TADIL: Task-Agnostic Domain-Incremental Learning through Task-ID Inference using Transformer Nearest-Centroid Embeddings

Gusseppe Bravo-Rocca
Barcelona Supercomputing Center
Barcelona, Spain
gusseppe.bravo@bsc.es

Peini Liu
Barcelona Supercomputing Center
Barcelona, Spain
peini.liu@bsc.es

Jordi Guitart
Barcelona Supercomputing Center
Universitat Politècnica de Catalunya
Barcelona, Spain
jordi.guitart@bsc.es

Ajay Dholakia
Lenovo Infrastructure Solutions Group
Morrisville, NC, USA
adholakia@lenovo.com

David Ellison
Lenovo Infrastructure Solutions Group
Morrisville, NC, USA
dellison@lenovo.com

Abstract

Classical Machine Learning (ML) models struggle with data that changes over time or across domains due to various factors such as noise, occlusion, illumination or frequency, unlike humans who can learn from such non independent and identically distributed data. Consequently, a Continual Learning (CL) approach is indispensable, particularly, Domain-Incremental Learning, as classical static ML approaches are inadequate to deal with data that comes from different distributions. In this paper, we propose a novel pipeline for identifying tasks in domain-incremental learning scenarios without supervision. The incremental pipeline comprises four primary steps. First, we obtain a base embedding from the raw data using a transformer-based model. Second, we group the embedding densities based on their similarity to obtain the nearest points to each cluster centroid. Third, we train an incremental task classifier using only these points. Finally, thanks to the lightweight computational requirements of the pipeline, we use it to devise an algorithm that can decide in an online fashion when to learn a new task using the task classifier and a drift detector. We evaluate our approach by conducting experiments using the SODA10M real-world driving dataset and several CL strategies. We demonstrate that the performance of these CL strategies when using our pipeline can match the ground-truth approach, both in experiments assuming task boundaries using a traditional approach, and also in more realistic task-agnostic scenarios that require detecting new tasks on-the-fly.

1. Introduction

Machine Learning (ML) has advanced significantly, but real-life applications often present non-IID data, leading to domain shift problems. Continual Learning (CL) addresses these limitations by enabling models to learn continuously after deployment. However, existing methods, such as regularization [16], replay [7], and architecture modifications [1,11], assume rigid task boundaries and known tasks, which may not hold in many real-world scenarios.

We propose a novel task-agnostic approach for domain-incremental learning that can detect task drift and classify tasks without supervision. Our approach segments the data stream into meaningful domains and classes, and applies appropriate CL strategies to each domain. To the best of our knowledge, no other works have proposed unsupervised approaches for identifying and classifying tasks in task-agnostic domain-incremental learning scenarios for driving datasets.

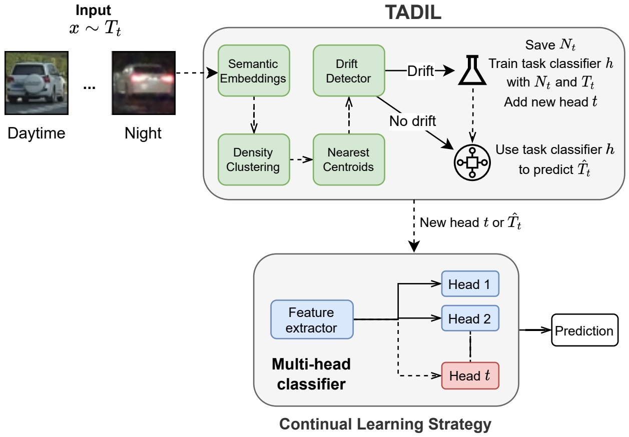

Figure 1. Our method processes a batch of images x associated with a task T_t, calculates the nearest-centroid embeddings N_t, and checks for drift. If drift is found, N_t is stored in memory $\mathcal{M}$ and a task classifier h is incrementally trained with N_t and T_t unsupervised. If no drift is detected, classifier h estimates the task $\hat{T}_t$. The multi-head classifier then chooses the correct classifier based on $\hat{T}_t$ to generate the final prediction y

2. Problem definition

Let X be the input space, Y be the output space, and T be the space of task IDs. We consider a sequence of K tasks, where each task k is associated with a joint distribution $P_k(X, Y)$ over $X \times Y$. The goal of domain-incremental learning is to learn a sequence of K classifiers $f_1, f_2, ..., f_K$, where $f_k : X \rightarrow Y$ is the classifier for task k (they are not necessarily different from each other), such that each classifier can be learned incrementally from data without forgetting the previous tasks, i.e., when learning f_k, the classifiers $f_1, f_2, ..., f_{k-1}$ should be preserved.

During inference, the task ID $t \in T$ will be unknown. The classifier f_t for each task t will be used to predict the output $y \in Y$, that is, $f_t = p_t(y|x)$. More formally, we can define domain-incremental learning as:

$$\arg \min_{f_1, f_2, \ldots, f_K} \sum_{k=1}^{K} \mathcal{L}(f_k, P_k) \tag{1}$$

where the goal is to minimize the loss function $\mathcal{L}(f_k, P_k)$ over a set of K functions $f_1, f_2, ..., f_K$, subject to the constraint that each of the K functions can be learned incrementally. As for our experiments, we consider the set of functions $f_1, f_2, \ldots, f_K$ as multi-head classifiers in the following way:

$$f_k(\mathbf{x}) = g_k(e(\mathbf{x})), \quad k = 1, 2, \ldots, K \tag{2}$$

where $e(\mathbf{x})$ denotes the shared feature extractor network, $g_k(\cdot)$ denotes the classifier for the k-th task, and $\mathbf{x}$ denotes the input sample. Particularly, we use the ResNet18 [4] architecture as the feature extractor network and linear classifiers on top of it.

To improve the performance of the multi-head classifier at inference time and enable strategies that require the task ID, such as EWC, ER, LwF, among others, it is necessary to have a task classifier that can learn the task ID with no supervision. This task classifier should take the input sample $\mathbf{x}$ and predict the corresponding task ID $t \in T$, which can then be used to select the appropriate classifier f_t for inference.

Let g_t be the classifier for task t learned from data. We define a task classifier $h_t : X \rightarrow T$ that takes an input sample $\mathbf{x} \in X$ and predicts the corresponding task ID $t \in T$. This task classifier can be learned without supervision, as it simply needs to predict the correct task ID associated with each input sample.

During inference, given an input sample $\mathbf{x} \in X$ and the predicted task ID $\hat{t} = h(\mathbf{x})$, the multi-head classifier $f_{\hat{t}}$ is used to predict the output $y \in Y$. Thus, the final prediction can be written as:

$$y = f_{\hat{t}}(\mathbf{x}) = g_{\hat{t}}(e(\mathbf{x})) \tag{3}$$

Therefore, incorporating a task classifier into the domain-incremental learning framework can be expressed as:

$$\arg \min_{h, g_1, g_2, \ldots, g_K} \sum_{k=1}^{K} \mathcal{L}(f_k, P_k) \tag{4}$$

where h is the task classifier, g_k is the classifier for task k, and $\mathcal{L}(f_k, P_k)$ is the loss function for task k. This objective function ensures that each of the K classifiers can be learned incrementally without forgetting the knowledge they previously learned, while also incorporating the task classifier into the learning process.

3. Related work

This section introduces related works in the field of Continual Learning relevant to our research.

3.1. Multimodal transformers

Multimodal transformers, such as CLIP [10], are important in CL for generating rich embeddings. TransFuser [9] and Huang et al.'s neural prediction framework [5] are examples of transformer-based models for autonomous driving.

3.2. Domain-incremental learning

Domain-incremental learning (Domain-IL) focuses on learning multiple tasks sequentially. DISC [8] is an online zero-forgetting approach that requires task ID, while an autoencoder-based method [2] uses reconstruction error to identify domains. Domain-aware categorical representations [15] address stability-plasticity and imbalance challenges.

3.3. Task-Agnostic Continual Learning (TACL)

TACL aims to learn from non-stationary distributions without known task identity [17]. Generative replay [12], Learning without Forgetting (LwF) [6] and Rebuffi's work [11] are some examples. However, they lack model-agnosticism and unbiased (they use the same model for identification and training) task identification.

In this paper, we propose a more robust approach using a lightweight, independent model for unbiased task identification and adaptation while retaining performance on previous tasks.

4. Components of the pipeline for task-agnostic domain-incremental learning

We present a pipeline for task-agnostic domain-incremental learning, which includes the Nearest Centroid Algorithm for training an incremental task classifier and a drift detector to identify when to incrementally train the task classifier.

The task classifier is obtained through the following pipeline: **Semantic embedding**. Given a batch of inputs $X = x_1, x_2, ..., x_m$, we obtain their embeddings $E = e_1, e_2, ..., e_m$ using the pretrained transformer-based model CLIP ViT-B/32 [10], represented as $E = f_{emb}(X)$.

Density-based clustering. We cluster the embeddings E using the DBSCAN density clustering algorithm. Let the clustering labels be $C = c_1, c_2, ..., c_m$, and the clustering function be $f_{clust}(E; \epsilon, minPts)$.

Nearest-cluster centroids. We obtain the nearest centroids $M = m_1, m_2, ..., m_j$ of the j distinct clusters present in C. Each centroid m_i is calculated using the Nearest Centroids Algorithm [13], represented as $M = f_{cent}(E, C)$.

Nearest-centroid Incremental classifier. The task classifier h_t for task T_t is obtained by combining individual classifiers using a majority vote, represented as $h_t = f_{cls}(M_{t^d}, t^d)$, where t^d is the new task ID detected by the drift detector R.

Drift detector. We define a drift function R that measures the dissimilarity between the nearest neighbors at different time points:

$$R(N_t, N_{t'}) = \frac{1}{k} \sum_{s=1}^{k} d(N_{t[s]}, N_{t'[s]}) \tag{5}$$

A larger value for the drift function suggests a possible shift in the data distribution and a new task.

5. Online algorithm for task-agnostic domain-incremental learning

We present an online pipeline algorithm for task-agnostic domain-incremental learning using the components discussed in the previous section (refer to Algorithm 1 for details).

For each batch of images, our algorithm calculates the nearest-centroid embeddings N_t and checks for drift with the known tasks stored in memory $\mathcal{M}$. Drift is evaluated using the drift detector. If the batch drifts from all tasks (i.e., it is a new task), we save N_t in memory, incrementally train the task classifier h_t with N_t and the new task label T_t, and add a new head to the multi-head classifier for this new task.

If a non-drifting task is found in memory, the classifier h_t estimates the task ID, which is used to select the appropriate classifier in the multi-head classifier for inference until a domain change occurs.

Algorithm 1: Online Task-Agnostic algorithm for Domain-Incremental Learning that yields decisions using the drift detector R and task classifier h

```
Data: Memory $\mathcal{M}$, dataset $D_t$
Function online_TADIL($\mathcal{M}$, $D_t$):
    $N_t \leftarrow$ get_nearest_centroid_embeddings($D_t$)
    for $N_{t'} \in \mathcal{M}$.reversed() do
        if (not R($N_t$, $N_{t'}$)) then
            use the task classifier $h_{t'}$ to predict the
              task ID $T_t$;
            if ($T_t \neq T_{t'}$) then
                raise warning
            use head $g_{t'}(D_{t'})$ from the multi-head
              classifier for inference;
            return;
    save $N_t$ into memory $\mathcal{M}$;
    train incrementally the task classifier $h_t$ using
      $N_t$ and a new task label $T_t$;
    add a new head $g_t(D_t)$ to the multi-head
      classifier;
    use head $g_t(D_t)$ for inference;
    return;
```

6. Experimental evaluation

6.1. Testbed

The testbed used in the experiments is as follows. Platform: Ubuntu 22.04 (64 bits). Hardware: 2x Intel(R) Xeon(R) Platinum 8360Y CPU @ 2.40GHz, 256 GB RAM. Datasets: SODA10M, it contains 10M unlabeled images and 20k labeled images [3]. We use the objects of the labeled images (20,000 1920×1080 color images of 6 differ-

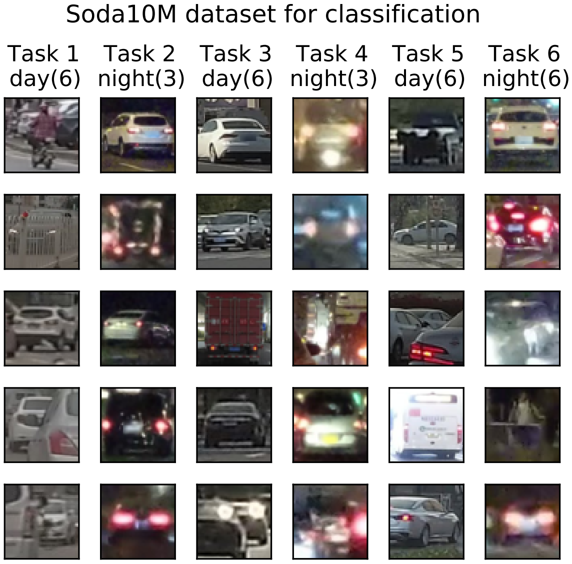

Figure 2. Soda10M for the CLAD-C benchmark.

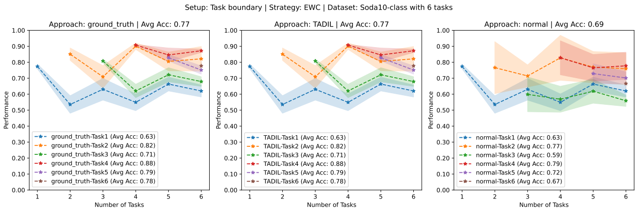

Figure 3. Comparison using the EWC strategy. This figure shows how a regularization-based algorithm behaves with task boundaries.

ent objects) to evaluate our experiments. We created a modified version of the dataset used in the CLAD-C challenge for online classification [14] (see Figure 2). We split up the 6 tasks into training (80%) and testing data (20%) so that we obtain a domain incremental setup for classification. Besides, we tested with different metrics which are aligned to our experiments.

6.2. Performance of the CL multi-head models

In this subsection, we compare the performance of CL strategies with different task ID approaches in a classical CL setup with task boundaries. We implement a multi-head model using the Adam optimizer, learning rate of 0.01, and cross-entropy loss. The evaluated CL strategies include EWC, Experience Replay, and LwF. All strategies use 4 epochs, a batch size of 200, and the same optimizer and criterion.

The model is provided with a task ID for each task, allowing it to switch between different output heads or parameters. We compare three approaches for task ID: ground-truth, our approach (TADIL), and normal. Figure 3 present the accuracy of EWC strategy. Our proposed method performs on par with the ground-truth approach.

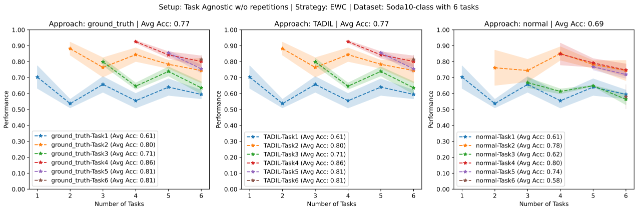

Figure 4. EWC strategy. This figure shows how this strategy behaves with no task boundaries.

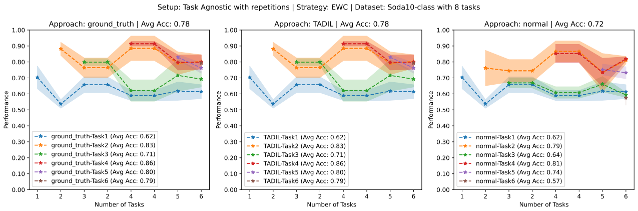

Figure 5. EWC strategy. This figure shows how this strategy behaves with no boundaries and repeated tasks.

Next, we evaluate the three CL strategies in task-agnostic scenarios, using our drift detector component to detect new tasks. Figures 4 presents the accuracy of EWC strategy in this scenario. TADIL outperforms the task ID-oblivious method and matches the ground-truth approach.

Lastly, we evaluate the CL strategies in a task-agnostic scenario with task repetitions. Figure 5 present the accuracy of EWC strategy. Our proposed method continues to perform on par with the ground-truth approach, showcasing its adaptability and effectiveness in addressing real-world challenges.

7. Conclusions

In this paper, we introduced TADIL, a novel pipeline for task-agnostic domain-incremental learning without supervision. Utilizing a transformer-based model, TADIL obtains base embeddings from raw data and groups them by similarity. A task classifier is then incrementally trained using these points, and in conjunction with a drift detector, facilitates learning new tasks.

The performance of TADIL was showcased through experiments on the SODA10M dataset, where it was demonstrated that our pipeline could match the ground truth performance in both traditional and realistic task-agnostic scenarios.

Future work aims to enhance models for task-agnostic continual learning scenarios, specifically developing an Experience Replay strategy using nearest-centroid embeddings, and exploring other continual learning scenarios and modalities.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory Aware Synapses: Learning What (not) to Forget. In *Proc. European Conf. on Computer Vision, ECCV 2018*, pages 144–161. Springer International Publishing, Sep. 8–14 2018. 1

[2] Camila González, Georgios Sakas, and Anirban Mukhopadhyay. What is wrong with continual learning in medical image segmentation? *CoRR*, abs/2010.11008, 2020. 3

[3] Jianhua Han, Xiwen Liang, Hang Xu, Kai Chen, Lanqing Hong, Jiageng Mao, Chaoqiang Ye, Wei Zhang, Zhenguo Li, Xiaodan Liang, and Chunjing Xu. SODA10M: A Large-Scale 2D Self/Semi-Supervised Object Detection Dataset for Autonomous Driving, 2021. arXiv preprint arXiv:2106.11118. 3

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proc. 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'16)*, pages 770–778, Jun. 27–30 2016. 2

[5] Zhiyu Huang, Xiaoyu Mo, and Chen Lv. Multi-modal Motion Prediction with Transformer-based Neural Network for Autonomous Driving. In *Proc. 39th Int. Conf. on Robotics and Automation (ICRA)*, pages 2605–2611, May 23–27 2022. 2

[6] Zhizhong Li and Derek Hoiem. Learning without Forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(12):2935–2947, 2018. 3

[7] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient Episodic Memory for Continual Learning. In *Advances in Neural Information Processing Systems, vol. 30 (NIPS 2017)*, pages 6470–6479. Curran Associates Inc., 2017. 1

[8] Muhammad Jehanzeb Mirza, Marc Masana, Horst Possegger, and Horst Bischof. An Efficient Domain-Incremental Learning Approach to Drive in All Weather Conditions. In *Proc. 2022 IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW'22)*, pages 3000–3010, Jun. 19–20 2022. 2

[9] A. Prakash, K. Chitta, and A. Geiger. Multi-Modal Fusion Transformer for End-to-End Autonomous Driving. In *Proc. 2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR'21)*, pages 7073–7083. IEEE Computer Society, Jun. 19–25 2021. 2

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proc. 38th Int. Conf. on Machine Learning (ICML'21)*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, Jul. 18–24 2021. 2, 3

[11] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental Classifier and Representation Learning. In *Proc. 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'17)*, pages 5533–5542, Jul. 21–26 2017. 1, 3

[12] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual Learning with Deep Generative Replay. In *Advances in Neural Information Processing Systems, vol. 30 (NIPS 2017)*, pages 2994–3003. Curran Associates Inc., 2017. 3

[13] Robert Tibshirani, Trevor Hastie, Balasubramanian Narasimhan, and Gilbert Chu. Diagnosis of Multiple Cancer Types by Shrunken Centroids of Gene Expression. *Proceedings of the National Academy of Sciences of the United States of America*, 99(10):6567–6572, 2002. 3

[14] Eli Verwimp, Kuo Yang, Sarah Parisot, Lanqing Hong, Steven McDonagh, Eduardo Pérez-Pellitero, Matthias De Lange, and Tinne Tuytelaars. CLAD: A Realistic Continual Learning Benchmark for Autonomous Driving. *Neural Networks*, 161:659–669, 2023. 4

[15] J. Xie, S. Yan, and X. He. General Incremental Learning with Domain-aware Categorical Representations. In *Proc. 2022 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR'22)*, pages 14331–14340. IEEE Computer Society, Jun. 21–24 2022. 3

[16] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual Learning through Synaptic Intelligence. In *Proc. 34th Int. Conf. on Machine Learning (ICML'17)*, volume 70 of *Proceedings of Machine Learning Research*, pages 3987–3995. PMLR, Aug. 6–11 2017. 1

[17] Haoran Zhu, Maryam Majzoubi, Arihant Jain, and Anna Choromanska. TAME: Task Agnostic Continual Learning using Multiple Experts, 2022. arXiv preprint arXiv:2210.03869. 3