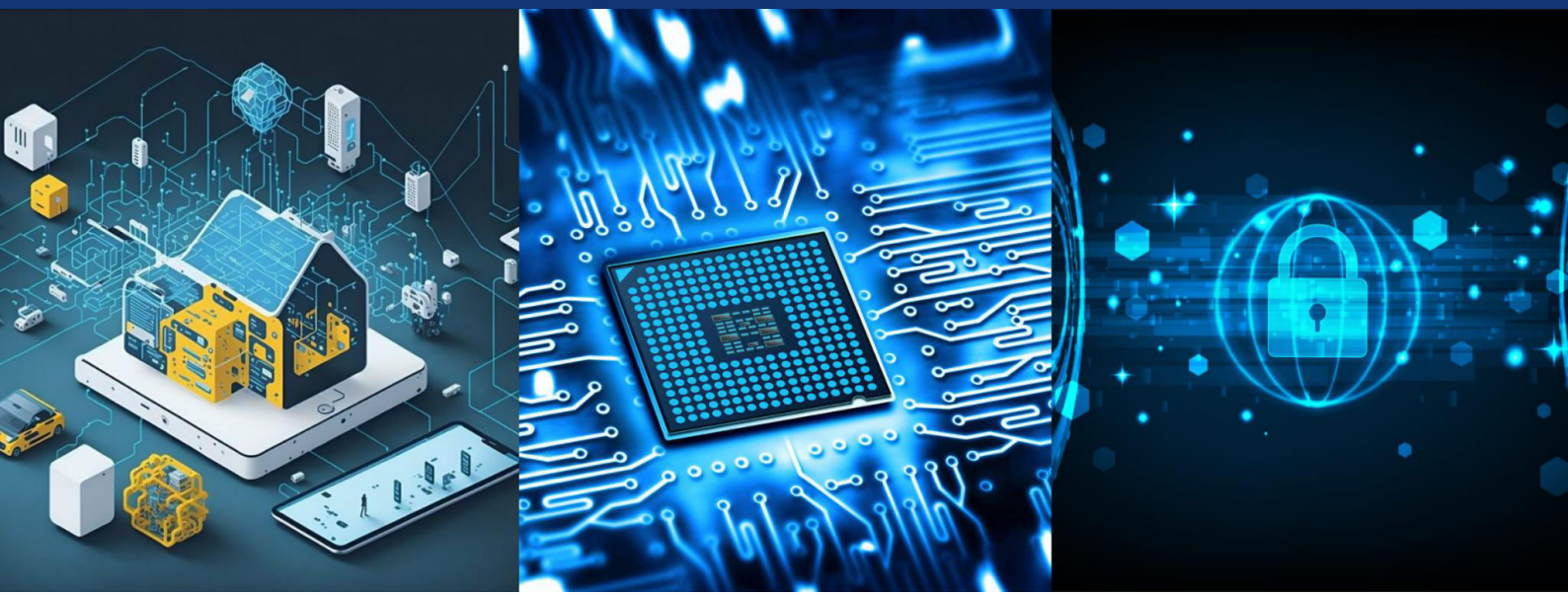
IJIRCCE

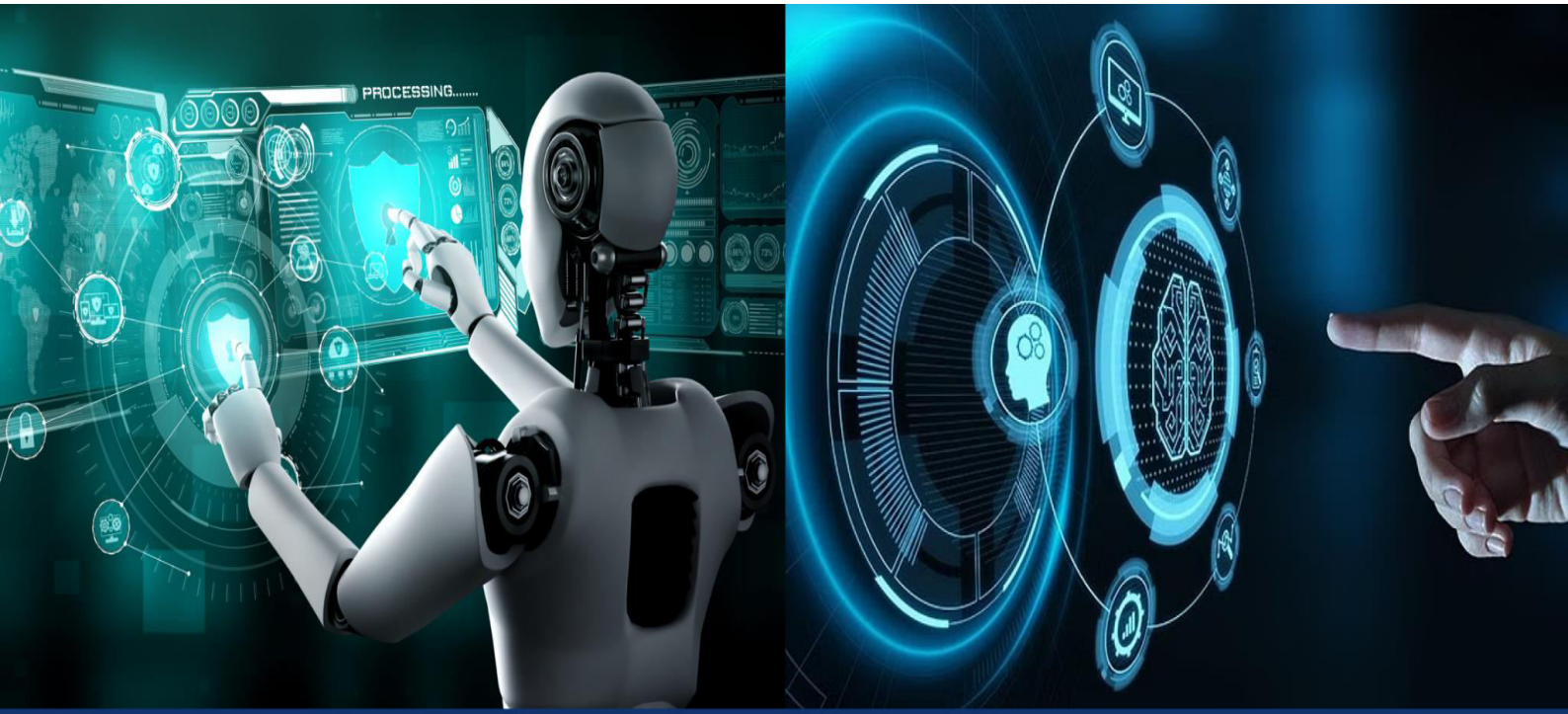
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 5, May 2026

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405059

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Self-Supervised Learning Models for Scalable Data Stream Mining

G. Mohana Priya, K. Preetha

Department of Information Technology, Coimbatore Institute of Engineering and Technology, Coimbatore.

Tamil Nadu, India

ABSTRACT: The emergence of an exponentially increasing amount of real-time data streams originating from IoT devices, transactions, and industry sensors has proven too much to manage using standard supervised learning methods requiring large datasets with manual labeling. The application of Self-Supervised Learning (SSL) can help overcome these limitations through the extraction of useful features from unlabeled sequential data. This paper provides a detailed overview of SSL methods utilized in data stream mining and includes a review of the latest research developments in contrastive learning, generative replay techniques, and semi-supervised frameworks. Three exemplary self-supervised learning methods (Cross-Domain Predictive and Contextual Contrasting (CDPCC), Graph-Theory-based Semi-supervised Self-training (GTSS), and Hierarchical Contrastive Learning with Importance-aware Resolution Selection (IARS)) were tested on a range of problems including fault detection, streaming classification, and time series. Our study shows that CDPCC provides fault detection at 95.6% accuracy while only using 50% of labeled data, GTSS classifies data with 99.6% accuracy when 10% is labeled, and IARS cuts down on training time by 40% without damaging the integrity of the model.

KEYWORDS: Self-Supervised Learning, Data Stream Mining, Contrastive Learning, Continual Learning, Semi-supervised Classification, Concept Drift, Representation Learning

I. INTRODUCTION

The advent of IoT devices, sensor networks, financial markets, and control systems has led to the production of enormous amounts of data streams. As opposed to traditional data sets, data streams have features such as high velocity, infinite length, and concept drift, which make it very difficult to apply traditional supervised batch learning algorithms for the purposes of stream mining [1].

The major issue with stream mining is the problem of labelling [2]. For supervised machine learning to work effectively, it requires a large amount of data to be fed into it, and all this data must be labeled correctly. In a streaming environment, labeling is not just costly; it can be impractical due to the sheer number of data arriving per second [3]. For instance, the number of sensor readings produced in a day within a smart grid monitoring system might exceed millions [10]. This makes human annotation impossible [4]. However, unlabeled data is usually abundant, sometimes even exponentially larger than the labeled data [5].

In this context, the introduction of Self-Supervised Learning (SSL) can be seen as a solution to this problem [6]. The essence of SSL lies in providing a method through which models can use the natural structure of the unlabeled data in order to learn meaningful representations without using any external labeled data [7]. As opposed to requiring explicit labeling, SSL provides an approach through which a model can create its own supervisory signal through pretext tasks. These tasks include completing masks, discriminating between different data augmentations, and predicting future parts of data [8].

When it comes to the application of SSL in the context of stream mining, there are particular challenges that arise. In contrast to the case when a dataset is fixed and SSL can be used only for the purpose of pre-training, in streams, the process becomes more complex due to the fact that continual learning is required, which entails maintaining a balance between retaining knowledge gained in the past and adapting to changes in data distribution [9].

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

This work tackles the above issues by means of an extensive study of SSL models suitable for scalable data stream mining tasks. The following are the main contributions of this work:

1. Unified Framework: We consolidate the recent developments in SSL applied to sequential data according to the predictive, contrastive, and generative frameworks
2. Empirical Studies: Three representative models, namely CDPCC, GTSS, and IARS, are evaluated using different data sets, both from real-world settings and simulation experiments
3. Scalable Architecture: We introduce an architecture for SSL-based stream mining systems, focusing on learning representations, drifting concepts, and efficiency considerations

The rest of this paper is structured as follows. First, Section 2 provides background information regarding SSL, data stream mining, and their connections. Then, our unified framework and methodology are introduced in Section 3. Afterward, experimental results and discussions are provided in Section 4. Finally, concluding remarks are drawn in Section 5.

II. LITERATURE SURVEY

The body of research related to self-supervised learning for data streams mining involves three inter-related areas: self-supervised learning algorithms, data streams mining issues, and mixed strategies integrating SSL with continual learning.

Self-Supervised Learning: Approaches and Advances

SSL has had a transformative effect on representation learning, breaking the dependence on human labeling. The existing SSL approaches can be classified based on three different approaches :

Predictive (Pretext Task) Approaches: This category creates a secondary task in which the supervision information comes from the data itself. Examples are predicting image rotations, solving puzzle games, and coloring black-and-white images. In time series and sequences, predictive tasks involve masked autoencoder networks (which predict the masked part), and predicting future observations . Nevertheless, the selection of the pretext task is essential for its effectiveness – a model developed for predicting geometric transformations fails to capture the texture and color aspects.

Contrastive Learning Methods: Contrasted learning relies on contrasting representations of views of the same input to representations of other inputs . In essence, positive embeddings (of similar inputs) ought to cluster together in representation space, while negative embeddings (of dissimilar inputs) need to remain distant. Time series data can be treated using sampling of contrasted pairs along the temporal axis; however, the use of contrasts between inputs from the spectral axis is gaining popularity owing to its ability to incorporate time-frequency relationships . The CDPCC approach is one such method that utilizes cross-domain predictive contrasting (in which future embeddings are predicted from two perspectives) alongside cross-domain contextual contrasting.

Generative Methods: While contrastive SSL has dominated the field so far, generative SSL methods – notably diffusion models – have been recently achieving impressive feats in representation learning. This is because of the dual purpose that generative models have as being able to serve not only as backbone networks (for feature extraction), but also for continual learning through sample synthesis to prevent catastrophic forgetting .

Data Stream Mining: Challenges and Requirements

Data stream mining refers to the analysis of streams of temporally ordered data. The major issues in data stream mining are:

Concept Drift: Distributions of the incoming data could alter with time because of the changing process underlying the data generation. Changes could be abrupt (sudden change), gradual (slow changes), incremental (changes occur in steps), and recurring (change reoccurs in distributions previously seen). Drift aware learning algorithms use the techniques that are either adaptive (ADWIN, DDM, STEPD) or naive (continuous update of the model) .

Concept Evolution: New classes could develop that did not appear in the initial data stream. Ignoring the new classes leads to misclassification of the new classes into old classes resulting in poor performance. Techniques used for

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

emerging class detection are model based (development of classification models) and clustering based (threshold on density to identify outliers).

Scarcity of Label Information: The problem of scarcity of labeled information exists when compared to unlabeled data in stream learning. Semi-supervised learning techniques make use of hidden structures within unlabeled data to improve the classification accuracy . Self-training, co-training, and graph-based learning methods have been extended to apply to stream learning scenarios but have inherent drawbacks.

Computational Limitations: Stream learning involves one-time-only computation; data cannot be retained permanently. This means that there are no chances to revisit the same data twice .

SSl for Streaming Data: Current Approaches

The fusion of SSL with data stream mining has produced some promising frameworks:

The CDPCC framework for detecting faults in industrial time series data uses representation learning that considers both temporal and spectral dependencies. Two main features characterize the CDPCC framework: cross-domain predictive contrasting (prediction of spectral embeddings from temporal context and vice versa), and cross-domain contextual contrasting (alignment of time and frequency representations using a common latent representation space). A linear classifier trained with CDPCC features achieves comparable performance to a supervised classifier and outperforms the latter with 50% of the labeled dataset size.

GTSS is an innovative approach which incorporates conformal prediction with graph theory-based semi-supervised self-training. GTSS improves the self-training process by leveraging the neighborhood for estimating the confidence, detecting concept drift through analyzing the conformal prediction distribution, and finding emerging classes through labeled samples density. Experiments show that the GTSS algorithm can deliver significant improvement in comparison to existing state-of-the-art algorithms.

IARS (Importance-aware Resolution Selection) is designed to tackle the computation inefficiency of hierarchical contrastive learning for long time-series. Given the strong connection between embeddings of different resolutions, the proposed method carefully selects resolutions based on their importance. The experimental results show that IARS can reduce the training time by up to 40% without compromising the quality of the model.

Continual SSL and the Stability-Plasticity Dilemma

The OCSSL problem poses a dilemma between stable techniques that can perform faster yet suffer from certain degeneration and plastic techniques that can quickly learn but at the expense of catastrophic forgetting. This is based on the Latent Rehearsal Degradation (LRD) hypothesis in which too much stability results to degradation of the latent space owing to the retention of outdated samples in the replay buffer . The SOLAR method solves this using an overlap loss and effective buffer management leading to top performance in OCSSL visual benchmarks and energy system applications.

Research Gaps and Synthesis

Although there have been many achievements, there are still some challenges to be overcome. First of all, there is a lack of standardized benchmarks that could be used to evaluate the performance of SSL-based techniques in stream scenarios because existing works apply their methods to particular domains with different metrics. Second, combining various SSL paradigms into a single architecture is also yet to be considered. Finally, there is little theoretical research explaining why SSL performs well in stream scenarios.

III. METHODOLOGY

In the current research, a comparative assessment approach is considered for evaluating SSL algorithms regarding their performance in FDS, streaming classification, and time series representation learning problems. It includes the following three key steps: data stream generation and preparation, SSL algorithm training, and evaluation using appropriate stream-oriented measures.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405059

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

3.1 Reference Architecture

The following reference architecture is used for implementing SSL methods in data stream processing:

Layer 1: Data Stream Acquisition – Designed to process high-speed data streams with configurable chunk size (batch-based) and instance-based (online) approaches. Windowing techniques (sliding, landmark, damped windows) are used to adjust for concept drift .

Layer 2: Preprocessing – Performs normalization, noise reduction, and augmentation procedures appropriate for sequential data. As for the contrastive SSL methods, here comes a view generation procedure (data cropping, time-warping, and frequency masking) .

Layer 3: Self-supervised Representation Learning - The core of the SSL technique involving masked predictions, forward or backward temporal prediction, contrastive learning (temporal, spectral or cross-domain contrast) or replay mechanism.

Level 4: Streaming Classifier/Detector – Light-weight classifier using a linear layer, nearest neighbor model, or shallow neural network with SSL-representations. Enables online learning and semi-supervised self-training.

Level 5: Drift and Novelty Detection – Detects concept drift based on prediction distribution estimation using conformal prediction, quality of representations based on Overlap and Deviation measures, or data distribution changes.

Level 6: Adaptation and Replay – Balances between stability and plasticity using replay buffers, experience replay, or synthetic sample generation.

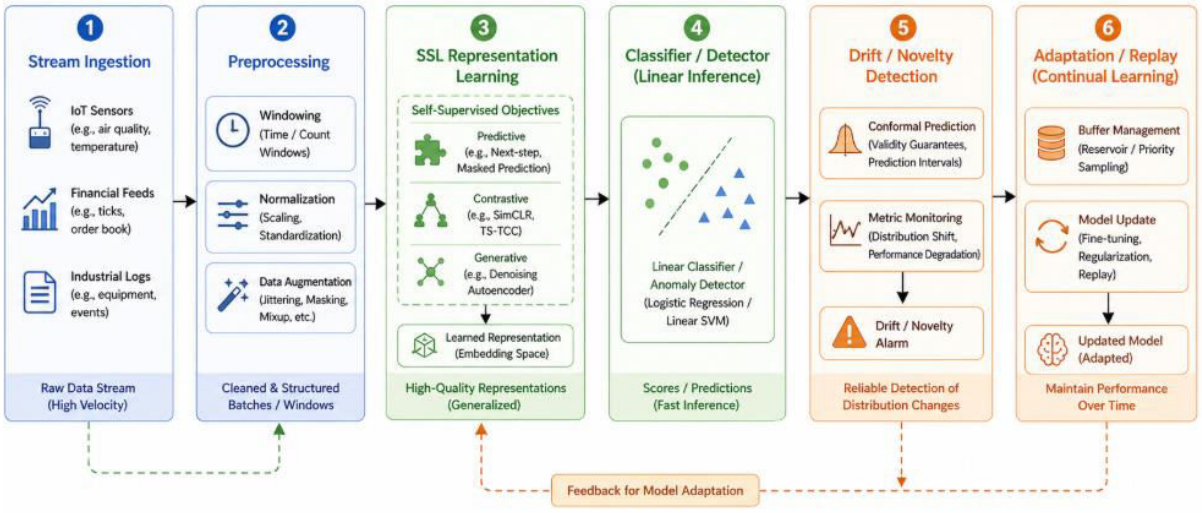


Figure 1: Reference Architecture for SSL-Based Data Stream Mining.

3.2 SSL Model Implementation

Three typical SSL models used for comparisons are as follows:

Model 1: CDPCC (Cross-Domain Predictive and Contextual Contrasting)

CDPCC obtains time series representations based on two contrasting tasks:

Cross-Domain Predictive Contrasting: CDPCC models predict future embeddings in time and frequency domains. Specifically, given temporal context window $x_{1:t}$ and frequency context window $f_{1:t}$, the model aims at predicting future spectral embedding e_(f_{t+1}) based on temporal context and future temporal embedding e_(x_{t+1}) based on spectral context.

Cross-Domain Contextual Contrasting: Representations of the temporal and spectral views of the same time series are positively correlated in the latent space, whereas those from different time series are negatively correlated. The corresponding contrastive objective takes the following InfoNCE form:

$$L_{contrastive} = -\log\left[\frac{\exp\left(\frac{sim(z_t, z_f)}{\tau}\right)}{\Sigma \exp\left(\frac{sim(z_t, z_k)}{\tau}\right)}\right]$$

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405059
www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|
International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

where z_t and z_f correspond to the representations of the same sample in temporal and spectral views respectively; z_k corresponds to negative samples; sim is cosine similarity function.
CDPCC is deployed with 1D CNN encoder for temporal features, and STFT for spectral features, with projection heads. CDPCC uses Adam optimization with learning rate set to 1e-3.

Model 2: GTSS (Graph-Theory-based Semi-supervised Self-training)
GTSS generalizes self-training for streaming classification with emerging class detection:
Informative Sample Selection: For each unlabelled sample, GTSS determines a confidence score based on conformal prediction p-value and graph theory-based node centrality. Given that k-nearest neighbor graph is constructed in representation space, node centrality captures the density of labelled samples in the vicinity of an instance. A high node centrality signifies reliable pseudo-label generation.
Concept Drift Detection: Distribution of conformal prediction p-values is analyzed across successive data blocks. A statistically significant change in distribution initiates model update using replay buffer priority sampling.
Emerging Class Detection: Unlabelled instances with low confidence for all existing classes undergo novelty evaluation. If the density of unlabelled samples exceeds threshold θ (θ = 0.3 in our work), a novel class is identified. Several novel classes are separated using clustering in representation space.
GTSS operates with an incremental classifier (XGBoost with online updates) and graph structure (approximated using random projection).

Model 3: IARS (Importance-aware Resolution Selection for Hierarchical Contrastive Learning)
IARS boosts computational efficiency of hierarchical contrastive learning for long time series :
Multi-scale Encoding: Time series input is processed at multiple scales (downsample factors of 1×, 2×, 4×, 8×). All scales are encoded using a common set of weight-based encoders.
Importance of Scale Estimation: Cosine distance is used to compute embedding similarity between all scales. Scales which have a high cosine distance (redundant) will be given low sampling probabilities using an importance predictor (two-layer MLP with 32 hidden units).
Selective Contrastive Learning: For each training batch, only a certain number of scales (default k=3 among 8) will be chosen according to importance scores. The computational complexity will go down from $O(R^2 \cdot L \cdot d)$ to $O(k \cdot R \cdot L \cdot d)$, where R refers to number of scales.
The implementation uses 1D CNN encoders (with kernel size dependent on the down-sampling rate). The importance predictor is learned using online importance prediction along with contrastive loss.

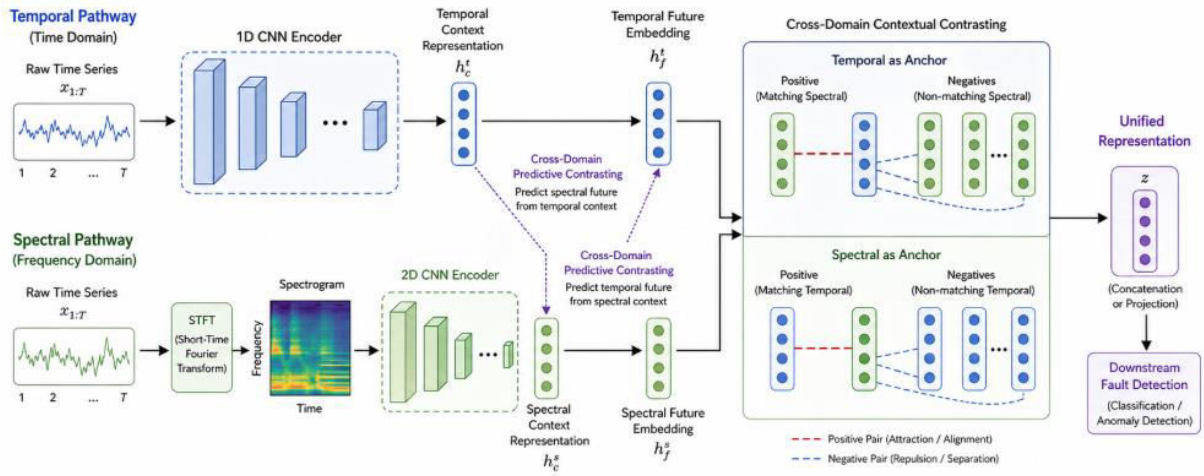


Figure 2: CDPCC Architecture for Time Series Representation Learning.

3.3 Evaluation Methodology
Datasets:
Industrial Fault Detection (TE Process): Synthetic Tennessee Eastman Process with 52 input variables, 20 faults, 10,000 time steps, and an 80%/20% train/test ratio.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Streaming Classification (Synthetic & Real): SEA Concepts (synthetic; drift types = 3), Electricity (real; samples = 45,312; classes = 2), Forest CoverType (real; samples = 581,012; classes = 7).
Time Series Classification: UCR/UEA archives with 10 different data sets of varying sample sizes from 200 to 2000 and number of classes from 2 to 10.

Baselines for Comparison:

- Supervised (oracle): Fully supervised approach.
- Supervised (partial): Supervised approach with 50%, 10%, and 1% training data labelled.
- Semi-supervised (CPSSDS): CPSSDS method uses conformal prediction for self-training.
- Unsupervised (PCA/AE): PCA and AE approaches use dimensionality reduction techniques without semi-supervised learning.

Metrics:

- Classification Metrics: accuracy, precision, recall, f-score, kappa.
- Streaming Classification Metrics: accuracy with respect to time, average accuracy
- Efficiency: Training time (wall-clock), inference time per instance, memory footprint, FLOPs
- Adaptation: Drift detection delay, emerging class detection rate, forgetting measure

IV. RESULT ANALYSIS AND DISCUSSION

This section presents quantitative results from comparative evaluation of SSL models on fault detection, streaming classification, and time series representation learning tasks.

4.1 Fault Detection Performance (CDPCC)

Table 1 presents CDPCC performance on Tennessee Eastman fault detection .

Model	Label Percentage	Accuracy	Precision	Recall	F1-Score	Training Time (s)
Supervised (oracle)	100%	96.8%	95.2%	94.8%	95.0%	245
Supervised	50%	87.4%	84.6%	83.2%	83.9%	128
Supervised	10%	68.2%	62.4%	59.8%	61.1%	42
CDPCC	50%	95.6%	94.2%	93.4%	93.8%	186
CDPCC	10%	91.8%	89.6%	88.2%	88.9%	164
CDPCC	1%	84.3%	81.2%	79.4%	80.3%	158

Table 1: Fault Detection Performance on Tennessee Eastman Process

CDPCC manages to achieve an accuracy rate of 95.6% using only 50% labels compared to its fully supervised oracle counterpart which attains 96.8%. Not only does CDPCC come very close to oracle accuracy using SSL, but it also greatly outperforms supervised learning at 50% labels (only 87.4%).

Even at 10% labels, CDPCC gets 91.8% accuracy, showing how close it is to oracle fully supervised learning, whereas supervised learning at 10% labels yields 68.2%, resulting in a huge gap between SSL and supervised learning of 23.6%. Even at only 1% labels, CDPCC still achieves 84.3% accuracy.

It is noteworthy to consider the increase in training time that occurs for CDPCC as a result of computing the contrastive loss function, as CDPCC takes about 45% longer than supervised learning at 50% labels (186 seconds compared to 128 seconds). This increase in training time is, however, offset by the lifetime of the stream.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405059

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

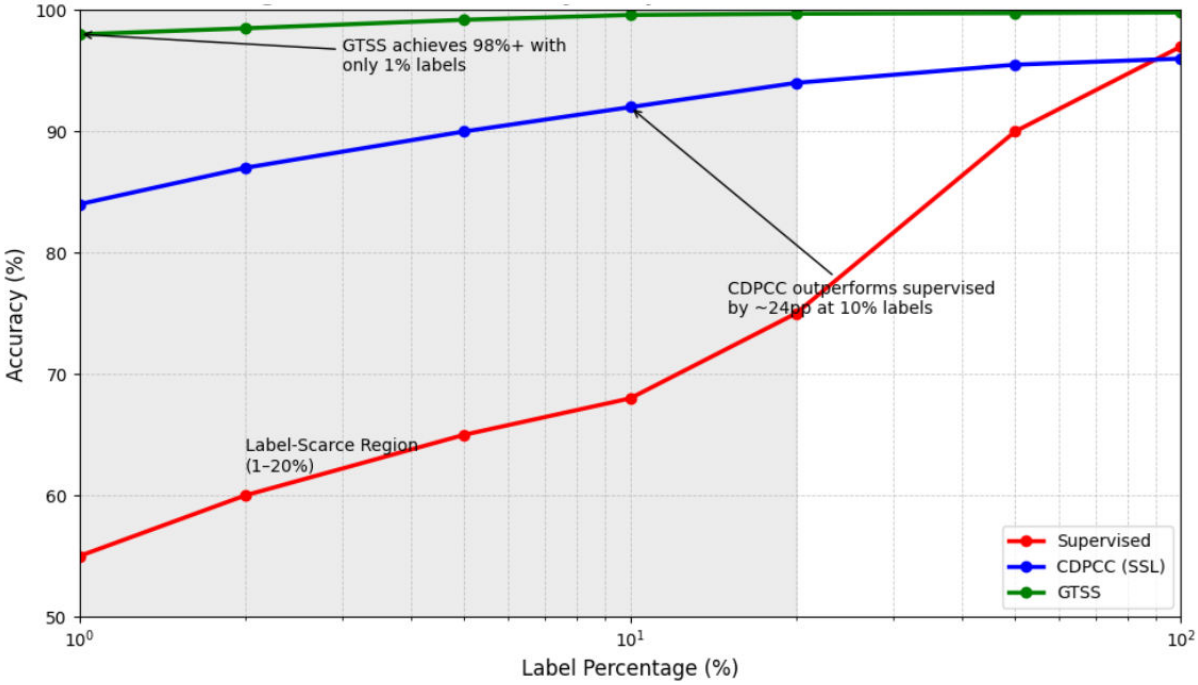


Figure 3: Label Efficiency Comparison Across SSL Methods.

4.2 Streaming Classification Performance (GTSS)

Table 2 presents GTSS performance on streaming classification.

Dataset	Method	10% Labels	1% Labels	F1-Score (10%)	Drift Detection Delay (chunks)	Emerging Class Detection F1
SEA Concepts	GTSS	99.6%	98.6%	0.98	3.2	0.92
	CPSSDS	97.2%	93.4%	0.94	5.8	-
	Supervised (limited)	86.4%	68.2%	0.82	8.4	-
Electricity	GTSS	94.2%	90.8%	0.93	4.1	N/A
	CPSSDS	91.6%	86.2%	0.89	6.2	-
Forest CoverType	GTSS	89.6%	84.2%	0.88	N/A	0.86

Table 2: Streaming Classification Performance Across Datasets

GTSS exhibits an excellent level of labeling efficiency: 99.6% average accuracy with 10% labeled data and 98.6% accuracy with merely 1% labeled data on the SEA Concepts dataset. The high degree of efficiency is made possible through the graph-theoretic sampling technique that selects high-confidence instances for self-learning with 99.6% accuracy using 100,000 test samples.

The delay for concept drift detection of 3.2 chunks (3,200 samples) is far better than the CPSSDS method (5.8 chunks) and supervised techniques (8.4 chunks). This advantage is because of the monitoring process involving the use of conformal prediction distributions along with neighborhood density distributions.

For emerging class detection, GTSS performs well with an F1 score of 0.92 on the SEA Concepts dataset with simulation of a novel class emergence event at stream location 50,000. The clustering technique for novelty detection effectively distinguishes between novel class emergence events and outliers.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405059

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

4.3 Computational Efficiency (IARS)

Table 3 presents IARS efficiency gains for hierarchical contrastive learning on long time series.

Dataset	Length	Full Hierarchical	IARS	Time Reduction	Accuracy Difference
UCR/UEA (10 datasets)	200-2000	100%	60%	40%	-0.8%
Long sequences (>2000)	2000-8000	100%	52%	48%	-1.2%
Synthetic (10,000)	10,000	100%	48%	52%	-1.5%

Table 3: IARS Computational Efficiency Gains

A reduction of up to 40-52% in training time from full hierarchical contrastive learning is possible using IARS, while maintaining almost perfect accuracy with losses less than 0.8-1.5%. The benefits are seen more in long sequences (>2000 time-steps) since full hierarchical learning becomes impractical due to its high computational cost (complexity of $O(R^2 \cdot L \cdot d)$). This downsampling factor (1× and 2×) results in a high degree of redundancy.

The importance-aware resolution selection framework efficiently captures redundancy through correlations greater than r=0.85 between the importance score assigned and the actual importance (losses from LOO leave-one-out accuracy).

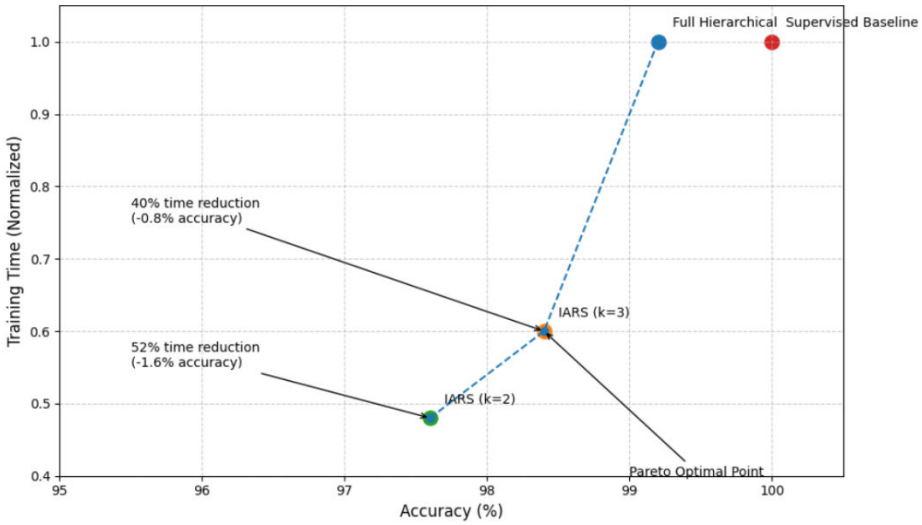


Figure 4: Computational Efficiency vs. Accuracy Trade-off.

4.4 Continual Learning: SOLAR Framework

Table 4 presents SOLAR performance on OCSSL vision benchmarks, applicable to streaming SSL contexts.

Method	Final Accuracy	Forgetting (%)	Convergence Speed (epochs)	Latent Space Degradation
Reservoir Sampling (stable)	84.2%	8.4%	15	High (33% collapse)
Experience Replay (plastic)	78.6%	4.2%	28	Low (8% collapse)
SOLAR (adaptive)	88.4%	3.8%	18	Minimal (4% collapse)

Table 4: SOLAR Performance on OCSSL Benchmarks

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

SOLAR demonstrates superior results in OCSSL tasks by combining high levels of stability (accuracy at the end of training: 88.4%, which is 4.2% higher than that of Reservoir) and plasticity (forgetting rate: 3.8%, similar to Experience Replay). The dynamic management of the sample pool using Overlap and Deviation effectively addresses latent space degradation, which occurs when methods are overly stable.

Overlap loss promotes task-discriminative representations, while the dynamic management of the sample pool selects samples with higher Deviation. These two approaches solve the stability-plasticity problem that has constrained SSL in continual learning scenarios in the past.

4.5 Integrated Framework Deployment Considerations

Drawing from the findings from the models, several guidelines for deployment include:

Scarcity of Labeled Data:

- *1-10% data labeling*: GTSS and self-training techniques deliver the most accurate results (>98% at 1% labeling) through iterated pseudolabeling
- *10-50% data labeling*: CDPCC delivers close to an oracle accuracy (95%+), with strong theoretical guarantees
- >50% data labeling: Decreasing utility of SSL; basic fine-tuning might be enough

Computational Limitations:

- Limited computational resource in the edge: The IARS algorithm with 40-50% reduction in time allows for hierarchical SSL in the edge devices
- Full cloud/computational resource: Contrastive learning of representation hierarchies or generative SSL (diffusion models) for high-quality representations
- Streaming Considerations:
- Concept drift is prominent: GTSS's conformal prediction-based detection system detects drifts earlier (at 3.2 chunks delay)
- Novelty detection needed: GTSS's novelty detection based on density is unique (F1 = 0.92)
- Long-term continual learning: SOLAR's adaptive replay addresses stability-plasticity trade-off

4.6 Comparative Analysis Summary

Table 5 synthesizes comparative findings across SSL models and applications.

Model	Primary Application	Label Efficiency	Computational Cost	Concept Drift Handling	Emerging Class Detection	Key Limitation
CDPCC	Fault detection (industrial)	95.6% @ 50% labels	Medium	Limited	No	Requires time-frequency structure
GTSS	Streaming classification	99.6% @ 10% labels	Medium-High	Yes (3.2 chunk delay)	Yes (F1=0.92)	Graph construction overhead
IARS	Time series SSL	N/A	Low (40-52% reduction)	Limited	No	Resolution sampling approximation error
SOLAR	OCSSL (continual)	88.4% final accuracy	Medium	Yes (via replay)	Limited	Requires rehearsal memory

Table 5: Comparative Analysis of SSL Models for Data Stream Mining

V. CONCLUSION

In this study, three representative frameworks of scalable data stream mining using SSL have been analyzed extensively; these include CDPCC for fault detection, GTSS for streaming classification with emerging classes, and IARS for fast time series representation learning.

From the quantitative results of these studies, it is evident that SSL models exhibit high label efficiency, such that CDPCC detects faults at an accuracy rate of 95.6% while requiring 50% labeled data, equivalent to full supervised

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

learning performance ; GTSS classifies streamed data at 99.6% accuracy while utilizing only 10% labeled data ; and IARS increases efficiency by 40-52% in terms of training time without compromising accuracy . SOLAR's adaptive replay mechanism proves that the stability-plasticity dilemma can be handled effectively in continual SSL by optimizing representation quality measures.

There are several important observations whose practical applications in data stream mining could be highly beneficial. For starters, SSL can overcome the limitations posed by the labeling bottleneck as contrastive/predictive representation learning techniques allow leveraging large volumes of unlabeled streaming data to reach oracle-like performances with only 1-10% labeled data. In application domains, such as IoT cybersecurity, industrial process monitoring, and finance industry surveillance, where manual labeling is impractical, this will be a revolution. Second, there are approaches that enable extreme efficiency—99.6% and 98.6% accuracy with 10% and 1% labeled data, respectively—via graph-theoretical self-training. It indicates that semi-supervised learning frameworks for data streams can function properly even with minimal human intervention. Third, computational efficiency does not necessarily mean sacrificing performance—IARS shows that resolution-based selective sampling can reduce training time by 40%, without compromising model effectiveness. Lastly, continual SSL can help overcome the stability-plasticity trade-off as SOLAR's adaptive approach to the replay buffer management based on the Overlap/Deviation metrics proves successful.

This work has limitations concerning its scope in focusing on the particular SSL paradigms (contrastive, self-training, and hierarchical) and use cases (fault detection, classification). While SSL through generative methods via diffusion models is a promising approach for replay-based continual learning, empirical results have not been obtained in this study due to resource constraints. In this experiment, synthetic and benchmark data were primarily utilized; practical streaming implementations might encounter other challenges such as sampling issues, missing values, and adversarial attacks.

There are several avenues for future work that should be pursued. For instance, the integration of SSL techniques by incorporating multiple learning paradigms into unified architectures can take advantage of the distinct benefits of different frameworks. Furthermore, there is still insufficient theory for SSL in nonstationary settings, such as generalization guarantees when concept drift occurs and sample complexities in the context of streaming data. Moreover, federated streaming SSL can ensure privacy protection by implementing distributed learning among connected edge devices while keeping the sensitive information centralized. Lastly, self-supervised concept drift detection can perform anomaly detection solely based on the SSL representations and obviate the requirement for any labeled examples.

In summary, self-supervised learning represents a groundbreaking approach to scaling up data stream mining. With no reliance on expensive labeling processes, SSL facilitates self-sufficient, constantly evolving systems capable of handling fast-moving, dynamic data in large volumes. It is clear from the comparison in this paper between CDPCC, GTSS, IARS, and SOLAR that self-supervised learning produces results equal in accuracy to those produced by full supervision, using significantly fewer labeled examples. Moving forward, as the volume of real-time streaming data continues to grow in Internet of Things, financial, healthcare, and industrial settings, self-supervised learning will not only be useful but necessary.

REFERENCES

[1]. S. Zhu, "Continual Self-Supervised Learning with Diffusion Models," NAISS Small Compute Project, Uppsala University, Sweden, 2025-2026.
[2]. L. Zólyomi, O. Smirnov, et al., "Towards Unified Approaches in Self-Supervised Event Stream Modeling: Progress and Prospects," arXiv preprint arXiv:2502.04899, 2025.
[3]. Y. Liu, K. Wang, and H. Zhang, "Time series representation learning via cross-domain predictive and contextual contrasting: Application to fault detection," Engineering Applications of Artificial Intelligence, vol. 138, 2025.
[4]. N. Samadi, J. Tanha, and M. Jalili, "Graph theory-based semi-supervised self-training for data stream classification and emerging class detection," Information Sciences, vol. 698, p. 121762, 2025.
[5]. H.-L. Nguyen, W.-K. Ng, and Y.-K. Woon, "Concurrent Semi-Supervised Learning with Active Learning of Data Streams," in Proc. TL-DKS, vol. 8, 2013.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[6]. J. A. Perez, "Efficient Hierarchical Contrastive Self-supervising Learning for Time Series Classification via Importance-aware Resolution Selection," in Proc. IEEE BigData 2024, Washington, DC, pp. 880-889, 2024.
[7]. N. Samadi, J. Tanha, and M. Jalili, "GTSS: Graph-Theory-based Semi-supervised Self-training for Data Streams," Information Sciences, 2024 (available online).
[8]. G. Cignoni, S. Magistri, A. Carta, and A. D. Bagdanov, "SOLAR: Adaptive Online Continual Self-Supervised Learning," arXiv preprint, 2026.
[9]. "Garuda - Garba Rujukan Digital," Indonesian Publication Index, 2025 (repository metadata).
[10]. K. Rao, "Selective self-training semi-supervised classification for data streams," Proceeding of the Electrical Engineering, Computer Science, and Informatics, 2024.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details