Kaggle Inclusive Images Challenge

Pavel Ostyakov, Samsung AI

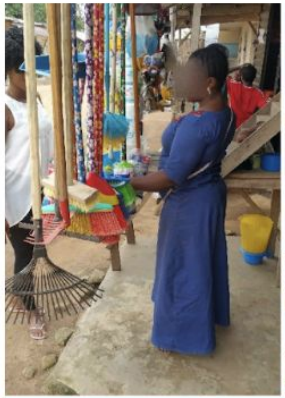
In the money | Gold | Silver | Bronze

#	△pub	Team Name	Kernel	Team Members	Score	Entries	Last
1	▲ 1	[ods.ai] Pavel Ostyakov			0.39151	9	15d
2	▲ 19	vecxoz			0.37374	5	11d
3	▲ 25	WorldWideInclusive			0.37102	3	13d
4	▲ 18	Azat Davletshin			0.33184	5	13d
5	▼ 1	[ods.ai] Artyom Palvelev			0.30785	1	15d
6	▲ 8	cybercore.co.jp			0.30479	5	14d
7	▲ 23	Anastasia Karpovich			0.28790	5	14d
8	▲ 19	novxin			0.27885	1	15d
9	—	Giba			0.27745	1	15d
10	▼ 9	chicm			0.25649	8	13d
11	▲ 26	Matt Motoki			0.22570	2	11d
12	—	RUC1996			0.22522	5	11d
13	▲ 25	Philip Popien			0.22173	3	11d
14	▲ 31	Victor Zhao			0.22018	8	11d
15	▼ 12	[ods.ai] Sergey Malyutin			0.21720	2	15d

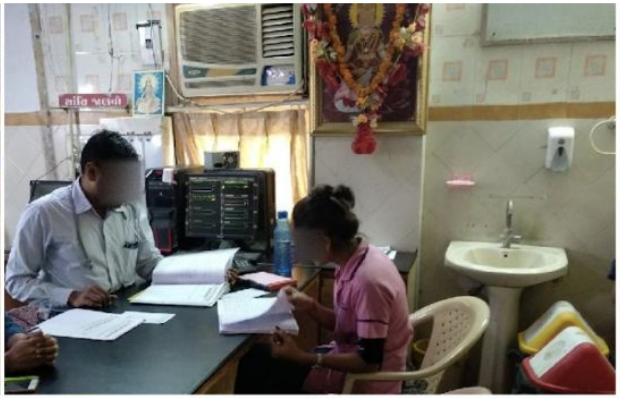
Problem statement

Given a large set of images from one geographic region train a model generalized on another regions.

Evaluation metric: F2 score (like F1, but recall is more important than precision).

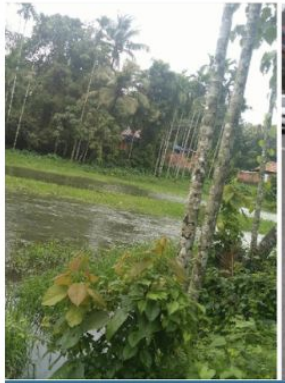


Shopkeeper, Person

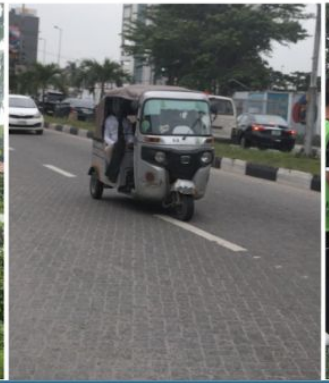


Physician, Hospital, Person

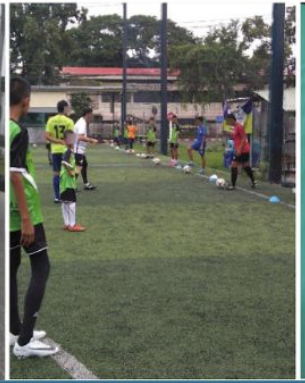
Drummer, Person



Nature, Beauty



Transport, Person, Public transport



Sports, Child, Soccer, Training, Person

Food

Dataset

Train

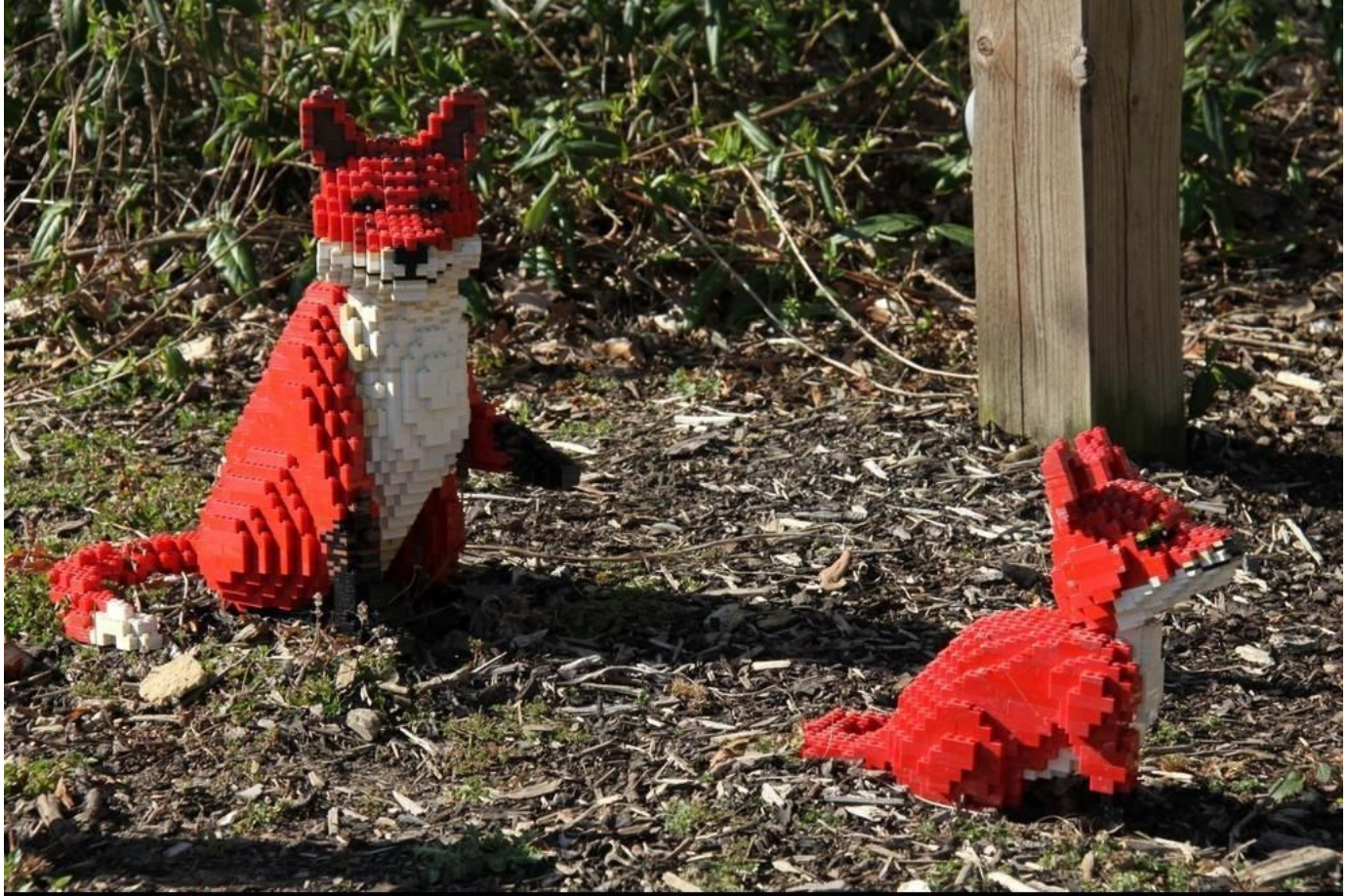
- 1.7 millions images
- More than 7000 classes
- Wikipedia text data

Test (stage1)

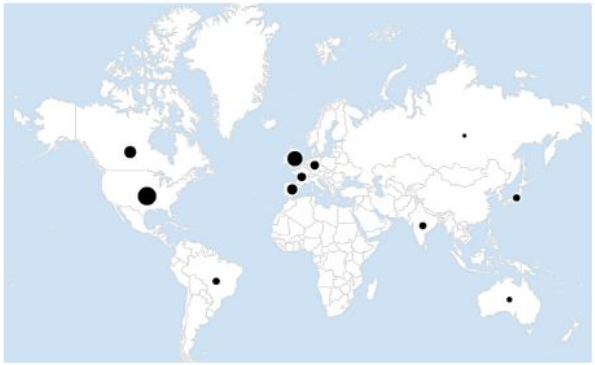
- 32 thousand images
- Ground truth labels for 1 thousand images for tuning models locally

Restrictions

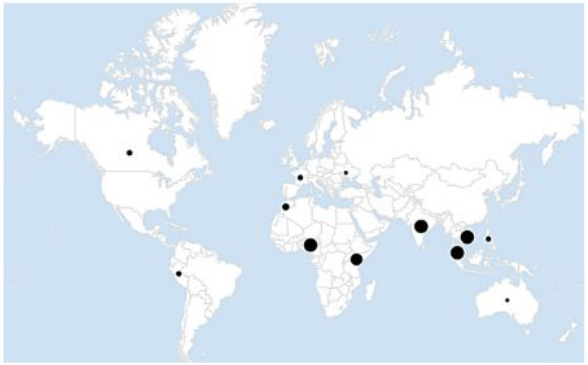
- Pretrained models are not allowed;
- No other data allowed;
- Predictions should be made based on image input only (not metadata);
- Model should be uploaded by the Stage1 deadline;


Foxes after convolution
Right one is pooled version of the left one

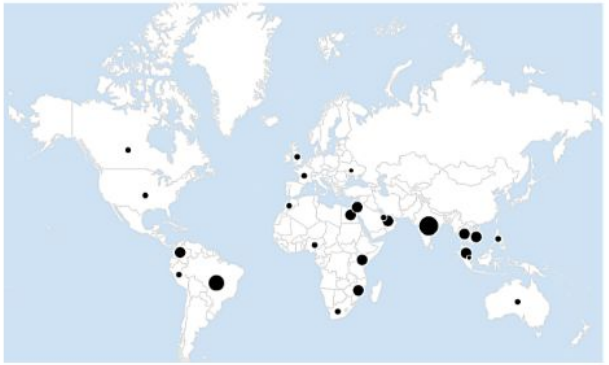
Data distribution



OpenImages Distribution (See Shankar et al., 2017)



Challenge Stage 1 Distribution (Illustrative)



Challenge Stage 2 Distribution (Illustrative)

Key idea

1. Train convolutional neural network on the train dataset;
2. Freeze all layers except the last one and use 1000 images from Stage1 for tuning;
3. Ensemble different models;

Training details

- Input image size: 224x224;
- 100k images for validation;
- Reduce learning rate on plateau;
- Architecture is important;
- Larger batch size is better;

How to use 1000 labels from Stage1

1. Freeze a network and adapt only the last layer to perform better on these images;
2. Adapt the last layers to perform better on both validation and images from Stage1 test.

Model architectures

- PNASNet-5-Large
- NASNet-A-Large
- SENet154
- ResNet152
- ResNet101
- Densenet121

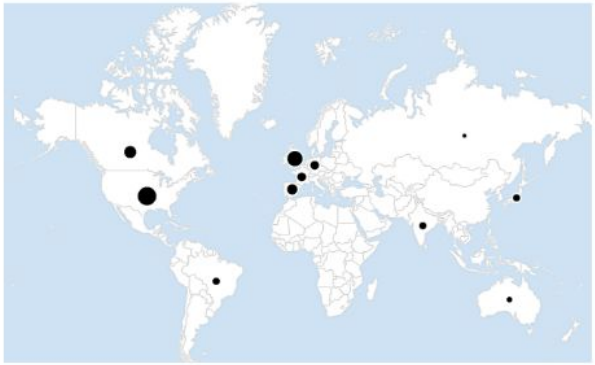
Other findings

- Best single model PNASNet-5-Large gives 4th place;
- The final solution contains ~300 models;
- The distribution on images doesn’t matter while distribution of targets does;

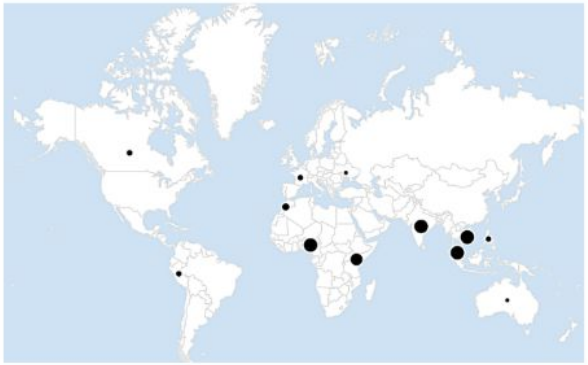
How not to overfit

- Use cross-validation while tuning the last layer;
- Use dropout and strong augmentation while tuning the last layer;
- Choose correct weights in ensemble to perform well both on validation and public leaderboard;

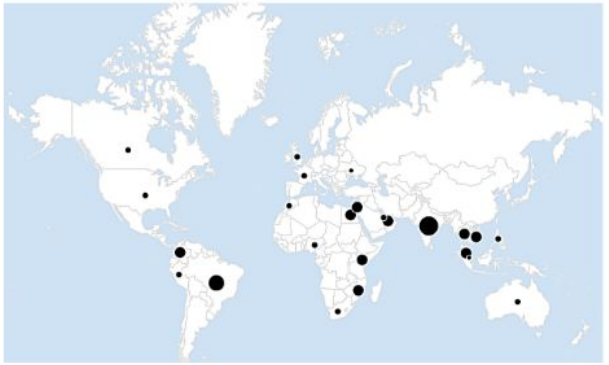
Data distribution



OpenImages Distribution (See Shankar et al., 2017)



Challenge Stage 1 Distribution (Illustrative)



Challenge Stage 2 Distribution (Illustrative)

Pavel Ostyakov

Deep Learning Engineer at Samsung AI Center
Moscow, Russia
Joined 3 years ago · last seen in the past day

Followers 85
Following 12

Competitions Grandmaster

Home | Competitions (23) | Kernels | Discussion (58) | Datasets | ••• | Edit Profile

Competitions Grandmaster

Rank
11
of 93,421

6	2	2

Inclusive Images Challenge · 10 days ago · Top 1%	1st of 468
The 2nd YouTube-8M Vide... · 4 months ago · Top 1%	2nd of 312
Google Landmark Recognit... · 6 months ago · Top 1%	4th of 477

Kernels Contributor

Unranked

0	0	0

No kernel results

Discussion Contributor

Unranked

6	4	27

A simple technique for exte... · 10 months ago	163 votes
My single model [0.057 on ... · 10 months ago	102 votes
Our solution [4th place] · 6 months ago	54 votes

How to win on Kaggle

- Write clean code which could be reused in the future;
- Read all messages on forum;
- Merge with other people;
- Generate many different ideas and discuss then with your teammates/friends;

How to win on Kaggle [2]

- Use ods.ai community;
- Practice, more practice. You should spend as much time as possible;
- Read papers and their implementations;
- Don’t think you cannot get a gold medal, you can!

Contacts

- ODS: ostyakov
- pavelosta@gmail.com