OOD-CV Challenge Report

September 18, 2023

1 Team details

- Challenge track: OOD-CV Workshop SSB Challenge (OSR Track - ImageNet1k)
- Team name: DAIU
- Team leader name: Mengjia Wang
- Team leader address, phone number, and email: 266 Xinglong Section, Xifeng Road, Xi'an City, Shaanxi Province, China. (+86)13102818603. 3230724499@qq.com
- Rest of the team members: Min Gao, Jingwen Zhang
- Team website URL: None
- Affiliation: School of Artificial Intelligence, Xidian University, Xi'an, China
- User names on the OOD-CV Codalab competitions: DAIU

- Link to the codes of the solution(s): `https://github.com/wmj183363206/osr-imagenet-1k-1st`

2 Contribution details

- Title of the contribution : Efficient Implementation of Open-Set Recognition Task

- General method description: 1. Our Team used Test Time Augmentation (TTA) [1] in inference stage, like 'HorizintalFlip'; 2. Our Team used lost of models, such as "deit3-base-patch16-384.fb-in1k" and "deit3-large-patch16-224.fb-in1k" [2]; 3. Our Team also use a common fusion strategy, which is Hard Voting Classifier (not use because lower scores).

- Description of the particularities of the solutions deployed for each of the tracks : 1. As for TTA Strategy, we used "Scale", "HorizontalFlip" and "VerticalFilp"; 2. The models we used are "deit3-base-patch16-224.fb-in1k", "deit3-base-patch16-384.fb-in1k", "deit3-large-patch16-224.fb-in1k", "deit3-large-patch16-384.fb-in1k" and "deit3-huge-patch14-224.fb-in1k"; 3. The Hard Voting Classifier, which we brought lower scores and finally did not use, is taking the average probability of all model prediction samples in a certain category as the standard, and the corresponding type with the highest probability is the final prediction result.

- References:

[1] Eugene Khvedchenya. Pytorch toolbelt. `https://github.com/BloodAxe/pytorch-toolbelt`, 2019.

[2] Hugo Touvron, Matthieu Cord, and Herve Jegou. Deit iii: Revenge of the vit. *arXiv preprint arXiv:2204.07118*, 2022.

- Representative image / diagram of the method(s): The method are shown in the following two tables.

Table 1: Hard Data Scores of Different Methods on Local not on Leaderboard

Method	Hard Data						
	AVG	AUROC	FPR	OSCR	ACC	AUOUT	DTERR
deit3-base-patch16-224.fb-in1k	0.8368	0.6890	0.6875	0.7079	0.3360	0.8209	0.5853
deit3-base-patch16-384.fb-in1k	0.8476	0.6838	0.7052	0.7167	0.3281	0.7965	0.6010
deit3-large-patch16-224.fb-in1k	0.8463	0.6697	0.6932	0.7032	0.3360	0.8097	0.5888
deit3-large-patch16-384.fb-in1k	0.8560	0.7022	0.7136	0.7289	0.3169	0.7894	0.6181
deit3-huge-patch14-224.fb-in1k	0.8515	0.6990	0.6986	0.7182	0.3307	0.808	0.6050

Table 2: Easy Data Scores of Different Methods on Local not on Leaderboard

Method	Easy Data						
	AVG	AUROC	FPR	OSCR	ACC	AUOUT	DTERR
deit3-base-patch16-224.fb-in1k	0.8368	0.7097	0.7614	0.7562	0.2888	0.7162	0.6312
deit3-base-patch16-384.fb-in1k	0.8476	0.7173	0.7776	0.7724	0.2780	0.6923	0.6492
deit3-large-patch16-224.fb-in1k	0.8463	0.6862	0.7615	0.7519	0.2883	0.7162	0.6311
deit3-large-patch16-384.fb-in1k	0.8560	0.7101	0.7793	0.7706	0.2781	0.6852	0.6548
deit3-huge-patch14-224.fb-in1k	0.8515	0.6992	0.7663	0.7590	0.2873	0.7037	0.6400

3 Global Method Description

[* Indicates method used in competition test results.]

- Total method complexity: 1. deit3-base-patch16-224.fb-in1k with 2078MB peak memory and 17.5 GFLOPs; 2. deit3-base-patch16-384.fb-in1k with 8956MB peak memory and 55.5 GFLOPs; 3. deit3-large-patch16-224.fb-in1k with 3789MB peak memory and 61.6 GFLOPs; 4. deit3-large-patch16-384.fb-in1k with 12866MB peak memory and 191.2 GFLOPs; 5. deit3-huge-patch14-224.fb-in1k with 6984MB peak memory and 167.4 GFLOPs;

- Model Parameters: 1. deit3-base-patch16-224.fb-in1k with 86.6Mb model parameter; 2. deit3-base-patch16-384.fb-in1k with 86.9Mb model parameter; 3. deit3-large-patch16-224.fb-in1k with 304.4Mb model parameter; 4. deit3-large-patch16-384.fb-in1k with 304.8Mb

model parameter; 5. deit3-huge-patch14-224.fb-in1k with 632.1Mb model parameter;

- Run Time: 1. deit3-base-patch16-224.fb-in1k with 9min 14s total run time; 2. deit3-base-patch16-384.fb-in1k with 29min 50s total run time; 3. deit3-large-patch16-224.fb-in1k with 27min 24s total run time; 4. deit3-large-patch16-384.fb-in1k with 88min 55s total run time; 5. deit3-huge-patch14-224.fb-in1k with 71min 41s total run time;

- Which pre-trained or external methods / models have been used: 1. deit3-base-patch16-224.fb-in1k; 2. deit3-base-patch16-384.fb-in1k; 3. deit3-large-patch16-224.fb-in1k; 4. deit3-large-patch16-384.fb-in1k; 5. deit3-huge-patch14-224.fb-in1k;

- Training description : We simply trained the model with basic data augmentation, like Resize, RandomCrop, RandomFlip and Normalization. The training optimizer is Adamw with 0.00005 learning rate with 6 epochs. But we found its training scores was lower than the directly inference on the local evaluation.

- Testing description: We inferenced the model with TTA like "Scale", "HorizontalFlip" and "VerticalFilp". Then we get the results.

- Quantitative and qualitative advantages of the proposed solution : In this challenge, our team tried 5 different pertrained models based on Imagenet-1k dataset. The Training only used the deit3-base-patch16-224.fb-in1k, and we found its performance was not very good. In addition, with the limitation of the time, our team did not go to deep.

- Results of the comparison to other approaches (if any) : None

- Novelty of the solution and if it has been previously published: We add TTA in the inference stage and found that the result was in good

performance. This strategy is previously published.

4 Ensembles and fusion strategies

- Describe in detail the use of ensembles and/or fusion strategies (if any).: The Hard Voting Classifier, which we brought lower scores and finally did not use, is taking the average probability of all model prediction samples in a certain category as the standard, and the corresponding type with the highest probability is the final prediction result.

- What was the benefit over the single method? : No benefit.

- What were the baseline and the fused methods? : The baseline is the deit3-base-patch16-224.fb-in1k, and the fusion method is Hard Voting Classifier.

5 Technical details

- Language and implementation details (including platform, memory, parallelization requirements) : We used Pytorch, single GPU training and testing, totally used 2 GeForce RTX 3090.

- Human effort required for implementation, training and validation?: The mainly human effort for implementation was in the downloading data; for training was in the training code; not much effort in the validation.

- Training/testing time? Runtime at test per image : The whole training/testing time was shown above, and the runtime at test per image: 1. deit3-base-patch16-224.fb-in1k with 831 im/s; 2. deit3-base-patch16-384.fb-in1k with 190 im/s; 3. deit3-large-patch16-224.fb-in1k with 277 im/s; 4. deit3-large-patch16-384.fb-in1k with 67 im/s; 5.

deit3-huge-patch14-224.fb-in1k with 112 im/s;

- Comment the efficiency of the proposed solution(s)? : The training was time consuming and worse performance. As for the inference, the directly inference with TTA was the best solution we have found.

6 Other details

- General comments and impressions of the OOD-CV challenge. : The SSB challenge in the osr track has great room for development in the future. We are very grateful for OOD-CV official hosting such a competition.

- Other comments: None