Fuzzy Shared Representation Learning for Multistream Classification

En Yu, Jie Lu *IEEE Fellow*, Guangquan Zhang

***Abstract*—Multistream classification aims to predict the target stream by transferring knowledge from labeled source streams amid non-stationary processes with concept drifts. While existing methods address label scarcity, covariate shift and asynchronous concept drift, they focus solely on the original feature space, neglecting the influence of redundant or low-quality features with uncertainties. Therefore, the advancement of this task is still challenged by how to: 1) ensure guaranteed joint representations of different streams, 2) grapple with uncertainty and interpretability during knowledge transfer, and 3) track and adapt the asynchronous drifts in each stream. To address these challenges, we propose an interpretable Fuzzy Shared Representation Learning (FSRL) method based on the Takagi-Sugeno-Kang (TSK) fuzzy system. Specifically, FSRL accomplishes the non-linear transformation of individual streams by learning the fuzzy mapping with the antecedents of the TSK fuzzy system, thereby effectively preserving discriminative information for each original stream in an interpretable way. Then, a multistream joint distribution adaptation algorithm is proposed to optimize the consequent part of the TSK fuzzy system, which learns the final fuzzy shared representations for different streams. Hence, this method concurrently investigates both the commonalities across streams and the distinctive information within each stream. Following that, window-based and GMM-based online adaptation strategies are designed to address the asynchronous drifts over time. The former can directly demonstrate the effectiveness of FSRL in knowledge transfer across multiple streams, while the GMM-based method offers an informed way to overcome the asynchronous drift problem by integrating drift detection and adaptation. Finally, extensive experiments on several synthetic and real-world benchmarks with concept drift demonstrate the proposed method's effectiveness and efficiency.**

***Index Terms*—Concept Drift, Multistream Classification, Transfer Learning, Fuzzy Systems**

I. Introduction

In machine learning, there is a common assumption that optimal model performance is contingent upon the training and test datasets adhering to identical distributions, thereby enabling the model to generalize effectively. However, in real-world applications, various data are always generated continuously and sequentially with unpredictable changes in their underlying distribution. The phenomenon is referred to as concept drift [1], which results in a decline in the performance of models trained on historical data once drift occurs. Consequently, researchers have shown significant interest in devising effective learning techniques to analyze streaming data in non-stationary environments. The objective of this research is to address the challenges posed by concept drift and enhance the model's adaptability to the continuously evolving data distributions in dynamic real-world environments.

Existing studies have provided empirical evidence regarding the efficacy of concept drift adaptation methods for handling dynamic distributions in data streams [2]–[5]. Research in this domain predominantly falls into two categories: informed and blind. Informed methods typically leverage supervised or unsupervised drift detection strategies to dynamically monitor data streams. Upon detecting a drift, an adaptation mechanism will be triggered, enabling the model to quickly adapt to the new concept [6], [7]. In contrast, blind methods continuously update the model with new incoming data without explicitly employing drift detection mechanisms [8], [9]. It's important to note, however, that a considerable number of these approaches are specifically designed for single-stream scenarios with delayed labels.

In the realm of advanced intelligent systems, it is quite typical to encounter the concurrent generation of multiple data streams simultaneously. For example, in a healthcare monitoring system, multiple data streams like patient vital signs, electronic health records, and wearable device data vary over time but interact to facilitate critical medical decisions. These various data streams enable personalized patient care and real-time health assessments, despite exhibiting distinct distributions due to varying data sources [10], [11]. In addition, the labeling of this multifaceted data, essential for facilitating critical medical decisions and enabling personalized patient care, presents significant challenges in terms of time and labor costs. This has led to the emergence of hybrid data environments, where voluminous streams of both labeled and unlabeled data are processed concurrently [12].

To address this situation, multistream classification has been introduced, involving both labeled and unlabeled data streams with concept drifts [13]–[15]. This task aims to predict the labels of the target stream by transferring knowledge from one or multiple labeled source streams, while also handling the concept drift problem. Generally, there are three common challenges in this task [16]: *1) Label scarcity*, which refers to the lack of labels for the target data stream, unlike the source streams that have labeled data; *2) Covariate shift*, which indicates that each data stream exhibits a unique distribution, differing from others and *3) Asynchronous concept drift*, which occurs when source and target streams independently

The work was supported by the Australian Research Council (ARC) under Laureate project FL190100149 and discovery project DP200100700.

En Yu, Jie Lu, and Guangquan Zhang are with the Decision Systems and e-Service Intelligence Laboratory, Australian Artificial Intelligence Institute (AAII), Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: En.Yu@student.uts.edu.au; Jie.Lu@uts.edu.au; Guangquan.Zhang@uts.edu.au.)

Corresponding author: Jie Lu

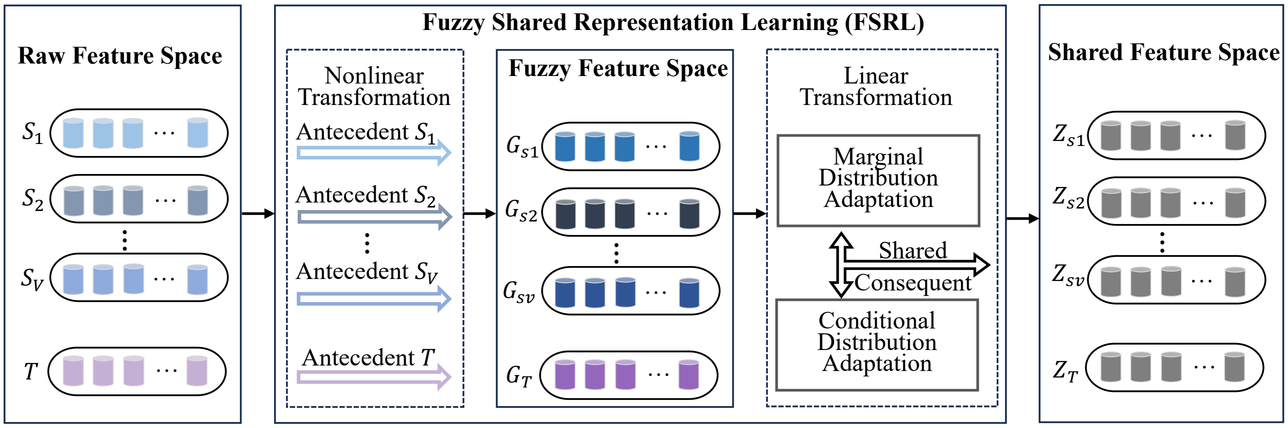


Fig. 1: The high-level illustration of FSRL. It first collects data batches from all data streams and constructs a fuzzy mapping with the antecedent part of the TSK fuzzy system individually to realize nonlinear transformation. Then, the consequent components are optimized by aligning the conditional and marginal distributions simultaneously.

experience concept drift at different times, uniquely impacting the model's performance. In recent years, some methods have been proposed to address these challenges by integrating online domain adaptation and drift-handling techniques. Most of them predominantly concentrate on single-source streams, which can hamper model performance due to the constraints in source data quality. Additionally, these approaches, being reliant on a single source, often face the risk of overfitting. To address these issues, a multi-source configuration has been introduced, which taps into diverse source streams to gather supplementary data, thereby enriching the model's accuracy and robustness [8], [17].

However, transferring knowledge among multiple non-stationary data streams presents additional challenges that are not considered in current works. Firstly, most current methodologies construct classifiers based on information from the original feature space, overlooking the impact of redundant or low-quality features, which can detrimentally affect the final decision-making process [12]. Therefore, a method **to exploit guaranteed joint representations of different streams** becomes particularly crucial during knowledge transfer. Secondly, various data streams with asynchronous drifts inevitably contain inherent uncertainties and dynamic relationships. Consequently, a method **to address these uncertainties and provide a reasonable interpretability analysis** is also a crucial focus of this task.

Since fuzzy systems have shown robust learning capabilities and transparent interpretability for various applications [18]–[20], we propose a novel Fuzzy Shared Representation Learning (FSRL) method based on TSK fuzzy system to address the newly exposed challenges. TSK fuzzy system is a data-driven system comprised of multiple IF-THEN fuzzy rules, offering high interpretability. It can learn model parameters in a data-driven manner similar to other machine learning methods. The output of a TSK fuzzy system is based on linear functions of the inputs, providing a more flexible way to learn shared representations. This flexibility allows for better adaptation to various tasks and data types. Many methods based on TSK fuzzy systems have been proposed to enhance interpretability and address uncertainty in transfer learning, which also provides a guaranteed foundation for the design of my model [21]–[23]. Therefore, FSRL employs a multioutput TSK fuzzy system as a transformation method to learn a shared fuzzy space. Specifically, as shown in Figure 1, we collect a data batch from each data stream and construct a fuzzy mapping with the antecedent part of the TSK fuzzy system individually to realize nonlinear transformation. It preserves discriminative information for each original stream in an interpretable way. Then, an advanced method is proposed to learn the consequent parameters of the TSK fuzzy system by considering all data streams simultaneously. Following that, blind window-based (BFSRL) and informed GMM-based (IFSRL) adaptation strategies are designed to address the various drifting situations over time. Specifically, BFSRL utilizes a fixed-size sliding window to aggregate incoming data, implementing FSRL at each step. This method not only effectively demonstrates FSRL's capability in facilitating knowledge transfer across multiple streams but also provides a direct solution to the challenges in multistream classification. However, the continuous variation in the frequency and types of data drifting in different datasets makes it sensitive to window size.

To enhance the robustness and generalization, we further introduce the IFSRL, which integrates drift detection and adaptation mechanisms, providing a more nuanced response to the concept drift problem. As shown in Figure 2, we initially train a Gaussian Mixture Model (GMM) [24], [25] using collected data, which is employed to estimate the conditional distributional relationship between new data and current data. Additionally, we employ the Drift Detection Method (DDM) [26] to detect the drift in each labeled source stream, as it offers a stable and accurate detection approach. Simultaneously, we utilize the weighted ensemble strategy for asynchronous drift adaptation leveraging the conditional distributions derived from the GMM. For the unlabeled target stream, we design two sliding windows and continuously monitor their distribution

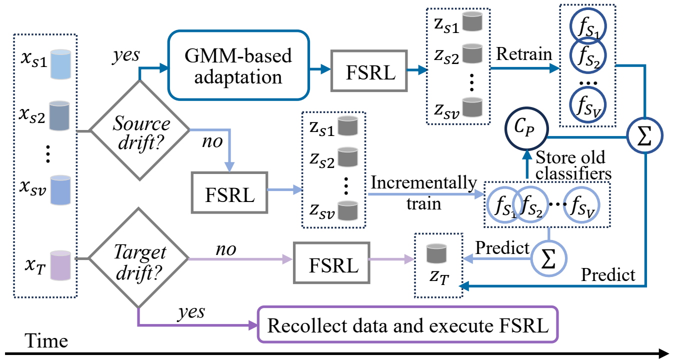


Fig. 2: The IFSRL framework involves using DDM for drift detection in source streams. New samples are mapped to the learned fuzzy shared space for incremental base classifier training when there is no drift. If drift occurs, the GMM-based adaptation module adjusts the data and retrains a new base classifier, while preserving old base classifiers in a pool for retaining previous knowledge. Drift in the target stream is monitored using two sliding windows based on the mean conditional distribution, triggering data re-collection and new fuzzy shared space learning once target drift is detected.

changes to detect drift based on the mean conditional distribution effectively. When drift is detected in the target stream, it indicates that the previously established fuzzy shared space, learned through FSRL, is no longer valid. This necessitates the re-collection of data and the learning of a new fuzzy shared space, ensuring our model's adaptability and relevance in the face of evolving data characteristics. The main contributions of this work are summarized as follows,

- This paper firstly introduces the TSK-FS into multistream classification under concept drift and proposes a novel FSRL method to learn a shared fuzzy feature space. It not only addresses uncertainties and interpretability during knowledge transfer, but also enhances model adaptability and robustness in the shared fuzzy space.
- An advanced optimization method for multistream joint distribution adaptation is proposed to learn the consequent parameters of the TSK fuzzy system. This addresses inter-stream shifts, simultaneously mitigating the impact of redundant features.
- We design an online blind window-based adaptation method that utilizes a fixed-size sliding window to aggregate incoming data, implementing FSRL at each step. This approach provides a straightforward demonstration of FSRL's effectiveness in facilitating knowledge transfer across multiple streams.
- A novel informed online adaptation method is proposed, integrating drift detection and GMM-based adaptation mechanisms to solve the asynchronous drift problem, thus providing a more general and robust solution to multistream classification.

II. Related Work

This section offers a comprehensive survey of the literature related to our research. We first delve into the fundamental definition of concept drift and popular concept drift adaptation methods designed for single-stream scenarios. Subsequently, we provide an overview of existing research on the multistream classification task, highlighting the shortcomings of current approaches. Finally, we introduce the definitions, preliminaries, and prevalent optimization methods associated with TSK fuzzy systems.

A. Concept drift

The field of data stream classification has garnered significant attention in research, primarily due to the dynamic nature of real-world data streams characterized by the phenomenon of concept drift. Concept drift is delineated as a shift in the data distribution over time, observable when the joint distribution at time $t+1$, $P_{t+1}(X, y) \neq P_t(X, y)$ [1]. This shift poses formidable challenges to maintaining classifier accuracy and ensuring rapid adaptability. In response to these challenges, various strategies have been developed to enhance model effectiveness and reliability by employing concept drift adaptation strategies [27]. For example, Window-based approaches stand out prominently, such as DDM [26], Adaptive Windowing (ADWIN) [28], Dynamic Extreme Learning Machine (DELM) [29] and so on. These methods operate by monitoring changes in data statistics or prediction errors across different data windows, adapting to new concepts upon detecting drift. Additionaly, Instance-based lazy learning methodologies, including Just-In-Time adaptive classification (JIT) [30] and Stepwise Redundancy Removal (SRR) [31], have found extensive application. In addition, ensemble learning mechanisms have been integrated into this domain, such as Adaptive Random Forest (ARF) [27], Dynamic Weighted Majority (DWM) [32], Learn++NES [33]. These diverse approaches collectively contribute to the arsenal of techniques available for handling concept drift in dynamic data stream scenarios. However, these approaches are specifically designed for single stream with delayed labels, which can not be used for multistream scenarios.

B. Multistream classification

To address the multistream problem, Chandra et al. [13] introduced a multistream classification framework using ensemble classifiers, enhanced by Kernel Mean Matching. The FUSION algorithm [16] further improves this approach with effective density ratio estimation. Neural network-based models like Autonomous Transfer Learning (ATL) [34] and meta-learning frameworks [35], [36] provide additional solutions for handling drifting data streams. However, only depending on one single stream may degrade the model performance due to the constraints in source data quality.

To bolster robustness, multisource stream classification seeks to leverage synergistic information from diverse source streams. Du et al. [37] introduced Melanie, a notable approach that utilizes a weighted ensemble classifier for knowledge transfer across multiple sources, pioneering the handling of concept drift from various streams simultaneously. However, it is essential to note that Melanie is constrained to supervised scenarios and is not applicable to unlabeled data.

This highlights the need for further advancements in accommodating diverse data types and scenarios in multi-source stream classification. Furthermore, AutOmatic Multi-Source Domain Adaptation (AOMSDA) [9] introduces a central moment discrepancy-based regularizer, facilitating information integration across multi-source streams and addressing covariate shifts through a node weighting strategy in an unsupervised way. Despite its efficacy on a chunk basis, AOMSDA lacks dynamic stream change detection. In response to this limitation, Jiao et al. [12] propose a reduced-space Multistream Classification based on Multi-objective Optimization (MCMO). This approach aims to identify common feature subsets for minimizing distribution shifts and incorporates a GMM for drift adaptation. However, these methods still fail to leverage guaranteed joint representations of different streams and address uncertainties. Moreover, they do not provide a reasonable interpretability analysis for knowledge transfering in a dynamic environment. Further research is warranted to enhance the understanding and interpretability of model decisions in evolving multistream scenarios.

C. Takagi-Sugeno-Kang fuzzy system

The TSK fuzzy system [38], an advanced model within the realm of fuzzy logic, stands out for its data-driven methodology and application of IF-THEN fuzzy rules [23]. This approach facilitates the construction of models that are not only robust in learning from complex data patterns but also excel in providing transparent and interpretable results. The versatility of the TSK fuzzy system is evidenced by its successful deployment across a wide range of fields, from control systems to pattern recognition and beyond. Specifically, it can be can be formulated based on "IF-THEN" rules as follows:

$$\begin{aligned} &\mathrm{IF}: x_1 \text{ is } A_1^k \wedge x_2 \text{ is } A_2^k \wedge \cdots \wedge x_d \text{ is } A_d^k \\ &THEN: f^k(\mathbf{x}) = p_0^k + p_1^k x_1 + \cdots + p_d^k x_d, \end{aligned} \tag{1}$$

where $k \in \{1, 2, \ldots, K\}$ denotes the index of the k-th rule. Let $\mathbf{x} \in \mathbb{R}^{d\times 1}$ represent the sample vector, where d is the dimensionality of the samples. The function $f^k(\mathbf{x})$ signifies the output of the k-th rule. Additionally, A_i^k denotes the fuzzy set associated with the i-th feature under the k-th rule, and $\wedge$ symbolizes a fuzzy conjunction operator.

In contrast to crisp sets where membership values are strictly confined to either 0 or 1, fuzzy sets allow for membership degrees ranging continuously from 0 to 1. A frequently employed representation of fuzzy membership is through the Gaussian function, which is formulated as:

$$\mu_{A_i^k}(x_i) = \exp\left(-(x_i - c_i^k)/2\delta_i^k\right)^2, \tag{2}$$

where parameters c_i^k and δ_i^k represent the center and width of the Gaussian function and they are also the antecedent parameters of the TSK fuzzy system. The estimation of these parameters can be effectively achieved through various methodologies, such as the Fuzzy C-Means Clustering (FCM) [21] and Var-Part [39].

Once the antecedent parameters are established, it becomes feasible to compute the membership value for each feature in the specific fuzzy set A_i^k, as delineated in Eq.(2). When multiplication serves as the conjunction operator, the firing level of each sample's k-th rule can be determined via Eq.(3a). Eq.(3b) presents the normalized version of this calculation. Further, the output of the TSK fuzzy system is represented as the weighted mean of $f_k(\mathbf{x})$, as elucidated in Eq.(3c).

$$\mu^k(\mathbf{x}) = \prod_{i=1}^{d} \mu_{A_i^k}(x_i), \tag{3a}$$

$$\tilde{\mu}^k(\mathbf{x}) = \mu^k(\mathbf{x}) \Bigg/ \sum_{k'=1}^{K} \mu^{k'}(\mathbf{x}), \tag{3b}$$

$$f(\mathbf{x}) = \sum_{k=1}^{K} \tilde{\mu}^k(\mathbf{x}) f^k(\mathbf{x}). \tag{3c}$$

Upon acquiring the antecedent parameters, the expression of the TSK fuzzy system's output, as indicated in Eq.(3c), can be represented as linear regression, as delineated in Eq.(4).

$$y = f(\mathbf{x}) = \mathbf{p}_g^{\mathrm{T}} \mathbf{x}_g, \tag{4}$$

where

$$\mathbf{x}_e = [1, \mathbf{x}^{\mathrm{T}}]^{\mathrm{T}} \in R^{(d+1)\times 1}, \tag{5a}$$

$$\tilde{\mathbf{x}}^k = \tilde{\mu}^k(\mathbf{x})\mathbf{x}_e \in R^{(d+1)\times 1}, \tag{5b}$$

$$\mathbf{x}_g = [(\tilde{\mathbf{x}}^1)^{\mathrm{T}}, (\tilde{\mathbf{x}}^2)^{\mathrm{T}}, \ldots, (\tilde{\mathbf{x}}^K)^{\mathrm{T}}]^{\mathrm{T}} \in R^{K(d+1)\times 1}, \tag{5c}$$

$$\mathbf{p}^k = [p_0^k, p_1^k, \cdots, p_d^k]^{\mathrm{T}} \in R^{(d+1)\times 1}, \tag{5d}$$

and

$$\mathbf{p}_g = [(\mathbf{p}^1)^{\mathrm{T}}, (\mathbf{p}^2)^{\mathrm{T}}, \ldots, (\mathbf{p}^K)^{\mathrm{T}}]^{\mathrm{T}} \in R^{K(d+1)\times 1}. \tag{6}$$

III. PROPOSED METHOD

In this section, we first define the multisource stream classification problem and analyze the challenges inherent in this task, as well as its objectives. Then, we provide an overview of our proposed Fuzzy Shared Representation Learning (FSRL) method. Subsequently, we delve into the detailed description of the FSRL algorithm and the optimization strategy. Furthermore, we discuss in depth two online adaptation strategies designed to handle different drifting scenarios over time, i.e., blind window-based and informed GMM-based approaches.

A. Problem definition and overall framework

Multisource stream classification is an online process involving V distinct labeled source streams $S_v, v \in [1, V]$ and a single target stream T with *scarcity of labels*. Denoting P_{S_v} and P_T as the distributions from S_v and T. Each arrived data sample from source streams at time i is represented by $\mathbf{x}_{S_{vi}} \in \mathcal{D}^d$ with true label $y_{S_{vi}}$, while only d-dimensional features $\mathbf{x}_{T_i}$ of target stream can be obtained. All streams at the same time step i are related but with *joint domain shift*, i.e., 1) marginal shift $P_{S_v}(\mathbf{x}_{S_{vi}}) \neq P_{S_{v^*}}(\mathbf{x}_{S_{v^*i}}) \neq P_T(\mathbf{x}_{T_i})$ and 2) conditional shift $Q_{S_v}(y_{S_{vi}} \mid \mathbf{x}_{S_{vi}}) \neq Q_{S_{v^*}}(y_{S_{v^*i}} \mid \mathbf{x}_{S_{v^*i}}) \neq Q_T(y_{T_i} \mid \mathbf{x}_{T_i})$ and . In addition, another challenging issue of this setup is the *asynchronous concept drift* over time, which can manifest in three primary scenarios:

- Source Drift: $\exists\, i$ if $P_{S_v}(\mathbf{x}_{S_{vi}}) \neq P_{S_v}(\mathbf{x}_{S_{v(i+1)}}), v \in [1, V]$ but $P_T(\mathbf{x}_{T_i}) = P_T(\mathbf{x}_{T_{(i+1)}})$, the drift only occurs in the source stream.

Algorithm 1 The learning process of FSRL

Input: Input data $\{(\mathbf{x}_{S_{vi}}, y_{S_{vi}})\}_{i=1}^{n}$, $\{\mathbf{x}_{T_i}\}_{i=1}^{n}$; the number of fuzzy rules K; hyperparameters λ

Output: Antecedent parameters of the TSK fuzzy systems; consequent transformation matrix $\mathbf{P}$.

1: Estimate the antecedent parameters of the TSK fuzzy systems by using the Var-Part clustering algorithm for each stream.
2: Construct the new fuzzy representations $\{\mathbf{g}_{S_{v_i}}\}_{i=1}^{n}$ and $\{\mathbf{g}_{T_i}\}_{i=1}^{n}$ generated by fuzzy rules for each stream by Eq. (5a)-(5c).
3: Construct MMD matrix $\mathbf{M}^0$ by Eq.(12)
4: **repeat**
5: Solve the generalized eigendecomposition problem in Eq.(17) and select the m smallest eigenvectors to construct $\mathbf{P}$.
6: Train a classifier f using new fuzzy representations $\{(\mathbf{z}_{S_{v_i}}, y_{S_{vi}})\}_{i=1}^{N}$ to update pseudo target labels.
7: Construct MMD matrices $\mathbf{M}^c$ by Eq.(14).
8: **until** Convergence

- Target Drift: $\exists\, i$ if $P_{Sv}(\mathbf{x}_{S_{vi}}) = P_{Sv}(\mathbf{x}_{S_{v(i+1)}}), v \in [1, V]$ but $P_T(\mathbf{x}_{T_i}) \neq P_T(\mathbf{x}_{T_{(i+1)}})$, the drift only occurs in the target stream.
- Concurrent Drifts: $\exists\, i$ if $P_{Sv}(\mathbf{x}_{S_{vi}}) \neq P_{Sv}(\mathbf{x}_{S_{v(i+1)}}), v \in [1, V]$ and $P_T(\mathbf{x}_{T_i}) \neq P_T(\mathbf{x}_{T_{(i+1)}})$, it means drift occurs in both source and target streams.

Our primary goal is to precisely predict the labels of the target stream by adeptly harnessing the insights gleaned from the labeled source streams. This entails not only utilizing the available knowledge from the source streams but also adeptly navigating and adapting to various drifting scenarios. Furthermore, our attention is centered on developing methods to secure more robust and dependable joint representations in the midst of transferring knowledge across diverse streams. Simultaneously, we aim to identify and implement strategies to circumvent potential uncertainties, thereby enhancing the overall efficacy and reliability of our approach. This focus is integral to advancing the precision and effectiveness of our methods within the context of dynamic stream environments.

Therefore, we propose the FSRL method based on the multioutput TSK fuzzy system. It learns a common feature subspace for various data streams, in which both the unique discriminative information of each original stream and the distribution-adapted common features are simultaneously preserved and emphasized. Furthermore, FSRL delves into deciphering nonlinear interactions among multiple data streams via the rule-based TSK fuzzy system. It facilitates the achievement of promising performance in terms of robustness and interpretability. Following that, blind window-based (BFSRL) and informed GMM-based (IFSRL) online adaptation strategies are designed to address the various drifting situations over time based on the learned common features.

B. Fuzzy Shared Representation Learning

As previously discussed, prevalent approaches in multi-stream classification predominantly rely on the original feature space for constructing classifiers. This often leads to the inadvertent inclusion of redundant or low-quality features, potentially compromising the efficacy of the decision-making process [12]. To address this issue, our methodology involves the creation of a fuzzy shared space, which encompasses a two-step process for deriving fuzzy shared representations. This process comprises an initial nonlinear transformation, followed by a linear dimensionality reduction.

In our proposed method, the nonlinear transformation within the multioutput TSK fuzzy system is facilitated by its antecedent parameters. For this purpose, the Gaussian function is employed as the membership function, where the antecedent parameters are specifically designed to represent the center and the width of the Gaussian curve. Following this, the Var-Part clustering method [39] is applied to estimate the clusters $\mathbf{C_{Sv}}$ and $\mathbf{C_T}$, corresponding to each source and target stream, respectively. Subsequently, the kernel width matrices $\mathbf{D_{Sv}}$ and $\mathbf{D_T}$ are derived, utilizing a procedure analogous to that used in the FCM [21] approach as follows:

$$(\mathbf{D}_{Sv})_p^k = \sum_{i=1}^{n_{Sv}} \left(x_{S_{vip}} - (\mathbf{C}_{Sv})_p^k \right)^2, v = 1, 2, \cdots, V \tag{7a}$$

$$(\mathbf{D}_{T})_p^k = \sum_{i=1}^{n_{T}} \left(x_{T_{ip}} - (\mathbf{C}_{T})_p^k \right)^2, \tag{7b}$$

where $k = 1, 2, \ldots, K$ represents the total number of rules, while $p = 1, 2, \ldots, d$ denotes the dimension of samples. n_{Sv} and n_T signify the respective numbers of samples within each source and target stream. Notably, the estimation of clusters and kernel widths is conducted individually across different streams. This approach ensures that the discriminative information pertinent to each original stream is not only preserved but also rendered in an interpretable manner.

Once all the antecedent parameters of the TSK fuzzy system are established, any sample ($\mathbf{x}_{S_{vi}}$ or $\mathbf{x}_{Ti}$) from either the source stream or the target stream is first mapped into a stream-specific fuzzy space via Eq. (5a)–(5c). The new representations of the source and target samples in the fuzzy space can be formulated as follows:

$$\mathbf{g}_{S_{vi}} = [(\tilde{\mathbf{x}}_{S_{vi}}^1)^{\mathrm{T}}, (\tilde{\mathbf{x}}_{S_{vi}}^2)^{\mathrm{T}}, \ldots, (\tilde{\mathbf{x}}_{S_{vi}}^K)^{\mathrm{T}}]^{\mathrm{T}} \in R^{K(d+1)\times 1}, \quad v = 1, 2, \cdots, V \tag{8a}$$

$$\mathbf{g}_{T_i} = [(\tilde{\mathbf{x}}_{T_i}^1)^{\mathrm{T}}, (\tilde{\mathbf{x}}_{T_i}^2)^{\mathrm{T}}, \ldots, (\tilde{\mathbf{x}}_{T_i}^K)^{\mathrm{T}}]^{\mathrm{T}} \in R^{K(d+1)\times 1}. \tag{8b}$$

Subsequently, the consequent components of the TSK fuzzy system are employed as the transformation function $\phi(*)$. This function is instrumental in facilitating linear dimensionality reduction within the resultant shared fuzzy feature space, which is formulated as follows:

$$\phi(\mathbf{x}_{S_{vi}}) = \mathbf{z}_{S_{v_i}} = \mathbf{P}^T \mathbf{g}_{S_{vi}}, v = 1, 2, \ldots V, \tag{9a}$$

$$\phi(\mathbf{x}_{T_i}) = \mathbf{z}_{T_i} = \mathbf{P}^T \mathbf{g}_{T_i}, \tag{9b}$$

$$\mathbf{P} = [\mathbf{p}_g^1, \mathbf{p}_g^2, \ldots, \mathbf{p}_g^m] \in R^{K(d+1)\times m}. \tag{9c}$$

In this study, distinct from the design for antecedent parameters, we postulate that the consequent parameters across different streams are shared. This assumption aids in streamlining the feature set, thereby enhancing computational efficiency and improving the overall performance of the model. Additionally, it facilitates the discovery of common representations among various data streams.

Due to the presence of joint domain shifts among different data streams, it becomes imperative to consider this factor while learning the consequent parameters. This consideration aims to minimize the distribution differences: i.e., 1) marginal distribution $P_{Sv}(\mathbf{x}_{S_{vi}})$ and $P_T(\mathbf{x}_{T_i})$, and 2) conditional distribution $Q_{Sv}(y_{S_{vi}} \mid \mathbf{x}_{S_{vi}})$ and $Q_T(y_{T_i} \mid \mathbf{x}_{T_i})$, in the final shared fuzzy space. Therefore, we further propose a multistream joint distribution adaptation method in our study. Given the transformation function $\phi(*)$, it aims at matching the joint expectations between each source stream and target stream,

$$\begin{aligned}&\min_{\phi} \sum_{v=1}^{V} \left(\|\mathbb{E}_P[\phi(\mathbf{x}_{S_{vi}}), y_{Svi}] \ -\mathbb{E}_P\left[\phi(\mathbf{x}_{T_i}), y_{Ti}\right]\|^2\right) \\ &\approx \sum_{v=1}^{V} \left(\|\mathbb{E}_{P_{Svi}}\left[\phi(\mathbf{x}_{S_{vi}})\right] \ -\mathbb{E}_{P_T}\left[\phi(\mathbf{x}_{T_i})\right]\|^2\right. \\ &\left.+ \|\mathbb{E}_{Q_{Si}}\left[y_{Svi} \mid \phi(\mathbf{x}_{S_{vi}})\right] \ -\mathbb{E}_{Q_T}\left[y_{Ti} \mid \phi(\mathbf{x}_{T_i})\right]\|^2\right).\end{aligned} \tag{10}$$

1) Marginal distribution adaptation: To align marginal distributions between source streams and target stream, we regard the Maximum Mean Discrepancy (MMD) [40] as the distance:

$$\begin{aligned}&\sum_{v=1}^{V}\left(\left\|\frac{1}{n_{S_v}}\sum_{i=1}^{n_{S_v}}\mathbf{P}^{\mathrm{T}}\mathbf{g}_{S_{vi}} - \frac{1}{n_T}\sum_{j=1}^{n_T}\mathbf{P}^{\mathrm{T}}\mathbf{g}_{T_j}\right\|^2\right) \\ &= \operatorname{tr}\left(\mathbf{P}^{\mathrm{T}}\mathbf{G}_X\mathbf{M}^0\mathbf{G}_X^{\mathrm{T}}\mathbf{P}\right),\end{aligned} \tag{11}$$

where $\operatorname{tr}(\cdot)$ denotes the trace of a matrix. $\mathbf{G}_X = [\mathbf{G}_{S_1}, \ldots, \mathbf{G}_{S_V}, \mathbf{G}_T] \in R^{K(d+1)\times(V\times n_{Sv}+n_T)}$ and $\mathbf{G}_{S_v} = [\mathbf{g}_{S_{v_1}}, \mathbf{g}_{S_{v_2}}, \ldots, \mathbf{g}_{S_{v_{n_{S_v}}}}] \in R^{K(d+1)\times n_{S_v}}$ and $\mathbf{G}_T = [\mathbf{g}_{T_1}, \mathbf{g}_{T_2}, \ldots, \mathbf{g}_{T_{n_T}}] \in R^{K(d+1)\times n_T}$. $\mathbf{M}^0 \in R^{(V\times n_{Sv}+n_T)^2}$ is the MMD matrix, which is formulated as:

$$\left(\mathbf{M}^0\right)_{ij} = \begin{cases}\frac{1}{n_S n_S}, & i,j \le n_S \\ \frac{1}{n_T n_T}, & i,j > n_T \\ \frac{-1}{n_S n_T}, & \text{otherwise}\end{cases} \tag{12}$$

where $n_S = V \times n_{S_v}$ is the total number of V source streams.

2) Conditional distribution adaptation: Although minimizing the discrepancies in the marginal distributions across domains can be beneficial, this alone doesn't ensure the alignment of the conditional distributions $Q_{Sv}(y_{S_{vi}} \mid \mathbf{x}_{S_{vi}})$ and $Q_T(y_{T_i} \mid \mathbf{x}_{T_i})$. It is imperative to focus on reducing the differences between these conditional distributions for effective and robust distribution adaptation in the context of domain adaptation [41]. However, due to the scarcity of labels in the target stream, it is impractical to directly estimate the conditional distribution $Q_T(y_{T_i} \mid \mathbf{x}_{T_i})$ of the target stream. Consequently, we propose an assumption: $Q_{Sv}(y_{S_{vi}} \mid \mathbf{x}_{S_{vi}}) \approx Q_T(y_{T_i} \mid \mathbf{x}_{T_i})$. This allows us to apply the classifiers trained on source streams to predict pseudo labels $\hat{y}_T$ for the target stream [40]. To enhance accuracy, we adopt an iterative approach to update the classifier and the feature transformation $\phi(*)$. Since the true labels for source streams and the pseudo labels for target stream have been obtained, the MMD distance between class-conditional distributions $Q_{Sv}(\mathbf{x}_{S_{vi}} \mid y_{S_{vi}} = c)$ and $Q_T(\mathbf{x}_{T_i} \mid y_{T_i} = c)$ can be measured by:

$$\begin{aligned}&\sum_{v=1}^{V}\sum_{c=1}^{C}\left\|\frac{1}{n_{S_v}^{(c)}}\sum_{y_{S_{vi}}=c}\mathbf{P}^{\mathrm{T}}\mathbf{g}(\mathbf{x}_{S_{vi}}) - \frac{1}{n_T^{(c)}}\sum_{\hat{y}_{T_i}=c}\mathbf{P}^{\mathrm{T}}\mathbf{g}(\mathbf{x}_{T_i})\right\|^2 \\ &= \operatorname{tr}\left(\mathbf{P}^{\mathrm{T}}\mathbf{G}_X\sum_{c=1}^{C}\mathbf{M}^c\mathbf{G}_X^{\mathrm{T}}\mathbf{P}\right),\end{aligned} \tag{13}$$

where $c = 1, 2, \ldots C$ is the class numbers. $n_S^{(c)}$ and $n_T^{(c)}$ represent the number of examples belonging to the c-th class in all source streams and the target stream, respectively. $\mathbf{M}^c$ is defined as:

$$(\mathbf{M}^c)_{ij} = \begin{cases}\frac{1}{n_S^{(c)}n_S^{(c)}}, & i,j \le n_S \wedge y_{S_i}, y_{S_j} = c \\ \frac{1}{n_T^{(c)}n_T^{(c)}}, & i,j > n_S \wedge \hat{y}_{T_{(i-n_S)}}, \hat{y}_{T_{(j-n_S)}} = c \\ \frac{-1}{n_S^{(c)}n_T^{(c)}}, & \begin{cases} i \le n_S, j > n_S \wedge \hat{y}_{S_i}, \hat{y}_{T_{(j-n_S)}} = c \\ i > n_S, j \le n_s \wedge \hat{y}_{T_{(i-n_S)}}, \hat{y}_{S_j} = c\end{cases} \\ 0, & \text{otherwise}\end{cases} \tag{14}$$

3) Overall objective function and optimization: The subsequent parameters in P can be derived using the conventional Principal Component Analysis (PCA) methodology by incorporating both Eq.(11) and Eq.(13). Therefore, the overall objective function can be formulated as:

$$\begin{aligned}&\min_{P}\sum_{c=0}^{C}\operatorname{tr}\left(\mathbf{P}^{\mathrm{T}}\mathbf{G}_X\mathbf{M}^c\mathbf{G}_X^{\mathrm{T}}\mathbf{P}\right) + \lambda\|\mathbf{P}\|_F^2, \\ &s.t. \qquad \mathbf{P}^{\mathrm{T}}\mathbf{G}_X\mathbf{H}\mathbf{G}_X^{\mathrm{T}}\mathbf{P} = \mathbf{I}\end{aligned} \tag{15}$$

where $\|\mathbf{P}\|_F^2$ is a constraint term to avoid overfitting and λ is the corresponding regularization parameter and it is set as 1 in this paper. $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}$ represents the centering matrix and $n = V \times n_{S_v} + n_T$. $\mathbf{I} \in R^{K(d+1)\times K(d+1)}$ is a identity matrix.

According to the constrained optimization theory, we denote $\mathbf{\Phi} = \operatorname{diag}(\phi_1, \ldots, \phi_m) \in \mathbb{R}^{m\times m}$ as the Lagrange multiplier. Consequently, the Lagrangian function related to Eq.(15) can be derived as:

$$\begin{aligned}L = &\operatorname{tr}\left(\mathbf{P}^{\mathrm{T}}\left(\mathbf{G}_X\sum_{c=0}^{C}\mathbf{M}^c\mathbf{G}_X^{\mathrm{T}} + \lambda\mathbf{I}\right)\mathbf{P}\right) \\ &+ \operatorname{tr}\left(\left(\mathbf{I} - \mathbf{P}^{\mathrm{T}}\mathbf{G}_X\mathbf{H}\mathbf{G}_X^{\mathrm{T}}\mathbf{P}\right)\mathbf{\Phi}\right).\end{aligned} \tag{16}$$

By setting $\frac{\partial L}{\partial \mathbf{P}} = 0$, we obtain generalized eigendecomposition form:

$$\left(\mathbf{G}_X\sum_{c=0}^{C}\mathbf{M}^c\mathbf{G}_X^{\mathrm{T}} + \lambda\mathbf{I}\right)\mathbf{P} = \mathbf{G}_X\mathbf{H}\mathbf{G}_X^{\mathrm{T}}\mathbf{P}\mathbf{\Phi}. \tag{17}$$

Ultimately, the task of identifying the optimal transformation matrix $\mathbf{P}$ simplifies to resolving Eq.(17), specifically targeting the m smallest eigenvectors. In this context, $\phi_1, \ldots, \phi_m$ represent the minimal eigenvalues, with the corresponding eigenvectors denoted as $\mathbf{P} = [\mathbf{p}_g^1, \mathbf{p}_g^2, \ldots, \mathbf{p}_g^m]$. Once the consequent parameters $\mathbf{P}$ are acquired, it becomes feasible to derive new representations for each source stream and the target stream in the final fuzzy shared representation space. The detailed learning process is summarized in Algorithm 1.

Algorithm 2 The online learning process of IFSRL

Input: Input data $\{(\mathbf{x}_{S_{vi}}, y_{S_{vi}})\}_{i=1}^{N}$, $\{\mathbf{x}_{T_i}\}_{i=1}^{N}$; Window size n.
Output: Predicted target labels.

```
Read first n instances from each stream.
Obtain the Antecedent parameters of the TSK fuzzy systems and
  consequent transformation matrix P via Algorithm 1.
Create the source drift detectors DDM_{S_v} and GMM models for
  each source and target stream.
while there is incoming data do
  for v = 1 : V do
    if Source drift is detected then
      GMM-based adaptation by Eq.(24).
      Construct the fuzzy shared representation z_{S_{v_i}} by Eq.
        (8a) and Eq.(9a).
      Move the current classifier to the classifier pool C_p and
        retrain a new classifier f_{S_v} using (z_{S_{v_i}}, y_{S_{vi}}).
    else
      Construct the fuzzy shared representation z_{S_{v_i}} by Eq.
        (8a) and Eq.(9a).
      Incrementally training using (z_{S_{v_i}}, y_{S_{vi}}).
    end if
  end for
  Move detection window and calculate μ_det, μ_ref
  if Target drift is detected then
    Remove all base classifiers and return to line 1.
  else
    Construct the fuzzy shared representation z_{T_i} by Eq. (8b)
      and Eq.(9b).
    Predict the target label.
  end if
end while
```

In Algorithm 1, it is assumed that each stream comprises n samples, resulting in a total of $N = (V+1)n$ samples. The computational complexity of step 1 and step 2 is denoted as $O(2dNK)$ and $O((1+d)NK)$, respectively. Subsequently, assuming I iterations for multistream joint distribution adaptation and a final data dimensionality of $\hat{d}$ in the fuzzy shared representation space, the computational complexities in steps 5, 6, and 7 are denoted as $\hat{d}m^2d^2$, $N\hat{d}d$, and CN^2, respectively. Therefore, the overall complexity of the proposed FSRL is expressed as $O(NK(3d+1)+\hat{d}m^2d^2+N\hat{d}d+CN^2)$.

C. Blind window-based adaptation (BFSRL)

The proposed BFSRL is a straightforward yet highly effective approach. In this strategy, a set of sliding windows W_{S_v} and W_T with size n are crafted to collect the incoming data from source and target streams, respectively. This framework facilitates the joint representation learning of the collected data via FSRL, paving the way for subsequent label prediction tasks for the target stream. Specifically, within any given sliding window, the source data $W_{S_v} = [\mathbf{x}_{S_{v_1}}, \mathbf{x}_{S_{v_2}}, \ldots, \mathbf{x}_{S_{v_n}}], v = [1, V]$ and target data $W_T = [\mathbf{x}_{T_1}, \mathbf{x}_{T_2}, \ldots, \mathbf{x}_{T_n}]$ are procured. Subsequently, stream-specific fuzzy representations are derived using Eq.(3a) and Eq. (3b) as follows:

$$\mathbf{G}_{W_{S_v}} = [\mathbf{g}_{S_{v_1}}, \mathbf{g}_{S_{v_2}}, \ldots, \mathbf{g}_{S_{v_n}}], v = [1, V] \tag{18a}$$

$$\mathbf{G}_{W_T} = [\mathbf{g}_{T_1}, \mathbf{g}_{T_2}, \ldots, \mathbf{g}_{T_n}]. \tag{18b}$$

Subsequently, the consequent parameters P are optimized by employing the optimization criteria defined in Equation (15). Therefore, the final fuzzy shared representations can be calculated by:

$$\mathbf{Z}_{W_{S_v}} = \mathbf{P}^T\mathbf{G}_{W_{S_v}}, v = 1, 2, \ldots V \tag{19a}$$

$$\mathbf{Z}_{W_T} = \mathbf{P}^T\mathbf{G}_{W_T}. \tag{19b}$$

Ultimately, classifiers f_{S_v} are trained on the entirety of the fuzzy shared representations derived from the source streams. The aggregate predictive probability is then obtained by an ensemble, where each classifier's contribution is weighted and averaged, formalized as:

$$f^E(\mathbf{Z}_{W_T}) = \frac{1}{V}\sum_{v=1}^{V} f_{S_v}(\mathbf{Z}_{W_T}). \tag{20}$$

This approach not only efficaciously showcases the potential of the proposed FSRL in enabling knowledge transfer across various streams but also offers a direct resolution to the complexities inherent in multistream classification.

D. Informed GMM-based adaptation (IFSRL)

Although BFSRL offers a straightforward and effective approach, it is highly sensitive to the window size due to the varying frequencies and types of concept drift in different datasets. This variability can lead to challenges in consistently applying this method across diverse drifting conditions. To enhance the generalization and robustness of the proposed method, we introduce the IFSRL, which integrates drift detection and fuzzy shared representation learning, providing a more nuanced response to the concept drift problem.

In this process, similar to window-based approaches, sliding windows W_{S_v} and W_T with size n are crafted to collect the incoming data from source and target streams, respectively. Subsequently, we obtain an initial set of antecedent parameters and the shared consequent parameters $\mathbf{P}$ as well as base classifiers f_{S_v} via FSRL. Then, we conduct online detection and adaptation based on different asynchronous drift scenarios.

1) Drift detection: The selection of appropriate drift detection methods hinges on the availability of labels. For labeled source streams, we use the DDM [26] to detect the drift due to its accuracy and stability. When a new source sample $\mathbf{x}_{S_{vi}}$ arrives, its predicted label updates the drift detector based on prediction error. Drift is detected if the error rate exceeds a set warning threshold. This provides a reliable and efficient mechanism for monitoring drift in dynamic data streams.

However, due to the absence of labels in the target data, we are constrained to employing unsupervised methods for drift detection. Our strategy for identifying drift in the target stream involves vigilantly tracking alterations in its probability distribution. Recognizing the efficacy of GMM in accurately representing data probability distributions, we leverage the archived target data to initialize a GMM model based on the Expectation-Maximization algorithm. The underlying premise of GMM is its capacity to approximate real-world data through a finite number of mixture components, and it is formulated as follows:

$$P(x) = \sum_{k=1}^{K} P(x \mid C_i) \cdot w_k, \tag{21}$$

where K signifies the total number of Gaussians or mixture components which is set the same as the number of fuzzy rules, and x is the observed multivariate. Each mixture component C_k is associated with a weight w_k, which is determined based on the observations encompassing C_k, and $0 \leq w_k \leq 1, \sum_{k=1}^{K} w_k = 1$. $P(x \mid C_k)$ defines the likelihood of the observation x being allocated to the mixture component C_k. It can be represented by using the mean μ_k and the covariance Σ_k of each mixture component C_k as follows:

$$P(x \mid C_k) = \frac{1}{\left(2\pi^{d/2}\sqrt{|\Sigma_k|}\right)} \exp\left(-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)\right). \tag{22}$$

Then the correlation between target samples can be assessed by the conditional probability $P(\mathbf{x}_{T_i}|C_k)$. If there is a temporal variation in the conditional probability of target samples, it may indicate the emergence of a new concept. However, detection based on a single sample is prone to sensitivity towards outliers. To mitigate this, we employ two sliding windows: the Reference Window $W_{ref} = [\mathbf{x}_{T_1}, \mathbf{x}_{T_2}, \ldots, \mathbf{x}_{T_n}]$ and the Detect Window $W_{det} = [\mathbf{x}_{T_{(n+1)}}, \mathbf{x}_{T_{(n+2)}}, \ldots, \mathbf{x}_{T_{(2n)}}]$, where n represents the number of instances in each window. The average conditional probability for the reference window is then determined using point estimation of the mean within the normal distribution by:

$$\mu_{\text{ref}} = \frac{1}{n}\sum_{k=1}^{n} \max_{k\in\{1,2,\ldots,K\}} P(\mathbf{x}_{T_i} \mid C_k). \tag{23}$$

The confidence interval estimation of the μ_{ref} is known to be $[\mu_{\text{ref}} - z_\alpha(\sigma/\sqrt{n}), \mu_{\text{ref}} + z_\alpha(\sigma/\sqrt{n})]$, where σ is the standard deviation and z_α is the significance level which is set as 3 [42]. When the point estimation by the mean μ_{ref} in the detection window satisfies $\mu_{\text{det}} \geq \mu_{\text{ref}} + z_\alpha \times \sigma/\sqrt{n}$, the decision is made that the change has occurred. Otherwise, W_{ref} and W_{det} move step by step to receive new incoming data. If μ_{det} in the detection window meets or exceeds $\mu_{\text{ref}} + z_\alpha \times \sigma/\sqrt{n}$, it is inferred that a change has taken place. Otherwise, both W_{ref} and W_{det} are incrementally shifted to incorporate new incoming data.

2) Drift adaptation: As discussed before, there are various drifting scenarios in multistream classification. Here, we systematically categorize them into three distinct scenarios for comprehensive analysis.

- No drift: when no drift is detected in either the source or target streams, the process advances with the incremental training of the base classifiers. During the initialization phase, the implementation of FSRL equips us with optimal antecedent parameters and shared consequent parameters $\mathbf{P}$. This allows for the newly arriving data $\mathbf{x}_{S_{vi}}$ to be effectively mapped into the fuzzy shared space. Subsequently, this mapped data $\mathbf{z}_{S_{v_i}}$ is employed to refine and update the base classifiers f_{S_v}.
- Source-only drifting: once a drift is detected within any source stream, an adaptation module should be deployed to handle new concepts. Similarly, we utilize the GMM to evaluate the distributions of the old and new concepts. For a newly incoming instance $\mathbf{x}_{S_{vi}}$, its importance weight $w_{S_{vi}}$ can be calculated by maximizing the conditional probability of GMM as follows:

$$w_{S_{vi}} = \max_{k\in\{1,2,\ldots,K\}} P(\mathbf{x}_{S_{vi}} \mid G_i). \tag{24}$$

When a new concept emerges in any source stream, adaptation is achieved by applying a multiplicative factor $w_{S_{vi}}$ to the data. This weighted data is then mapped into the fuzzy shared space. Subsequently, a new base classifier is created and trained using this mapped data $\mathbf{z}_{S_{v_i}}$. It is important to note that old base classifiers are not updated with new samples. Instead, they are preserved within a classifier pool, denoted as C_p, to maintain the old knowledge. Finally, the ensemble of joint predictive probabilities is formulated as follows:

$$\begin{aligned} f^E(\mathbf{z}_{T_i}) = &\frac{w_{Si}}{\sum_{n=1}^{N} w_{Si} + \sum_{l=1}^{|C_p|} w_P} f_{S_v}(\mathbf{z}_{T_i}) \\ &+ \frac{w_P}{\sum_{n=1}^{N} w_{Si} + \sum_{l=1}^{|C_p|} w_P} f_P(\mathbf{z}_{T_i}), \end{aligned} \tag{25}$$

where w_P is the weight of l-th classifer in C_p, and $w_{Si} = \frac{1}{N}\sum_{i=1}^{N} w_{S_{vi}}$.
- Target-inclusive drifting: upon the detection of a target drift, the antecedent parameters and the shared consequent parameters $\mathbf{P}$, as well as the base classifiers f_{S_v} that were established in the initial phase, become inadequate for the classification of target samples. As a result, all base classifiers are removed from the classifier pool. This necessitates a re-initialization of the model, enabling it to adapt efficiently to the newly concepts.

The online learning process of IFSRL is detailed in Algorithm 2. During the online process, there are four main modules: FSRL, GMM, DDM, and the Hoeffding Tree classifier. The overall complexity of FSRL is given by $O(NK(3d+1) + \hat{d}m^2d^2 + N\hat{d}d + CN^2)$. In this method, we employ the EM algorithm to estimate the GMM parameters, and its complexity can be regarded as linear, i.e., $O(n)$, where n is the window size. DDM also exhibits linear complexity, $O(n)$. The time complexity of each Hoeffding Tree classifier is $O(n\log(n))$. Assuming there are V source streams and 1 target stream, the time complexities of GMM, DDM, and Hoeffding Tree are $O((V+1)n)$, $O(Vn)$, and $O(Vn\log(n))$, respectively. Consequently, the overall complexity of IFSRL is $O(NK(3d+1)+\hat{d}m^2d^2+N\hat{d}d+CN^2)+O((2V+1+V\log(n))n)$. In fact, since $N=(V+1)n$, and V and K are quite small compared to n, the complexity of IFSRL depends on the window size n. Hence, we can adjust n to strike a balance between the performance of IFSRL and available resources.

IV. EXPERIMENTS

In the experiment, we first demonstrated that BFSRL and IFSRL consistently outperform current methods in multistream classification, highlighting both robustness and superiority. Second, because IFSRL performs a more comprehensive perspective, we validated the effectiveness of each proposed component when facing different challenges by ablation study. Finally, we established the scalability of IFSRL across diverse data streams, corroborating its stable predictive capabilities. In

TABLE I: Characteristics of all datasets.

	Dataset	Type	Features	Class	Instance number	Drift type
Sythetic	**SEA**	Single stream	3	2	100K	Abrupt/Recurring
	RBF	Single stream	10	2	20K	Incremental
	Tree	Single stream	20	2	20K	Abrupt/Gradual
	Hyperplane	Single stream	4	2	120K	Incremental
Real-world	**Weather**	Single stream	8	2	18K	Unknown
	Electricity	Single stream	8	2	45K	Unknown
	Kitti	Single stream	55	8	25K	Unknown
	CNNIBN	Multistream	124	2	120K	Unknown
	BBC	Multistream	124	2	120K	Unknown

TABLE II: Classification accuracy (%) with the variance of various methods on all benchmarks. The best results are highlighted in bold, while the second-best results are marked with an underline.

		Synthetic Datasets				Real-World Datasets				
Methods		**SEA**	**RBF**	**Tree**	**Hyperplane**	**Weather**	**Electricity**	**Kitti**	**CNNIBN**	**BBC**
Fusion	**S1**	$85.04_{\pm 0.84}$	$82.03_{\pm 1.41}$	$76.98_{\pm 1.11}$	$83.29_{\pm 0.67}$	$71.04_{\pm 1.50}$	$73.82_{\pm 2.56}$	$54.21_{\pm 2.61}$	$66.76_{\pm 0.74}$	$61.76_{\pm 0.09}$
	S2	$85.78_{\pm 0.92}$	$83.46_{\pm 1.20}$	$76.74_{\pm 1.00}$	$84.05_{\pm 0.52}$	$70.65_{\pm 1.32}$	$73.07_{\pm 3.01}$	$52.36_{\pm 2.72}$	$67.54_{\pm 1.11}$	$61.26_{\pm 0.43}$
	S3	$84.31_{\pm 1.13}$	$81.03_{\pm 1.73}$	$75.21_{\pm 1.07}$	$82.17_{\pm 0.57}$	$72.17_{\pm 1.17}$	$74.31_{\pm 2.73}$	$50.38_{\pm 2.43}$	$65.34_{\pm 0.92}$	$59.86_{\pm 0.19}$
ATL	**S1**	$88.42_{\pm 1.70}$	$84.53_{\pm 2.01}$	$76.43_{\pm 2.17}$	$86.17_{\pm 1.04}$	$74.57_{\pm 1.94}$	$75.07_{\pm 1.81}$	$52.78_{\pm 3.78}$	$62.78_{\pm 1.44}$	$62.78_{\pm 1.16}$
	S2	$88.74_{\pm 1.75}$	$85.21_{\pm 1.85}$	$76.71_{\pm 1.86}$	$87.07_{\pm 1.21}$	$75.03_{\pm 2.01}$	$75.83_{\pm 1.77}$	$54.01_{\pm 3.09}$	$65.74_{\pm 1.76}$	$62.34_{\pm 0.83}$
	S3	$87.62_{\pm 1.01}$	$83.16_{\pm 2.13}$	$76.07_{\pm 2.42}$	$86.01_{\pm 1.49}$	$74.62_{\pm 1.77}$	$73.96_{\pm 2.13}$	$53.26_{\pm 3.21}$	$62.65_{\pm 1.38}$	$60.76_{\pm 0.77}$
Melanie		$89.18_{\pm 0.77}$	$86.04_{\pm 0.39}$	$78.93_{\pm 0.61}$	$86.38_{\pm 0.57}$	$77.74_{\pm 0.89}$	$77.45_{\pm 2.95}$	$50.29_{\pm 1.34}$	$68.79_{\pm 0.31}$	$\mathbf{68.04}_{\pm 0.01}$
AOMSDA		$\mathbf{90.23}_{\pm 1.42}$	$85.26_{\pm 2.89}$	$76.87_{\pm 3.47}$	$87.66_{\pm 1.74}$	$76.55_{\pm 1.41}$	$78.02_{\pm 3.32}$	$\underline{67.79}_{\pm 3.16}$	$69.07_{\pm 1.40}$	$63.36_{\pm 1.07}$
MCMO		$87.46_{\pm 2.12}$	$86.26_{\pm 0.77}$	$77.64_{\pm 1.47}$	$84.04_{\pm 1.42}$	$76.02_{\pm 3.43}$	$78.79_{\pm 2.17}$	$64.82_{\pm 4.17}$	$68.83_{\pm 0.89}$	$60.12_{\pm 1.51}$
BFSRL (Ours)		$87.62_{\pm 0.01}$	$\underline{86.32}_{\pm 0.42}$	$\underline{79.42}_{\pm 0.32}$	$\mathbf{87.78}_{\pm 0.14}$	$\underline{78.93}_{\pm 0.11}$	$\mathbf{80.27}_{\pm 0.58}$	$62.32_{\pm 1.21}$	$\underline{69.13}_{\pm 1.74}$	$60.15_{\pm 0.79}$
IFSRL (Ours)		$\underline{89.53}_{\pm 2.01}$	$\mathbf{87.94}_{\pm 1.39}$	$\mathbf{79.89}_{\pm 1.06}$	$\underline{87.75}_{\pm 1.71}$	$\mathbf{80.04}_{\pm 2.12}$	$\underline{79.46}_{\pm 1.62}$	$\mathbf{69.06}_{\pm 1.14}$	$\mathbf{73.4}_{\pm 1.45}$	$\underline{63.48}_{\pm 2.07}$

addition, we also analyzed parameter sensitivity and convergence.

A. Benchmarks

In the experiment, we use four synthetic: SEA [43], RBF [44], Tree [45], Hyperplane [28], and five real-world datasets: Weather [46], Electricity [45], Kitti [47], BBC [48] and CNNIBN [48] to simulate the multistream classification task and test our proposed method. The detailed characteristics of all datasets are elaborated in Table I. To simulate the multistream classification scenario, we first sort all samples in descending order according to the probability of each sample $P(x) = \exp\frac{(x-\bar{x})^2}{2\sigma^2}$ in a Gaussian distribution, which induces the problem of covariate shift. Then the construction of source streams follows a sequential order, with the first source stream being built upon the top N_i samples, followed by the second source stream, the third source stream, and so on up to $(N-1)$-th source stream. The remaining data samples are then assigned to the target stream. All samples selected in each stream will be recovered to the original chronological order to maintain the raw temporal relationship (i.e., asynchronous drift). Only source streams exclusively consist of labels, whereas the target stream lacks labels, resulting in the scarcity of labels problem.

In addition, BBC and CNNIBN are constructed based on the *TV News Channel Commercial Detection* Dataset[1] [48]. It comprises prominent audio-visual features collected from 150 hours of television news broadcasts, including 30 hours each from five news channels (i.e., BBC, CNNIB, CNN, NDTV, and TIMESNOW). All the video shots are recorded consecutively and used for commercial or non-commercial detection. Specifically, the original dataset is multimodal and contains five sets of video features (i.e., video shot length, screen text distribution, motion distribution, frame difference distribution, and edge change ratio) and seven sets of audio features (i.e., short-term energy, zero crossing rate, spectral centroid, spectral flux, spectral roll-off frequency, fundamental frequency and bag of audio words), for 4125 dimensions in total. In this experiment, we remove the bag of audio words feature and just use the other 11 sets of features. In addition, to retain as much of the original data as possible, we re-sampled all data streams to 30,000 samples. We designate CNNIBN and BCC as the target streams while treating the remaining streams as source streams to simulate a multistream classification task.

B. Baselines and experiment settings

To validate the effectiveness of our proposed approach, we conducted experiments comparing it with five state-of-the-art methods. Specifically, FUSION [16] and ATL [34] are

[1] https://archive.ics.uci.edu/dataset/326/tv+news+channel+commercial+detection+dataset

single-source stream-based algorithms, whereas Melanie [37], AOMSDA [9], and MCMO [12] are designed for multi-source classification scenarios. For FUSION and ATL, we formed three distinct groups by pairing each source stream with the target stream: FUSIONs1, FUSIONs2, and FUSIONs3 and ATLs1, ATLs2, and ATLs3, respectively. To keep a fair comparison, we referenced their respective papers for optimal parameter selection or adjusted parameters to ensure optimal performance on each dataset.

In this study, we implemented the framework using the scikit-multiflow learning library [49] in Python. All experimental evaluations were conducted on a server equipped with 187GB of memory and powered by an Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz.

C. Overall performance

Our main experiments used a 3-source and 1-target setup, and Table II compares the classification accuracy of our proposed methods, i.e., BFSRL and IFSRL, against all baselines on three synthetic and four real-world datasets.

First, in comparison to single-source-based techniques, such as Fusion and ATL, the multi-source-based methods consistently exhibit substantial advancements. This observation underscores the efficacy of leveraging multiple labeled source streams, as they contribute more discriminative and complementary information. Consequently, it emphasizes the significance of incorporating diverse data sources for achieving superior performance.

Secondly, compared to multi-source-based methods, our approach consistently achieves top-tier performance on most datasets. Specifically, BFSRL outperforms the MCMO algorithm across all datasets, and even when compared to the supervised method (Melanie), it demonstrates superior results on seven out of nine datasets. This directly attests to the effectiveness of our proposed FSRL algorithm. However, when contrasted with another blind adaptation method, AOMSDA, BFSRL exhibits weaker performance on the SEA and Kitti datasets. This discrepancy can be attributed to the fact that different methods respond differently to various types and frequencies of drift. Despite this, BFSRL outperforms AOMSDA significantly on the other seven datasets. This highlights the robust performance of BFSRL across diverse datasets, showcasing its effectiveness in addressing knowledge transfer.

In addition, our proposed informed adaptation method, IFSRL, consistently exhibits superior performance on almost all datasets. This is attributed to the integration of drift detection, allowing for a more nuanced response to different types and frequencies of drift. Moreover, the method employs a GMM-based ensemble weighting prediction strategy, enabling the retention of prior knowledge for replay during predictions of new concepts. This approach proves to be more effective compared to previous methods, such as direct retraining or fine-tuning, in mitigating catastrophic forgetting. The incorporation of drift detection and ensemble weighting enhances the adaptability and knowledge preservation capabilities of IFSRL, contributing to its superiority across diverse datasets.

TABLE III: Classification accuracy (%) of variants.

	$\mathbf{V}_1$	$\mathbf{V}_2$	$\mathbf{V}_3$	$\mathbf{V}_4$	**IFSRL**
SEA	85.76	87.78	87.04	88.42	**89.53**
RBF	85.39	85.91	85.03	86.92	**87.94**
Tree	76.57	77.49	77.91	78.72	**79.89**
Hyperplane	86.31	87.21	87.04	86.64	**87.75**
Weather	76.35	78.89	78.04	78.71	**80.04**
Electricity	75.87	76.72	77.21	78.33	**79.46**
Kitti	61.76	65.85	67.01	67.78	**69.06**
CNNIBN	69.47	71.71	70.42	71.84	**73.40**
BBC	60.41	62.05	62.57	63.04	**63.48**

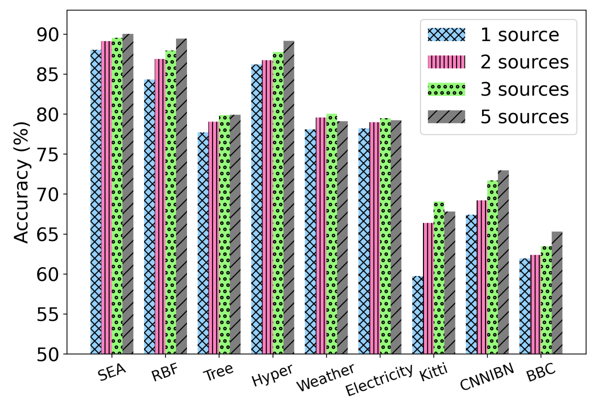


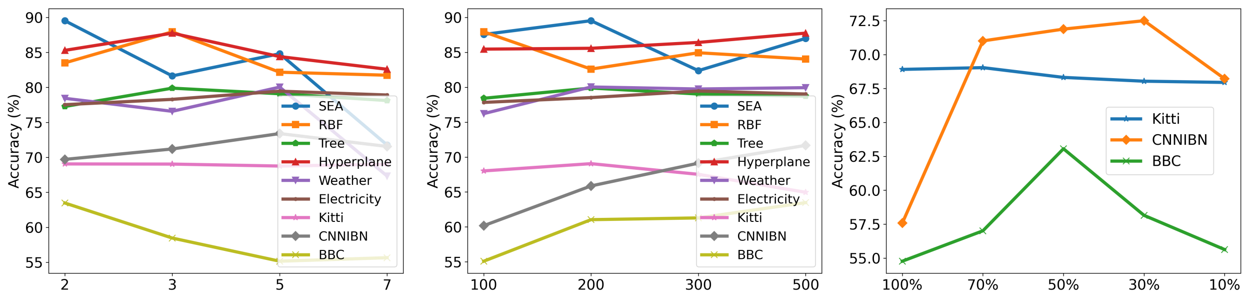
Fig. 3: The influence of the different number of sources.

D. Ablation study

To validate the rationality of each component and its impact on the overall classification results, we designed four variants of IFSRL. As shown in Table III, the baseline (V_1) directly removes the FSRL component, providing a reference point for comparison. Subsequent variants explore the effects of individual modifications. V_2 disregards the shared consequent parameters and independently considers each stream. V_3 omits the use of GMM to address asynchronous concept drift in source streams. V_4 eliminates the classifier pool, i.e., it employs only one specific classifier for each stream without considering prior knowledge.

The classification accuracy results across multiple datasets provide several insights. Concretely, IFSRL consistently outperforms all variants across all datasets, demonstrating the effectiveness of the integrated components, especially the significance of fuzzy shared representation during the knowledge transfer. Specifically, V_2 and V_3 exhibit competitive accuracy compared to the baseline V_1, suggesting that shared consequent parameters and GMM-based handling of asynchronous concept drift individually contribute to improved performance. V_4 demonstrates the importance of the classifier pool, showcasing that considering multiple classifiers for each stream and preserving prior knowledge through the ensemble approach in IFSRL significantly enhances adaptability and performance.

These findings underscore the synergistic impact of the proposed modifications in IFSRL, affirming its robustness in addressing various challenges in the multistream classification.



(a) Number of fuzzy rules (b) Window size (c) Feature dimension

Fig. 4: The effect of main parameters on classification accuracy.

The superior performance across all ablation variants reaffirms the efficacy of our proposed method in handling complex dynamic correlations.

E. Supplementary experiments

1) Influence of Source Numbers: In this section, we explore the influence of the number of source streams on the performance of IFSRL. Specifically, we assess IFSRL's performance using 1, 2, 3, and 5 source streams, respectively. For configurations with more than 3 sources, we evenly distribute the source samples into V streams and re-sample them to ensure an equivalent number of samples as in the target stream. Our primary objective is to determine if leveraging multiple source streams enhances predictive capability compared to a single-source stream. As depicted in Figure 3, the results indicate a consistent improvement in performance with multi-source streams across all datasets, surpassing the performance of a single-stream setup. It underscores the ability of multi-source streams to provide supplementary and complementary information, thereby enhancing overall predictive performance.

However, it is noteworthy that as the number of source streams increases, there might be a decline in performance. For instance, on the Weather and Kitti datasets, the performance with 3 source streams surpasses that with 5 source streams. This trend could be attributed to the increased complexity of the model as the number of source streams grows, potentially affecting its overall performance. Despite these fluctuations, the performance of IFSRL remains stable across various source configurations. This stability suggests that our proposed method can effectively adapt to different numbers of data streams, showcasing its versatility and robustness in handling diverse source stream scenarios.

2) Parameters: In the proposed IFSRL method, three key parameters significantly influence the classification results: the number of fuzzy rules K, the window size n, and the dimensions after feature mapping. To analyze their respective impacts on the prediction performance, experiments were conducted on all datasets using different parameter values. Specifically, we set the fuzzy rules number to be $\{2, 3, 5, 7\}$, the window size to be $\{100, 200, 300, 500\}$, and the dimensions after reduction to be $\{100\%, 70\%, 50\%, 30\%, 10\%\}$ of the original dimensions, respectively. During the experiments, each parameter was tuned independently while keeping the others fixed, and the diverse performances are depicted in Figure 4.

Different datasets exhibit varying optimal fuzzy rules primarily due to the diverse fuzzy relationships among multiple data streams. Additionally, each dataset possesses unique drift frequencies and periods, and the choice of window size significantly impacts the results. For datasets with higher frequency drifts, a more flexible window size may be necessary to adapt to changes in data distribution. Finally, the impact of data dimensionality on results is paramount, primarily due to the substantial influence of redundant features. This is why our approach considers drift adaptation in a low-dimensional fuzzy shared space, effectively mitigating the influence of redundant features and thereby enhancing the algorithm's robustness. The optimal parameters configured in our experiments are also listed in Table IV.

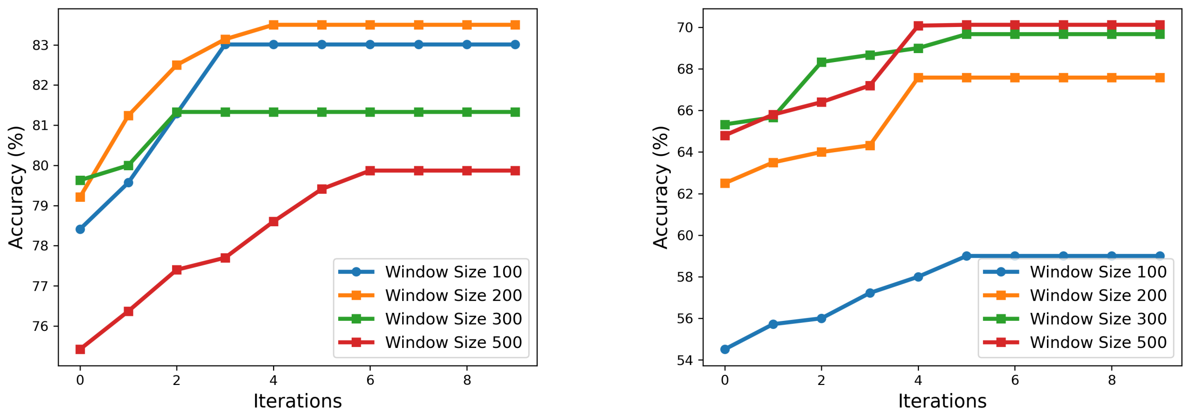
3) Convergence Analysis: To further validate the convergence of this algorithm, we conducted experiments on both a synthetic (SEA) and a real-world (Weather) dataset. We tested the trend of accuracy with increasing iterations under different window sizes (i.e., window size $\in \{100, 200, 300, 500\}$). As depicted in Figure 5, we observed a increase in accuracy with the number of iterations, stabilizing within approximately five iterations for both datasets. These experimental results demonstrate the guaranteed convergence of our method.

TABLE IV: Parameter settings on different datasets.

	Fuzzy rules	Window size	Feature dimension
SEA	2	200	3
RBF	3	100	10
Tree	3	200	20
Hyperplane	3	500	4
Weather	5	200	8
Electricity	5	300	8
Kitti	2	200	30
CNNIBN	5	400	50
BBC	2	500	50

V. Conclusion & Future works

This paper introduces Fuzzy Shared Representation Learning (FSRL) as an innovative method for multistream classification, specifically tailored to address challenges associated



(a) SEA dataset (b) Weather Dataset

Fig. 5: Convergence analysis on SEA and Weather datasets.

with non-stationary data streams that exhibit concept drift. FSRL integrates a TSK fuzzy system for non-linear transformation and introduces a joint distribution adaptation for inter-stream shift alignment and feature reduction. It enhances the model's interpretability and adaptability, and promotes robust knowledge transfer across multiple streams. Moreover, our proposed method features two online adaptation strategies: 1) BFSRL empirically demonstrates the effectiveness of FSRL; 2) IFSRL incorporates a drift detection module and GMM-based distribution adaptation to handle asynchronous drifts. These contributions present practical solutions for managing the dynamic nature of real-world data streams, underscoring the significance of FSRL in fortifying the robustness and interpretability of multistream classification.

Although FSRL has demonstrated good performance in the multistream classification task, it still has some limitations. For example, 1) The number of fuzzy rules is still pre-defined, and we need to adjust this parameter to seek optimal performance on specific data. Our future work will focus on how to automatically estimate the values of fuzzy rules through learning algorithms; 2) When estimating class-conditional distributions, we assume that the source stream and target stream share the same label space. However, in open-world scenarios, new classes may emerge, which is also one of our concerns.

References

[1] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 2346–2363, 2018.

[2] B. Jiao, Y. Guo, D. Gong, and Q. Chen, "Dynamic ensemble selection for imbalanced data streams with concept drift," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[3] A. Liu, J. Lu, and G. Zhang, "Concept drift detection: dealing with missing values via fuzzy distance estimations," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 11, pp. 3219–3233, 2020.

[4] F. Hinder, V. Vaquet, and B. Hammer, "One or two things we know about concept drift–a survey on monitoring evolving environments," *arXiv preprint arXiv:2310.15826*, 2023.

[5] G. Bai, C. Ling, and L. Zhao, "Temporal domain generalization with drift-aware dynamic neural networks," in *The Eleventh International Conference on Learning Representations*, 2022.

[6] S. Abadifard, S. Bakhshi, S. Gheibuni, and 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.

[7] J. Gama and G. Castillo, "Learning with local drift detection," in *International conference on advanced data mining and applications*. Springer, 2006, pp. 42–55.

[8] C. Yang, Y.-m. Cheung, J. Ding, and K. C. Tan, "Concept drift-tolerant transfer learning in dynamic environments," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3857–3871, 2021.

[9] X. Renchunzi and M. Pratama, "Automatic online multi-source domain adaptation," *Information Sciences*, vol. 582, pp. 480–494, 2022.

[10] K. Wang, J. Lu, A. Liu, and G. Zhang, "An augmented learning approach for multiple data streams under concept drift," in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2023, pp. 391–402.

[11] Y. Song, J. Lu, H. Lu, and G. Zhang, "Learning data streams with changing distributions and temporal dependency," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[12] B. Jiao, Y. Guo, S. Yang, J. Pu, and D. Gong, "Reduced-space multistream classification based on multi-objective evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, 2022.

[13] S. Chandra, A. Haque, L. Khan, and C. Aggarwal, "An adaptive framework for multistream classification," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 1181–1190.

[14] B. Dong, Y. Gao, S. Chandra, and L. Khan, "Multistream classification with relative density ratio estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3478–3485.

[15] H. Tao, Z. Wang, Y. Li, M. Zamani, and L. Khan, "Comc: A framework for online cross-domain multistream classification," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[16] A. Haque, Z. Wang, S. Chandra, B. Dong, L. Khan, and K. W. Hamlen, "Fusion: An online method for multistream classification," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 919–928.

[17] Z. Wang, C. Zhou, B. Du, and F. He, "Self-paced supervision for multi-source domain adaptation," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2022.

[18] J. Lu, H. Zuo, and G. Zhang, "Fuzzy multiple-source transfer learning," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 12, pp. 3418–3431, 2019.

[19] H. Yu, J. Lu, and G. Zhang, "Topology learning-based fuzzy random neural networks for streaming data regression," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 2, pp. 412–425, 2020.

[20] K. Li, J. Lu, H. Zuo, and G. Zhang, "Source-free multi-domain adaptation with fuzzy rule-based deep neural networks," *IEEE Transactions on Fuzzy Systems*, 2023.

[21] C. Yang, Z. Deng, K.-S. Choi, and S. Wang, "Takagi–sugeno–kang transfer learning fuzzy logic system for the adaptive recognition of epileptic electroencephalogram signals," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 5, pp. 1079–1094, 2015.

[22] P. Xu, Z. Deng, J. Wang, Q. Zhang, K.-S. Choi, and S. Wang, "Transfer representation learning with tsk fuzzy system," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 3, pp. 649–663, 2019.

[23] W. Zhang, Z. Deng, T. Zhang, K.-S. Choi, and S. Wang, "Multi-view fuzzy representation learning with rules based model," *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[24] G. Oliveira, L. L. Minku, and A. L. Oliveira, "Tackling virtual and real concept drifts: An adaptive gaussian mixture model approach," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[25] E. Yu, J. Lu, B. Zhang, and G. Zhang, "Online boosting adaptive learning under concept drift for multistream classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 15, 2024, pp. 16 522–16 530.

[26] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in *Brazilian Symposium on Artificial Intelligence*. Springer, 2004, pp. 286–295.

[27] H. M. Gomes, A. Bifet, J. Read, J. P. Barddal, F. Enembreck, B. Pfharinger, G. Holmes, and T. Abdessalem, "Adaptive random forests for evolving data stream classification," *Machine Learning*, vol. 106, no. 9, pp. 1469–1495, 2017.

[28] A. Bifet and R. Gavalda, "Learning from time-changing data with adaptive windowing," in *Proceedings of the 2007 SIAM International Conference on Data Mining*. SIAM, 2007, pp. 443–448.

[29] S. Xu and J. Wang, "Dynamic extreme learning machine for data stream classification," *Neurocomputing*, vol. 238, pp. 433–449, 2017.

[30] C. Alippi and M. Roveri, "Just-in-time adaptive classifiers—part i: Detecting nonstationary changes," *IEEE Transactions on Neural Networks*, vol. 19, no. 7, pp. 1145–1153, 2008.

[31] N. Lu, G. Zhang, and J. Lu, "Concept drift detection via competence models," *Artificial Intelligence*, vol. 209, pp. 11–28, 2014.

[32] J. Z. Kolter and M. A. Maloof, "Dynamic weighted majority: An ensemble method for drifting concepts," *The Journal of Machine Learning Research*, vol. 8, pp. 2755–2790, 2007.

[33] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1517–1531, 2011.

[34] M. Pratama, M. de Carvalho, R. Xie, E. Lughofer, and J. Lu, "Atl: Autonomous knowledge transfer from many streaming processes," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 269–278.

[35] H. Yu, Q. Zhang, T. Liu, J. Lu, Y. Wen, and G. Zhang, "Meta-add: A meta-learning based pre-trained model for concept drift active detection," *Information Sciences*, vol. 608, pp. 996–1009, 2022.

[36] E. Yu, Y. Song, G. Zhang, and J. Lu, "Learn-to-adapt: Concept drift adaptation for hybrid multiple streams," *Neurocomputing*, vol. 496, pp. 121–130, 2022.

[37] H. Du, L. L. Minku, and H. Zhou, "Multi-source transfer learning for non-stationary environments," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[38] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE transactions on systems, man, and cybernetics*, no. 1, pp. 116–132, 1985.

[39] T. Su and J. G. Dy, "In search of deterministic methods for initializing k-means and gaussian mixture clustering," *Intelligent Data Analysis*, vol. 11, no. 4, pp. 319–338, 2007.

[40] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *International conference on machine learning*. PMLR, 2017, pp. 2208–2217.

[41] J. Wang, C. Lan, C. Liu, Y. Ouyang, T. Qin, W. Lu, Y. Chen, W. Zeng, and P. Yu, "Generalizing to unseen domains: A survey on domain generalization," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[42] Y. Kim and C. H. Park, "An efficient concept drift detection method for streaming data under limited labeling," *IEICE Transactions on Information and systems*, vol. 100, no. 10, pp. 2537–2546, 2017.

[43] W. N. Street and Y. Kim, "A streaming ensemble algorithm (sea) for large-scale classification," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 2001, pp. 377–382.

[44] Y. Song, J. Lu, A. Liu, H. Lu, and G. Zhang, "A segment-based drift adaptation method for data streams," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[45] A. Liu, J. Lu, and G. Zhang, "Diverse instance-weighting ensemble based on region drift disagreement for concept drift adaptation," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 293–307, 2020.

[46] G. Ditzler and R. Polikar, "Incremental learning of concept drift from streaming imbalanced data," *IEEE transactions on knowledge and data engineering*, vol. 25, no. 10, pp. 2283–2301, 2012.

[47] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 3354–3361.

[48] A. Vyas, R. Kannao, V. Bhargava, and P. Guha, "Commercial block detection in broadcast news videos," in *Proceedings of the 2014 Indian Conference on Computer Vision Graphics and Image Processing*, 2014, pp. 1–7.

[49] J. Montiel, J. Read, A. Bifet, and T. Abdessalem, "Scikit-multiflow: A multi-output streaming framework," *Journal of Machine Learning Research*, vol. 19, no. 72, pp. 1–5, 2018. [Online]. Available: http://jmlr.org/papers/v19/18-251.html

En Yu received his Bachelor's from the School of Information Science and Engineering, Shandong Normal University, Jinan, Shandong, China, in 2016 and his Master's degree in 2020. From January 2019 to January 2020, he was a Visiting Student with the Language Technology Institute, School of Computer Science, Carnegie Mellon University (CMU), Pittsburgh, PA, USA. He is currently working toward a Ph.D. degree with the Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. His research interests include adaptive learning under concept drift, data stream mining and fuzzy learning.

Jie Lu (F'18) is an Australian Laureate Fellow, IFSA Fellow, Distinguished Professor and the Director of Australian Artificial Intelligence Institute (AAII) at the University of Technology Sydney, Australia. She received a PhD degree from Curtin University in 2000. Her main research expertise is in transfer learning, concept drift, fuzzy systems, decision support systems and recommender systems. She has published over 500 papers in IEEE Transactions and other leading journals and conferences. She is the recipient of two IEEE Transactions on Fuzzy Systems Outstanding Paper Awards (2019 and 2022), NeurIPS2022 Outstanding Paper Award, Australia's Most Innovative Engineer Award (2019), Australasian Artificial Intelligence Distinguished Research Contribution Award (2022), Australian NSW Premier's Prize on Excellence in Engineering or Information & Communication Technology (2023), and the Officer of the Order of Australia (AO) 2023.

Guangquan Zhang is an Australian Research Council (ARC) QEII Fellow, Associate Professor and the Director of the Decision Systems and e-Service Intelligent (DeSI) Research Laboratory at the Australian Artificial Intelligence Institute, University of Technology Sydney, Australia. He received his PhD in applied mathematics from Curtin University, Australia, in 2001. From 1993 to 1997, he was a full Professor in the Department of Mathematics, Hebei University, China. His main research interests lie in fuzzy multi-objective, bilevel and group decision making, fuzzy measures, transfer learning and concept drift adaptation. He has published six authored monographs and over 500 papers including some 300 articles in leading international journals. He has supervised 40 PhD students to completion and mentored 15 Postdoc fellows. Prof Zhang has won ten very competitive ARC Discovery grants and many other research projects. His research has been widely applied in industry.