Team GARCH
Approach

CSC Hackathon 2023

Track 1: Image Deduplication
task by ЛУН

GlobalLogic
A Hitachi Group Company

REVENUEGRID

ЛУН

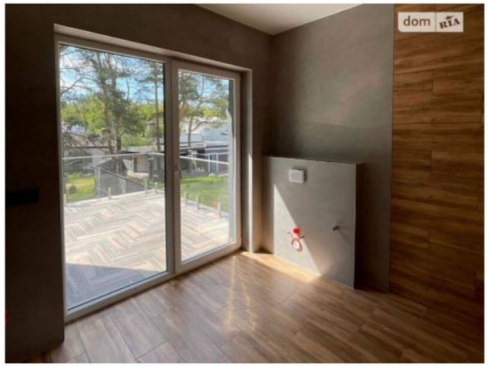
HACKATHON CSC

2023

30/06 - 09/07

Hackathon Expert

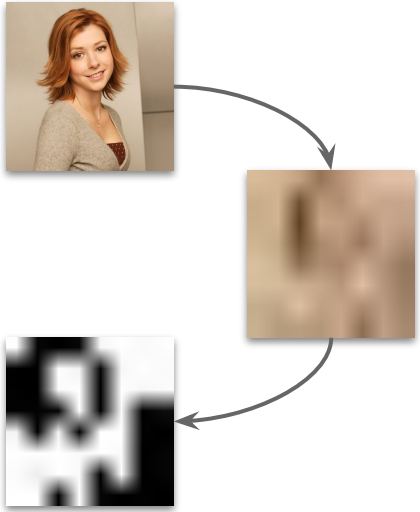
Task & Key observation



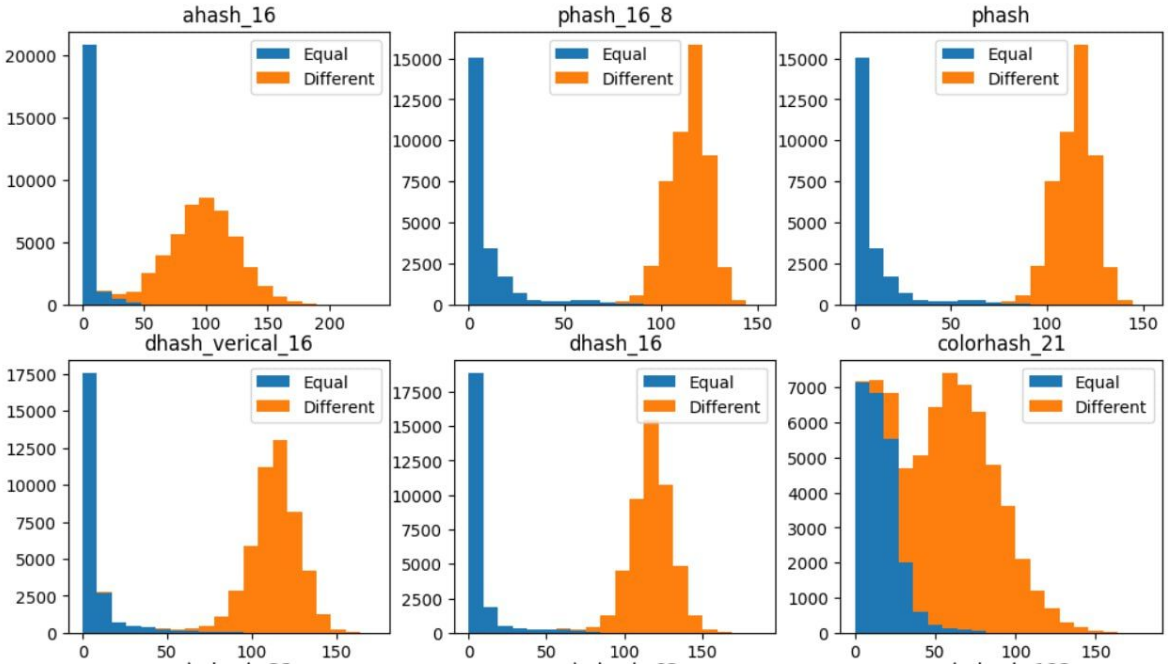
- Images in pairs are extremely similar
- 18% of pairs are identical pixelwise
- Sources of divergency: crops, watermarks, slight difference in angles or illumination

- perceptual hashes
- keypoints detection and matching (SIFT, ORB + FLANN)

Perceptual Hashes

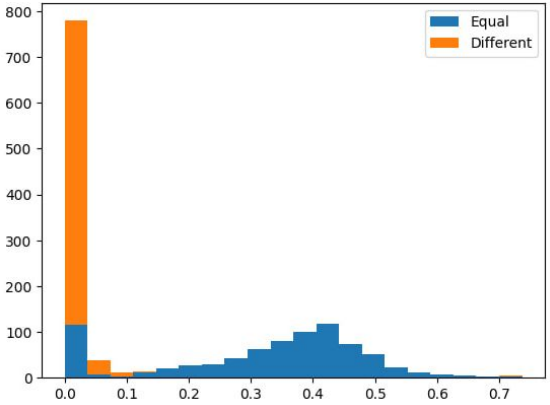


https://hackerfactor.com/blog/index.php?/archives/432-Looks-Like-It.html

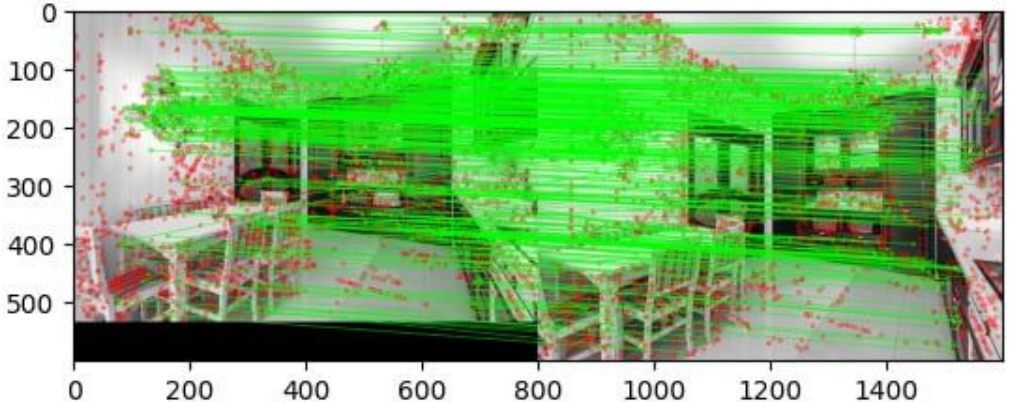


Keypoints detection: SIFT & ORB

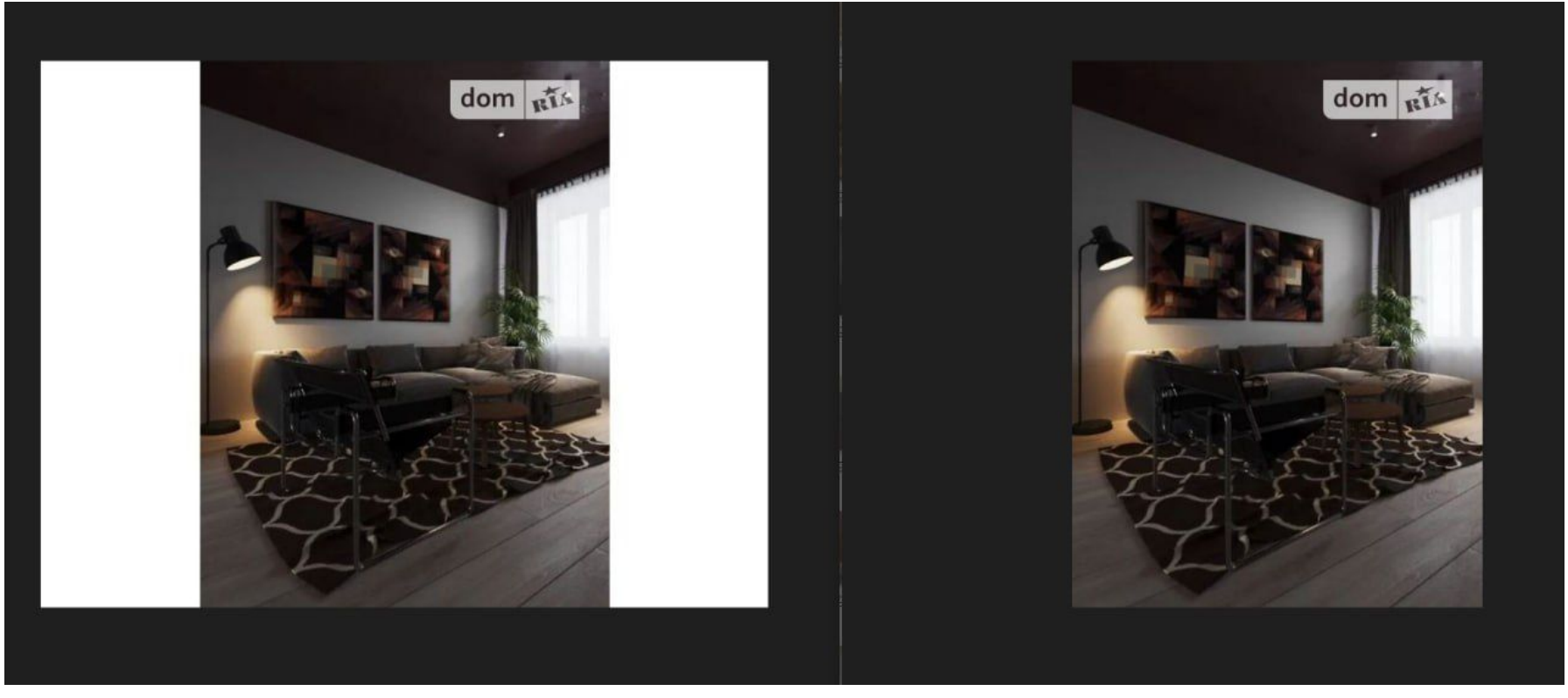
- Needed to handle shrinks, crops etc.
- Used as 1 number: similarity score
- Tuning results: more points is better
- ORB: 9 times faster alternative to SIFT
- Keypoints matching using FLANN



ORB similarity distribution



Preprocessing: unpadding



Preprocessing: garbage in – garbage out

- Review errors, and manually fix mislabeling

True label is 0, predicted 1

Table-driven data science goes brrrr...

- preprocessed images

- hash differences
- ORB/SIFT similarity %
- image sizes

_id	is_same	unpadded.ahash			
0	0.0				
1	1.0				
2	0.0				
3	1.0				
4	0.0				
...	...				
90633	0.0				
90634	1.0				
90635	0.0	2.0		12.0	71.0
90636	0.0	2.0		21.0	74.0
90637	0.0	2.0		11.0	73.0

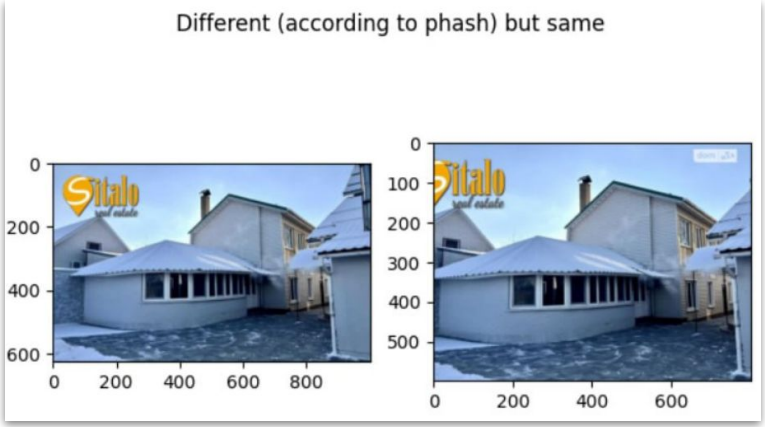
```
features_v2 = [
    "unpadded.ahash_4",
    "unpadded.ahash_8",
    "unpadded.colorhash_21",
    "unpadded.dhash_4",
    "unpadded.dhash_8",
    "unpadded.height_diff",
    "unpadded.height_ratio",
    "unpadded.left_height",
    "unpadded.left_width",
    "unpadded.phash_4",
    "unpadded.phash_8",
    "unpadded.right_height",
    "unpadded.right_width",
    "unpadded.sift_similarity",
    "unpadded.whash_4_haar",
    "unpadded.whash_8_haar",
    "unpadded.width_diff",
    "unpadded.width_ratio",
]
```

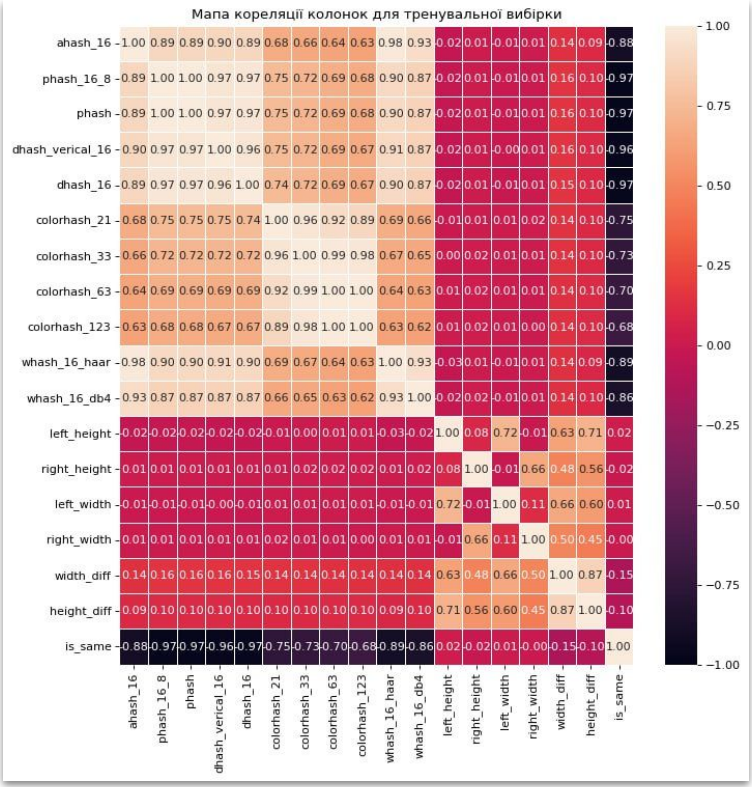
LightGBM

- 5 folds cross-val
- Grid-search threshold for the best F1 macro

Diversity

- Boosting tends to overfit to good predictors
- Features should be diverse, not precise
- Lower dimensional hashes - better results in the final model





“Light” model

- faster
- better on validation

pair prediction time: **0.264** s (on 1 CPU)
(no batches, no image loading time included)

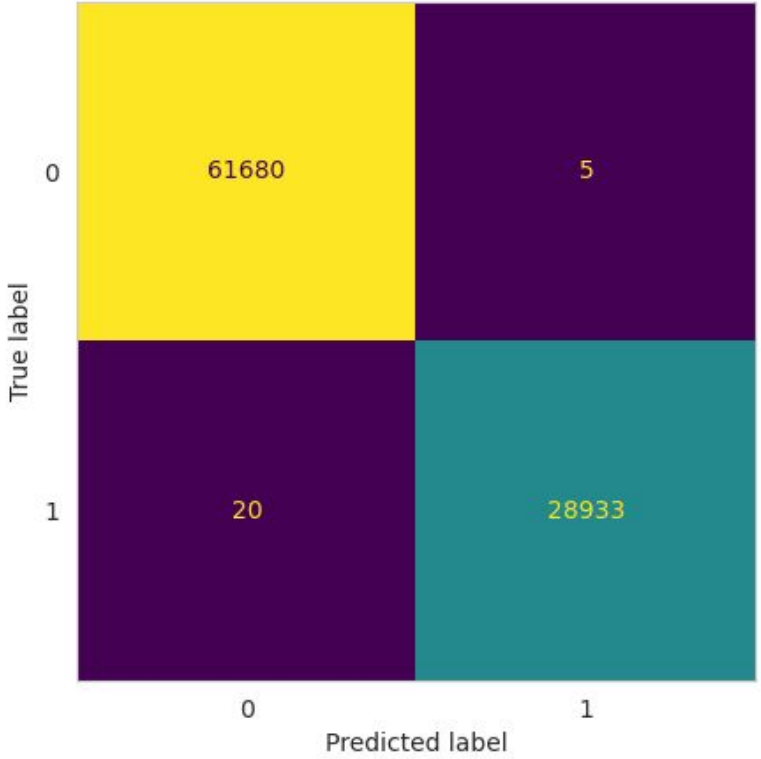
Features:

- `"unpadded.ahash_8",`
- `"unpadded.colorhash_21",`
- `"unpadded.dhash_8",`
- `"unpadded.phash_8",`
- `"unpadded.orb_similarity",`
- `"unpadded.whash_8_haar"`

LightGBM

- num_iterations: 38
- threshold: 0.5

Results: out of fold validation (train)

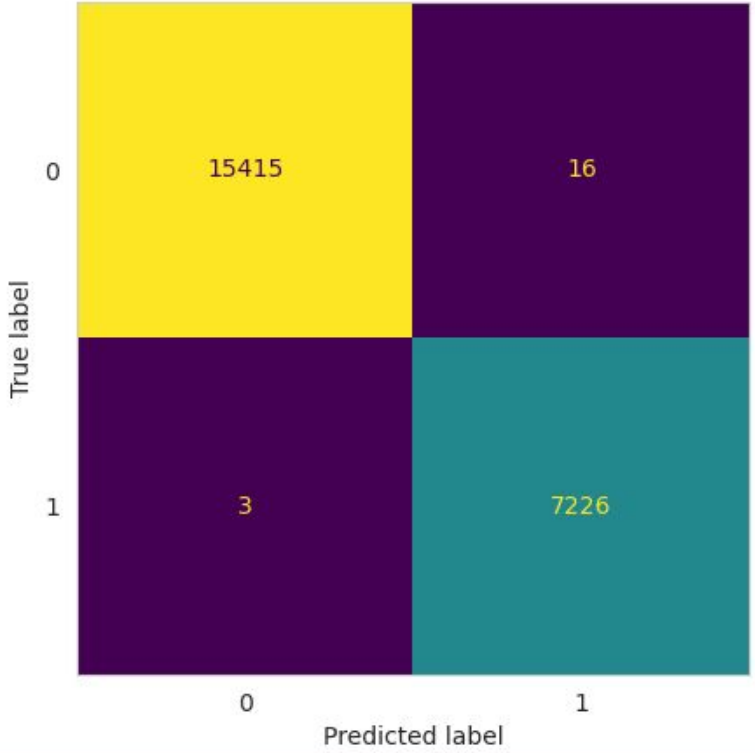


	precision	recall	f1-score	support
0.0	0.999676	0.999919	0.999797	61685
1.0	0.999827	0.999309	0.999568	28953
accuracy			0.999724	90638
macro avg	0.999752	0.999614	0.999683	90638
weighted avg	0.999724	0.999724	0.999724	90638

roc_auc_score(y, y_pred)=0.9999010115426035

pair prediction time: 0.264 s (on 1 CPU)
(no batches, no image loading time included)

Results: test



	precision	recall	f1-score	support
0	0.999805	0.998963	0.999384	15431
1	0.997791	0.999585	0.998687	7229
accuracy			0.999162	22660
macro avg	0.998798	0.999274	0.999036	22660
weighted avg	0.999163	0.999162	0.999162	22660

pair prediction time: 0.264 s (on 1 CPU)
(no batches, no image loading time included)

“Heavy” model

- slower
- better on test

pair prediction time: **0.9836** s (on 1 CPU)
(no batches, no image loading time included)

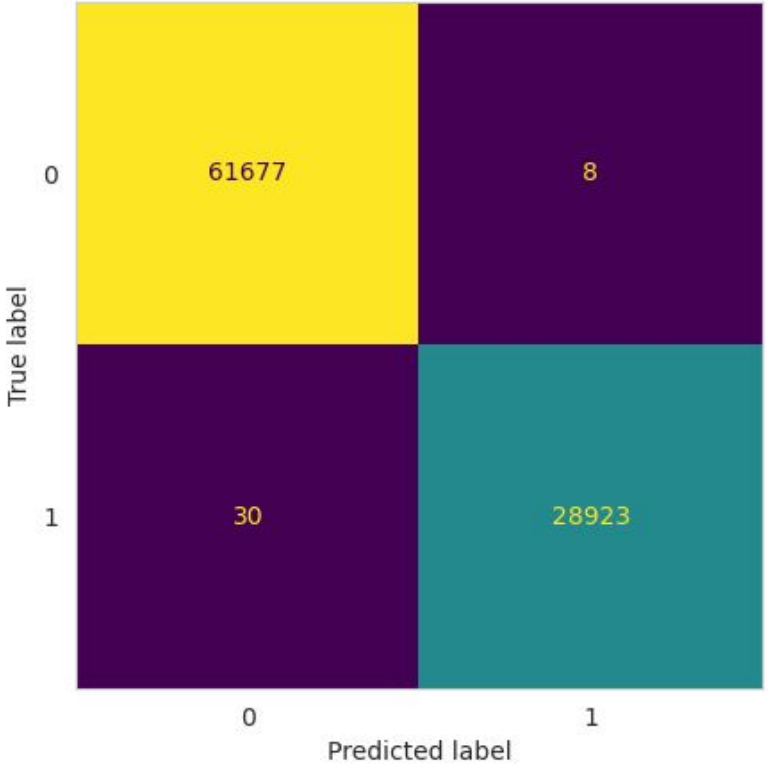
Features:

- "unpadded.ahash_4",
- "unpadded.ahash_8",
- "unpadded.colorhash_21",
- "unpadded.dhash_4",
- "unpadded.dhash_8",
- "unpadded.height_diff",
- "unpadded.height_ratio",
- "unpadded.left_height",
- "unpadded.left_width",
- "unpadded.phash_4",
- "unpadded.phash_8",
- "unpadded.right_height",
- "unpadded.right_width",
- "unpadded.sift_similarity",
- "unpadded.whash_4_haar",
- "unpadded.whash_8_haar",
- "unpadded.width_diff",
- "unpadded.width_ratio"

LightGBM

- num_iterations: 43
- threshold: 0.5

Results: out of fold validation (train)

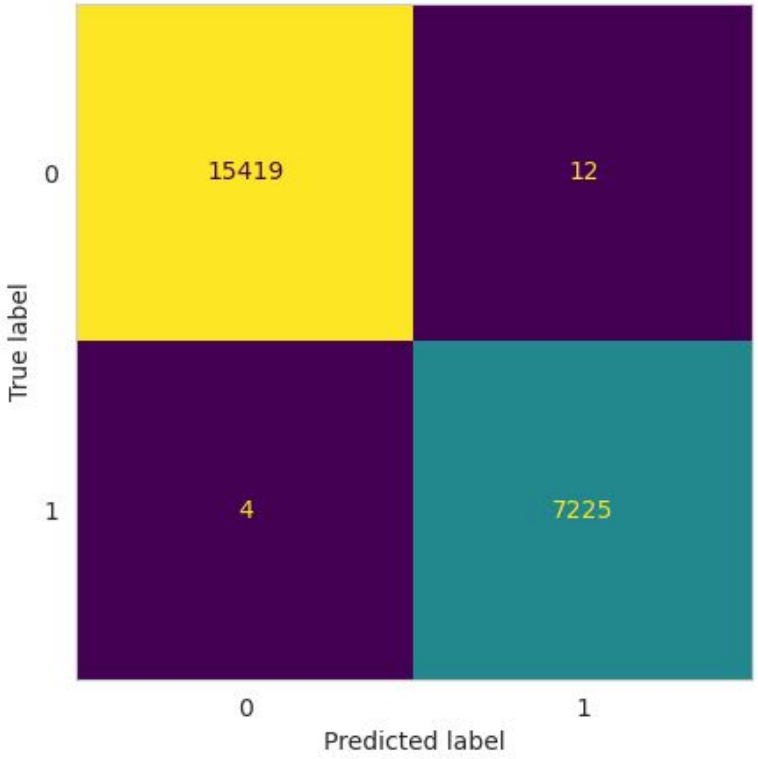


	precision	recall	f1-score	support
0.0	0.999514	0.999870	0.999692	61685
1.0	0.999723	0.998964	0.999344	28953
accuracy			0.999581	90638
macro avg	0.999619	0.999417	0.999518	90638
weighted avg	0.999581	0.999581	0.999581	90638

roc_auc_score(y, y_pred)=0.9998494195693741

pair prediction time: 0.9836 s (on 1 CPU)
(no batches, no image loading time included)

Results: test (“heavy” model)



	precision	recall	f1-score	support
0	0.999741	0.999222	0.999481	15431
1	0.998342	0.999447	0.998894	7229
accuracy			0.999294	22660
macro avg	0.999041	0.999335	0.999188	22660
weighted avg	0.999294	0.999294	0.999294	22660

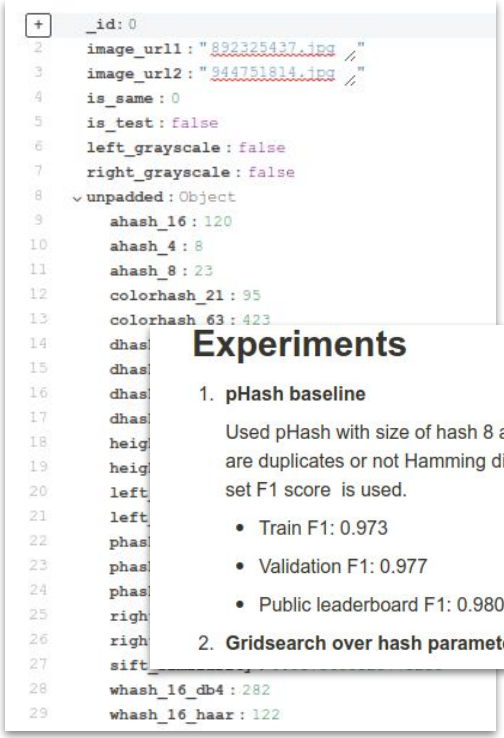
pair prediction time: 0.9836 s (on 1 CPU)
(no batches, no image loading time included)

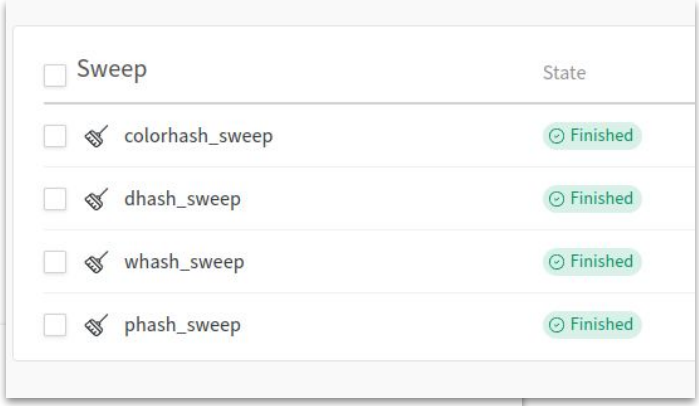
Further work

- ✨ thorough feature selection
- 🔧 better preprocessing
- 👍 data augmentation
- 🤔 crop resistant hash
- 🤔 neural networks

Side note: tools

- DB for features
- code in VCS
- Notion & WandB for experiments tracking





Experiments

1. **pHash baseline**

   Used pHash with size of hash 8 and high frequency factor of 4. To figure out whether images are duplicates or not Hamming distance between them is used. Best threshold in terms of train set F1 score is used.

   - Train F1: 0.973
   - Validation F1: 0.977
   - Public leaderboard F1: 0.9807

2. **Gridsearch over hash parameters**

Thank you for your attention!
Any questions?

github.com/nikiandr/csc_hackathon_lun

Time for demo