Utilizing an Ensemble Machine Learning Framework for Handling Concept Drift in Spatiotemporal Data Streams Classification

Ature Angbera[*1,2], Huah Yong Chan[*1]
[1]Universiti Sains Malaysia, School of Computer Sciences, Pulau Pinang 11800, Malaysia
[2]Joseph Sarwuan Tarka University, Department of Computer Science, Makurdi, Nigeria
E-mail: angberaature@student.usm.my, hychan@usm.my

Keywords: spatiotemporal data stream, concept drift, ensemble learning, machine learning, spatiotemporal data classification

Received: May 18, 2023

The number of systems and devices broadcasting spatiotemporal data has recently significantly increased. Streaming data analytics provides the foundation of various spatiotemporal data services and functions. The non-stationary characteristics of these platforms and the constantly altering trends of the spatiotemporal data streams present concept drift issues for spatiotemporal data analytics. As a result, when concept drift occurs, it harms the model. The model's performance will eventually decline. The learning algorithms need the proper adaptive techniques to deal with concept drift on the spatiotemporal data streams with accurate predictions. This paper proposes an average weighted performance ensemble model (AWPEM). The AWPEM framework is for drift adaptation for spatiotemporal data stream classification. The framework is evaluated using real-world spatiotemporal data and compared to other state-of-the-art methods. The results show that the proposed framework outperforms other methods in terms of classification accuracy and robustness to concept drift. Further research will focus on enhancing the adaptability of the proposed framework to diverse and dynamic spatiotemporal data environments, exploring mechanisms for automated parameter tuning, investigating computational efficiency and scalability to large-scale spatiotemporal datasets.

Povzetek: Za večjo robustnost učnih sistemov je razvit ansambel WPEM, povprečno uteženi performančni ansambel za prilagoditev konceptualnim spremembam pri klasifikaciji spatiotemporalnih podatkovnih tokov.

1 Introduction

In recent years, it has become evident that the volume of data generated by technologies such as social media, sensor data, and other sources is rapidly increasing. Particularly, spatiotemporal data is streamed and vastly outpaces analysis tools' memory and processing power. There are currently 2.7 zeta bytes of data in the digital realm, which is growing daily [1]. The amount of data produced by systems like email and network monitoring [2], forecasting air quality grade [3], assessing credit risk [4], and analysing mobile users' behaviour changes [5] are unfit for disc storage because they are so huge. In light of this, streaming algorithms are made to process data as it comes in, online, and without storing a sizable amount of data in the main memory. As a result, real-time analytics on non-stationary data have recently caught researchers' attention. Spatiotemporal data streams are data collections that flow continuously and alter as they enter a system. According to [6], data streams can be enormous, timely ordered, changing quickly, and potentially endless in duration. Due to the periodic data changes in the streaming spatiotemporal data, the typical mining method is faced with the problem of concept drift [7]. The mining algorithm needs to be upgraded. Concept drift is a problematic issue in online learning since it significantly affects the consistency of streaming data classification [8]. If it goes undiscovered, concept drift can negatively impact the accuracy of predictions. We can handle distributional changes and maintain great accuracy of the prediction over time by employing concept drift detection models.

Assume that X and Y are the random variables representing the streaming observations and the labels that go with them. According to [9], concept drift is analogous to a change in the joint probability P(Y, X) at different time steps $t, z \in \{1, \dots, T\}$, that is:

$$P_t(Y,X) \neq P_z(Y,X)$$
$$\Leftrightarrow P_t(Y|X)P_t(X) \neq P_z(Y|X)P_z(X)$$

At time step t, $P_t(Y,X)$ is referred to as the active concept. We also distinguish between drift in actual and drift in virtual concepts. A shift in $P(X)$, that is: $P_t(X) \neq P_z(X)$, is referred to as virtual concept drift (X). Virtual concept drift is, therefore, unrelated to the target distribution and has no impact on the decision boundary [9]. Real concept drift, also known as concept shift, on the other hand, refers to a change in the conditional target distribution, that is: $P_t(Y|X) \neq P_z(Y|X)$. Real concept drift moves the decision border, which could affect predictions in the future [9]. In order to prevent severe declines in prediction performance, it is imperative to spot changes in $P(Y|X)$ in a timely manner.

Sadly, concept drift does not exhibit a consistent trend in actual practice. Instead, we might see significant variations in concept drift's length and intensity. To achieve this, we distinguish various categories of concept drift as sudden or abrupt, gradual or reoccurring [1]. Figure 1 depicts the types or categories of concept drift in streaming data.

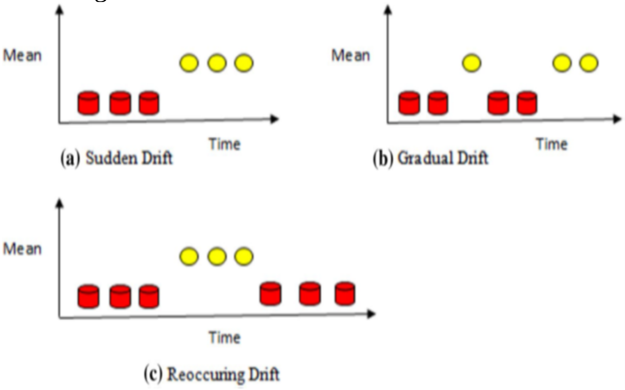

Figure 1: Types of concept drift [1]

This study investigates a precise and trustworthy concept drift approach in spatiotemporal data streams. The work suggests an average weighted performance ensemble model (AWPEM) for efficient concept drift detection and enhanced classification in spatiotemporal crime. The suggested model is an ensemble learning framework built on base learners and integrates two cutting-edge drift adaptation strategies. The two drift adaptation approaches employed are adaptive random forest (ARF) [10] and streaming random patches (SRP) [11]. As drift detectors, three popular drift detection approaches are used: adaptive windowing (ADWIN) [12], early drift detection method (EDDM) [13], and drift detection method (DDM) [14]. In order to build a strong spatiotemporal crime classification ensemble model with improved drift adaptability capability, the fundamental learners are combined and weighted based on their real-time success. The following are the study's ultimate goals:

i. Concept drift adaption techniques are examined.

ii. It suggests the AWPEM, a novel drift adaption technique, to solve the performance issues with the current concept drift techniques in spatiotemporal data streaming.

iii. Using two datasets, it assesses the proposed AWPEM framework for spatiotemporal crime predictions with concept drift detection and adaptation.

2 Related work

This part reviews previous efforts on spatiotemporal dataset (crime as a use case) analysis and provides an overview of cutting-edge techniques for concept drift detection and adaptation.

2.1 Spatiotemporal crime analysis

The interest in spatiotemporal modelling is a rising subject of open research [15]. The dynamic interaction between space and time allows for discovering relevant patterns through spatial-temporal data mining [16]. In [17] present a predictive strategy based on spatial analysis and auto-regressive models to effectively anticipate crime patterns in each place and automatically identify high-risk crime locations in urban areas. The algorithm's output is a spatiotemporal crime forecasting model, which includes several crime-prone areas and related crime predictors. Each of these predictor's functions as a predictive model to estimate the likelihood that crimes will occur to the extent it is allocated. In [18], the XGBoost machine learning algorithm was used to predict crime using actual crime and environmental data. The predictions were then interpreted using the SHAP method. With the aid of past crime statistics and environmental variables, XGBoost forecasts future crime. The prediction was then analyzed using SHAP, a machine learning interpreter, to show each variable's contributions from a global and local perspective. A cutting-edge deep learning technique called "Geographic-Semantic Ensemble Neural Network (GSEN)," which stacks a geographic prediction neural network with a semantic prediction neural network, was proposed [19]. The GSEN model combines the "Predictive Recurrent Neural Network (PredRNN), Graph Convolutional Predictive Recurrent Neural Network (GC-PredRNN)," and Ensemble Layer structures for spatiotemporal crime classifications in order to capture spatiotemporal dynamics from a variety of angles. The Fuzzy K-Nearest Neighbor algorithm and geospatial operations were utilised to create a crime prediction model in [20], and the study recommended integrating a Safe Route Travel app with a Crime Mapping system. Based on prior crime data, the model forecasts the type of crime a location is most likely to experience. In [21] suggested using deep inception-residual networks (DIRNet) to develop precise forecasts of theft-related crime based on data from non-emergency service requests (311 events). The outcomes demonstrated that the DIRNet outperforms alternative prediction models, averaging an F1 of 71% on average.

For spatiotemporal crime predictions, the strategies mentioned above produce excellent results. However, these are static machine-learning models created for offline learning. They are unable to adapt to real-time changes in spatiotemporal data. Due to this drawback, they are useless when used in real-world crime prediction systems.

2.2 Methods of concept drift

Concept drift problems are typically encountered in spatiotemporal streaming data analysis when data distribution changes over time due to the non-stationary environment. Concept drift problems often cause spatiotemporal crime prediction models to perform poorly, which has profound security implications. According to the rate at which the data distribution changes, concept drifts are categorized into abrupt and gradual drifts [12]. An efficient crime prediction model should swiftly adjust to the discovered drifts to address concept drift and retain high prediction accuracy [5], [22].

i. Detection of Concept Drift

A well-known prototype result-based methodology called the drift detection method (DDM) establishes dual parameters, a "*warning level*" and a "*drift level*," to track changes in the prototype's *"error rate and standard deviation"* for drift detection [18]. The concept drift presence is noticed by a substantial rise in the prototype's overall error rate and DDM standard deviation. Although a learner will be modified if its results dramatically worsen, the drift detection method is easy to work with and can prevent pointless model modifications. DDM can detect abrupt drifts efficiently but frequently responds slowly to gradual drifts; this is because memory overflows are brought on by the necessity to retain a sizable number of data samples to reach the drift level of a long, gradual drift [23], [24].

The Early Drift Detection Method (EDDM) keeps track of the separation between two successive errors to identify gradual drift. When there is little space between two subsequent errors, gradual drift occurs [25]. The separation between two misclassification errors should grow as predictions get more accurate. The process for resizing a window is the same as with DDM. The main problem with EDDM is that at least 30 mistakes must be included in the calculation, making it difficult to utilize with unbalanced datasets [23]. The EDDM is an improvement of the DDM; hence the DDM could not detect gradual drift.

The Adaptive sliding window (ADWIN) algorithm separates the data stream into two sequential sub-windows within a variable-size sliding window and compares the means [26]. Drift is discovered when the difference between the means exceeds a threshold value determined using the Hoeffding bound [27]. All previous data samples from before the discovered drift point are discarded once the drift point is identified [23]. Due to the sliding window's capacity to be stretched to a large windowing size to notice long shifts, ADWIN successfully identifies progressive drifts. The mean value, however, is often not a good indicator of change.

ii. Concept Drift Adaptation:

After drift detection, a suitable drift adaptation strategy should be employed to handle the found drifts and maintain strong learning performance. Incremental learning and ensemble procedures are the two most common types of current drift adaptation strategies. The learning model is incrementally modified by studying each sample one at a time in chronological order. The Hoeffding tree (HT), which employs the Hoeffding constraint, is a particular kind of "decision tree (DT)" whereby data streams may incrementally adapt [23]. Instead of using a decision tree to choose the optimal schism, the hoeffding tree uses the "Hoeffding bound" to determine how many samples are needed to choose the split node. Now that its node has been updated, the HT can adapt to new samples. However, the HT lacks any processes for handling certain types of drift. A modernized version of the HT, the Hoeffding Anytime Tree (HATT), often referred to as the "Extremely Fast Decision Tree (EFDT)" [28], instead than waiting to find the best split in the HT, splits nodes when it reaches the confidence threshold. The EFDT can respond better to concept drifts than the hoeffding tree thanks to this division mechanism, albeit its efficiency could be enhanced [29].

Ensemble learning approaches have been suggested as a way to improve concept drift adaption and create reliable learners for data stream analytics. Block-based and online ensembles are two additional categories for ensemble approaches [30], [31]. Block-based ensembles partition the data streams into fixed-size blocks, after which each block is trained with a base learner. The base learners will be assessed and modified each time a new block is released. Although they typically take longer to react to abrupt drifts, block-based ensembles react to drifts gradually and properly. Deciding on the right block dimension to allow for a drift reaction speed and the learning trade-off between the base learners' results is another issue with block-based ensemble systems [31]. Three popular block-based ensembles are "Accuracy Updated Ensemble (AUE), Accuracy Weighted Ensemble (AWE), and Streaming Ensemble Algorithm (SEA)" [5], [32]. Online ensembles include different incremental learning models, such as HTs, to enhance learning performance. The adaptive random forest (ARF) approach, developed by Gomes et al. [10], uses HTs as base learners and ADWIN as each tree's drift detector. The drift detection mechanism replaces the underperforming base trees with new trees that better fit the new concept. Since the random forest is also an effective machine learning algorithm, ARF frequently outperforms many other approaches. ARF also includes a powerful resampling method and the flexibility to accommodate various drifts. For streaming data analytics, Gomes et al. [11] also put forth the Streaming Random Patches (SRP) innovative adaptive ensemble approach. To create predictions, SRP combines the online bagging method and random subspace. SRP and ARF use the same technology, but SRP uses a global subspace randomization strategy, while ARF uses a local subspace randomization approach. The more adaptable global subspace randomization strategy increases the diversity of base learners. SRP frequently has better prediction accuracy than ARF, although its execution time is often longer [29].

Although numerous concept drift adaptation techniques are now in use, their prediction accuracy and drift reaction time are performance limited. Due to their weak capacity to react to drift and low model complexity, incremental learning approaches frequently perform poorly. In contrast, block-based ensembles face significant difficulties in determining block size and drift reaction speed. Online ensembles like ARF and SRP consistently outperform block-based ensembles and incremental learning; however, their randomization strategies also cause unstable learning models by adding more unpredictability to building their models. In order to increase drift adaptation performance, this research proposes an ensemble model that is stable and reliable.

3 Proposed model framework

Figure 2 depicts a high-level view of the proposed system for spatiotemporal crime prediction based on data stream analytics. The primary steps are listed below. First, generated streams of spatiotemporal data are pre-processed. Second, the critical learners for the first classification of crimes and drift adaptation are constructed using six drift adaptation techniques of concept drift: "ARF+DDM, ARF+EDDM, ARF+ADWIN, SRP+EDDM, SRP+DDM, and SRP+ADWIN." The ensemble approach is then created by merging the six base learners' probabilities of their prediction according to the specified AWPEM architecture. The ensemble model is employed, which can predict crime and adjust to concept drifts.

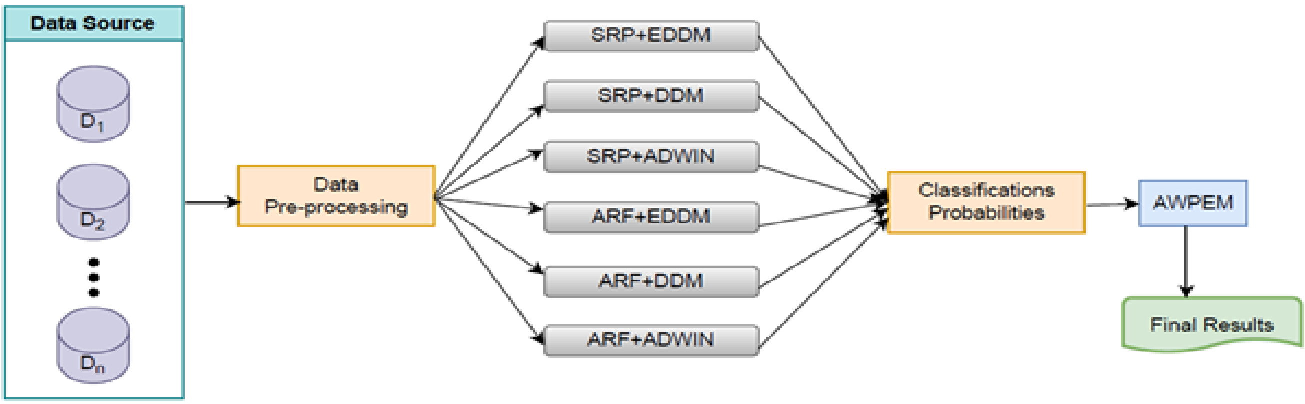

Figure 2: Proposed model framework

Three fundamental approaches for detecting drift (ADWIN, DDM, and EDDM) and two cutting-edge techniques for adjusting drift (ARF and SRP) are combined to create a robust ensemble model. Innovative methods for drift adaption, "ARF and SRP," have demonstrated superior performance to other drift adaptation techniques in experimental tests observed from literature [33], [34]. This motivated us to adapt and use them in our proposed model as our concept drift adaption techniques. Additionally, as was already mentioned, the concept drift detectors (ADWIN, DDM, and EDDM) employed in this work have the advantage of detecting both sudden and gradual drift. As a result, the suggested ensemble model can identify both types of drift effectively thanks to both drift detection techniques.

The average weighted performance ensemble model (AWPEM), a unique ensemble technique, is proposed in this study as a means of integrating the base learners for spatiotemporal data stream analytics. Many other ensemble techniques combine fixed weights, whereas AWPEM gives learners adjustable weights based on how they perform at the moment. Assuming that the goal attribute contains c various labels, $y \in 1, \ldots, c$ for each x data input and that the data stream $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. It is possible to write the target class predicted by AWPEM as seen in Equation 1.

$$\hat{y} = \underset{i \in \{1,\ldots,c\}}{argmax} \frac{\sum_{j=1}^{k} w_j P_j(y = i | L_j, x)}{k} \tag{1}$$

$P_j(y = i | L_j, x)$ is the likelihood that the value of the class I will appear in the data sample x utilizing, j_{th} base learner, L_j. Each weight of the L_j is w_j and the suggested model base learners' number is k (i.e., $k = 6$).

After each data sample has been processed, the instantaneous error rate (see equation 2) is determined by dividing the sum of incorrectly classified labels by the overall number of samples analyzed.

$$ErrorRate = \frac{Total\ number\ of\ misclassified\ samples}{Total\ number\ of\ processed\ samples} \tag{2}$$

Each base learner's weight w_j, is obtained by taking the inverse of the real-time error rate. The w_j is denoted as given in Equation 3.

$$w_j = \frac{1}{ErrorRate + \epsilon} \tag{3}$$

Where ϵ (epsilon) represents a little constant that serves as a safeguard against the denominator falling to zero.

The idea of real-time error rates is employed in our proposed AWPEM model to create the base learners' weights for further dependable adaptation to data changes as opposed to the "mean square error rates" of the blocks of data utilized in the accuracy updated ensemble model. The ensemble model can account for the outcome of the base learners individually on a particular job due to the actual-time error rate on all compiled data.

The AWPEM provides several benefits since it can enhance the weights of base learners who perform at the top of their class while also considering other base learners. The suggested model's inverse-based weighting operation has improved AUE's weighting operation.

Additionally, the flexible weights generated with "real-time error rates" can be utilized to change the relevance of the base learners concerning their instantaneous results, in contrast to the fixed weights used in many existing ensemble techniques. These shifting weights guarantee that the present base learners who excel at the highest levels will be awarded higher weights.

4 Experiments and results

The Scikit-Multiflow [35] framework was expanded to implement the suggested framework using Python 3.9 on a computer with a core i5 processor.

4.1 Data pre-processing

We used two datasets for our model training and testing. These are crime datasets; the first dataset (Dataset1) has 319073 rows × 16 columns. The prediction class is the OFFENSE, whether it was violent or nonviolent. The second dataset (Dataset2) has 28303 rows × 20 columns. The prediction class here is whether there was an attack or no attack. In the hold-out evaluation, the initial model training took up 10% of the data, while the testing portion took up the remaining 90%. Before using the learning model for model training and updating, the learning model is tested via prequential validation, sometimes referred to as test-and-train validation. The five criteria, "accuracy, precision, recall, f1-score, and execution time," were utilized to assess the effectiveness of the suggested approach.

4.2 Results and discussion of the experiment

Prediction accuracies of the base learners (six) in the suggested model were lower than that of the proposed AWPEM on both datasets used in the experiments.

Figure 3 depicts the accuracies of the crime predictions of the base learning models of ARF+ADWIN and ARF+DDM on both datasets used in this study. Figures 3(a) and 3(b) show that the base learners' model accuracies were 86.33% and 98.43% on both datasets, respectively. Figure 3(c) and 3(d) prediction accuracies were 86.93% and 98.70% on both datasets, respectively.

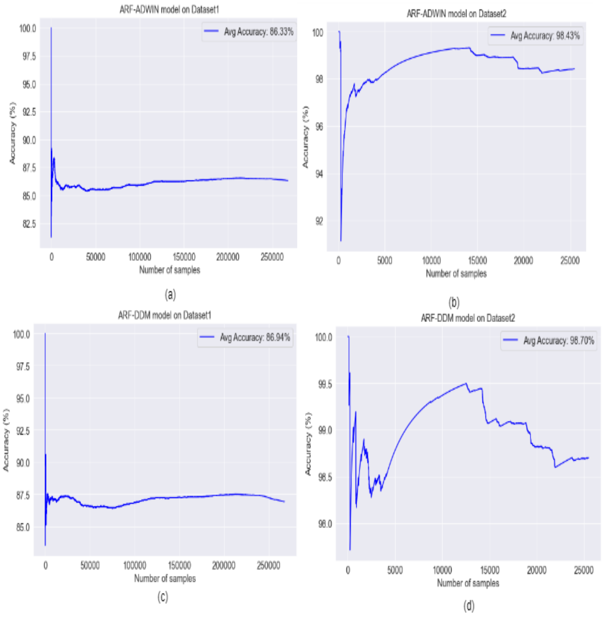

Figure 3: Accuracies of ARF+ADWIN and ARF+DDM, both datasets.

Figure 4 depicts the accuracies of the crime predictions of the base learning models of ARF+EDDM and SRP+ADWIN on both datasets used in this study. Figures 4(a) and 4(b) show that the base learners' model accuracies were 87.38% and 98.75% on both datasets, respectively. Figure 4(c) and 4(d) prediction accuracies were 80.56% and 98.56% on both datasets, respectively.

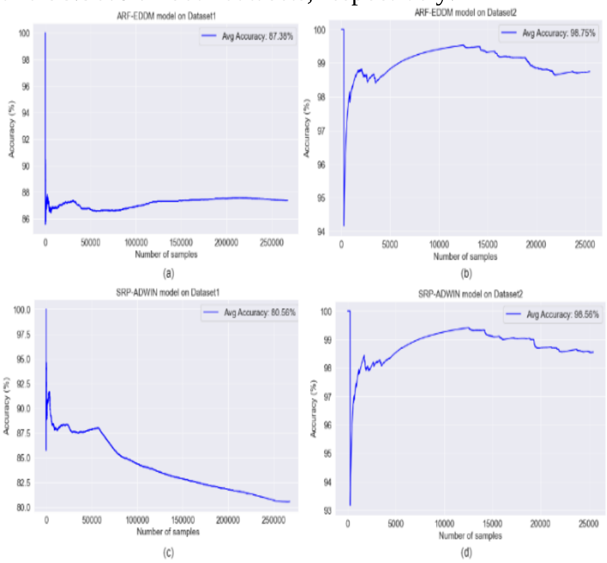

Figure 4: Accuracies of ARF+EDDM and SRP+ADWIN, both datasets

Figure 5 depicts the accuracies of the crime predictions of the base learning models of ARF+EDDM and SRP+ADWIN on both datasets used in this study. Figures 5(a) and 5(b) show that the base learners' model accuracies were 95.01% and 98.43% on both datasets, respectively. Figure 5(c) and 5(d) prediction accuracies were 95.00% and 98.64% on both datasets, respectively.

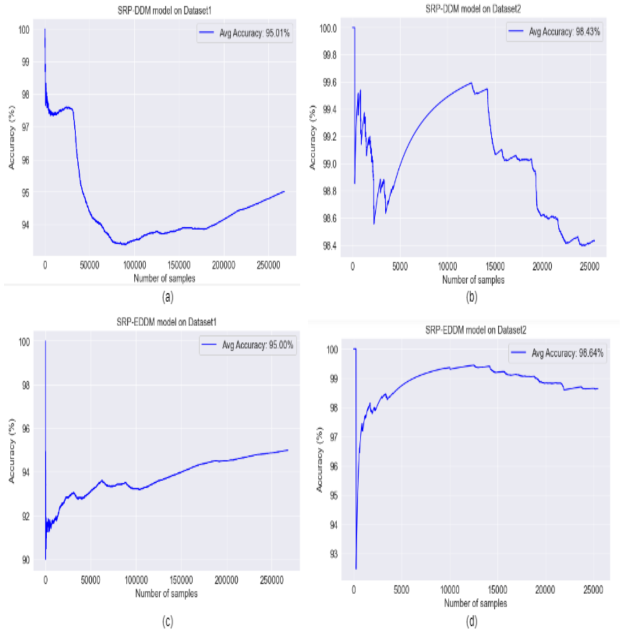

Figure 5: Accuracies of SRP+DDM and SRP+EDDM both datasets

Table 1: Comparison of the effectiveness of drift adaptation techniques on Dataset1

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Average time (ms)
ARF+ADWIN	86.33	76.58	86.80	81.37	6.07
ARF+DDM	86.93	77.17	88.08	82.26	11.87
ARF+EDDM	87.38	78.13	87.90	82.73	12.4
SRP+ADWIN	80.56	70.92	73.66	72.26	22.28
SRP+DDM	95.01	91.33	94.47	92.87	20.88
SRP+EDDM	95.00	91.84	93.77	92.80	22.18
HT	84.49	71.09	92.49	80.39	1.17
LB	92.48	86.29	92.87	89.46	29.07
Proposed AWPEM	98.45	97.87	97.63	97.75	35.08

Table 2: Comparison of the effectiveness of drift adaptation techniques on Dataset2

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Average time (ms)
ARF+ADWIN	98.43	96.15	94.63	95.38	0.20
ARF+DDM	98.70	97.36	95.04	96.18	0.25
ARF+EDDM	98.75	96.71	95.98	96.34	0.23
SRP+ADWIN	98.56	96.52	95.02	95.76	1.02
SRP+DDM	98.43	96.34	94.45	95.39	1.33
SRP+EDDM	98.64	96.62	95.43	96.02	1.12
EFDT	91.02	69.35	85.54	76.60	0.41
HT	91.61	73.03	81.11	76.86	0.09
LB	97.79	92.76	94.54	93.64	0.02
Proposed AWPEM	99.24	98.54	97.01	97.77	4.20

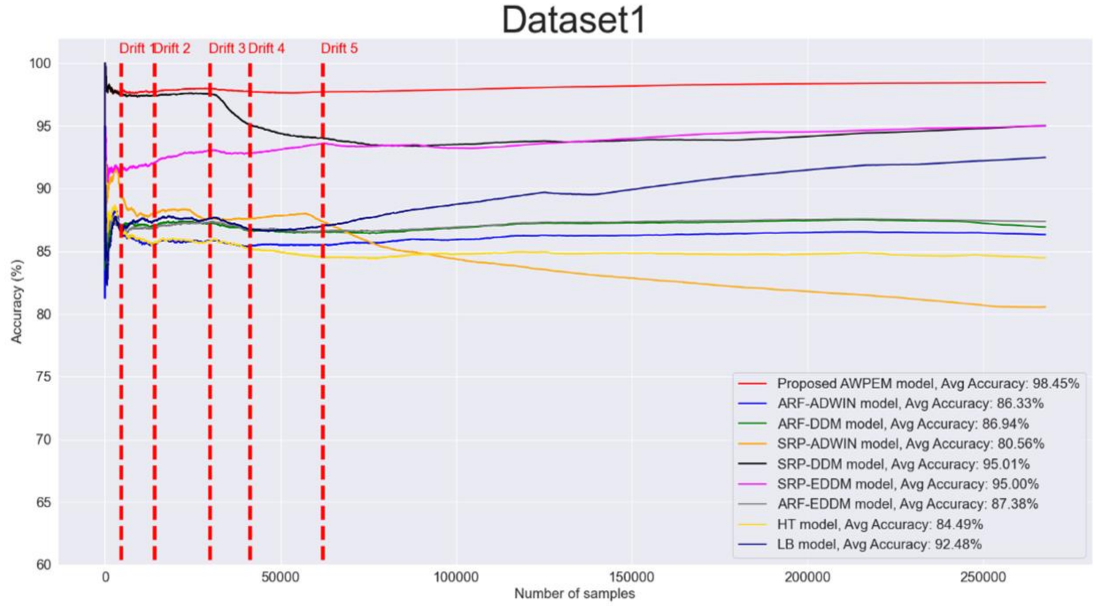

Figure 6: Accuracy comparison of drift adaptation methods on dataset1

The suggested AWPEM framework's performance is compared in Tables I and II to other cutting-edge drift adaptive methods presented in this paper, such as ARF, SRP, HT, EFDT, & LB. The suggested AWPEM technique performs better than all other models, as demonstrated in Tables I and II. On dataset1, from Figure 6, five drifts were seen early on; these drifts were noticed as a result of an increase in crime occurrences. Despite having different levels of adaptability, all the methods used could adjust to the drifts swiftly. However, our proposed AWPEM method swiftly adapted to the drifts and maintained a higher accuracy of 98.45%.

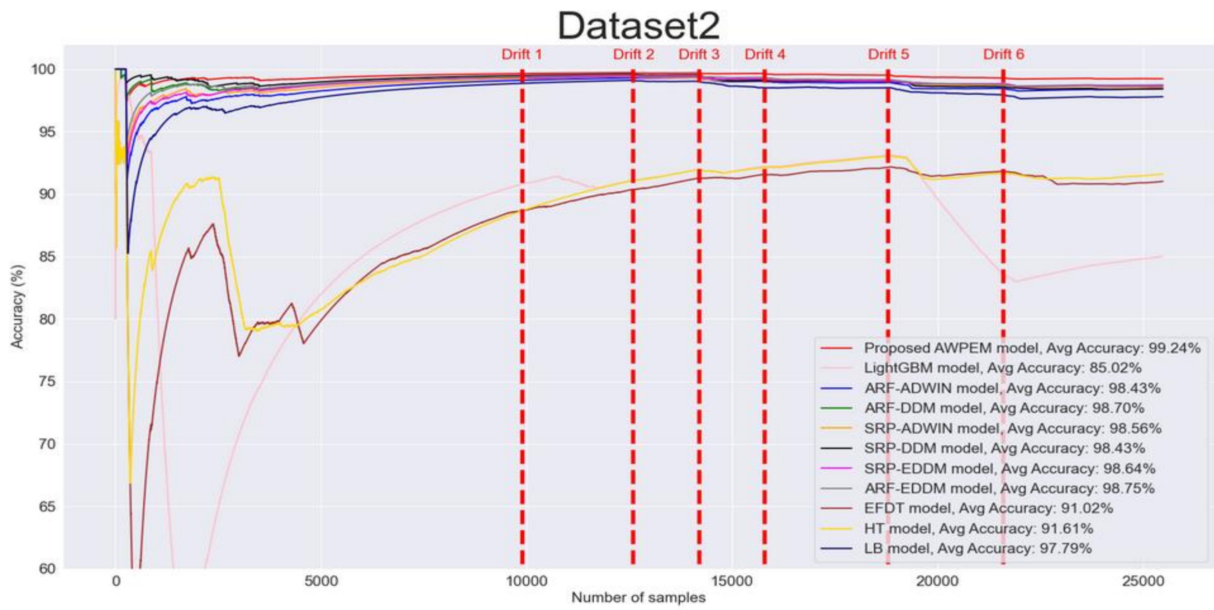

Figure 7: Comparison of drift adaption techniques' accuracy on dataset 2

There were six concept drifts in the tests for dataset2, as depicted in Figure 7. Both sudden (drifts 1, 3, and 5) and gradual drifts were seen in this series (drifts 2, 4, 6); these changes resulted from increased crime committed at various points. Once more, the suggested AWPEM approach quickly corrected for drifts and preserved a greater accuracy of 99.24 %. The justification for selecting the six base learners as base learners is further strengthened by their superior performance compared to other cutting-edge drift adaption techniques.

Figure 8 illustrates how the suggested AWPEM strategy takes longer to execute on datasets 1 and 2 than the other tested methods, but the mean time execution remains reasonable for each occurrence. M1, M2, M3, M4, M5, and M6 in figure 8 stand for, respectively, "ARF+ADWIN, ARF+DDM, ARF+EDDM, SRP+ADWIN, SRP+DDM, and SRP+EDDM".

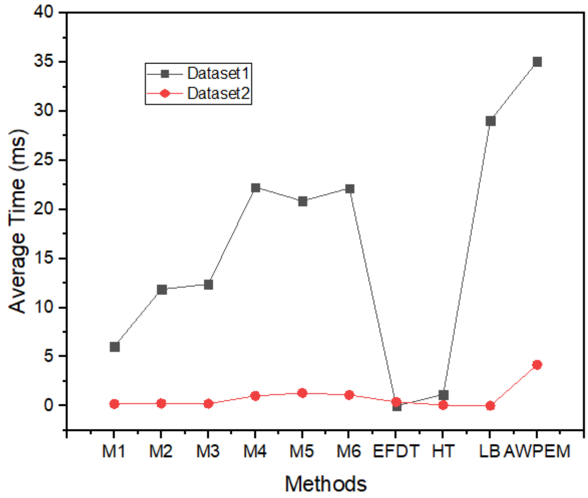

Figure 8: Average time comparison of the models on the two datasets.

Overall, while the average time taken by AWPEM may be higher compared to some other models, its outstanding performance across multiple metrics demonstrates its efficacy in addressing concept drift. Additionally, the difference in processing time between the two datasets highlights the importance of considering dataset characteristics and computational resources when selecting an appropriate drift adaptation technique.

Certainly, addressing the computational efficiency of the AWPEM model can be a promising avenue for future research. While AWPEM demonstrates exceptional performance in handling concept drift, its relatively longer average processing time, especially in Dataset1, suggests room for improvement in terms of computational efficiency. One potential direction for future studies could involve optimizing the algorithmic implementation of AWPEM to reduce its computational overhead without compromising its predictive accuracy. This optimization may involve refining the model architecture, streamlining the feature selection process, or implementing more efficient data processing techniques. Furthermore, exploring parallelization and distributed computing strategies could help accelerate the processing speed of AWPEM, making it more scalable and suitable for handling large-scale spatiotemporal datasets in real-time applications.

In terms of precision, recall, and f1-score, Figures 9 and 10 compare the suggested method to state-of-the-art techniques. The suggested model's precision shows that it can correctly recognize and predict positive data samples even when drift occurs. On the other hand, current methods consider the classification model's response to the input data at hand, resulting in a wrong prediction when concept drifts occur.

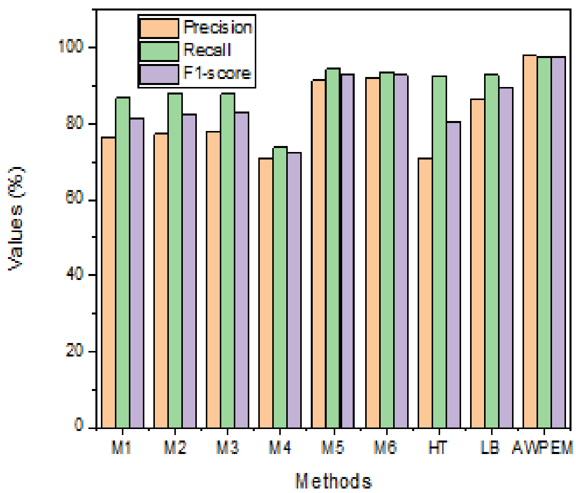

Figure 9: Performance comparison with precision, recall, and f1-score of the methods on dataset1

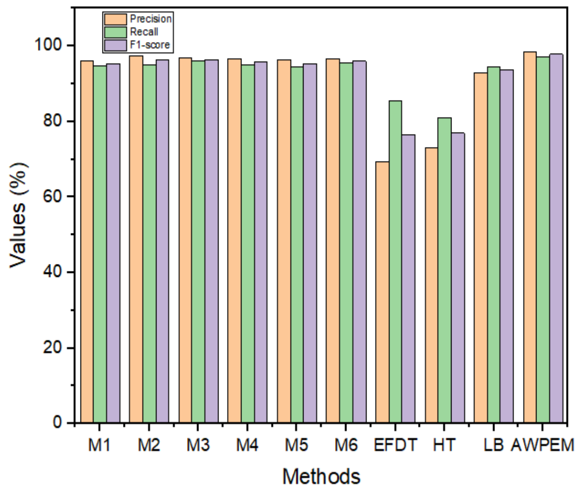

Figure 10: Performance comparison with precision, recall, and f1-score of the methods on dataset2

Recall demonstrates the capacity of the suggested model to generate fewer false negatives. For dataset1 and dataset2, the proposed model outperformed earlier techniques in the recall by 97.63% and 97.01%, respectively. When false negatives and false positives are critical, and the class distribution in the real-world dataset is uneven, the F1-score is calculated and compared. For datasets 1 and 2, the proposed model beat existing techniques by 97.75% and 97.77% f1-score.

5 Conclusion

This research offers the AWPEM framework for drift adaptive for spatiotemporal data streams classification, built on a collection of cutting-edge drift adaptation techniques. The proposed framework predicted crime in response to adaptation to the changes in data (concept drift) by attaining 98.45% and 99.24% accuracies on the two datasets. These accuracies' performance is significantly higher than other cutting-edge algorithms' accuracies based on the experimental performance of the two spatiotemporal data streams. To expand it, more drift adaption strategies that offer better performance, diversity, and speed can be added to the proposed framework in subsequent studies. Also, it can be further optimized to reduce its high computational efficiency.

Data availability statement

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

[1] S. Priya and R. A. Uthra, "Comprehensive analysis for class imbalance data with concept drift using ensemble based classification," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 5, pp. 4943–4956, 2020, http://dx.doi.org/10.1007/s12652-020-01934-y.

[2] K. Pradeep Mohan Kumar, M. Saravanan, M. Thenmozhi, and K. Vijayakumar, "Intrusion detection system based on GA-fuzzy classifier for detecting malicious attacks," Concurrency and Computation: Practice and Experience, vol. 33, no. 3, pp. 5–10, 2019, http://dx.doi.org/10.1002/cpe.5242.

[3] W. Liu, H. Zhang, and Q. Liu, "An Air Quality Grade Forecasting Approach Based on Ensemble Learning," Proceedings - 2019 International Conference on Artificial Intelligence and Advanced Manufacturing, AIAM 2019, pp. 87–91, 2019, http://dx.doi.org/10.1109/aiam48774.2019.00024 .

[4] Z. Li, Y. Tian, K. Li, F. Zhou, and W. Yang, "Reject inference in credit scoring using Semi-supervised Support Vector Machines," Expert Systems with Applications, vol. 74, pp. 105–114, 2017,. http://dx.doi.org/10.1016/j.eswa.2017.01.011

[5] 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, 2019, http://dx.doi.org/10.1109/tkde.2018.2876857.

[6] J. Gama, "Knowledge Discovery from Data Streams (1st ed.)," in Chapman and Hall/CRC, New York, 2010, p. 255, http://dx.doi.org/10.1201/ebk1439826119-c1.

[7] C. Raab, M. Heusinger, and F. M. Schleif, "Reactive Soft Prototype Computing for Concept Drift Streams," Neurocomputing, vol. 416, pp. 340–351, 2020, http://dx.doi.org/10.1016/j.neucom.2019.11.111.

[8] L. Yang, Y.-M. Cheung, and Y. Yan Tang, "Adaptive chunk-based dynamic weighted majority for imbalanced data streams with concept drift," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 8, pp. 2764–2778, 2020, http://dx.doi.org/10.1109/tnnls.2019.2951814.

[9] J. Gama, I. Zliobaite, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A Survey on Concept Drift Adaptation," *ACM Computing Surveys*, vol. 1, no. 1, p. 35, 2013, http://dx.doi.org/10.1145/2523813

[10] H. M. Gomes *et al.*, "Adaptive random forests for evolving data stream classification," *Machine Learning*, vol. 106, no. 9–10, pp. 1469–1495, 2017, http://dx.doi.org/10.1007/s10994-017-5642-8.

[11] H. M. Gomes, J. Read, and A. Bifet, "Streaming random patches for evolving data stream classification," *Proceedings - IEEE International Conference on Data Mining, ICDM*, vol. 2019-Novem, no. Icdm, pp. 240–249, 2019, http://dx.doi.org/10.1109/icdm.2019.00034.

[12] A. Bifet and R. Gavaldà, "Learning from time-changing data with adaptive windowing," in *Proceedings of the 7th SIAM International Conference on Data Mining*, 2007, pp. 443–448, http://dx.doi.org/10.1137/1.9781611972771.42.

[13] M. Baena-García, J. del Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavaldà, and R. Morales-Bueno, "Early Drift Detection Method," *4th ECML PKDD International Workshop on Knowledge Discovery from Data Streams*, vol. 6, pp. 77–86, 2006, [Online]. Available: http://www.lsi.upc.edu/~abifet/EDDM.pdf.

[14] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3171, pp. 286–295, 2004, http://dx.doi.org/10.1007/978-3-540-28645-5_29.

[15] R. de Medrano and J. L. Aznarte, "A spatio-temporal attention-based spot-forecasting framework for urban traffic prediction," *Applied Soft Computing Journal*, vol. 96, 2020, http://dx.doi.org/10.1016/j.asoc.2020.106615.

[16] A. Hamdi, K. Shaban, A. Erradi, A. Mohamed, S. K. Rumi, and F. D. Salim, *Spatiotemporal data mining: a survey on challenges and open problems*, no. 0123456789. Springer Netherlands, 2021, http://dx.doi.org/10.1007/s10462-021-09994-y.

[17] C. Catlett, E. Cesario, D. Talia, and A. Vinci, "Spatio-temporal crime predictions in smart cities: A data-driven approach and experiments," *Pervasive and Mobile Computing*, vol. 53, pp. 62–74, 2019, http://dx.doi.org/10.1016/j.pmcj.2019.01.003.

[18] X. Zhang, L. Liu, M. Lan, G. Song, L. Xiao, and J. Chen, "Interpretable machine learning models for crime prediction," *Computers, Environment and Urban Systems*, vol. 94, no. January, p. 101789, 2022, http://dx.doi.org/10.1016/j.compenvurbsys.2022. 101789.

[19] G. Jin, H. Sha, Y. Feng, Q. Cheng, and J. Huang, "GSEN: An ensemble deep learning benchmark model for urban hotspots spatiotemporal prediction," *Neurocomputing*, vol. 455, pp. 353–

367, 2021, http://dx.doi.org/10.1016/j.neucom.2021.05.008.

[20] S. Sharma, A. Uniyal, P. Srinivasan, and S. Chaudhari, "Fuzzy Based Geo-Spatial Crime Category Prediction for Crime Mapping and Safe Route Travel," in *2022 IEEE Region 10 Symposium (TENSYMP)*, 2022, pp. 1–6, http://dx.doi.org/10.1109/tensymp54529.2022.9864342.

[21] X. Ye, L. Duan, and Q. Peng, "Spatiotemporal Prediction of Theft Risk with Deep Inception-Residual Networks," *Smart Cities*, vol. 4, no. 1, pp. 204–216, 2021, http://dx.doi.org/10.3390/smartcities4010013.

[22] R. C. Samant, S. H. Patil, R. N. Sinha, and A. K. Kadam, "A Systematic Ensemble Approach for Concept Drift Detector Selection in Data Stream Classifiers," *International Journal of Engineering Trends and Technology*, vol. 70, no. 9, pp. 119–130, 2022, http://dx.doi.org/10.14445/22315381/ijett-v70i9p212.

[23] S. Wares, J. Isaacs, and E. Elyan, "Data stream mining: methods and challenges for handling concept drift," *SN Applied Sciences*, vol. 1, no. 11, pp. 1–19, 2019, http://dx.doi.org/10.1007/s42452-019-1433-0.

[24] P. Wang, N. Jin, W. L. Woo, J. R. Woodward, and D. Davies, "Noise tolerant drift detection method for data stream mining," *Information Sciences*, vol. 609, pp. 1318–1333, 2022, http://dx.doi.org/10.1016/j.ins.2022.07.065.

[25] C. H. Tan, V. C. S. Lee, and M. Salehi, "Information resources estimation for accurate distribution-based concept drift detection," *Information Processing and Management*, vol. 59, no. 3, p. 102911, 2022, http://dx.doi.org/10.1016/j.ipm.2022.102911.

[26] S. G. T. C. Santos, R. S. M. Barros, and P. M. Gonçalves, "A differential evolution based method for tuning concept drift detectors in data streams," *Information Sciences journal*, vol. 485, pp. 376–393, 2019, http://dx.doi.org/10.1016/j.ins.2019.02.031.

[27] J. Tanha, N. Samadi, Y. Abdi, and N. Razzaghi-asl, "CPSSDS : Conformal prediction for semi-supervised classification on data streams," *Information Sciences*, vol. 584, pp. 212–234, 2022, http://dx.doi.org/10.1016/j.ins.2021.10.068.

[28] C. Manapragada, G. I. Webb, and M. Salehi, "Extremely fast decision tree," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, no. August, pp. 1953–1962, 2018, http://dx.doi.org/10.1145/3219819.3220005.

[29] L. Yang, D. M. Manias, and A. Shami, "PWPAE: An Ensemble Framework for Concept Drift Adaptation in IoT Data Streams," *2021 IEEE Global Communications Conference, GLOBECOM 2021 - Proceedings*, pp. 1–6, 2021, http://dx.doi.org/10.1109/globecom46510.2021.9685338.

[30] H. Guo, S. Zhang, and W. Wang, "Selective ensemble-based online adaptive deep neural networks for streaming data with concept drift," *Neural Networks,* vol. 142, pp. 437–456, 2021, http://dx.doi.org/10.1016/j.neunet.2021.06.027.

[31] Y. Sun, Z. Wang, H. Liu, C. Du, and J. Yuan, "Online Ensemble Using Adaptive Windowing for Data Streams with Concept Drift," *International Journal of Distributed Sensor Networks*, vol. 2016, 2016, http://dx.doi.org/10.1155/2016/4218973.

[32] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," *Information Fusion*, vol. 37, pp. 132–156, 2017, http://dx.doi.org/10.1016/j.inffus.2017.02.004.

[33] H. M. Gomes, J. Read, and A. Bifet, "Streaming Random Patches for Evolving Data Stream Classification," in *2019 IEEE International Conference on Data Mining (ICDM)*, 2019, pp. 240–249, http://dx.doi.org/10.1109/icdm.2019.00034.

[34] H. M. Gomes, J. P. Barddal, A. F. Enembreck, and A. Bifet, "A survey on ensemble learning for data stream classification," *ACM Computing Surveys*, 50(2), 2017, http://dx.doi.org/10.1145/3054925.

[35] J. Montiel, J. Read, A. Bifet, and T. Abdessalem, "Scikit-multiflow: A Multi-output Streaming Framework," *Journal of Machine Learning Research*, vol. 19, pp. 1–5, 2018, https://doi.org/10.48550/arXiv.1807.04662.