Fashion Classification Model

HyperParameter Hunters

Hema Motiani & Umnia Hameed

Introduction

Why is fashion classification important?

- Personalized shopping experiences through recommendation systems
- Helps brands and retailers automate inventory management
- Trend analysis and visual search tools
- Tackles real-world classification challenges using computer vision
- Opportunity to apply technical skills to a creative, high-impact field

Who does it serve?

- E-commerce platforms, fashion retailers, consumers, and tech companies in fashion

Past work

Previous Dataset:

- We were previously working with the small fashion dataset.
- Baseline CNN model trained from scratch
- Achieved high accuracy quickly due to dataset simplicity

Key Takeaways:

- Limited complexity and poor real-world applicability
- Motivated us to use a more realistic dataset

Next Step:

- Switched to the **DeepFashion** dataset.
- Richer, more complex, better-labeled.

DataSet

Now working with the DeepFashion dataset.

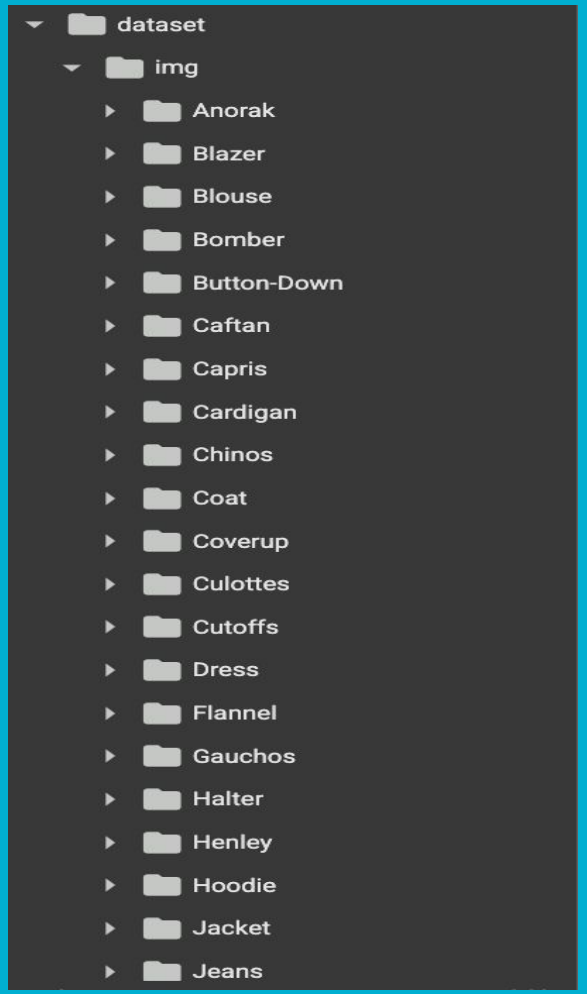
Large-scale benchmark from CUHK's Multimedia Lab.

Key Characteristics:

- 800,000+ images of diverse clothing items in various settings
- Annotated with category labels, bounding boxes, landmarks, and attributes
- Captures real-world variations in pose, lighting, and background
- Contains both consumer and shop images.
- The raw dataset was organized in highly specific directory image names.
- We wrote a Python script to:
    - Map detailed hyper specific categories into 50 high-level classes
    - Selected the top 20 most represented classes
    - Performed a stratified 80-20 train/test split for balanced evaluation

Why This Matters:
This richer, more challenging dataset helps train a robust classification model.

Ideal for building models with better generalization ability

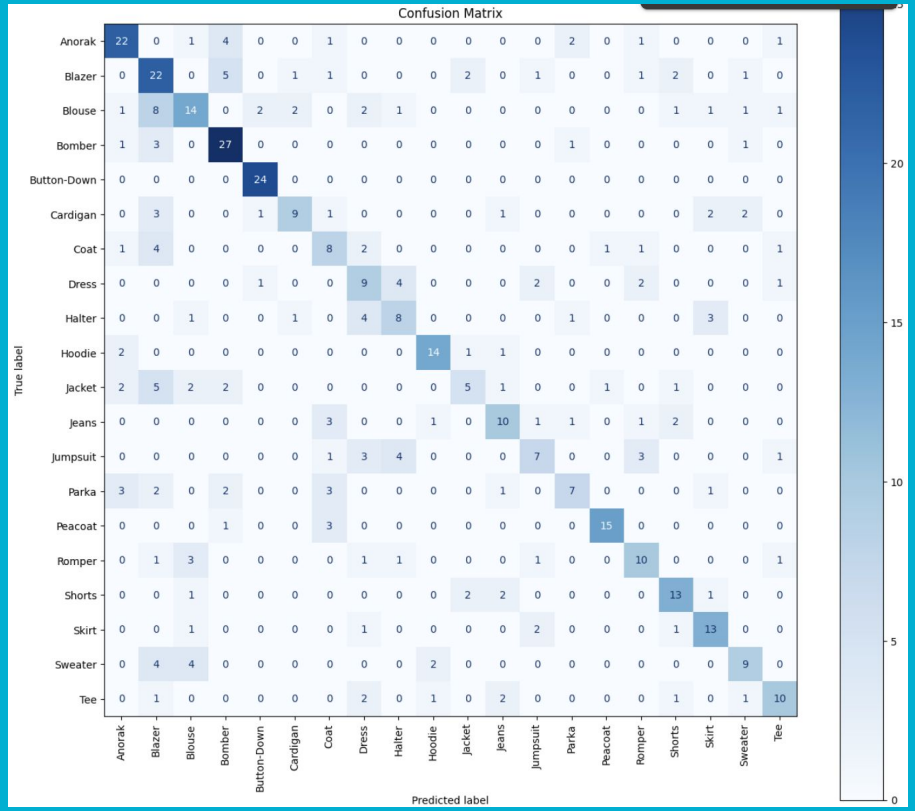


Class Distribution

Method

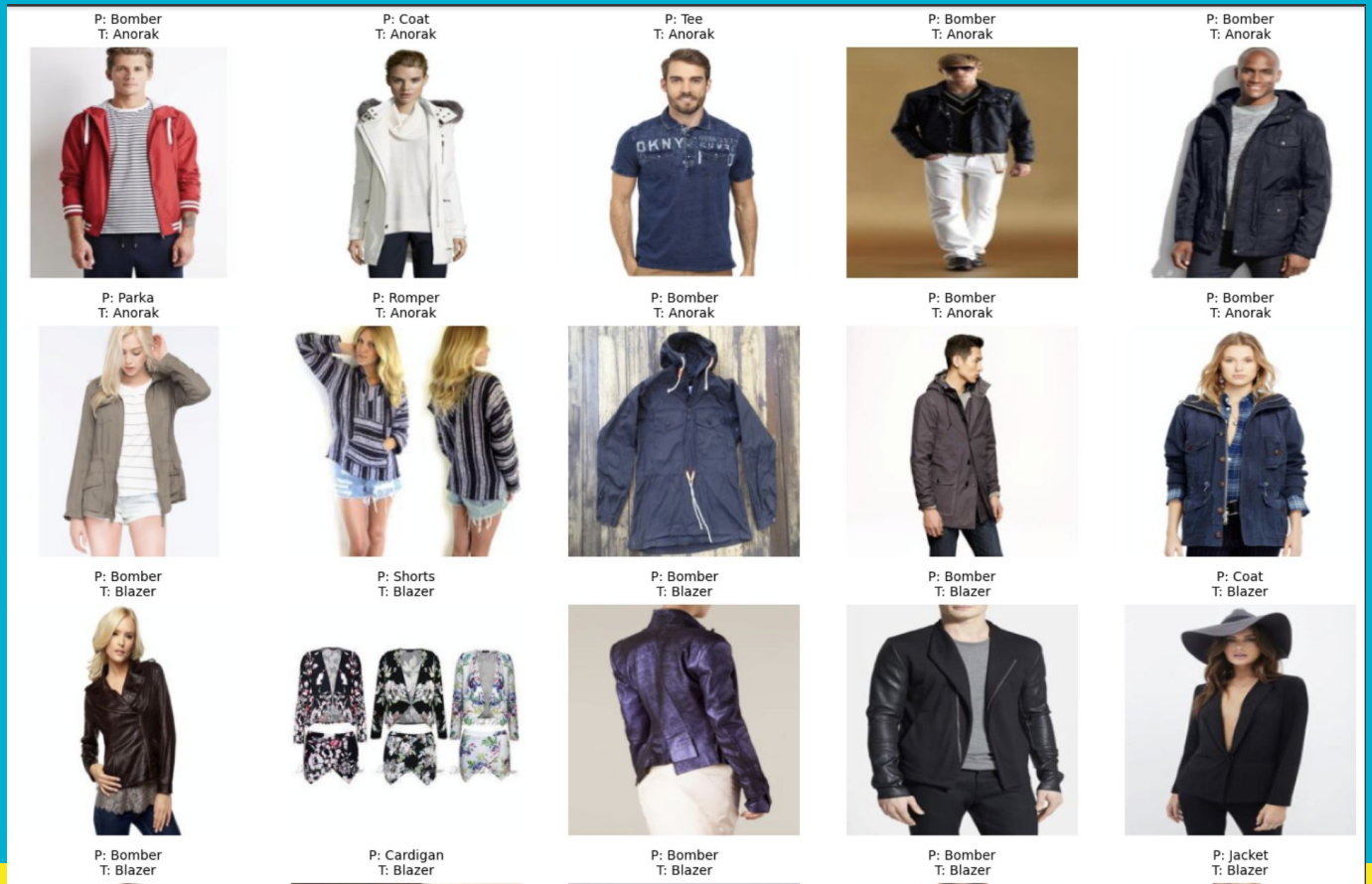
Step	Model 1: 18%	Model 2: 60.27%
Data	Top 20 classes, 80-20 stratified split into train_data and test_data	Top 20 classes, 80-20 stratified split into train_data and test_data
Preprocessing	No Data augmentation	Data Augmentation
Normalization	ImageNet mean & std	ImageNet mean & std
Pretrained Model	EfficientNet-B0	EfficientNet-B0
Transfer Learning	Fully connected Layer trained	Fine tuning last 2 blocks as well as FC Layer
Training	Adam (lr=0.001), CrossEntropyLoss, 10 epochs, Batch size = 32	Adam (lr=0.0005), CrossEntropyLoss, 10 epochs, Batch size = 32 L2 Regularization, LR scheduler

Evaluation



	precision	recall	f1-score
Anorak	0.76	0.69	0.72
Blazer	0.44	0.61	0.51
Blouse	0.53	0.53	0.53
Bomber	0.64	0.85	0.73
Button-Down	0.89	1.00	0.94
Cardigan	0.82	0.47	0.60
Coat	0.42	0.44	0.43
Dress	0.44	0.63	0.52
Halter	0.55	0.61	0.58
Hoodie	0.93	0.78	0.85
Jacket	0.33	0.16	0.21
Jeans	0.39	0.47	0.43
Jumpsuit	0.57	0.42	0.48
Parka	0.64	0.47	0.55
Peacoat	0.89	0.89	0.89
Romper	0.47	0.50	0.49
Shorts	0.52	0.63	0.57
Skirt	0.69	0.61	0.65
Sweater	0.79	0.58	0.67
Tee	0.64	0.39	0.48
accuracy			0.60
macro avg	0.62	0.59	0.59
weighted avg	0.62	0.60	0.60

Misclassifications



Next steps

1. Use older small fashion model on new dataset by aligning the labels.
2. Merge categories that often get misclassified (e.g blazer and Anorak)
3. Run model on more epochs → effect on accuracy