**

**Table

Description automatically generated**

**Adding Dense Layers:**

**Text

Description automatically generated**

**# ADAM Optimizer with epochs=50 and batch\_size=32(Default), steps\_per\_epoch default one.**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**ADAM Optimizer with epochs=10 and batch\_size=64, steps\_per\_epoch=719.**

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**ADAM Optimizer with epochs=20 and batch\_size=32(Default), steps\_per\_epoch default one.**

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, application, Word

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, application, Word

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, application, Word

Description automatically generated**

**In a Nutshell when I compare the ADAM and SGD, ADAM optimizer works pretty well with training and validation data.**

**# Resnet Implementation.**

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface, table

Description automatically generated with medium confidence**

**# Model Buiding with 5 epochs, batch\_size=64 and steps\_per\_epoch=782.**

**Graphical user interface, text, application, email

Description automatically generated**

**# From the above, After epoch 2 I see val\_accuracy is decreasing, so we will stop training the model post that where val\_accuracy is increasing. The val\_accuracy is 0.95 and training accuracy is 0.99. It Clearly indicates that Resnet performs better than Simple CNN model, also overcomes the problem of Vanishing Gradient Descent.**

**Graphical user interface, text, application, email

Description automatically generated**

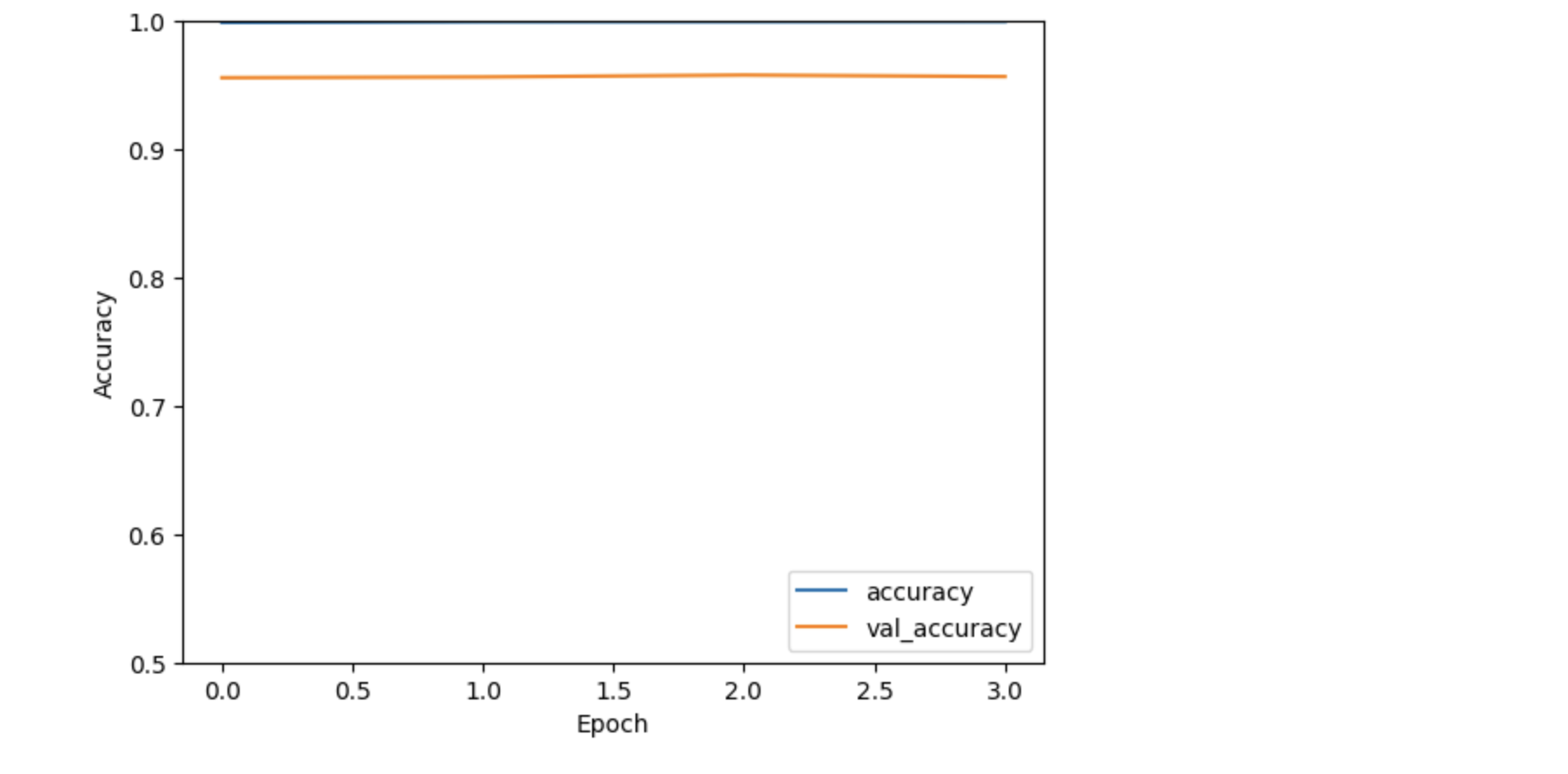
**Graphical user interface, application

Description automatically generated**

**# Model Buiding with 10 epochs, batch\_size=64 and steps\_per\_epoch=782.**

**Text

Description automatically generated**

****

**Here, In this case after 19 epochs we see the val\_accuracy remains same, so its better to stop the training as it may leads to overfitting of model.**

**# Implement Dropout technique to avoid overfitting of Model.**

**Graphical user interface, text, application

Description automatically generated**

**# In a Nutshell , Resnet works pretty well compare to Simple CNN model.**