



Google AI ML Winter Camp
谷歌 AI 机器学习应用冬令营

I Fashion

Pick up your favourite one!

Burn My TPU Team

Xia Li, Yang Hu, Chaopeng Zhang

Burn My TPU Team



I ♥ Fashion

Pick up your favourite one!

I saw a cool T-shirt!



But I can't find it in the Internet. So sad..



iFashion
<http://34.80.200.205>



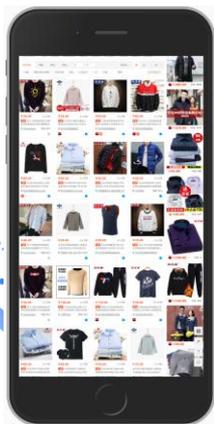
Wow, so cool!





I ♥ Fashion

Pick up your favourite one!



How to recommend??

I love sailor suits !



Career Dresses!



Emmmm,
I wanna...



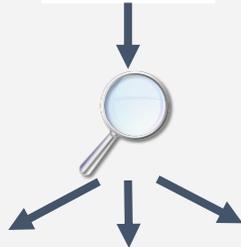
Yes! Sports
Clothes.



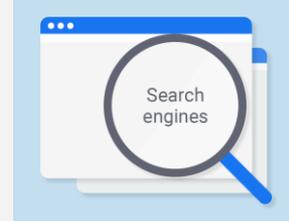
I ♥ Fashion

Pick up your favourite one!

What we do?



How to construct it?



Data Insight

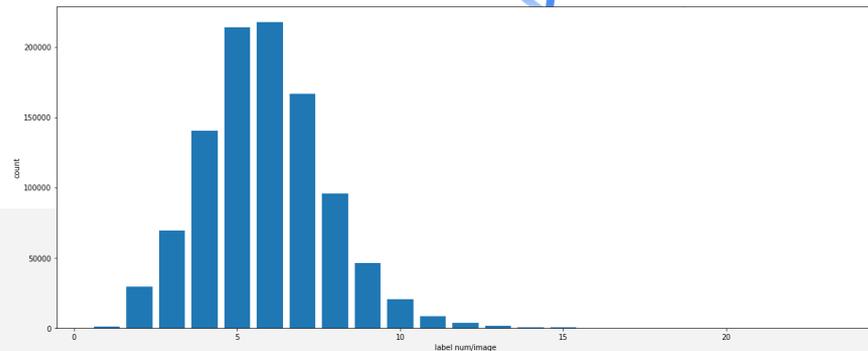
Dive into iMaterialist Dataset

- 1 million+ images
 - 1,014,544 train, 9,897 validation, 39,706 test (~100:1:4)
- 8 kinds of labels:
 - category, color, gender, material, neckline, pattern, sleeve, style
- 228 labels in total.

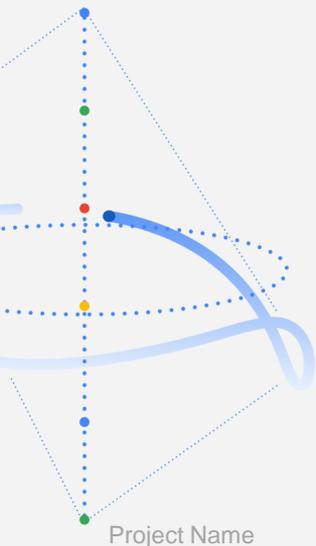


Data Insight

Dive into iMaterialist Dataset



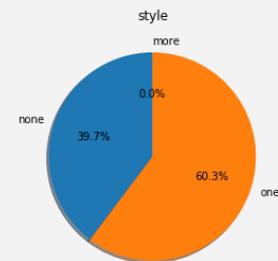
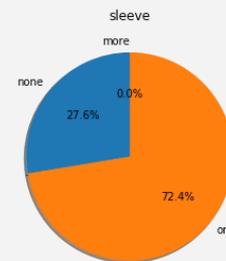
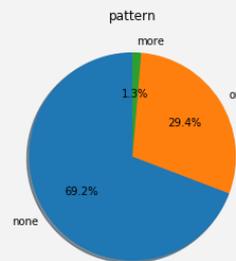
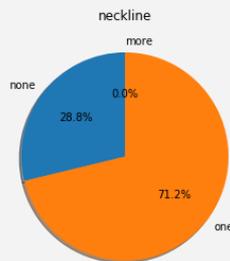
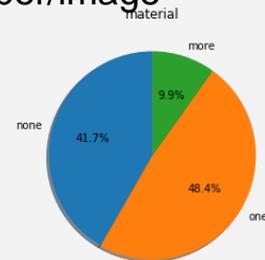
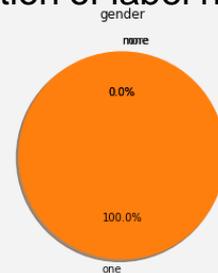
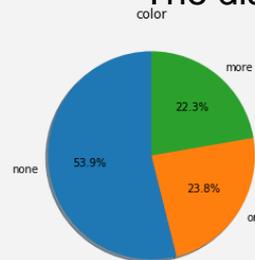
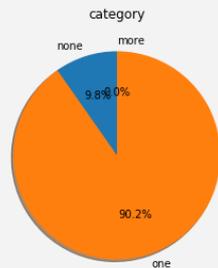
The distribution of label number/image



0 label

1 label

more labels



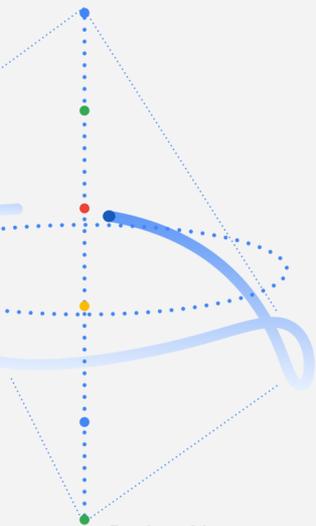
The distribution of label number/image for each task

Data Insight

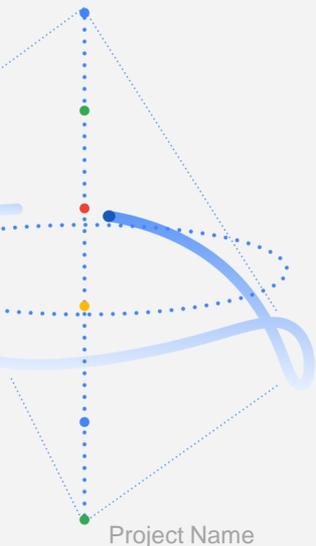
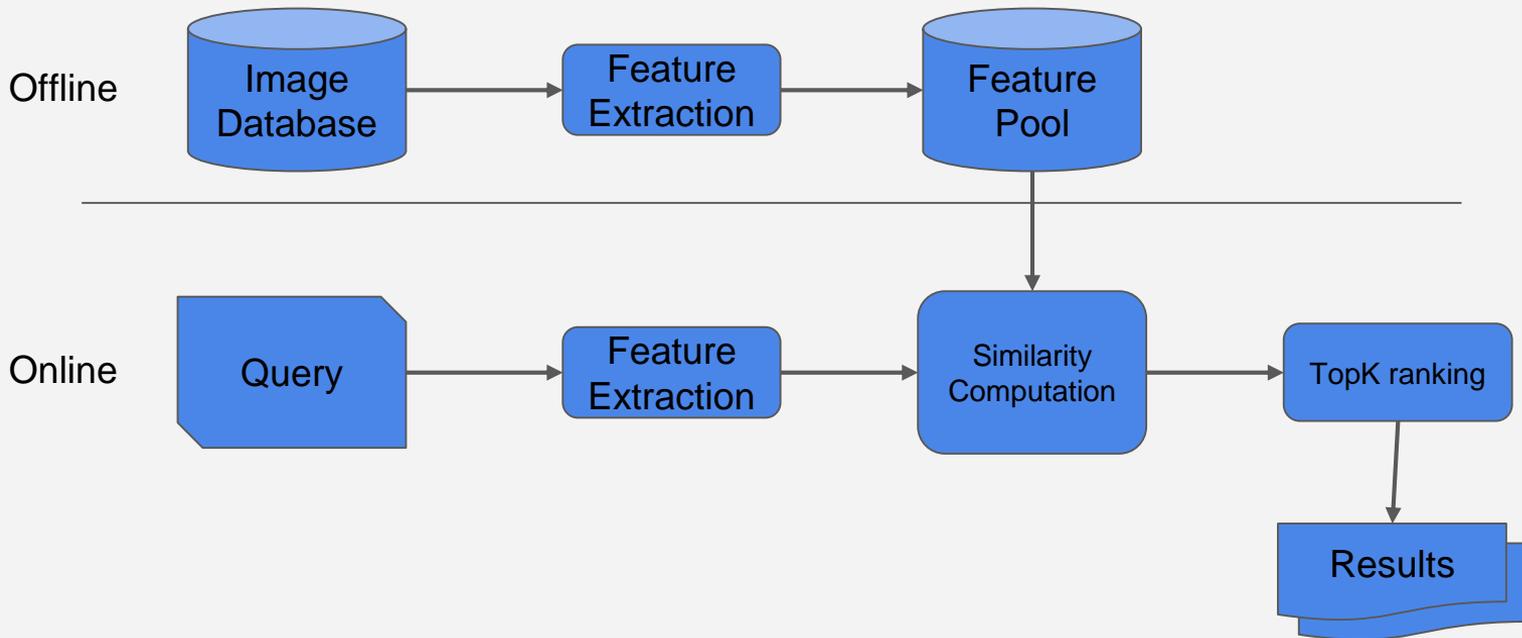
Dive into iMaterialist Dataset



Some non-related images like...

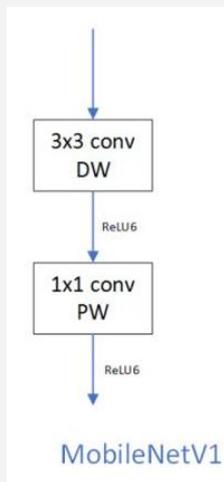
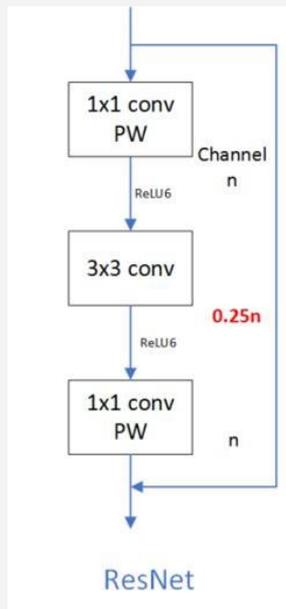


Naive Pipeline



Feature Extractor

Origin choice: ImageNet pretrained model



	ResNet50	MobileNet
top1 error	22.96	26.61
top5 error	6.85	8.95
Params	25,549K	4,232K
FLOPs	3,875M	580M

Feature Extractor

Sophisticated choice: finetune on iMaterialist Dataset

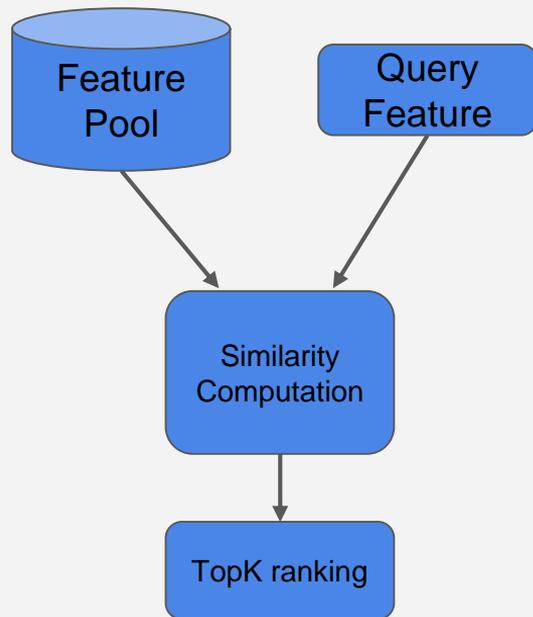
- Classification evaluation metric for iMaterialist
 - micro F1 (average on images)
- **Naive** method:
 - multi-label learning for all labels
 - use Binary Cross Entropy Loss (BCE Loss)
 - ResNet50 + 112*112 + 1 epoch: 45.27
- **Fancy** method: learning
 - learning on each separate task
 - share the same backbone
 - BCE Loss or CE Loss
 - ResNet50 + 112*112 + 1 epoch: 48.48
 - MobileNet + 112*112 + 1 epoch: 47.71
 - MobileNet + 224*224 + 1 epoch: 49.47

36		0.58050
37		0.58018
38		0.57640
39		0.57636
40		0.57552
41		0.57258
42		0.57252
43		0.57246
44		0.57163
45		0.56875
46		0.56555
47		0.56554
48		0.56383
49		0.55773
50		0.55770

Leaderboard of iMaterialist

Similarity Computation

- Similarity metric of features: **Cosine Distance**
- Brutally search:
 - Compare with each image in gallery
- Levelly search:
 - Clustering features using K-means, etc.
 - Compare with cluster center
 - then compare with images in one cluster
- Category search: (used in Alibaba)
 - Clustering images with their attributes
 - ...



Evaluation Method

Experiment settings:

gallery: 100000 images

query: 1000 images

Similarity Metric for images:

$$\text{IOU}(A, B) = |\text{L}(A) \cap \text{L}(B)| / |\text{L}(A) \cup \text{L}(B)|$$

Makeup the groundtruth

if $\text{IOU}(A, B) \geq 0.5$:

pair(A, B)

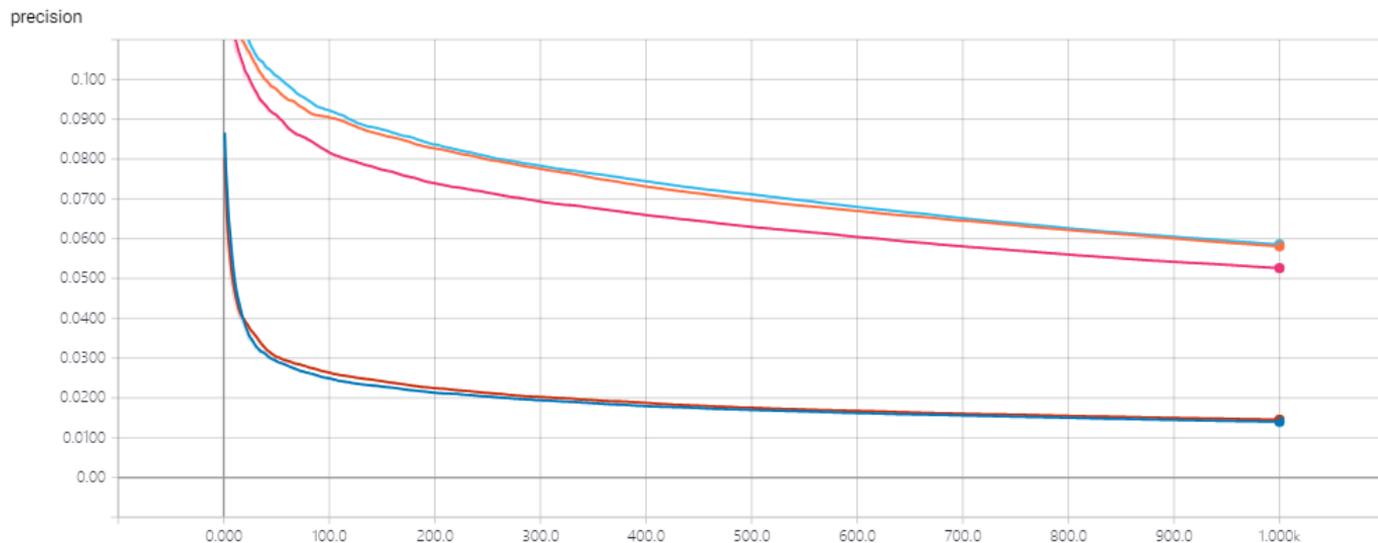
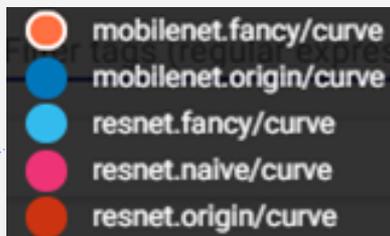
Evaluation metric for search:

precision, recall, F1 for topk results



Feature Extractor

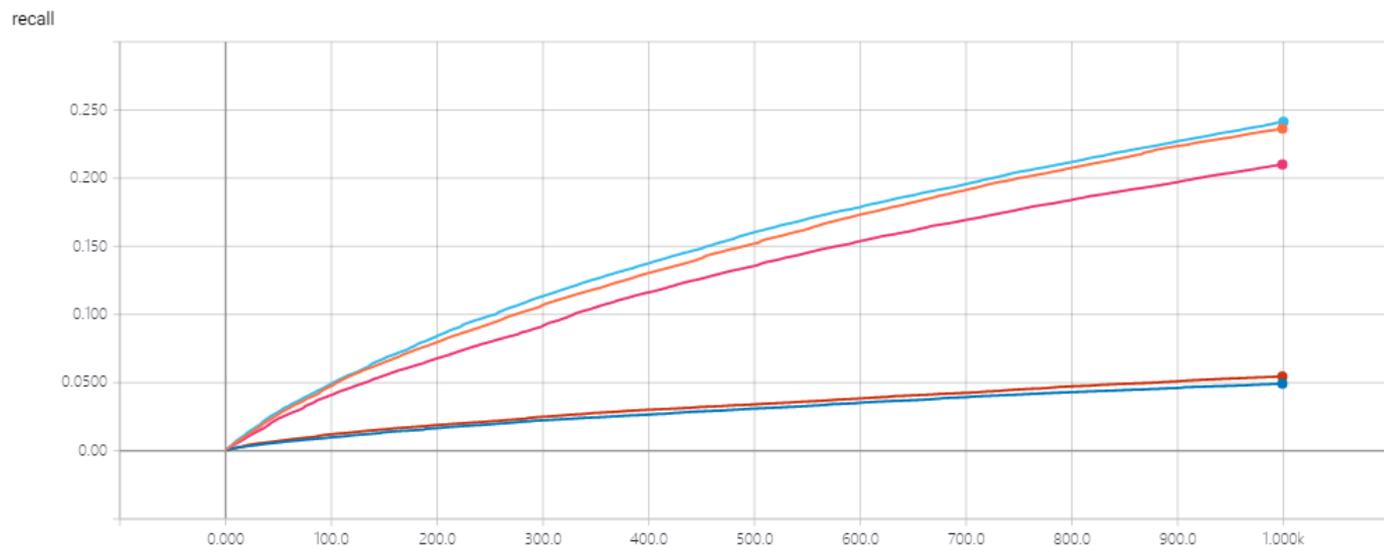
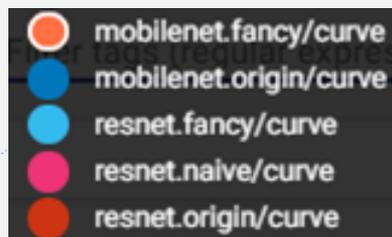
Evaluation upon precision



Project Name

Feature Extractor

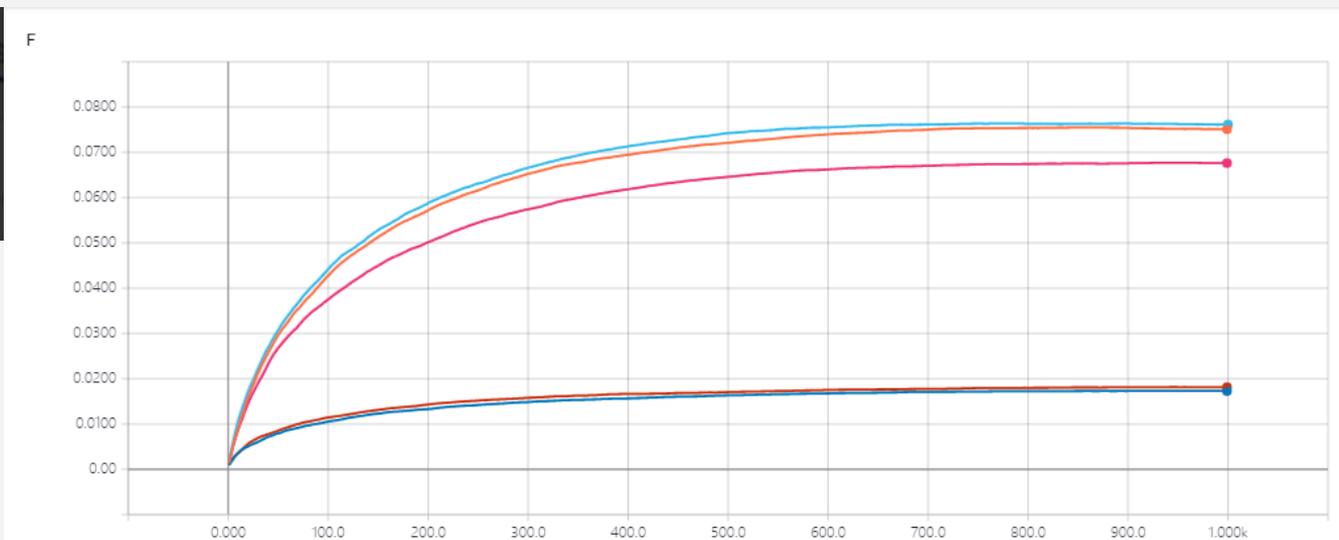
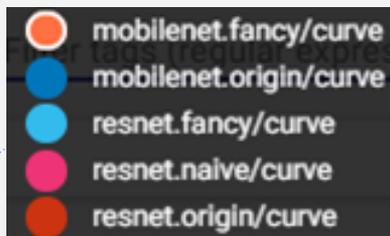
Evaluation upon recall



Project Name

Feature Extractor

Evaluation upon F1



Project Name

Feature Extractor

Visual example: mobilenet pretrained (**origin**)

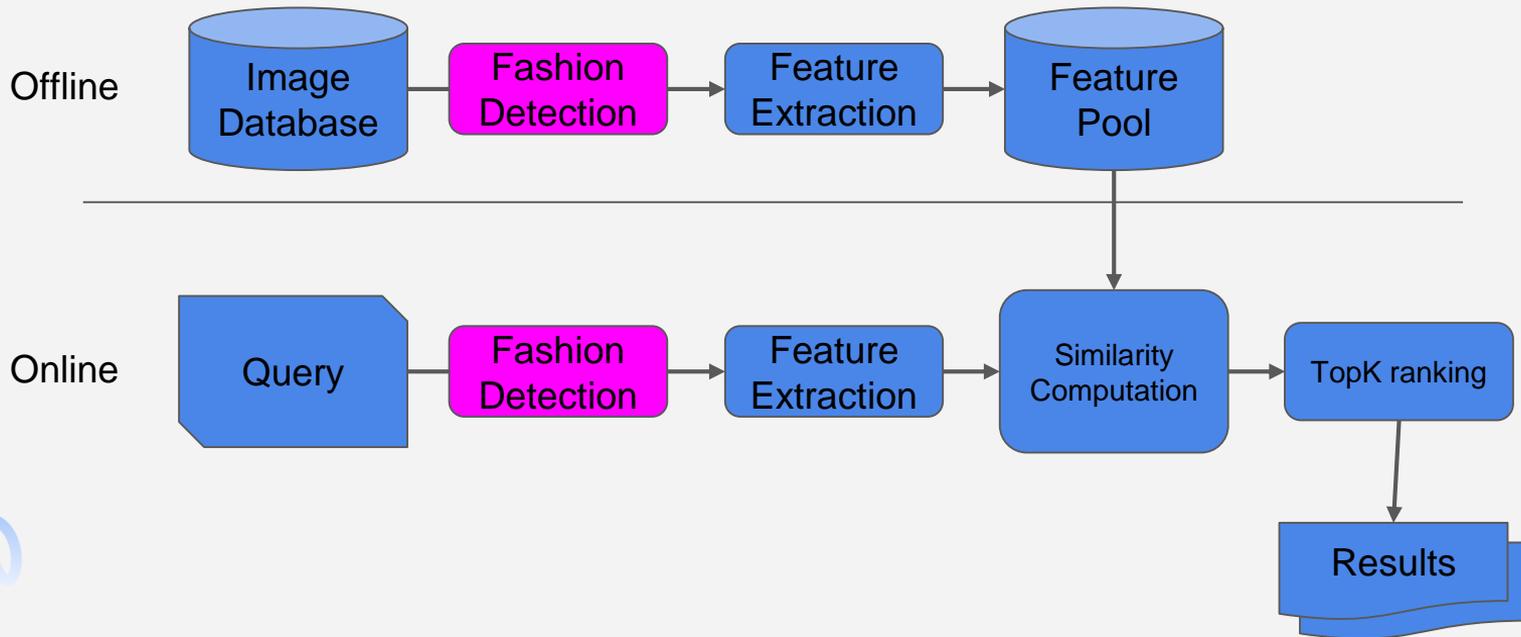


Feature Extractor

Visual example: mobilenet fine-tuned (**fancy**)



Sophisticated Pipeline



Fashion Detection

Mask rcnn-benchmark

<https://github.com/facebookresearch/maskrcnn-benchmark>

Result on Fashion Dataset →

Thinking

- 1: Can't detect fashion
- 2: Mask is not needed



Fashion Detection

Yolov3

<https://github.com/TencentYoutuResearch/ObjectDetection-OneStageDet>

1: In one stage

2: Faster

3: Good performance

4: The work of Our Youtu Lab

- Good performance

544x544	VOC2007 Test(mAP)	Time per forward (batch size = 1)
Yolov2	77.6%	11.5ms
Yolov3	79.6%	23.1ms

But ... can't detect fashion too



Train

Training

- **Dataset**

DeepFashion: Fashion Landmark Detection (By CUHK MMLab)

<http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion/LandmarkDetection.html>

- **VOC Format**

- **Time**

~ 50000 steps

~ 2 days

Project Name

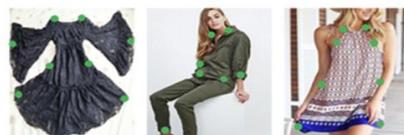
Upper-body Clothes



Lower-body Clothes



Full-body Clothes



Deploy

- Modify the code support the format from voc to single raw image
- Integrated with Classification

Result



Good cases



Bad cases

Deployment



- ❖ Back end
 - ❖ Framework
 - Django
 - ❖ Inference
 - Load Gallery Features, net and weights to GPU when start up
- ❖ Front end
 - ❖ Framework
 - AngularJS
 - ❖ Interaction
 - Send users' data and fetch results from back end

Demo

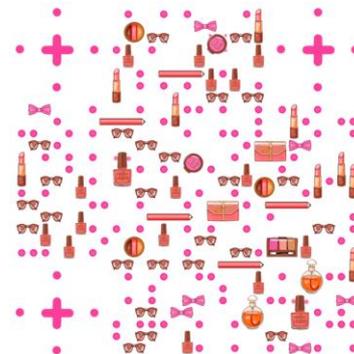
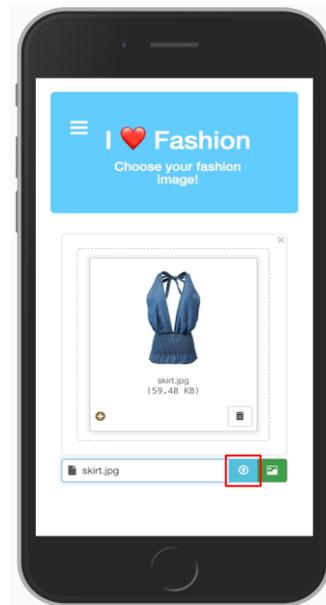
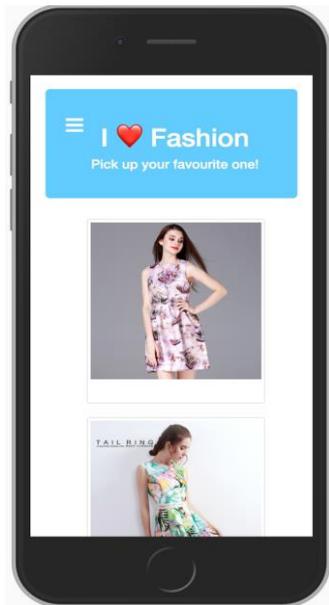
<http://34.80.200.205>

Pick your favorite

or

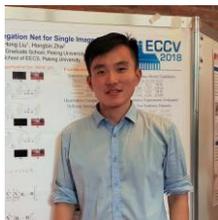
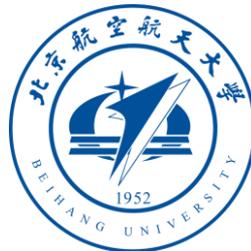
Upload your image

NOW, TRY IT YOURSELF!



Team member

BurnMyTPU



李夏

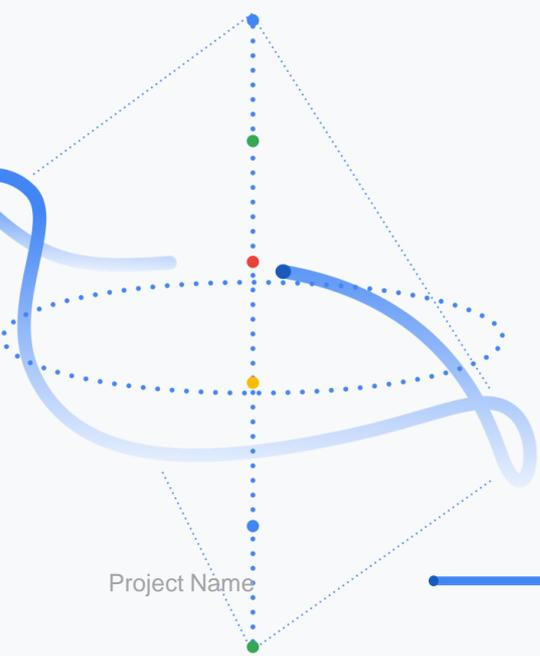


张潮鹏



胡杨

Thank You!



Project Name
