Team Name: Data Detectives

1. Introduction

Brief Summary

Our approach integrates information from the structured and image-based data to predict the product attributes of various categories in a fashion dataset. The method employs state-of-the-art convolutional neural network, EfficientNetB0 for the feature extraction from product images and encoded categorical features. The final output is a multi-output neural network; it predicts up to 10 attributes given a product.

---

2. Data Preprocessing

Data Cleaning

- Missing values in categorical attributes have been filled up by the dummy_value for meeting the submission requirements.
- Images that are not available in dataset directory have been replaced by zero matrix of target size - (224x224x3).

Data Exploration and Learnings

- Categories have different number of attributes; this has been handled using category_attributes.parquet file to adjust attributes with respect to their category.
- The characteristics were vastly different between classes and could not be encoded directly without independent label encoding of each characteristic.

Training and Validation Data Split Strategy

- The dataset was split through train_test_split to ensure 80:20 distribution of the training and validation sets with stratification. Classes were well-distributed in both the sets.

Feature Engineering

- The images of each product were resized into (224x224x3) dimensions and normalized with EfficientNet to aid in efficient performance.
- Label Encoding is applied to the categorical features in order to get integer representations, one-hot encoded for use within the model.

- "dummy_value" is added to other attributes above and beyond what is required to complete a category, making the dataset homogeneous.

---

3. Modeling Approach

Model Selection

EfficientNetB0 was selected because of its efficiency and really good performance in image-related tasks. The network was fine-tuned to do feature extraction from the images, along with the dense head added for multi-output classification.

Architecture

- **Backbone:** Pre-trained efficientnetb0 with the classification head removed.
- **Custom Head:** The input goes into a flattened layer followed by dropout (rate=0.3) and then 10 separate dense layers, of which each predicts one attribute with softmax activation.

Final Model and Hyperparameters

- **Model Details**:
  - Input: (224, 224, 3) image tensor
  - Base Model: EfficientNetB0
  - Output: 10 Dense layers with softmax activation (one for each attribute)
- **Hyperparameters**:
  - Optimizer: Adam (default settings)
  - Loss Function: Categorical Crossentropy
  - Batch Size: 32
  - Epochs: 10
  - Dropout: 30% to prevent overfitting.
- Hyperparameter tuning was computationally expensive; default settings were sufficient during experimentation.

---

4. Novelty and Innovation

- **Dynamic Attribute Processing:** Applied category_attributes.parquet to dynamically adapt the attribute prediction pipeline to each category
- **Image substitution:** Uses zero matrix to substitute missing images where data inconsistency occurs.

- **Multi-Output Modelling:** Applied shared image features to predict multi-attributes concurrently, thus reducing computational overhead

---

5. Training Details

- **Environment**:
  - GPU: NVIDIA Tesla K80 (colab environment)
  - RAM: 12 GB
  - Framework: TensorFlow/Keras
- **Training Time**: Approximately 45 minutes for 10 epochs on a dataset of 10,000 images.

---

6. Evaluation Metrics

Chosen Metrics

- Metrics:
  - Micro-F1 Score
  - Macro-F1 Score
  - Harmonic Mean of Micro and Macro F1 Scores
- The Loss Function is Categorical Crossentropy because it supports multi-class classification.

Results

Category-Level F1 Scores:

Category	Micro-F1 Score	Macro-F1 Score	Harmonic Mean
Sarees	0.65	0.62	0.63
Men T-Shirts	0.68	0.67	0.675
...	...	...	...

Attribute-Level F1 Scores (Optional):

Category	Attribute	Micro-F1 Score	Macro-F1 Score	Harmonic Mean
Sarees	attr_1	0.62	0.60	0.61
Men T-Shirts	attr_2	0.67	0.65	0.66

Error Analysis

- **Common Errors**:
    - Over-referring visually similar image patterns, such as print or woven kinds.
    - The performance of those classes who had less samples were low as data was biased.

---

7. Conclusion

Summary of Results

For all categories, the model gave high F1 scores, meaning that the integration of both structured and image data was successful. Shared attributes from the EfficientNet backbone allowed for multi-output predictions in an efficient way with good accuracy.

Limitations and Future Improvements

- **Limitations**:
    - Lower accuracy for classes having lesser sample counts.
    - Limited hyperparameter tuning due to computational constraints.
- **Future Improvements**:
    - Incorporate attention mechanisms to better align the image attributes.
    - Perform data augmentation to enhance generalization.

---

8. Appendix

Additional Observations

- There were dependencies between several categories that can be taken advantage of in subsequent runs by the conditional predictions.

---

9. How to Run the Code

1. Import necessary libraries for environment setup.
2. Load the training files.
3. Inference and generation of the submission.csv file.