H2ST: Hierarchical Two-Sample Tests for Continual Out-of-Distribution Detection

Supplementary Material

7. Supplementary

In this supplementary, we include more details on the following aspects:

- We report experimental results and conduct analysis of single C2ST performance in Section 7.1.
- We provide detailed experimental results of C2ST in Section 7.2.
- We present visualizations of performance sensitivity to depth in Section 7.3.
- We provide detailed results of different memory sizes in Section 7.4.
- We discuss the source-target classifier architecture in Section 7.5.

7.1. Experimental Results and Analysis of single C2ST

In Section 3.3, we critically discussed the limitation of the sequential covariate shift detection method [24]. Their methodology presents a fundamental trade-off dilemma: While a unified classifier architecture demonstrates computational efficiency, its binary discriminative framework intrinsically lacks the capacity for task-id recognition. Conversely, maintaining specific classifiers per task effectively preserves discriminability inter tasks at the cost of increased resource overhead with increasing tasks. In the main text, we conducted a theoretical analysis of the second method and presented experimental results. Now, we turn to the first method, namely single-C2ST, for discussion.

In the single-C2ST, a unified source-target classifier is maintained. For the classifier, all previously learned tasks are treated as ID, so the source samples will be drawn as evenly as possible from the respective memory buffers of all learned tasks. This architecture enables each sample to undergo only a single evaluation pass through the classifier, thereby achieving significant computational resource efficiency. However, this method exhibits two significant limitations. Firstly, as previously discussed, it inherently lacks the capability for task-id prediction, making it unsuitable for open-world TIL. Secondly, as incremental learning progresses, the number of ID tasks increases substantially and the number of source samples drawn from each task decreases, resulting in degraded classifier performance. The detailed OOD and TIL detection performance are respectively presented in Table 5 and Table 6. The task-id in the tables refers to the identity of the most recently seen task. TP, FP, TN, and FN represent the IDs correctly predicted, OODs incorrectly predicted as IDs, OODs correctly predicted, and IDs incorrectly predicted as OODs, respectively. As shown in Table 5, almost all OOD samples can be detected, but the majority of ID samples will also be incorrectly identified as OOD. When the number of learned tasks increases, the composition of the source sample becomes more complex, and the ID samples only belong to one of the tasks, so it becomes more difficult to correctly predict the ID test sample as ID. The OOD target sample does not match any of them, so the classifier can still accurately identify them as OOD, providing sufficient training data for TIL. As depicted in Fig. 5, H2ST and single-C2ST exhibit comparable performance in TIL. However, the hierarchical architecture demonstrates considerable improvement in OOD detection, with an average increase of 62.06% in F1.

Task-id	OOD Detection Result				F1 Score
	TP	FP	TN	FN	
1	1920	30	5970	80	97.22
2	170	0	6000	3830	8.15
3	150	0	6000	3850	7.23
4	110	0	6000	3890	5.35
Average					29.49

Table 5. Detailed OOD detection performance of single-C2ST on MNIST dataset with GEM.

Task-id	Task Incremental Learning Accuracy				
	Task1	Task2	Task3	Task4	Task5
1	99.49	58.27	54.70	52.89	51.28
2	99.35	99.08	61.68	70.71	32.14
3	98.43	98.49	98.44	54.65	43.65
4	98.75	97.95	97.44	99.36	46.62
5	98.84	97.46	96.84	98.24	99.49
Average	ACC:	98.17		FT:	-1.25

Table 6. Detailed TIL performance of single-C2ST on MNIST dataset with GEM.

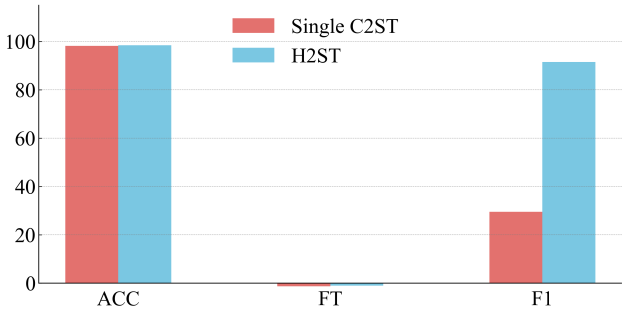


Figure 5. TIL and OOD detection performance of single C2ST and H2ST. H2ST demonstrates superior OOD detection performance.

7.2. Detailed Results of C2ST

We present a comparative analysis between H2ST and C2ST in Figure 2. Here we present comprehensive C2ST results, detailed in Table 8 and Table 9. While C2ST exhibits improved OOD detection performance compared to baseline methods, it still demonstrates a measurable performance gap relative to our proposed H2ST.

7.3. Visualization of Performance Sensitivity to Depth

While Fig. 3 visualizes the depth sensitivity analysis for CoRe50 with ER, we extend this analysis to other cases in Fig. 8. The results demonstrate that H2ST achieves superior overall performance and shows greater stability. Particularly in cases with a large number of classes per task, such as CIFAR-100 and Mini-ImageNet, the performance of C2ST deteriorates rapidly as the number of learned tasks increases. In contrast, H2ST consistently maintains stable performance throughout the incremental learning process.

7.4. Detailed Results of Different Memory Sizes

In Fig. 4, we present the trend of various metrics relative to memory size per task. We now provide detailed results of different memory sizes in Table 10 and Table 11. Memory size significantly impacts OOD detection, and we conduct an in-depth analysis. In H2ST, source samples are randomly drawn from memory buffers, with the fundamental assumption that these samples sufficiently represent the task distribution. An insufficient memory size leads to insufficient coverage of the task distribution, inadequate sample diversity, and increased distribution estimation bias. Conversely, simply increasing memory size is not an optimal solution, as the information gain from additional samples becomes negligible beyond a certain extent and larger memory directly leads to higher overhead. Therefore, finding an optimal memory size that balances representativeness and computational efficiency is crucial for effective OOD detection.

7.5. Different Source-Target Classifier Architectures

In Section 5.1, we employ a fully connected neural network with a single hidden layer of 128 ReLU units, denoted as MLP-I, as the source-target classifier. While this lightweight architecture demonstrates promising performance, we further explore alternative architectures. Specifically, we investigate deeper fully connected neural network with five hidden layers (MLP-II) and ten hidden layers (MLP-III), along with convolutional neural network with four convolutional layers (CNN-I), to provide comprehensive architectural comparisons. Fig. 6 illustrates the number of parameters of these models. We conduct experiments on CIFAR-10, with the F1 scores shown in Fig. 7 and the average metrics summarized in Table 7. The MLP-I with the fewest parameters achieves the best OOD detection effect, while the MLP-III with the most parameters performs the worst, misclassifying a large number of OOD samples as ID. This phenomenon is particularly relevant in continual learning characterized by non-stationary data streams, where source-target classifiers must be updated online to adapt to new distributions. While deeper models might theoretically offer greater representational capacity, their increased complexity hinders rapid adaptation to new distributions. In contrast, models with simpler architecture demonstrate superior adaptability, enabling faster adjustments in response to distributional shifts. Moreover, since each classifier only performs binary classification, simpler models are generally sufficient to meet the demands. Furthermore, their lower computational overhead makes them more practical for applications.

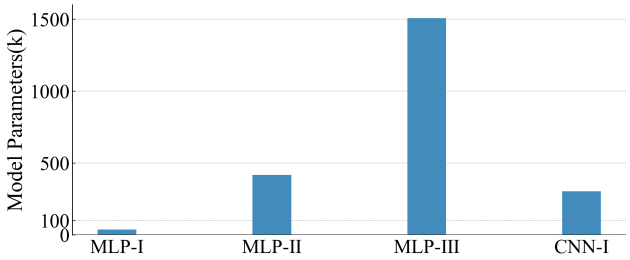


Figure 6. Parameters of different source-target classifier architectures.

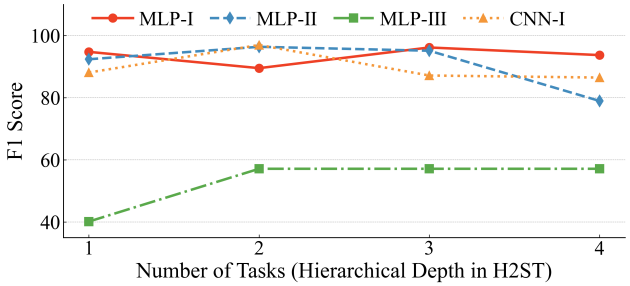


Figure 7. F1 score sensitivity to depth of different source-target classifier architectures.

Architecture	ACC↑	FT↑	F1↑	TA↑
MLP-I	**84.71**	**-7.90**	**93.54**	**92.82**
MLP-II	83.39	-6.83	90.72	78.67
MLP-III	51.14	-7.05	52.90	11.38
CNN-I	84.37	-6.83	89.68	79.16

Table 7. Performance comparison across different source-target classifier architectures.

TIL	Dataset															
	MNIST		SVHN		CIFAR-10		CIFAR-100		Mini-ImageNet		CoRe50		Stream-51		Average	
	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑
ER [45]	85.46	90.01	74.13	83.05	86.04	88.72	60.18	73.09	56.67	73.57	86.15	91.50	79.29	88.60	75.42	84.08
GEM [37]	86.34	90.31	74.92	83.65	87.88	89.35	64.65	74.32	57.84	73.54	90.77	93.64	79.72	87.89	77.45	84.67

Table 8. OOD detection performance of C2ST.

TIL	Dataset															
	MNIST		SVHN		CIFAR-10		CIFAR-100		Mini-ImageNet		CoRe50		Stream-51		Average	
	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑
ER [45]	96.14	-1.73	94.23	-3.16	83.72	-9.01	45.56	-14.06	31.02	-11.76	75.04	-4.70	67.56	-8.66	70.47	-7.58
GEM [37]	98.67	-0.56	92.64	-5.20	84.41	-7.80	44.97	-14.40	32.15	-10.86	77.03	-3.50	67.22	-11.68	71.01	-7.71

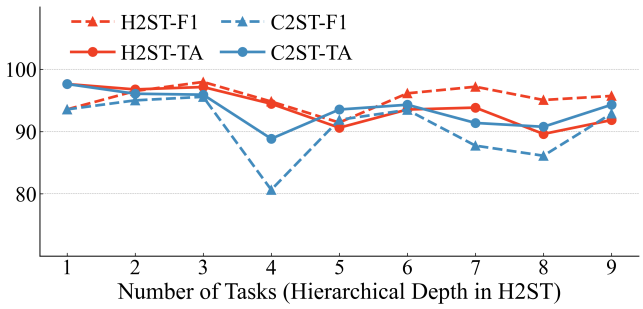
Table 9. TIL performance of C2ST.

TIL	Memory Size	Dataset															
		MNIST		SVHN		CIFAR-10		CIFAR-100		Mini-ImageNet		CoRe50		Stream-51		Average	
		F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑	F1↑	TA↑
ER [45]	40	23.82	68.16	63.65	78.27	52.89	73.78	31.31	66.69	31.60	67.23	44.68	76.09	39.11	73.77	41.01	72.00
	100	64.77	80.14	70.20	81.86	84.65	87.09	56.20	71.81	55.60	72.88	83.90	88.69	66.78	81.57	68.87	80.58
	200	92.03	93.78	77.60	84.60	88.89	89.59	84.21	82.02	79.34	81.59	94.11	94.06	89.24	90.68	86.49	88.05
	300	95.21	96.06	82.05	85.71	93.56	92.43	92.18	86.76	85.37	82.16	97.47	95.23	94.27	92.85	91.44	90.17
GEM [37]	40	23.53	68.03	69.45	79.92	57.41	75.13	34.03	66.42	30.89	66.78	52.18	77.67	41.05	74.39	44.08	72.62
	100	68.86	82.29	75.83	83.66	78.57	84.26	55.54	72.24	54.20	72.46	87.62	90.09	72.82	83.34	70.49	81.19
	200	91.55	93.43	77.87	85.20	93.54	92.82	89.98	84.37	83.88	82.69	95.38	93.94	91.63	91.23	89.12	89.10
	300	95.41	96.09	77.63	83.73	94.41	93.01	94.23	85.16	92.90	85.53	98.01	95.69	96.02	94.23	92.66	90.49

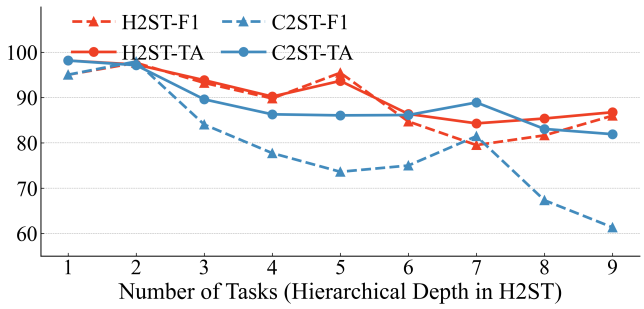
Table 10. Comparison of the OOD detection performance across different memory sizes.

TIL	Memory Size	Dataset															
		MNIST		SVHN		CIFAR-10		CIFAR-100		Mini-ImageNet		CoRe50		Stream-51		Average	
		ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑	ACC↑	FT↑
ER [45]	40	97.34	-2.21	91.95	-6.34	79.50	-14.15	41.12	-19.14	28.06	-16.79	72.56	-10.74	64.57	-11.23	67.87	-11.51
	100	97.80	-1.45	92.62	-5.67	79.30	-13.80	43.68	-15.70	29.99	-15.09	77.13	-4.31	74.37	-8.19	70.70	-9.17
	200	98.49	-0.80	93.45	-4.24	84.34	-8.98	45.09	-14.23	31.91	-10.56	78.26	-1.42	74.33	-5.50	72.27	-6.53
	300	98.71	-0.32	94.94	-2.59	85.76	-6.76	45.70	-12.45	33.27	-9.81	79.07	-2.93	75.22	1.98	73.24	-4.70
GEM [37]	40	97.37	-2.30	93.32	-4.76	74.78	-20.65	42.39	-17.74	29.56	-15.24	69.12	-11.56	66.31	-10.83	67.55	-11.87
	100	97.90	-1.51	94.04	-3.27	78.72	-15.54	43.17	-16.65	29.86	-15.15	77.21	-4.20	74.03	-8.87	70.70	-9.31
	200	98.43	-1.01	92.64	-5.20	84.71	-7.90	45.91	-13.90	30.80	-12.24	78.31	-1.64	69.36	-13.13	71.45	-7.86
	300	98.50	0.00	95.07	-2.48	85.37	-7.53	46.28	-12.25	33.66	-10.56	80.32	2.24	73.87	-4.90	73.29	-5.07

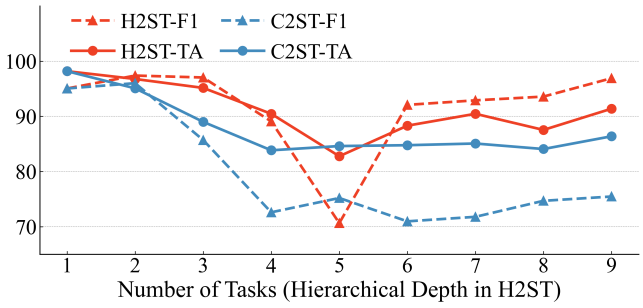
Table 11. Comparison of the TIL performance across different memory sizes.



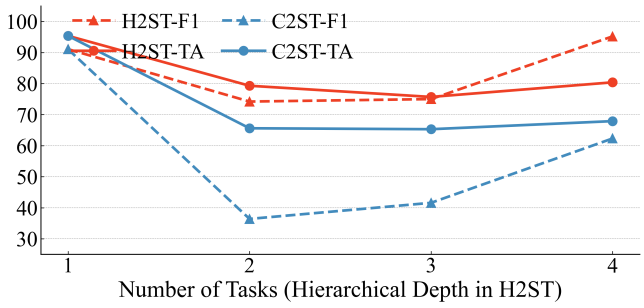
(a) CoRe50 dataset with GEM.



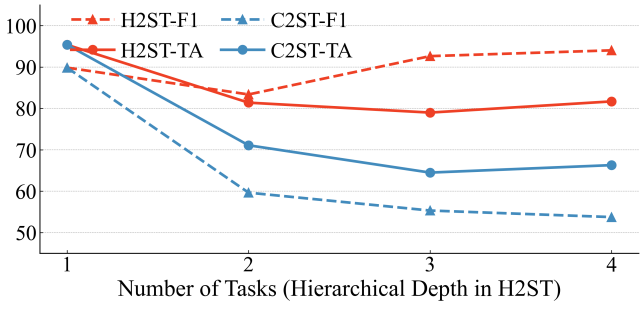
(b) Stream-51 dataset with ER.



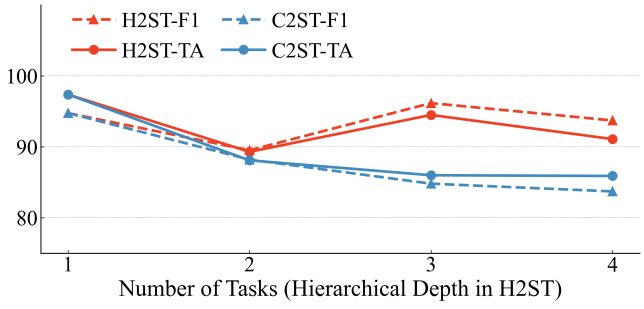
(c) Stream-51 dataset with GEM.



(d) Mini-ImageNet dataset with GEM.



(e) CIFAR-100 dataset with GEM.



(f) CIFAR-10 dataset with GEM.

Figure 8. Performance sensitivity to depth of (a) CoRe50 dataset with GEM, (b)Stream-51 dataset with ER, (c)Stream-51 dataset with GEM, (d)Mini-ImageNet dataset with GEM, (e)CIFAR-100 dataset with GEM and (f) CIFAR-10 dataset with GEM.