CAPSTONE PROJECT

PROJECT TITLE

PRESENTED BY

STUDENT NAME : HEMASHREE R

COLLEGE NAME : SESHADRIPURAM DEGREE COLLEGE

DEPARTMENT : BCA

EMAIL ID:SARASWATHIJB077@GMAIL.COM

AICTE STUDENT ID : STU67CECC3194952174160593 7

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

In this project, we aim to build a machine learning model that can accurately classify images of clothing items from the Fashion MNIST dataset. The dataset consists of 70,000 grayscale images of 10 different clothing categories, including T-shirts, trousers, dresses, and shoes. Each image is 28x28 pixels in size.

The main objectives are:

1. **Data Preprocessing**: We will load the dataset, normalize the pixel values to a range of 0 to 1, and visualize some sample images to understand the data better.
2. **Model Development**: We will create a neural network using TensorFlow and Keras. The model will consist of a flattening layer to convert the 2D images into 1D arrays, followed by a dense layer with 128 neurons and a softmax output layer for classification.
3. **Training and Evaluation**: The model will be trained on the training dataset for a specified number of epochs. After training, we will evaluate its performance on the test dataset to determine its accuracy.
4. **Prediction Visualization**: Finally, we will visualize the model's predictions on test images, highlighting correct predictions in blue and incorrect ones in red, to assess the model's performance visually.

PROPOSED SOLUTION

1. Data Collection
   We use the Fashion MNIST dataset, which includes 70,000 labeled images of different clothing items divided into training and testing sets.
2. Data Preprocessing
   Images are scaled to have pixel values between 0 and 1 by dividing by 255. This normalization helps the model learn better and faster.
3. Machine Learning Algorithm
   We build a neural network with three layers:

- Flatten layer to convert 2D images into 1D vectors.
- Dense layer with 128 neurons using ReLU activation.
- Output layer with 10 neurons using softmax activation to classify into 10 clothing categories.

4. Training and Evaluation
   The model is trained on the training data for 5 epochs and then tested on unseen test images to check its accuracy.
5. Deployment and Visualization
   We use the trained model to predict clothing categories on test images and visualize the results, showing correct predictions in blue and incorrect in red for better understanding.

SYSTEM APPROACH

1.System Requirements

- A computer with Python installed.
- TensorFlow and Keras libraries for building and training the neural network.
- NumPy for numerical operations and Matplotlib for plotting results.

2.Libraries Required

- **TensorFlow & Keras:** Used to create, train, and evaluate the deep learning model.
- **NumPy:** Helps handle and preprocess image data efficiently.
- **Matplotlib:** Used for visualizing images and prediction results to analyze model performance.

ALGORITHM & DEPLOYMENT

1. **Algorithm Selection**
   We use a simple neural network with layers that flatten the images, learn features using a dense ReLU layer, and classify with a softmax output layer into 10 clothing categories.

2. **Data Input**
   The input images are 28x28 pixel grayscale images from the Fashion MNIST dataset. Images are normalized to values between 0 and 1 for better learning.

3. **Training Process**
   The model learns patterns from the training images by adjusting weights over 5 epochs using the Adam optimizer and sparse categorical cross-entropy loss for multi-class classification.

4. **Prediction Process**
   After training, the model predicts the clothing category for new images by outputting probabilities. The highest probability determines the predicted class.

RESULT

```
from __future__ import absolute_import, division, print_function

# TensorFlow and tf.keras
import tensorflow as tf
from tensorflow import keras

# Helper libraries
import numpy as np
import matplotlib.pyplot as plt

print(tf.__version__)
```

```
2.18.0
```

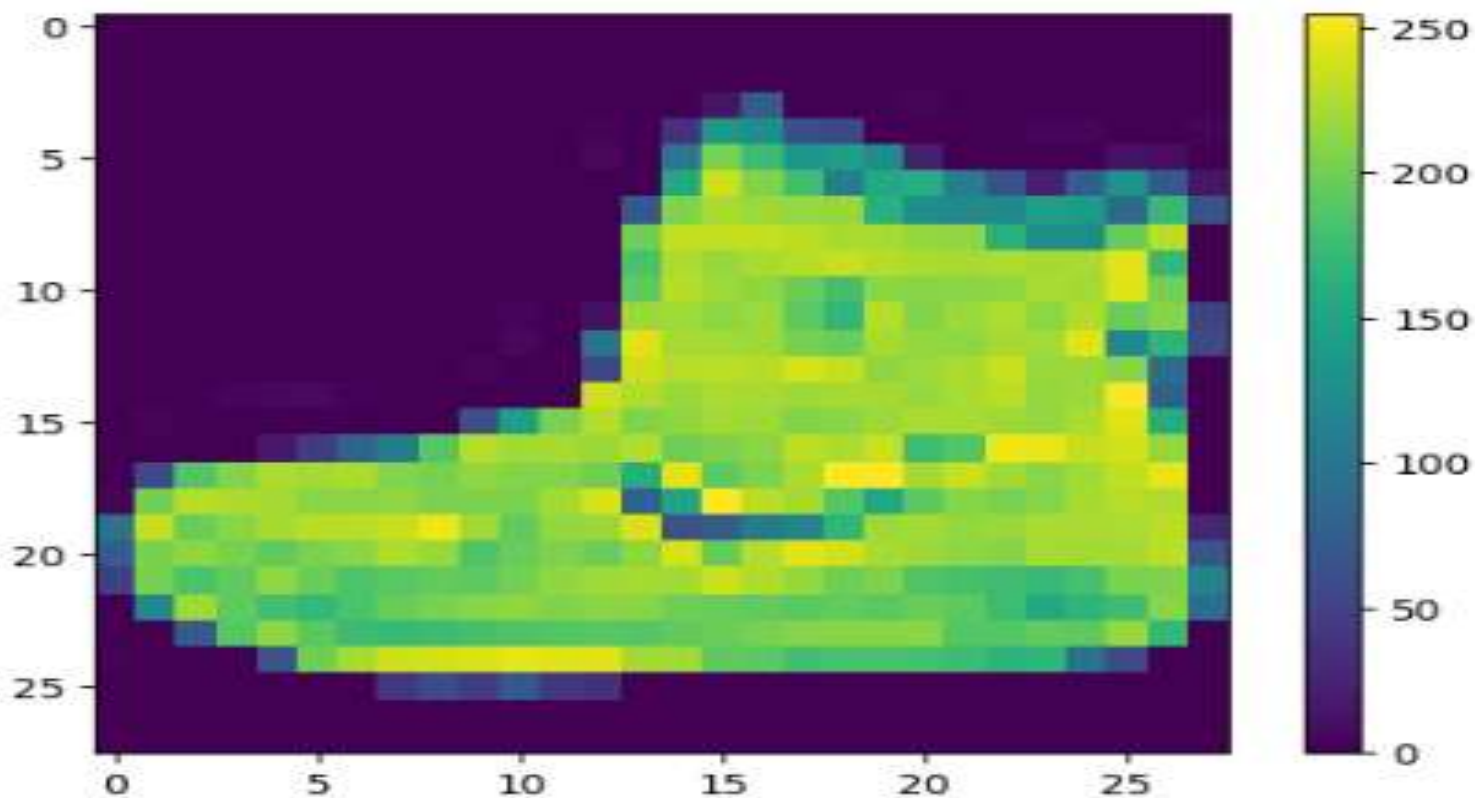
```
[ ] fashion_mnist = keras.datasets.fashion_mnist

    (train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz
29515/29515 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz
26421880/26421880 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz
5148/5148 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz
4422102/4422102 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
```

```
[ ] class_names = ['T-shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat',
                   'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']
```

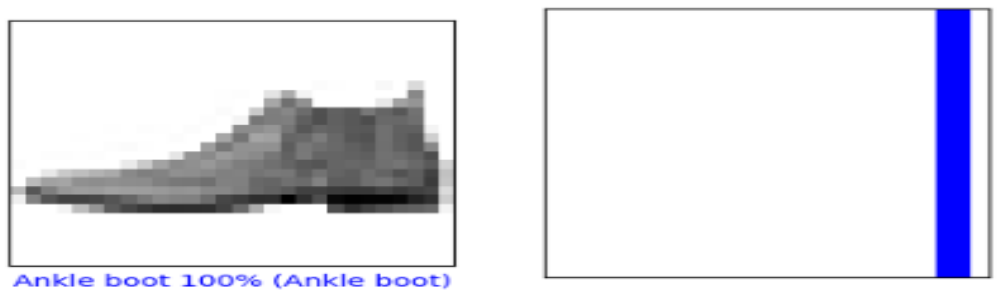
```
10000
```

```
plt.figure()
plt.imshow(train_images[0])
plt.colorbar()
plt.grid(False)
plt.show()
```

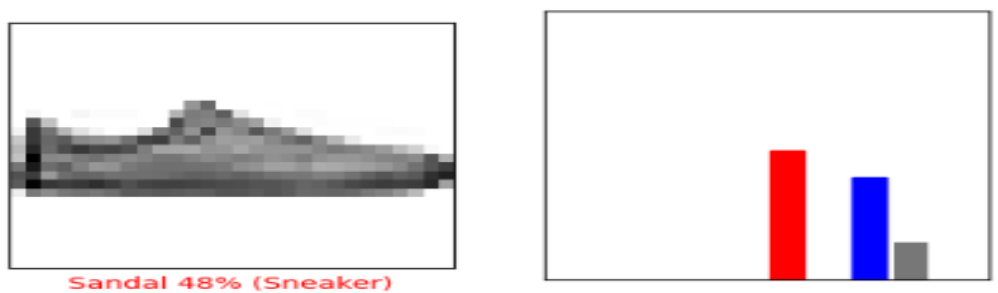


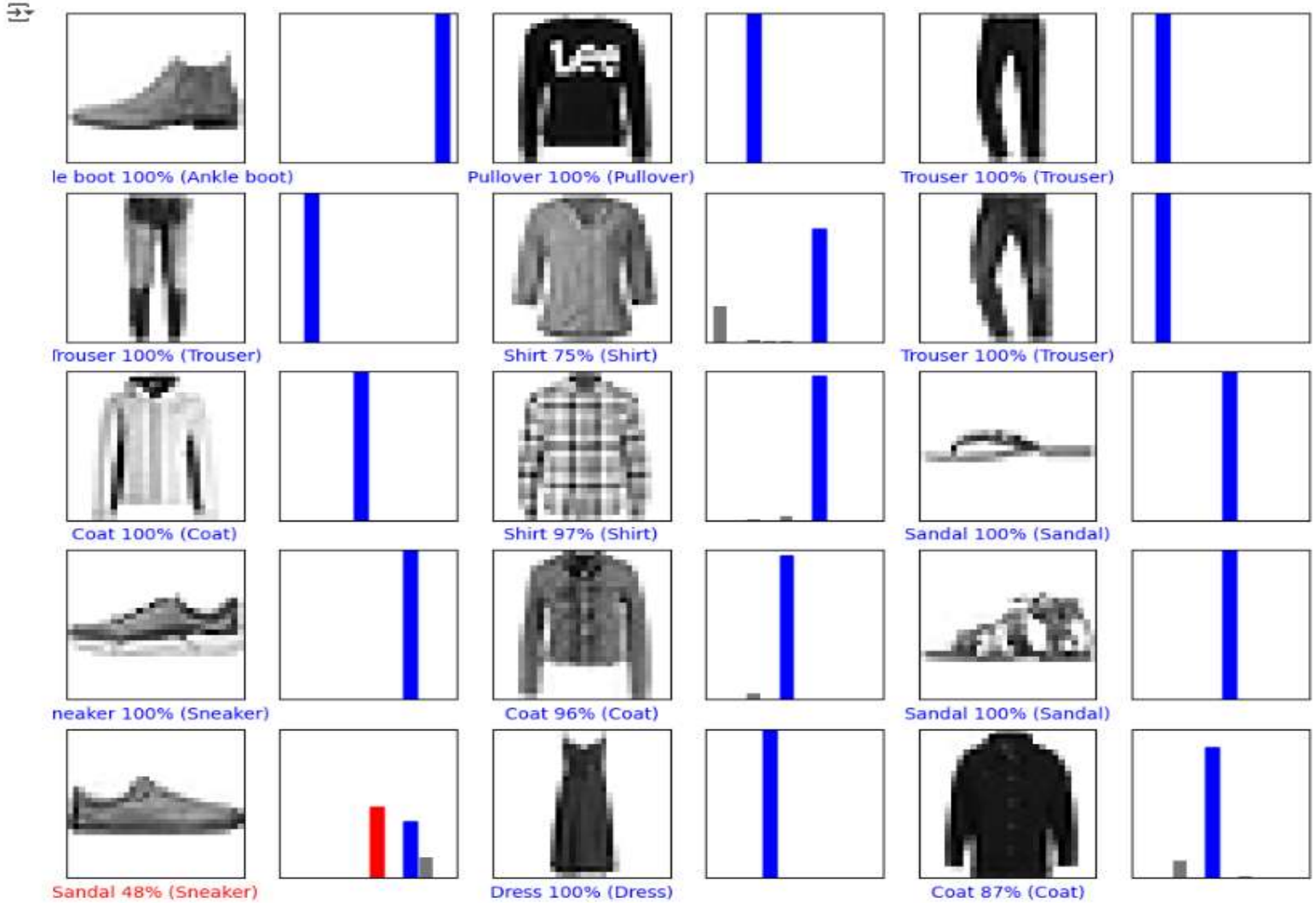

Ankle boot
T-shirt/top
T-shirt/top
Dress
T-shirt/top
Pullover
Sneaker
Pullover
Sandal
Sandal
T-shirt/top
Ankle boot
Sandal
Sandal
Sneaker
Ankle boot
Trouser
T-shirt/top
Shirt
Coat
Dress
Trouser
Coat
Bag
Coat

```
[ ]  i = 0
     plt.figure(figsize=(6,3))
     plt.subplot(1,2,1)
     plot_image(i, predictions, test_labels, test_images)
     plt.subplot(1,2,2)
     plot_value_array(i, predictions,  test_labels)
     plt.show()
```

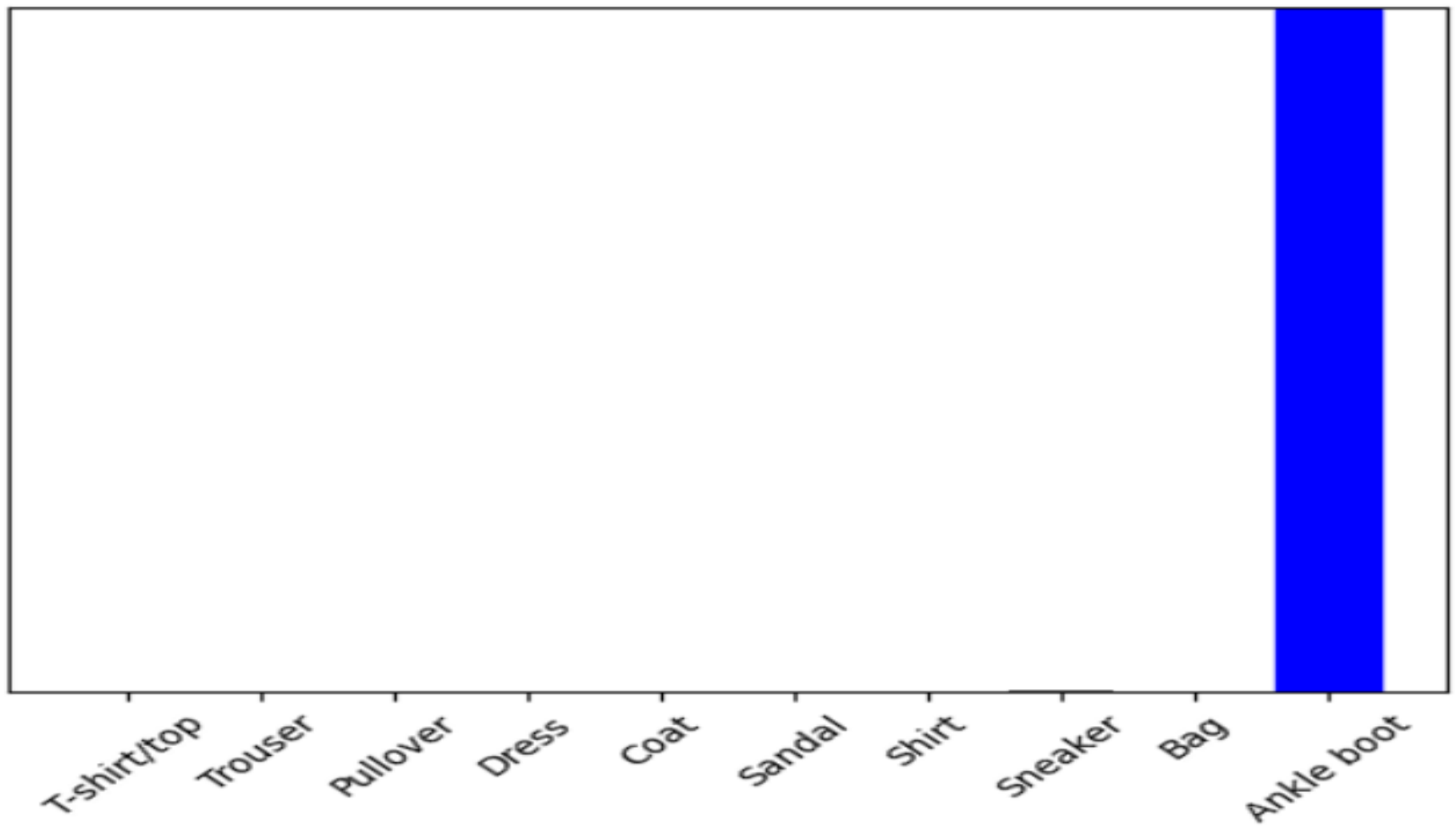



```
[ ]  i = 12
     plt.figure(figsize=(6,3))
     plt.subplot(1,2,1)
     plot_image(i, predictions, test_labels, test_images)
     plt.subplot(1,2,2)
     plot_value_array(i, predictions,  test_labels)
     plt.show()
```




le boot 100% (Ankle boot)
Pullover 100% (Pullover)
Trouser 100% (Trouser)
Trouser 100% (Trouser)
Shirt 75% (Shirt)
Trouser 100% (Trouser)
Coat 100% (Coat)
Shirt 97% (Shirt)
Sandal 100% (Sandal)
neaker 100% (Sneaker)
Coat 96% (Coat)
Sandal 100% (Sandal)
Sandal 48% (Sneaker)
Dress 100% (Dress)
Coat 87% (Coat)

```
plot_value_array(0, predictions_single, test_labels)
_ = plt.xticks(range(10), class_names, rotation=45)
```



CONCLUSION

In this project, we successfully developed a neural network model to classify images from the Fashion MNIST dataset. By leveraging TensorFlow and Keras, we built a straightforward architecture that effectively learned to recognize different clothing items.

Through data preprocessing, we ensured that the images were normalized, which significantly improved the model's performance. After training the model for 5 epochs, we achieved a commendable accuracy on the test dataset, demonstrating the model's ability to generalize well to unseen data.

- Visualization of predictions allowed us to assess the model's strengths and identify areas for improvement. Overall, this project highlights the power of deep learning in image classification tasks and provides a solid foundation for further exploration in computer vision applications.

FUTURE SCOPE

- **Incorporate More Diverse Data**
  Expand the dataset with more varied clothing images or real-world photographs to improve model robustness and accuracy.
- **Algorithm Optimization**
  Experiment with deeper neural networks, convolutional layers, or advanced techniques like transfer learning to enhance prediction quality.
- **Improve Performance**
  Optimize training with techniques such as data augmentation, dropout, or hyperparameter tuning to reduce overfitting and increase accuracy.
- **Deploy on Edge Devices**
  Implement the model on smartphones or edge devices for real-time, offline clothing classification applications.
- **Expand Application Use Cases**
  Apply the system to related tasks such as fashion recommendation, inventory management, or multi-city fashion trend analysis using emerging AI technologies.

REFERENCES

Other References links:

https://www.geeksforgeeks.org/

GitHub Link:

https://github.com/hemashreer11/Awesome_CV_with_Fashion_MNIST_Classification.git

Thank you