Bridging Streaming Continual Learning via In-Context Large Tabular Models

Afonso Lourenço[1,*], João Gama[2], Eric P. Xing[3,4] and Goreti Marreiros[1]

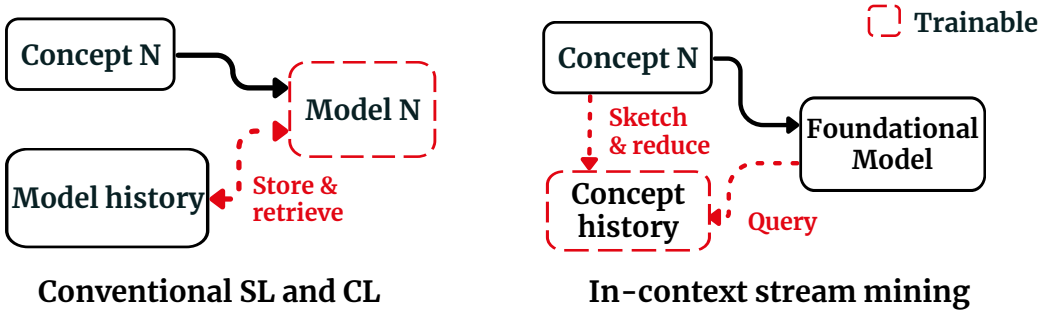
[1]GECAD, ISEP, Polytechnic of Porto, Rua Dr. António Bernardino de Almeida, Porto, 4249-015, Portugal
[2]INESC-TEC, FEP, University of Porto, Rua Dr. Roberto Frias, Porto, 4200-465, Portugal
[3]Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE
[4]Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

In streaming scenarios, models must learn continuously, adapting to concept drifts without erasing previously acquired knowledge. However, existing research communities address these challenges in isolation. Continual Learning (CL) focuses on long-term retention and mitigating catastrophic forgetting, often without strict real-time constraints. Stream Learning (SL) emphasizes rapid, efficient adaptation to high-frequency data streams, but typically neglects forgetting. Recent efforts have tried to combine these paradigms, yet no clear algorithmic overlap exists. We argue that large in-context tabular models (LTMs) provide a natural bridge for Streaming Continual Learning (SCL). In our view, unbounded streams should be summarized on-the-fly into compact sketches that can be consumed by LTMs. This recovers the classical SL motivation of compressing massive streams with fixed-size guarantees, while simultaneously aligning with the experience-replay desiderata of CL. To clarify this bridge, we show how the SL and CL communities implicitly adopt a divide-to-conquer strategy to manage the tension between plasticity (performing well on the current distribution) and stability (retaining past knowledge), while also imposing a minimal complexity constraint that motivates diversification (avoiding redundancy in what is stored) and retrieval (re-prioritizing past information when needed). Within this perspective, we propose structuring SCL with LTMs around two core principles of data selection for in-context learning: (1) distribution matching, which balances plasticity and stability, and (2) distribution compression, which controls memory size through diversification and retrieval mechanisms.

Keywords

Concept drift, data stream mining, foundational model

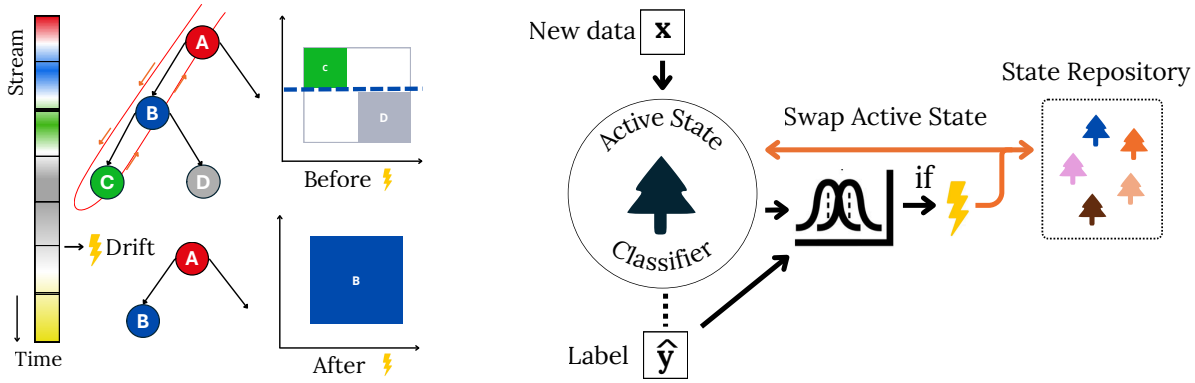


1. Introduction

For tabular stream learning (SL), ensembles of incremental decision trees (IDTs) have long been state-of-the-art [1]. They use statistical bounds to decide node splits and handle concept drift via subtree replacement (Fig. 1a). As shallow learners, IDTs converge quickly online due to their few trainable parameters. Yet, their learning capacity is limited by single-view features, plasticity loss from making locally optimal splits, catastrophic forgetting of class-conditional estimators, and the inability to model dependencies. Although various ad-hoc solutions have been proposed, mostly adding new candidate

1st Streaming Continual Learning Bridge at AAAI26, January 21, 2026, Singapore.
*Corresponding author.
fonso@isep.ipp.pt (A. Lourenço); jgama@fep.up.pt (J. Gama); epxing@cs.cmu.edu (E. P. Xing); mgt@isep.ipp.pt (G. Marreiros)
0000-0002-3465-3419 (A. Lourenço); 0000-0003-3357-1195 (J. Gama); 0000-0002-3683-4280 (E. P. Xing); 0000-0003-4417-8401 (G. Marreiros)

components for ensembling, they tend to be narrow, addressing one problem while assuming others are controlled. For example, ensembles may use drift detectors to swap to a more suitable model (Fig. 1b), but fail to evaluate stored models for relevance if the drift does not trigger an alarm [2].



(a) IDTs.

(b) IDTs swapped upon drift alarm.

Figure 1: SL relies on structural expansion, mostly adding new intelligence components for ensembling of IDTs.

Indeed, these ad-hoc intelligence components are only useful if we integrate them into a truly autonomous system. In this regard, continual learning (CL) allows more powerful deep learning (DL) schemas, adapting both through parameter addition and activation (Fig. 2). However, these struggle with tabular streams [3]. On one hand, the inductive biases of DL architectures assume structures which offer little advantage for irregular patterns typical of tabular data [4]. On the other hand, entangled architectures converge slowly due to stochastic updates, and unfixed weights make prior knowledge prone to being overwritten. Moreover, this plasticity does not guarantee learning efficiency, often requiring multiple data passes to reduce interference [5].

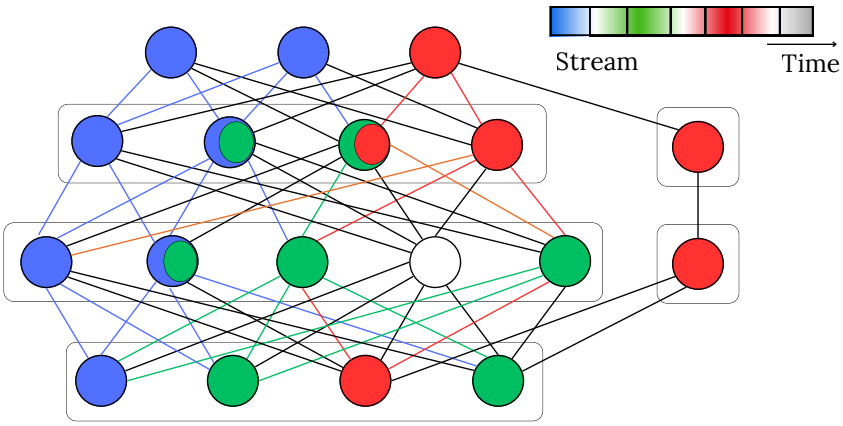


Figure 2: CL not only relies on both structural expansion, but also sophisticated parameter adaptation/activation DL schemas.

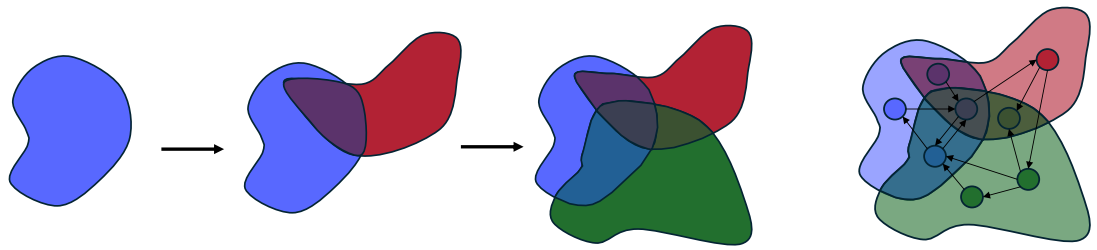
Due to these algorithmic differences, SL and CL prioritize different aspects of stateful learning [6]. CL focuses on long-term retention and mitigating forgetting, often without strict real-time constraints [7], whereas SL emphasizes rapid adaptation to high-frequency streams but typically ignores high-order dependencies and forgetting [8]. Recent efforts have tried to combine these paradigms [9, 10], yet no clear overlap exists. To address this, we propose to leverage the disruptive success of in-context large tabular models (LTMs) [11] as the unifying bridge for streaming continual learning (SCL): using on-the-fly techniques to summarize unbounded data streams before feeding them to LTMs.

In prior work [12], we showed that augmenting the TabPFN transformer [11] with a simple inference-time sketching mechanism consistently outperforms state-of-the-art methods such as Adaptive Random Forest [13], and Streaming Random Patches [14], on standard streaming benchmarks: NOAA, SmartMeter, Electricity, Rialto, Posture, CoverType, and PokerHand. Here, we explain how this data-centric view unifies the strengths of SL and CL: recovering the classical SL goal of compressing massive streams into compact sketches whose size and computational cost remain bounded [15], while simultaneously aligning with CL's experience-replay desiderata for retaining past concepts [16]. Our argument unfolds in two steps:

- **Extracting the shared desiderata of SL and CL.** Both communities follow a divide-to-conquer strategy driven by the underlying tension between **plasticity** (adapting to the current distribution) and **stability** (retaining past knowledge). Under a minimal memory constraint, this tension gives rise to two operational requirements: **diversification**, to avoid redundant or overlapping stored information, and **retrieval**, to re-activate relevant past experience when needed.
- **Mapping these desiderata to in-context stream mining with LTMs.** We show that SCL can be framed as selecting and organizing data for in-context learning, where **distribution matching** governs the plasticity–stability balance, and **distribution compression** governs memory efficiency through diversification and retrieval.

2. Current SL and CL state-of-the-art

Both SL and CL ultimately seek to maintain useful knowledge over time while adapting to new data. However, when learning occurs under streaming, non-stationary conditions, this requires a divide-to-conquer strategy. As new concepts accumulate, the feasible parameter space becomes progressively restricted, making it increasingly difficult to adjust the model without interfering with previously acquired knowledge (Fig. 3a). In the limit, finding parameter configurations that jointly satisfy all concepts is NP-hard [17]. Thus, the challenge is not merely retaining information, but doing so while preserving efficient access and reuse. A central lever in controlling interference is the degree of parameter sharing. Full sharing maximizes generalization but risks interference; no sharing avoids interference but scales poorly with the number of concepts [18]. A middle ground is modular compositionality, where knowledge is distributed across components that can be selectively re-used, enabling forward and backward transfer (Fig. 3b).



(a) Incrementally constrained solution space. (b) Modularity.

Figure 3: On the need for divide-to-conquer in SL and CL.

For tabular data streams, ensembles of IDTs are widely used and highly effective [1]. A single IDT can adapt online and converge to near-optimal splits using one-pass summary statistics and statistically grounded decision criteria, such as information gain tested with Hoeffding-style bounds [19]. However, predictive performance emerges primarily at the ensemble level. When concept drift occurs, trees respond differently depending on their local minima and data histories. This natural diversity can be leveraged: outdated or underperforming trees can be replaced, new trees introduced, and ensemble votes re-weighted based on recent performance. Research in this area has therefore progressed along two complementary axes. The first concerns the base learners themselves, balancing their ability to

adapt to new data (**plasticity**) while retaining useful structure from the past (**stability**). The second focuses on ensemble management, ensuring that the collection of learners remains diverse enough to cover different regions of the input space (**diversification**) and that previously useful models can be reactivated when similar conditions reappear (**retrieval**).

2.1. Stability

Despite their ability to continually store incrementally arriving data, IDTs are often biased toward recently observed distributions when new classes appear. Under strong temporal imbalance, where older classes do not reappear, performance on previously learned concepts deteriorates, resulting in catastrophic forgetting. This problem is amplified after splits, where conditional estimators are reset: classes absent from the current stream lose representation, are excluded from entropy-based split decisions, cannot be used in Naive Bayes classification, and have their priors removed at deeper nodes. Forgetting in IDTs occurs through three mechanisms: (1) exclusion of older classes from split evaluation, (2) failure of conditional classification due to missing estimates, and (3) disappearance of class priors in new branches (Fig. 4) [20]. Stability-oriented approaches mitigate this by preserving class information during updates. One strategy propagates class-conditional attribute estimators and maintains class priors in entropy and Bayesian computations [20]. Another relies on short-term memory replay to preserve representation continuity, e.g., through oversampling [21], or per-class balanced queues [22].

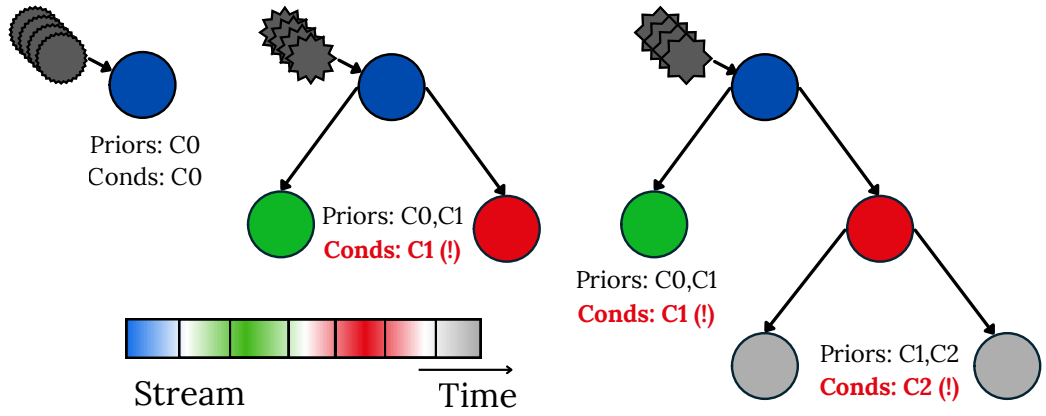


Figure 4: IDTs' conditional classification fails in class-incremental sequences, due to missing estimates.

Neural networks allow more expressive stability mechanisms. A common strategy is to approximate joint optimization over past tasks by penalizing updates to parameters deemed important for previous ones (Fig. 5). EWC [23] measures importance via the Fisher Information Matrix, MAS [24] via gradient or Hebbian activity with constant memory, and SI [25] accumulates importance online from loss reductions. RWalk [26] unifies these by computing importance in the Fisher-induced Riemannian parameter space. Since all of these operate incrementally in weight space, they extend naturally to streaming settings [27]. However, such methods typically exhibit temporary forgetting because SGD must pass through regions of high loss on old tasks to reach the regularized joint optimum [28]. This stability gap motivates modifying not only the objective, but also the optimization path [29]. To address this, gradient projection methods enforce updates orthogonal to gradient subspaces of previous tasks [30, 31]. However, while this reduces forgetting, strict orthogonality can overly limit knowledge transfer. Recent work relaxes these constraints, enabling controlled sharing, e.g., NCL [32] re-scales gradients using a Kronecker-factored Fisher approximation and combines projection with parameter regularization.

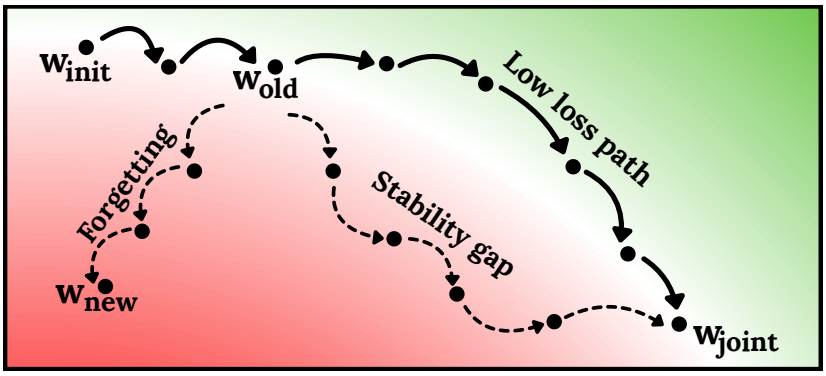
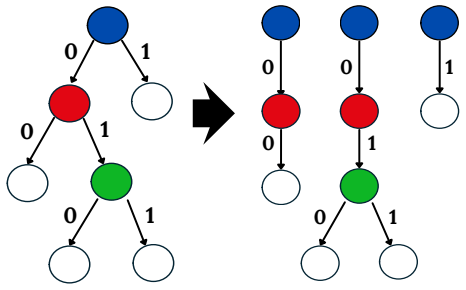


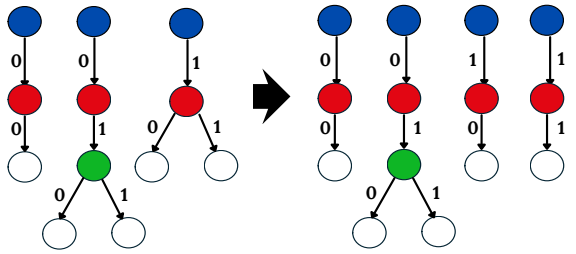
Figure 5: When optimizing the joint loss, SGD typically passes through a region of high loss on old tasks.

2.2. Plasticity

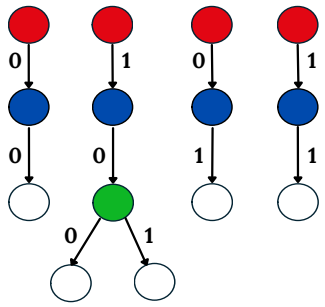
In streaming settings, models may lose not only past knowledge but also the capacity to learn new concepts. This occurs when parameters drift into regions of the loss landscape where optimization becomes slow, leading to reduced plasticity [33]. Under this view, while plasticity is often framed as freeing capacity in IDTs, e.g., via change detectors and post-pruning [34, 35], a more precise view lies in how well the current parameters serve as a starting point for further learning, independent of how much knowledge they store. In this regard, IDTs inherently exhibit low plasticity due to their conservative, history-dependent structure. Because they grow by making locally optimal splits (e.g., via thresholds, grace periods, tie-breaking rules), their ability to revise earlier decisions is limited. However, recent work has challenged this rigidity. PLASTIC [36] introduces a restructuring mechanism that allows an IDT to revisit and modify pruned subtrees (Fig. 6), exploiting the fact that a tree's structure can change without altering its predictive semantics. Similarly, DFDT [37] proposes reordering and pruning operations to promote informative attributes toward the root, enabling adaptation in trapezoidal data streams. DCFHT [38] extends this to capricious data streams.



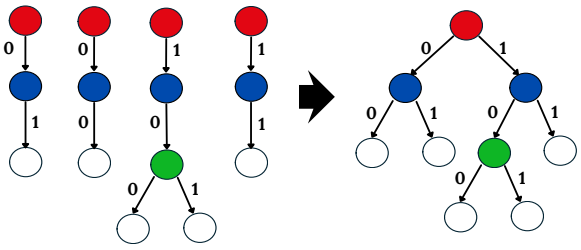
(a) Disconnect subtree.



(b) Desired splits at branches.



(c) Move splits to the root.



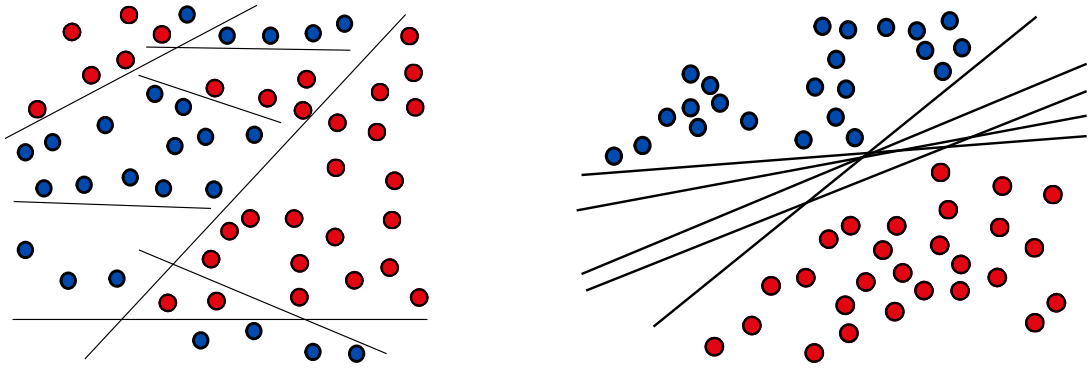
(d) Iteratively rebuild the subtree.

Figure 6: Restructuring IDTs by their intrinsic non-overlapping rule decomposition covering the full space.

Conversely, in neural networks, reduced plasticity often manifests as growing weight magnitudes and saturated units. Regularization-based stability methods such as EWC, MAS, and SI encourage parameters toward zero, but can unintentionally collapse weight matrix ranks and hinder adaptation. To retain plasticity, one can instead regularize toward initialization (preserving "how to learn") [39] or toward curvature-preserving parameter distributions, for example via Wasserstein-based order-statistic regularization [40]. Another direction focuses on reintroducing flexibility: S&P [41] combines weight decay with stochastic perturbations to restore movement in parameter space, though at the cost of increased forgetting. ReDo [42] improves stability by resetting only saturated units, yet still struggles with signal propagation issues [5]. Continual Backprop [43] makes this selective reset more principled by tracking utility scores over incoming and outgoing weights and protecting newly reset units until they mature. UPDG generalizes this by coupling gradient updates with adaptive perturbations, applying minimal changes to useful units and stronger rejuvenation to dormant ones [27].

2.3. Diversification

Under tight memory constraints, a divide-and-conquer strategy naturally calls for diversification, ensuring that models store complementary rather than redundant information (Fig. 7a). Even in cases where the data is relatively simple, one can explore different perspectives of the same patterns within a given computational budget (Fig. 7b). This can be imposed through hard boundaries, such as explicit output specialization [44]. More commonly, however, streaming ensemble methods rely on softer mechanisms that perturb the input or feature space: horizontally, e.g. through Poisson-based instance weighting [45], or selective instance filtering [46]; and vertically, e.g. through random subspace selection [14]. Beyond implicit diversification, several approaches explicitly manage the ensemble repository using diversity metrics such as double-fault [47], or the kappa statistic [48]. At a higher level, heterogeneous ensembles can be maintained via local search heuristics [49], evolutionary strategies [50], and meta-learning methods [51].



(a) In a complex space.

(b) In a simpler space.

Figure 7: Multiple non-redundant perspectives of the data.

In contrast, neural networks typically achieve diversification within a single model by controlling how representations are shared or separated. One approach is to learn domain-invariant features by sharing parameters across layers or models [52]. Another method allocates distinct feature-processing pathways while minimizing discrepancies between them, avoiding explicit mapping across domains [53]. Distillation-based methods support this by transferring feature-level knowledge from a classifier trained on past labels to one trained on new labels, enabling the model to adapt without retraining from scratch [54].

2.4. Retrieval

Balancing stability and plasticity in a shared representation is NP-hard [17]. Instead, a model must quickly recall past knowledge and decide which modules to update within a compositional framework [55, 56]. In SL, this is achieved in several ways. Neighborhood-based dynamic selection identifies supervised models with high competence in the local region around the query [57]. Referee meta-models detect recurrence without statistical comparisons across all stored models [58]. Sequence mining meta-models capture patterns in stored models, relying on expectations of transitions in their competence [59, 60]. Repository matching approaches reuse learned models when similar contexts reemerge, following drift detection (Fig. 1b) [61]. Finally, hybrid methods use Bayesian inference to compute posterior probabilities for all candidate states: likelihoods estimated with a weighted cosine distance of the the incoming data window to a meta-representation of each model, and priors estimated via transition matrices informed by drift detectors (Fig. 8) [2]. In unsupervised scenarios, novelty detection methods rely on clustering structures to find cohesive agglomeration of anomalies [62, 63].

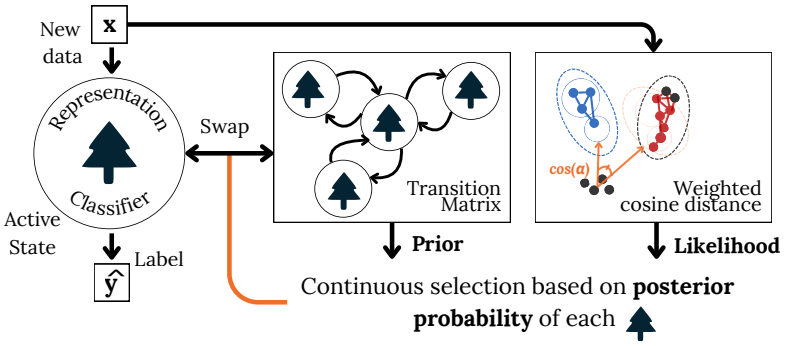
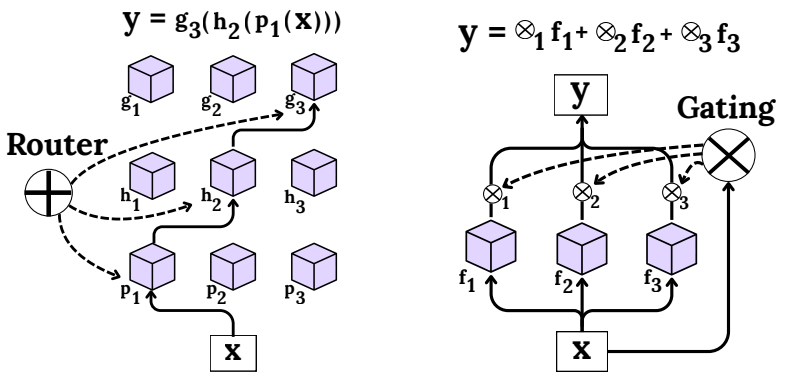


Figure 8: Posterior probabilities for all models, with signals of priors (when a state is expected to reappear) and likelihoods (how much a state matches current data).

In neural networks, retrieval can be facilitated through end-to-end optimization. A common approach involves using multiple expert branches that are selectively activated by a gating mechanism, with their outputs integrated via a data-dependent weighting scheme (Fig. 9b) [64]. Additionally, routing-based techniques allow for greater specialization by decomposing sub-concepts into sequential or parallel processing stages, enabling modules to be flexibly reused across different contexts (Fig. 9a) [65, 66]. In challenging unsupervised streaming scenarios, retrieval often relies on identifying latent concept boundaries. This can be achieved through methods such as cross-concept class discrimination [67], adversarial one-class classification [68], and reverse distillation from one-class embeddings [69]. Furthermore, cross-concept identification can be explicitly modeled using separate networks [70], learned binary masks [71], or various out-of-distribution detection approaches [72].



(a) Modular routing. (b) Mixture-of-experts.

Figure 9: End-to-end differentiable retrieval in CL, enabling modules to be flexibly reused.

3. In-context stream mining for SCL

Traditional SL and CL methods follow a two-stage process: optimizing models over a sliding window, followed by selecting the best configuration for the next window. Foundational models (FMs), however, enable instant model deployment, leveraging prior learning to bypass the need for extensive tuning [73]. Through pretraining on vast corpora, FMs acquire soft inductive biases, drawing from a wealth of prior experiences. This results in emergent abilities, such as few-shot in-context learning (ICL), which allows models to perform new tasks during inference by conditioning on a set of input-output examples, without requiring parameter updates. Consequently, this ICL capability of FMs has spurred a new research paradigm focused on designing architectures that are pre-trained on a wide range of synthetic tabular datasets, referred to as large tabular models (LTMs) [74]. Unlike traditional models, LTMs perform instant classification without fine-tuning [11, 75]. They adapt to unseen datasets in a single forward pass by using various training examples as context, similarly to how large language models (LLMs) use preceding tokens. Practically, an LTM is a transformer model (Fig. 10), trained on data simulated with Bayesian neural networks or structural causal models [76], inductive biases from decision trees [75], or DAG-based computational graphs [11], with the training set size acting as a regularizer on the network's expected complexity.

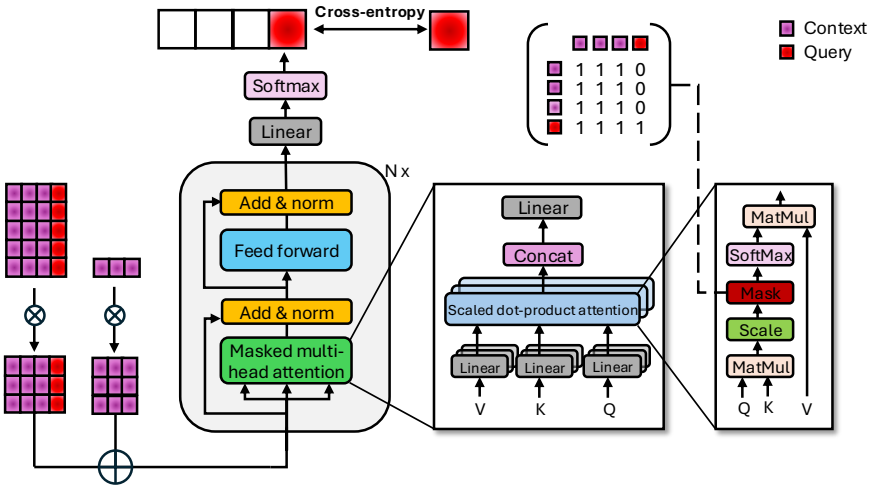


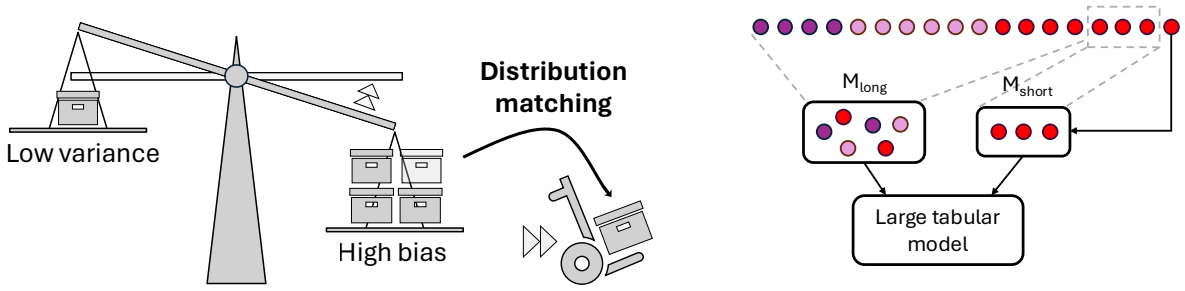
Figure 10: Exemplifying LTM's architecture and training.

Building on these developments, a new paradigm for SL and CL centers on stream-level context construction: summarizing unbounded data streams on-the-fly before providing them to LTMs [12]. Modern LTMs can process extremely large contexts, exceeding 500K samples [75] and 50K features [77], which provides unprecedented flexibility in how sketches of streaming data can be represented. Importantly, this data-centric perspective still allows us to leverage the aforementioned core insights from model-centric SL and CL research concerning **stability**, **plasticity**, **diversification**, and **retrieval**. In fact, these factors can now be controlled directly through context design rather than complex architectural or optimization interventions, enabling simpler and more explicit trade-offs.

Drawing from the literature on how FMs apply data selection strategies for pre-training, instruction-tuning, alignment, and in-context learning [78], we identify two complementary axes for organizing context design. The first concerns **distribution matching**: selecting data similar to the target, yielding **plasticity** when emphasizing the current distribution, and **stability** when maintaining support across prior distributions. The second concerns **distribution compression**: reducing redundancy while maintaining representational power, which supports **diversification** when filling memory with non-overlapping representative samples, and **retrieval** when dynamically constructing a task-specific context from a larger pool.

3.1. Distribution matching

To better understand the goal of distribution matching, one can adopt a frequentist perspective (Fig. 11a) [79]. From a variance standpoint, an LTM, pre-tuned but untrained, with many hyperparameters and multi-head attention, is highly sensitive to individual context samples, which increases its ability to select effective submodels and reduces predictor variance. From a bias standpoint, hyperparameters are optimized for the prior task distribution. If the prior is broad and not overly concentrated away from the true hypothesis, the posterior predictive distribution closely approximates the true predictive distribution. Consequently, the LTM's ability to learn at inference depends on its structural properties, with the optimal approximation characterized by a Kullback-Leibler criterion [79]. Intuitively, adding more context samples reduces sensitivity to minor input perturbations, lowering variance, while bias persists unless the context is concentrated near the target distribution.



(a) In-context learning from frequentist viewpoint.

(b) Dual-memory FIFO system.

Figure 11: Comparison of in-context learning and memory mechanisms.

With this perspective, when the data distribution shifts over time, the context may no longer reflect the current environment, leading the model to produce biased predictions based on outdated patterns. This requires a design trade-off:

- **Plasticity:** prioritizing recent examples to adapt quickly to new local patterns, at the risk of losing information about past classes and concepts.
- **Stability:** retaining examples from earlier concepts to maintain a global understanding of all classes, including those observed in the distant past.

A simple yet effective solution is a dual-memory FIFO system (Fig. 11b) [12]. The long-term memory stores a fixed set of older samples across all known classes, preserving rare or infrequently seen categories. In contrast, the short-term memory maintains the most recent portion of the stream, capturing local variations, transient sub-concepts, and evolving intra-class dynamics. By combining these two memories, the model achieves a balance between long-term stability and rapid adaptability to short-term fluctuations in the data distribution.

However, this approach is naive. Plasticity is largely reactive, implemented via fading strategies rather than proactive adaptation, while stability only addresses catastrophic forgetting of classes, without ensuring invariant representations across all concepts. Addressing this requires diversification and retrieval principles, adopting inductive biases to selectively match data distributions, such as: smoothness, where nearby points in high-density regions are assumed to produce similar outputs; clustering, where points in the same vicinity likely share a concept; and manifold, where high-dimensional data lies on shared low-dimensional latent representations. Notably, because in-context stream mining is formulated as a data selection problem rather than explicit model design, these inductive biases naturally align with the goal of distribution compression, i.e., promoting efficient and representative context construction.

3.2. Distribution compression

Distribution compression aims to prioritize heterogeneity and remove redundancies to reduce dataset size. Different communities approach this goal differently: SL uses synopses and sketches to summarize unbounded streams [80]; CL relies on experience replay to retain past concepts [81]; and FMs apply data selection strategies across pre-training, instruction-tuning, alignment, in-context learning, and fine-tuning [78]. Despite these differences, compression can generally be framed as a two-stage process:

- **Diversification:** populating and updating memory with representative, meaningful samples, through informed addition and deletion.
- **(Optional) Retrieval:** distinguishing between memory population (which samples to store) and sampling (which points to use for in-context learning).

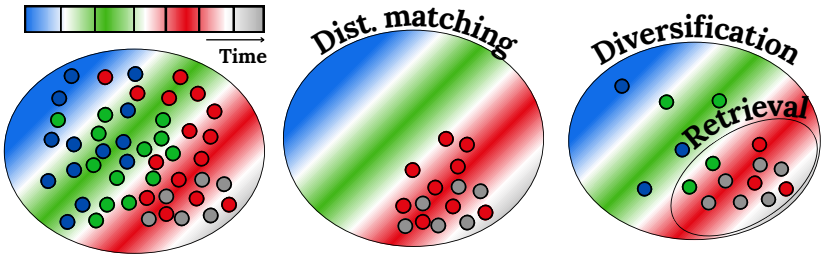


Figure 12: Distribution matching selects data similar to the target. Distribution compression maintains a diversified memory, and (optionally) retrieves data similar to the target.

Distribution compression is inherently linked to distribution matching, shaping how the model balances stability and plasticity when selecting context examples. However, the relationship between diversification and these objectives is subtler than it appears: although diversification is often associated with enhanced stability, it does not necessarily compromise plasticity. For example, selecting examples solely based on similarity to the query in embedding space [82] can lead to redundancy and omit less similar yet informative concepts that support contrastive learning [83]. In contrast, true distribution matching aims to maximize feature coverage [84], select examples according to difficulty [85], and consider sample sensitivity [86]. Empirical evidence supports this (Fig. 13): sequential methods that explicitly balance similarity to the query with diversity among selected examples consistently outperform naive strategies, such as choosing the K most similar examples, or selecting similar examples from a diversity-reduced subset [83].

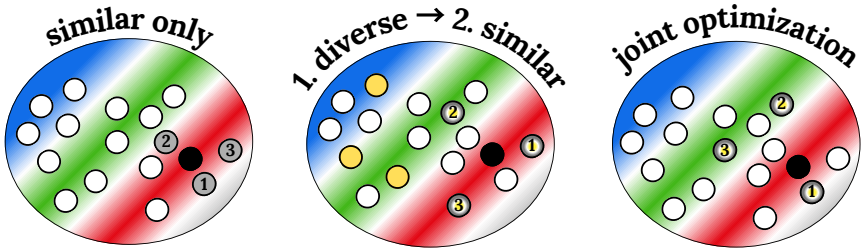


Figure 13: Selection criteria: (left) top 3 most similar from the full dataset; (middle) top 3 from a compressed set of 6 diversified instances; (right) top 3 from the full dataset using a metric that balances similarity and diversity.

Viewed from this perspective, diversification emerges as a mechanism that can simultaneously enhance both plasticity and stability. The benefits of diversification, however, are limited. Its effectiveness is constrained by the inherent difficulty of learning incremental concepts within a finite parameter space. In this context, retrieval serves as a complementary strategy: by separating points for new knowledge from those revisiting prior knowledge, retrieval allows for a divide-and-conquer approach [87]. Importantly, retrieval itself can be understood as a higher-level application of diversification.

While diversification maintains representative and informative examples in memory to support both plasticity and stability, retrieval applies the same principle to select query-specific context subsets, reducing interference between old and new concepts.

With this unified perspective, it becomes clear that heuristic or score-based methods benefiting diversification also enhance retrieval, they simply operate at different stages of the learning process. Critically, any such process must account for the fact that data points vary in potential: some are more representative or informative than others. Classical streaming data selection methods address this by framing the problem as online clustering, leveraging synopsis techniques such as histograms, wavelets, or sketches to construct geometric and statistical descriptors of the data [80]. Concept drift is typically detected by tracking assignment errors [88], or comparing recent and reference data within clusters using tests like the univariate k-sample Anderson-Darling for each principal component of each centroid [89]. However, these approaches are largely unsupervised, whereas our focus is on predictive performance. Predictive strategies may instead maintain a dynamic set of short-term and long-term prototypes based on error-driven representativeness learning and constrained clustering inspired by synchronization [90]. In this vein, numerous data selection methods have emerged across the fields of active learning [91], continual learning [92, 93], and training dynamics [94]. These methods generally assign each instance a scalar score, either reflecting informativeness or representativeness (Fig. 14) [95].

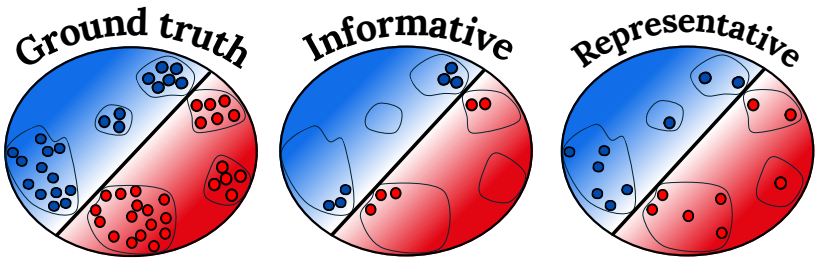


Figure 14: Selection criteria: informativeness prioritizes uncertain or difficult instances, while representativeness aims to capture the overall data distribution.

On one hand, **informativeness** measures how critical a sample is for learning, often favoring hard or uncertain instances. In active learning, this includes points where the model exhibits low confidence [96], high reconstruction errors [97] or high disagreement [98]. In continual learning, analogous metrics are employed, but with the implicit goal of mitigating catastrophic forgetting. Methods include herding selection [99, 93], which samples based on distance to the class mean; discriminative sampling [100], which targets decision boundary points; and entropy-based sampling [26], which selects high-uncertainty predictions. In training dynamics, sample quality is assessed via learnability heuristics such as forgetting frequency [92] or ease of learning [101]. Alternatively, separate scoring models [102, 103] or reinforcement learning strategies [104, 105] can optimize sample selection.

On the other hand, **representativeness** captures how well a subset reflects the overall data distribution, complementing informativeness. In active learning, these help balance exploration-exploitation trade-offs, capturing the structure of the raw data [106, 107] or the embedding space [108]. In contrast, continual learning faces the challenge of maintaining memory samples, where similarity is often be balanced with diversity [109, 110, 111], or informativeness with representativeness [112, 84]. Examples include CoPE [113], which maintains class prototypes in a shared latent space while minimizing intra-class variance and maximizing inter-class separation, and core-set methods such as cardinality-constrained bilevel optimization [114, 115, 116]. However, unlike standard core-set selection, in-context stream mining does not involve parameter updates.

All these principles naturally extend to in-context stream mining, with one key distinction: in LTMs, the memory population effectively is the model, since it directly defines the implicit decision boundaries that the LTM can express [12]. In contrast, in experience replay-based CL, the memory serves primarily as a support mechanism for a separate parametric model, where samples are selectively replayed to prevent forgetting, typically chosen using criteria such as max. loss, or min. margin confidence [117].

From this perspective, the statistical uncertainty of in-context stream mining is primarily determined by which observations are missing or underrepresented. While conventional SL emphasizes accumulated prequential model uncertainty, in-context stream mining focuses on how the current query shifts the hypothesis space in response to a context-query pairing. This is analogous to transductive reasoning, where predictions are made on a closed set of instances without constructing a general model.

Thus, while heuristic or score-based methods remain useful for evaluating individual samples, in-context stream learning benefits from assessing the joint influence of sample subsets. Inclusion or removal of combinations should be evaluated by their collective impact on learning [118], which can be operationalized by prompting the LTM to rate samples and observing performance changes, analogous to a leave-one-out procedure [119]. However, directly computing the performance drop for every possible instance not only may provide insufficient signal, but also is computationally infeasible, requiring 2^n inferences for a dataset of size n. Nonetheless, the framework is valuable for studying how combinations of training examples, rather than individual ones, affect generalization, for example through inclusion or exclusion of entire prototypical classes [120].

To make this approach tractable in practice, data selection can be framed as an online learning-to-rank (LTR) problem, which leverages the counterfactual effects of sample inclusion while minimizing LTM calls [121]. This naturally aligns with reinforcement learning or contextual bandit frameworks, where the system selects ranked lists of exemplars and receives observed performance rewards [122]. The challenge is balancing exploration (testing new rankings) with exploitation (using the best-known one).

4. Conclusions

LTMs emerge as a promising bridge between the CL and SL communities, warranting further investigation. While extreme real-time and edge constraints still requires smaller and faster LTMs, we set aside such engineering constraints, which are likely to be addressed by advances in the TinyML community [123]. Our focus is instead on the SCL context, where the central challenge is not model size or speed, but orchestrating the dynamic interplay of data arrival, training, recovery, and inference. As LTMs naturally become more efficient, research should prioritize algorithms that embody the core principles of streaming continual learning: stability (preserving past knowledge), plasticity (adapting to new data), diversification (reducing redundancy), and retrieval (enabling faster remembering). To illustrate the practical realization of these principles, we draw on insights from multiple communities: from SL, synopsis and sketches provide effective stream summarization [80]; from CL, experience replay preserves past concepts [81]; and from foundational models, data selection strategies enhance pre-training, fine-tuning, and in-context learning [78].

Looking further ahead, end-to-end solutions could selectively activate LTM components per instance, guided by specialized prompter contexts fine-tuned on the most relevant data [124] This strategy is already supported by evidence that fine-tuning improves retrieval-based performance [125, 126]. However, applying such strategies in continual learning remains a significant challenge across all modalities of foundational models [127, 128]. We therefore encourage the community to begin exploring data selection strategies as lightweight inference-time wrappers, which can guide future developments in a manner similar to the evolution of prompt engineering in NLP and vision. The authors also plan to pursue some of these directions and welcome feedback and collaboration from those interested in contributing to this line of research.

Acknowledgments

FCT funded under PhD scholarship PRT/BD/154713/2023 and project doi.org/10.54499/UIDP/00760/2020.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, M. Woźniak, Ensemble learning for data stream analysis: A survey, Information Fusion 37 (2017) 132–156.

[2] B. Halstead, Y. S. Koh, P. Riddle, M. Pechenizkiy, A. Bifet, A probabilistic framework for adapting to changing and recurring concepts in data streams, in: 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2022, pp. 1–10.

[3] D. Sahoo, Q. Pham, J. Lu, S. C. Hoi, Online deep learning: Learning deep neural networks on the fly, arXiv preprint arXiv:1711.03705 (2017).

[4] D. McElfresh, S. Khandagale, J. Valverde, V. Prasad C, G. Ramakrishnan, M. Goldblum, C. White, When do neural nets outperform boosted trees on tabular data?, Advances in Neural Information Processing Systems 36 (2023) 76336–76369.

[5] C. Lyle, Z. Zheng, E. Nikishin, B. A. Pires, R. Pascanu, W. Dabney, Understanding plasticity in neural networks, in: International Conference on Machine Learning, PMLR, 2023, pp. 23190–23211.

[6] A. Lourenço, J. Rodrigo, J. Gama, G. Marreiros, On-device edge learning for iot data streams: a survey, arXiv preprint arXiv:2502.17788 (2025).

[7] S. Lin, P. Ju, Y. Liang, N. Shroff, Theory on forgetting and generalization of continual learning, in: International Conference on Machine Learning, PMLR, 2023, pp. 21078–21100.

[8] J. Gama, P. P. Rodrigues, E. Spinosa, A. Carvalho, Knowledge discovery from data streams, in: Web Intelligence and Security, IOS Press, 2010, pp. 125–138.

[9] N. Gunasekara, B. Pfahringer, H. M. Gomes, A. Bifet, Survey on online streaming continual learning, in: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, 2023, pp. 6628–6637.

[10] F. Giannini, G. Ziffer, A. Cossu, V. Lomonaco, Streaming continual learning for unified adaptive intelligence in dynamic environments, IEEE Intelligent Systems 39 (2024) 81–85.

[11] N. Hollmann, S. Müller, L. Purucker, A. Krishnakumar, M. Körfer, S. B. Hoo, R. T. Schirrmeister, F. Hutter, Accurate predictions on small data with a tabular foundation model, Nature 637 (2025) 319–326.

[12] A. Lourenço, J. Gama, E. P. Xing, G. Marreiros, In-context learning of evolving data streams with tabular foundational models, arXiv preprint arXiv:2502.16840 (2025).

[13] H. M. Gomes, A. Bifet, J. Read, J. P. Barddal, F. Enembreck, B. Pfharinger, G. Holmes, T. Abdessalem, Adaptive random forests for evolving data stream classification, Machine Learning 106 (2017) 1469–1495.

[14] H. M. Gomes, J. Read, A. Bifet, Streaming random patches for evolving data stream classification, in: 2019 IEEE international conference on data mining (ICDM), IEEE, 2019, pp. 240–249.

[15] G. Cormode, S. Muthukrishnan, An improved data stream summary: the count-min sketch and its applications, Journal of Algorithms 55 (2005) 58–75.

[16] M. Riemer, I. Cases, R. Ajemian, M. Liu, I. Rish, Y. Tu, G. Tesauro, Learning to learn without forgetting by maximizing transfer and minimizing interference, in: International Conference on Learning Representations, 2019.

[17] J. Knoblauch, H. Husain, T. Diethe, Optimal continual learning has perfect memory and is np-hard, arXiv preprint arXiv:2006.05188 (2020).

[18] D. Risca, A. Lourenço, G. Marreiros, Continual learning for rotating machinery fault diagnosis with cross-domain environmental and operational variations, arXiv preprint arXiv:2504.10151 (2025).

[19] A. Lourenço, J. Rodrigo, J. Gama, G. Marreiros, Dfdt: Dynamic fast decision tree for iot data stream mining on edge devices, arXiv preprint arXiv:2502.14011 (2025).

[20] Ł. Korycki, B. Krawczyk, Streaming decision trees for lifelong learning, in: Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part I 21, Springer, 2021, pp. 502–518.

[21] Ł. Korycki, B. Krawczyk, Online oversampling for sparsely labeled imbalanced and non-stationary

data streams, in: 2020 international joint conference on neural networks (IJCNN), IEEE, 2020, pp. 1–8.
[22] K. Malialis, C. G. Panayiotou, M. M. Polycarpou, Online learning with adaptive rebalancing in nonstationary environments, IEEE transactions on neural networks and learning systems 32 (2020) 4445–4459.
[23] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., Overcoming catastrophic forgetting in neural networks, Proceedings of the national academy of sciences 114 (2017) 3521–3526.
[24] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, T. Tuytelaars, Memory aware synapses: Learning what (not) to forget, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 139–154.
[25] F. Zenke, B. Poole, S. Ganguli, Continual learning through synaptic intelligence, Proceedings of machine learning research 70 (2017) 3987–3995.
[26] A. Chaudhry, P. K. Dokania, T. Ajanthan, P. H. Torr, Riemannian walk for incremental learning: Understanding forgetting and intransigence, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 532–547.
[27] M. Elsayed, A. R. Mahmood, Addressing loss of plasticity and catastrophic forgetting in continual learning, arXiv preprint arXiv:2404.00781 (2024).
[28] M. De Lange, G. van de Ven, T. Tuytelaars, Continual evaluation for lifelong learning: Identifying the stability gap, arXiv preprint arXiv:2205.13452 (2022).
[29] T. Hess, T. Tuytelaars, G. M. van de Ven, Two complementary perspectives to continual learning: Ask not only what to optimize, but also how, arXiv preprint arXiv:2311.04898 (2023).
[30] M. Farajtabar, N. Azizan, A. Mott, A. Li, Orthogonal gradient descent for continual learning, in: International conference on artificial intelligence and statistics, PMLR, 2020, pp. 3762–3773.
[31] Y. Guo, W. Hu, D. Zhao, B. Liu, Adaptive orthogonal projection for batch and online continual learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, 2022, pp. 6783–6791.
[32] T.-C. Kao, K. Jensen, G. van de Ven, A. Bernacchia, G. Hennequin, Natural continual learning: success is a journey, not (just) a destination, Advances in neural information processing systems 34 (2021) 28067–28079.
[33] R. Pascanu, S. I. Mirzadeh, A study on the plasticity of neural networks, arXiv preprint arXiv:2106.00042 (2021).
[34] D. Nowak Assis, J. P. Barddal, F. Enembreck, Behavioral insights of adaptive splitting decision trees in evolving data stream classification, Knowledge and Information Systems (2025) 1–32.
[35] C. Manapragada, G. I. Webb, M. Salehi, Extremely fast decision tree, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 1953–1962.
[36] M. Heyden, H. M. Gomes, E. Fouché, B. Pfahringer, K. Böhm, Leveraging plasticity in incremental decision trees, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2024, pp. 38–54.
[37] C. Schreckenberger, T. Glockner, H. Stuckenschmidt, C. Bartelt, Restructuring of hoeffding trees for trapezoidal data streams, in: 2020 International Conference on Data Mining Workshops (ICDMW), IEEE, 2020, pp. 416–423.
[38] R. Zhao, Y. You, J. Sun, J. Gama, J. Jiang, Online learning from drifting capricious data streams with flexible hoeffding tree, Information Processing & Management 62 (2025) 104221.
[39] S. Kumar, H. Marklund, B. Van Roy, Maintaining plasticity via regenerative regularization, arXiv preprint arXiv:2308.11958 (2023).
[40] A. Lewandowski, H. Tanaka, D. Schuurmans, M. C. Machado, Directions of curvature as an explanation for loss of plasticity, arXiv preprint arXiv:2312.00246 (2023).
[41] J. Ash, R. P. Adams, On warm-starting neural network training, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 3884–3894.
[42] G. Sokar, R. Agarwal, P. S. Castro, U. Evci, The dormant neuron phenomenon in deep reinforce-

ment learning, in: International Conference on Machine Learning, PMLR, 2023, pp. 32145–32168.
[43] S. Dohare, A. Hernandez-Garcia, A. Lacoste, M. Weiss, Continual backprop: Stochastic gradient descent with persistent randomness, in: International Conference on Machine Learning, 2021, pp. 2660–2670.
[44] L. Neves, A. Lourenço, A. Cano, G. Marreiros, Online Hierarchical Partitioning of the Output Space in Extreme Multi-Label Data Streams, IOS Press, 2025. URL: http://dx.doi.org/10.3233/faia250975. doi:10.3233/faia250975.
[45] N. C. Oza, S. Russell, Experimental comparisons of online and batch versions of bagging and boosting, in: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, 2001, pp. 359–364.
[46] M. M. Idrees, M. Abulaish, A new combination of diversity techniques in ensemble classifiers for handling complex concept drift, Applied Soft Computing 96 (2020) 106613.
[47] S. Abadifard, S. Bakhshi, S. Gheibuni, F. Can, Dyned: Dynamic ensemble diversification in data stream classification, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, 2023, pp. 3707–3711.
[48] A. Cano, B. Krawczyk, Kappa updated ensemble for drifting data stream mining, Machine Learning 109 (2020) 175–218.
[49] B. Veloso, J. Gama, B. Malheiro, Self hyper-parameter tuning for data streams, in: Discovery Science: 21st International Conference, DS 2018, Limassol, Cyprus, October 29–31, 2018, Proceedings 21, Springer, 2018, pp. 241–255.
[50] A. R. Moya, B. Veloso, J. Gama, S. Ventura, Improving hyper-parameter self-tuning for data streams by adapting an evolutionary approach, Data Mining and Knowledge Discovery 38 (2024) 1289–1315.
[51] A. L. D. Rossi, A. C. P. de Leon Ferreira, C. Soares, B. F. De Souza, et al., Metastream: A meta-learning based method for periodic algorithm selection in time-changing data, Neurocomputing (2014).
[52] P. Wang, N. Jin, D. Davies, W. L. Woo, Model-centric transfer learning framework for concept drift detection, Knowledge-Based Systems 275 (2023) 110705.
[53] Z.-Y. Zhang, P. Zhao, Y. Jiang, Z.-H. Zhou, Learning with feature and distribution evolvable streams, in: International Conference on Machine Learning, PMLR, 2020, pp. 11317–11327.
[54] Z. Wang, L. Liu, D. Tao, Deep streaming label learning, in: International Conference on Machine Learning, PMLR, 2020, pp. 9963–9972.
[55] F. Tajwar, A. Kumar, S. M. Xie, P. Liang, No true state-of-the-art? ood detection methods are inconsistent across datasets, arXiv preprint arXiv:2109.05554 (2021).
[56] D. Risca, A. Lourenço, G. Marreiros, Boosting-inspired online learning with transfer for railway maintenance, arXiv preprint arXiv:2504.08554 (2025).
[57] R. Davtalab, R. M. Cruz, R. Sabourin, A scalable dynamic ensemble selection using fuzzy hyper-boxes, Information Fusion 102 (2024) 102036.
[58] J. Gama, P. Kosina, Recurrent concepts in data streams classification, Knowledge and Information Systems 40 (2014) 489–507.
[59] Y. Yang, X. Wu, X. Zhu, Mining in anticipation for concept change: Proactive-reactive prediction in data streams, Data mining and knowledge discovery 13 (2006) 261–289.
[60] O. Wu, Y. S. Koh, G. Dobbie, T. Lacombe, Probabilistic exact adaptive random forest for recurrent concepts in data streams, International Journal of Data Science and Analytics (2022) 1–16.
[61] P. M. Gonçalves Jr, R. S. M. de Barros, Rcd: A recurring concept drift framework, Pattern Recognition Letters 34 (2013) 1018–1025.
[62] E. R. Faria, J. Gama, A. C. Carvalho, Novelty detection algorithm for data streams multi-class problems, in: Proceedings of the 28th annual ACM symposium on applied computing, 2013, pp. 795–800.
[63] A. R. Paupério, A. Risca, Diogo Lourenço, G. Marreiros, R. Martins, Explainable anomaly detection for industrial iot data streams, arXiv preprint arXiv:2512.08885 (2025).
[64] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, J. Dean, Outrageously large

neural networks: The sparsely-gated mixture-of-experts layer, in: International Conference on Learning Representations, 2017.
[65] C. Rosenbaum, T. Klinger, M. Riemer, Routing networks and the challenges of modular and compositional computation, arXiv preprint arXiv:1904.12774 (2019).
[66] O. Ostapenko, D. Suris, A. Szabó, T. Mikolov, Attention for compositional modularity, in: NeurIPS'22 Workshop on All Things Attention: Bridging Different Perspectives on Attention, 2022.
[67] Y. Guo, B. Liu, D. Zhao, Dealing with cross-task class discrimination in online continual learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 20446–20455.
[68] M. Sabokrou, M. Khalooei, M. Fathy, E. Adeli, Adversarially learned one-class classifier for novelty detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3379–3388.
[69] H. Deng, X. Li, Anomaly detection via reverse distillation from one-class embedding, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 9737–9746.
[70] J. von Oswald, C. Henning, J. Sacramento, B. F. Grewe, Continual learning with hypernetworks, arXiv preprint arXiv:1906.00695 (2019).
[71] M. Wortsman, V. Ramanujan, R. Liu, A. Kembhavi, M. Rastegari, J. Yosinski, A. Farhadi, Supermasks in superposition, in: Advances in Neural Information Processing Systems, 2020, pp. 15173–15184.
[72] G. Kim, C. Xiao, T. Konishi, Z. Ke, B. Liu, A theoretical study on solving continual learning, Advances in neural information processing systems 35 (2022) 5065–5079.
[73] T. B. Brown, Language models are few-shot learners, arXiv preprint arXiv:2005.14165 (2020).
[74] B. van Breugel, M. van der Schaar, Why tabular foundation models should be a research priority, arXiv preprint arXiv:2405.01147 (2024).
[75] J. Qu, D. HolzmÃžller, G. Varoquaux, M. L. Morvan, Tabicl: A tabular foundation model for in-context learning on large data, arXiv preprint arXiv:2502.05564 (2025).
[76] N. Hollmann, S. Müller, K. Eggensperger, F. Hutter, Tabpfn: A transformer that solves small tabular classification problems in a second, arXiv preprint arXiv:2207.01848 (2022).
[77] C. Kolberg, K. Eggensperger, N. Pfeifer, Tabpfn-wide: Continued pre-training for extreme feature counts, arXiv preprint arXiv:2510.06162 (2025).
[78] A. Albalak, Y. Elazar, S. M. Xie, S. Longpre, N. Lambert, X. Wang, N. Muennighoff, B. Hou, L. Pan, H. Jeong, et al., A survey on data selection for language models, arXiv preprint arXiv:2402.16827 (2024).
[79] T. Nagler, J. Lützen, Statistical foundations of prior-data fitted networks, arXiv preprint arXiv:2305.11175 (2023).
[80] J. A. Silva, E. R. Faria, R. C. Barros, E. R. Hruschka, A. C. De Carvalho, J. Gama, Data stream clustering: A survey, ACM Computing Surveys (CSUR) 46 (2013) 1–31.
[81] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. K. Dokania, P. H. Torr, M. Ranzato, On tiny episodic memories in continual learning, arXiv preprint arXiv:1902.10486 (2019).
[82] Z. Wu, Y. Wang, J. Ye, L. Kong, Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering, arXiv preprint arXiv:2212.10375 (2022).
[83] W. Xiao, H. Zhao, L. Huang, The role of diversity in in-context learning for large language models, arXiv preprint arXiv:2505.19426 (2025).
[84] I. Levy, B. Bogin, J. Berant, Diverse demonstrations improve in-context compositional generalization, arXiv preprint arXiv:2212.06800 (2022).
[85] R. A. Cook, J. P. Lalor, A. Abbasi, No simple answer to data complexity: An examination of instance-level complexity metrics for classification tasks, in: Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), 2025, pp. 2553–2573.

[86] Y. Chen, C. Zhao, Z. Yu, K. McKeown, H. He, On the relation between sensitivity and accuracy in in-context learning, arXiv preprint arXiv:2209.07661 (2022).
[87] Z. Wang, Z. Zhang, S. Ebrahimi, R. Sun, H. Zhang, C.-Y. Lee, X. Ren, G. Su, V. Perot, J. Dy, et al., Dualprompt: Complementary prompting for rehearsal-free continual learning, in: European conference on computer vision, Springer, 2022, pp. 631–648.
[88] Y. Sakamoto, K.-i. Fukui, J. Gama, D. Nicklas, K. Moriyama, M. Numao, Concept drift detection with clustering via statistical change detection methods, in: 2015 Seventh International Conference on Knowledge and Systems Engineering (KSE), IEEE, 2015, pp. 37–42.
[89] J. S.-W. Wan, S.-D. Wang, Concept drift detection based on pre-clustering and statistical testing, Journal of Internet Technology 22 (2021) 465–472.
[90] J. Shao, Z. Ahmadi, S. Kramer, Prototype-based learning on concept-drifting data streams, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014, pp. 412–421.
[91] O. Sener, S. Savarese, Active learning for convolutional neural networks: A core-set approach, arXiv preprint arXiv:1708.00489 (2017).
[92] M. Toneva, A. Sordoni, R. T. d. Combes, A. Trischler, Y. Bengio, G. J. Gordon, An empirical study of example forgetting during deep neural network learning, arXiv preprint arXiv:1812.05159 (2018).
[93] F. M. Castro, M. J. Marín-Jiménez, N. Guil, C. Schmid, K. Alahari, End-to-end incremental learning, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 233–248.
[94] P. W. Koh, P. Liang, Understanding black-box predictions via influence functions, in: International conference on machine learning, PMLR, 2017, pp. 1885–1894.
[95] B. Pecher, I. Srba, M. Bielikova, J. Vanschoren, Automatic combination of sample selection strategies for few-shot learning, arXiv preprint arXiv:2402.03038 (2024).
[96] B. Krawczyk, Active and adaptive ensemble learning for online activity recognition from data streams, Knowledge-Based Systems 138 (2017) 69–78.
[97] Z. Yu, S. Huang, K. Yang, J. Lv, C. P. Chen, Ensemble approaches for dynamic data stream classification under label scarcity, IEEE Transactions on Big Data (2025).
[98] D. M. Cavalcanti, R. Cerri, E. R. Faria, Arm-stream: active recovery of miscategorizations in clustering-based data stream classifiers, Data Mining and Knowledge Discovery 39 (2025) 1–35.
[99] M. Welling, Herding dynamical weights to learn, in: Proceedings of the 26th annual international conference on machine learning, 2009, pp. 1121–1128.
[100] Y. Liu, Y. Su, A.-A. Liu, B. Schiele, Q. Sun, Mnemonics training: Multi-class incremental learning without forgetting, in: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, 2020, pp. 12245–12254.
[101] S. Swayamdipta, R. Schwartz, N. Lourie, Y. Wang, H. Hajishirzi, N. A. Smith, Y. Choi, Dataset cartography: Mapping and diagnosing datasets with training dynamics, arXiv preprint arXiv:2009.10795 (2020).
[102] O. Rubin, J. Herzig, J. Berant, Learning to retrieve prompts for in-context learning, arXiv preprint arXiv:2112.08633 (2021).
[103] X. Li, X. Qiu, Finding support examples for in-context learning, arXiv preprint arXiv:2302.13539 (2023).
[104] Y. Zhang, S. Feng, C. Tan, Active example selection for in-context learning, arXiv preprint arXiv:2211.04486 (2022).
[105] P. Lu, L. Qiu, K.-W. Chang, Y. N. Wu, S.-C. Zhu, T. Rajpurohit, P. Clark, A. Kalyan, Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning, arXiv preprint arXiv:2209.14610 (2022).
[106] D. Himaja, V. Dondeti, S. Uppalapati, S. Virupaksha, Cluster based active learning for classification of evolving streams, Evolutionary Intelligence 17 (2024) 2167–2191.
[107] C. Yin, S. Chen, Z. Yin, Clustering-based active learning classification towards data stream, ACM Transactions on Intelligent Systems and Technology 14 (2023) 1–18.
[108] Z. Zhang, Y. Jiang, S. Zhang, X. Xu, Nonstationary data stream classification with online active

learning and siamese neural networks, IEEE Transactions on Neural Networks and Learning Systems 33 (2021) 3087–3101.
[109] S. Sun, D. Calandriello, H. Hu, A. Li, M. Titsias, Information-theoretic online memory selection for continual learning, arXiv preprint arXiv:2204.04763 (2022).
[110] F. Wiewel, B. Yang, Entropy-based sample selection for online continual learning, in: 2020 28th European signal processing conference (EUSIPCO), IEEE, 2021, pp. 1477–1481.
[111] C. Qin, A. Zhang, C. Chen, A. Dagar, W. Ye, In-context learning with iterative demonstration selection, arXiv preprint arXiv:2310.09881 (2023).
[112] J. Bang, H. Kim, Y. Yoo, J.-W. Ha, J. Choi, Rainbow memory: Continual learning with a memory of diverse samples, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 8218–8227.
[113] M. De Lange, T. Tuytelaars, Continual prototype evolution: Learning online from non-stationary data streams, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 8250–8259.
[114] Z. Borsos, M. Mutny, A. Krause, Coresets via bilevel optimization for continual learning and streaming, Advances in neural information processing systems 33 (2020) 14879–14890.
[115] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, R. Iyer, Glister: Generalization based data subset selection for efficient and robust learning, in: Proceedings of the AAAI conference on artificial intelligence, volume 35, 2021, pp. 8110–8118.
[116] B. Mirzasoleiman, J. Bilmes, J. Leskovec, Coresets for data-efficient training of machine learning models, in: International Conference on Machine Learning, PMLR, 2020, pp. 6950–6960.
[117] A. Prabhu, P. H. Torr, P. K. Dokania, Gdumb: A simple approach that questions our progress in continual learning, in: European Conference on Computer Vision, Springer, 2020, pp. 524–540.
[118] S. Yang, Z. Xie, H. Peng, M. Xu, M. Sun, P. Li, Dataset pruning: Reducing training data by examining generalization influence, arXiv preprint arXiv:2205.09329 (2022).
[119] H. Tan, S. Wu, F. Du, Y. Chen, Z. Wang, F. Wang, X. Qi, Data pruning via moving-one-sample-out, Advances in neural information processing systems 36 (2023) 18251–18262.
[120] S. Jain, H. Salman, A. Khaddaj, E. Wong, S. M. Park, A. Madry, A data-based perspective on transfer learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 3613–3622.
[121] A. Grotov, M. De Rijke, Online learning to rank for information retrieval: Sigir 2016 tutorial, in: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, 2016, pp. 1215–1218.
[122] K. Purohit, V. Venktesh, S. Bhattacharya, A. Anand, Sample efficient demonstration selection for in-context learning, arXiv preprint arXiv:2506.08607 (2025).
[123] S. Somvanshi, M. M. Islam, G. Chhetri, R. Chakraborty, M. S. Mimi, S. A. Shuvo, K. S. Islam, S. A. Javed, S. A. Rafat, A. Dutta, et al., From tiny machine learning to tiny deep learning: A survey, arXiv preprint arXiv:2506.18927 (2025).
[124] D. Xu, O. Cirit, R. Asadi, Y. Sun, W. Wang, Mixture of in-context prompters for tabular pfns, arXiv preprint arXiv:2405.16156 (2024).
[125] F. den Breejen, S. Bae, S. Cha, T.-Y. Kim, S. H. Koh, S.-Y. Yun, Fine-tuning the retrieval mechanism for tabular deep learning, arXiv preprint arXiv:2311.17641 (2023).
[126] V. Thomas, J. Ma, R. Hosseinzadeh, K. Golestan, G. Yu, M. Volkovs, A. Caterini, Retrieval & fine-tuning for in-context tabular models, arXiv preprint arXiv:2406.05207 (2024).
[127] J. Bell, L. Quarantiello, E. N. Coleman, L. Li, M. Li, M. Madeddu, E. Piccoli, V. Lomonaco, The future of continual learning in the era of foundation models: Three key directions, arXiv preprint arXiv:2506.03320 (2025).
[128] E. N. Coleman, L. Quarantiello, Z. Liu, Q. Yang, S. Mukherjee, J. Hurtado, V. Lomonaco, Parameter-efficient continual fine-tuning: A survey, arXiv preprint arXiv:2504.13822 (2025).