

RESEARCH ARTICLE

Advancing Forest Fires Classification Using Neurochaos Learning

KUNAL KUMAR PANT¹, REMYA AJAI A S¹, (Member, IEEE),
AND NITHIN NAGARAJ², (Senior Member, IEEE)

¹Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Amritapuri Campus, Kollam 690525, India

²Complex Systems Programme, National Institute of Advanced Studies (NIAS), IISc Campus, Bengaluru, Karnataka 560012, India

Corresponding author: Remya Ajai A S (remya@am.amrita.edu)

The Article Processing Charge (APC) for this publication was funded by Amrita Vishwa Vidyapeetham.

ABSTRACT Forest fires are among the most dangerous and unpredictable natural disasters worldwide, with severe environmental and economic impacts motivating extensive research on fire-occurrence prediction using environmental variables. Although *Machine Learning (ML)* and *Deep Learning (DL)* methods are widely used for this task, their performance can degrade under limited training data, motivating nonlinear feature-transformation approaches for small and heterogeneous datasets. This work evaluates ChaosNet and Random Heterogeneous Neurochaos Learning (RHNL), two variants of *Neurochaos Learning (NL)*, a chaos-based, brain-inspired learning paradigm previously shown to capture nonlinear patterns from limited data and preserve causal feature behaviour under controlled settings. Experiments are conducted across the Algerian Forest Fires (AFF), Canadian Forest Fires (CFF), and Portugal Forest Fires (PFF) datasets under high-sample and low-sample regimes, using random, time-aware, and spatially grouped validation where appropriate. On AFF, RHNL_{50G50L} + SVM achieves a time-aware high-sample macro-F1 of 0.995 ± 0.010 , while ChaosNet attains 0.816 ± 0.04 macro-F1 using only 1% training data under random evaluation. On CFF, Random Forest is strongest in the high-sample setting (0.740 ± 0.024), whereas ChaosNet leads at the 1% fraction in the log-transformed setting (0.568 ± 0.03). On PFF, spatially grouped evaluation reduces performance, highlighting the importance of leakage-resistant validation for geographic generalization. Across datasets, ChaosNet trains in only 0.151–0.205(ms) at the 1% fraction, making it approximately 360–910× faster than Random Forest and 700–1250× faster than Bagging DT.

INDEX TERMS Neurochaos learning, forest fires, ChaosNet, machine learning, natural disasters, brain-inspired learning.

I. INTRODUCTION

Forest fires are a major global concern. They have enormous environmental impacts, such as loss of vegetation, endangering wildlife, disrupting air quality, pollution of water resources, etc. The composition and structure of forests are also extensively influenced by the fire regime (Heinselman, 1973; Wright and Bailey, 1982, as cited in [1]). Frequency, seasonality, size, type, intensity and severity are the six main components of forest fires. Study in [2] reports that there is a large amount of evidence that points to a trend of increase in forest fires (both in numbers and size). This

results from the relationship between El Niño and climate change. Variations in winds and sea surface temperatures over the tropical Pacific Ocean give rise to a type of global climatic phenomenon known as El Niño or El Niño–Southern Oscillation (ENSO). Evidence is mounting that the world is experiencing a positive feedback cycle whereby deforestation and forest fires made worse by climate change lead to a rise in the frequency of El Niño events, which in turn triggers more forest burning [3]. There may be an increase in El Niño’s frequency and intensity, which means that the world will experience warmer and more severe weather, which may increase the number of forest fires.

Each year, around 1.5 million square miles of land are impacted by fire, according to estimates from the European

The associate editor coordinating the review of this manuscript and approving it for publication was Ikramullah Lali¹.

Space Agency [4]. To put this into perspective, this area is larger than India and nearly four times the size of Nigeria. Figure 1 provides an overview of wildfire occurrences across the globe during 2024, and Figure 2 depicts the total burnt area associated with these events. Collectively, these figures highlight that regions including Africa, South America, Russia, and Australia exhibited both elevated wildfire frequency and substantial landscape degradation. In India, the SNPP-VIIRS (Suomi National Polar-Orbiting Partnership - Visible Infrared Imaging Radiometer Suite) identified 3,45,989 forest fires, whereas the MODIS (Moderate Resolution Imaging Spectroradiometer) sensor identified 52,785 forest fires during the forest fire season between November 2020 and June 2021 [5]. 35.71% of India's forests have not yet experienced fires of any meaningful size, nevertheless 54.40% of forests are subject to sporadic fires, 7.49% to moderately regular fires, and 2.40% to high incidence levels.

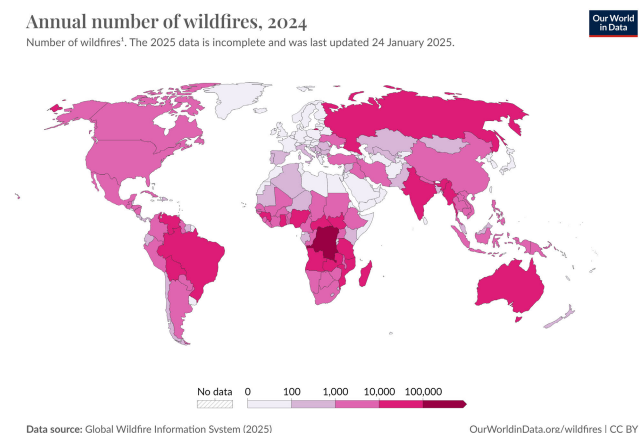


FIGURE 1. Annual number of wildfires (2024). Source: Our World in Data [6], licensed under CC BY 4.0.

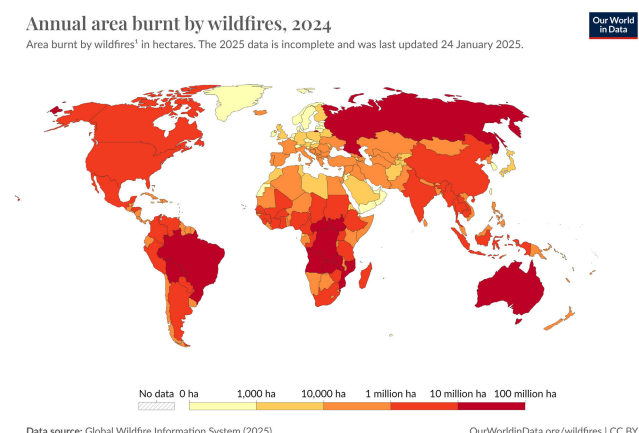


FIGURE 2. Annual area burnt by wildfires (2024). Source: Our World in Data [7], licensed under CC BY 4.0.

Machine-learning and deep-learning methods have been applied extensively to wildfire detection, susceptibility

assessment, occurrence classification, and spread prediction. Their performance, however, depends strongly on the available sample size, input modality, feature representation, class distribution, and evaluation protocol. These factors are especially relevant for the relatively small and heterogeneous tabular datasets considered in this study. We therefore evaluate Neurochaos Learning alongside tuned conventional classifiers to determine whether chaos-based representations offer an advantage under high-sample and data-constrained conditions.

A distinctive feature of Neurochaos Learning is its use of chaotic neuronal dynamics and controlled noise during feature generation. In particular, ChaosNet incorporates stochastic resonance, through which an appropriate level of noise may improve the response of the chaotic neurons to weak or variable input signals [8]. ChaosNet consists of a shallow layer of one-dimensional Generalized Lüroth Series maps that transform each input into dynamical features [9], [10]. These features may be classified using cosine similarity or supplied to a conventional classifier in a hybrid Neurochaos-ML pipeline [11].

Random Heterogenous Neurochaos Learning (RHNL) architecture is an updated structure of NL where randomness and the characteristics of neuronal heterogeneity in the human brain are taken into account together with chaotic behavior [12]. In RHNL, neurons based on GLS and the logistic map are positioned randomly within the input layer. There are three variations of RHNL structure namely $RHNL_{25L75G}$, $RHNL_{50L50G}$ and $RHNL_{75L25G}$. $RHNL_{25L75G}$ consists of 25% of the input locations assigned to logistic map neurons, with the remaining occupied by GLS map neurons. In $RHNL_{50L50G}$, 50% of the locations contain GLS map neurons, and the remaining positions are filled with logistic map neurons. 75% locations in $RHNL_{75L25G}$ are randomly assigned with logistic map function and the remaining with GLS function.

In this work, we focus on classification of forest fire occurrences as well as comparing the performance of traditional ML methods with NL. To carry out the same, three key regions were chosen: Algeria [13], Portugal [14], and Canada [15]. These regions were selected due to the availability of well-documented forest fires datasets that had been thoroughly collected, efficiently pre-processed, and extensively tested across various models, enabling a comparative analysis of model performance. The acronyms used in this study are summarized in Table 1 for ease of reference.

II. LITERATURE REVIEW

A. INITIAL DEVELOPMENTS: FUEL MODELS (1960-1980)

Researchers have long sought to understand the characteristics of fire, leading to advancements in forest fire prediction. Early efforts, whether focusing on fire occurrence, spread, or effects of different types of fuels and environmental conditions, can be traced back to the 1940s, as documented

TABLE 1. Acronyms used throughout the paper.

Item	Acronym
Neurochaos Learning	NL
Random Heterogeneous Neurochaos Learning	RHNL
<i>RHNL_{25L75G}</i>	RHNL architecture comprises 25%of neurons based on the logistic map, with the remaining neurons employing the GLS map. Features are fed to cosine similarity classifier
<i>RHNL_{25L75G} + SVM</i>	<i>RHNL_{25L75G}</i> with SVM classifier
<i>RHNL_{50L50G}</i>	RHNL architecture comprises 50%of neurons based on the logistic map, with the remaining neurons employing the GLS map Features are fed to cosine similarity classifier
<i>RHNL_{50L50G} + SVM</i>	<i>RHNL_{50L50G}</i> with SVM classifier
<i>RHNL_{75L25G}</i>	RHNL architecture comprises 75%of neurons based on the logistic map, with the remaining neurons employing the GLS map. Features are fed to cosine similarity classifier
<i>RHNL_{75L25G} + SVM</i>	<i>RHNL_{75L25G}</i> with SVM classifier
Deep Learning	DL
Decision Trees	DT
Random Forest	RF
European-Forest-Fire-Information System	EFFIS
Middle East and North Africa region	MENA
Geographic Information System	GIS
Artificial Neural Network	ANN
Fine Fuel Moisture Code	FFMC
Drought Code	DC
Duff Moisture Code	DMC
Fire Weather Index	FWI
Initial Spread Index	ISI
Buildup Index	BUI
Relative Humidity(%)	Rh
Rain Fall(mm)	Rain
Wind Speed(Km/Hr)	Ws
High Sample Training Regime	HSTR
Low Sample Training Regime	LSTR
Normalized Difference Vegetation Index	NDVI
Thermal Anomalies	TA
Land Surface Temperature	LST

in [16]. This study utilized controlled experimental environments to analyze the dynamics of fire spread, resulting in the development of mathematical models that provided theoretical frameworks for real-world applications. These models facilitated the prediction of key factors, such as the rate of spread, ignition time, and heat transfer processes. The study emphasized the growing necessity for empirical data to enhance the accuracy of forest fire predictions. The ideas and theories proposed in this were confirmed in [17].

The next most notable research was [18] in the 1970s. It described the intricate relation of fire spread to variables like wind, fuel etc. and came up with equations and mathematical models for forecasting how fast and how intensely fire will spread through a continuous layer of fuel lying on the ground. It discussed about the conversion of a forest ground fire to a crown fire, given that sufficient heat is generated. After that, in the 1980s, existing models were improved by developing tools to assist in selecting appropriate fuel models to predict fire behavior [19], improving the practical application of Rothermel's model.

Wildland-fire modelling was based physically on early studies that related fire spread and intensity to fuel properties, wind, slope and heat-transfer mechanisms. The Rothermel

model became a basis for several operational fire-behaviour systems and the creation of standardized fuel models made it applicable to a range of vegetation conditions. Later reviews have shown that these formulations continue to influence physical, quasi-physical and empirical fire-spread models developed with advances in computational simulation and spatial data processing [20], [21]. While physics-based fuel models continue to be relevant for understanding fire behaviour, data-driven approaches complement them by learning predictive relationships from observed environmental data.

B. INTEGRATION OF REMOTE SENSING AND GIS (1980-2000)

While the initial use of remote sensing for forest fire mapping began in the 1960s, the widespread adoption of remote sensing and GIS techniques emerged during the 1980s. With the advancement in computer and data storage services, GIS came up as a new possibility. GIS makes it possible to manage loads of spatial information and derive different sorts of models like cartographic models. This is done by combining these layers of stored information in different ways. Reference [22] takes on hazard mapping of forest fires.

This study was centered on the Mediterranean coast of Spain. It implements this by using two methods: high-resolution imagery and test areas already affected by the forest fire. This makes it easier to test the efficacy of the hazard mapping.

Advances in satellite sensing subsequently extended GIS-based fire analysis from static hazard mapping to active-fire detection, burned-area assessment, and near-real-time monitoring. MODIS and VIIRS products are widely used to detect thermal anomalies and active-fire locations. However, their detection performance is variable with fire size, vegetation cover, smoke, and sensor resolution [23]. The availability of higher-resolution multispectral imagery from Sentinel-2 has further supported detailed fire mapping using the visible, near-infrared and short-wave infrared bands and CNN-based enhancement of the lower-resolution spectral bands for active-fire detection [24].

The integration of remote sensing with GIS and machine-learning methods have been increasingly used to develop spatially explicit wildfire-susceptibility maps. Iban and Sekertekin [25] combined remotely sensed fire observations with environmental conditioning factors and GIS-based spatial analysis to identify areas with higher wildfire susceptibility. Such approaches commonly incorporate vegetation indices, land-surface temperature, topographic variables, climatic conditions, and proximity to anthropogenic factors. This development is relevant to the present study because the CFF dataset also includes satellite-derived variables, namely NDVI, land-surface temperature, and burned area.

C. EMERGENCE OF MACHINE LEARNING TECHNIQUES (1990-2025)

The rapid advancements in computational power and hardware during the 1990s propelled the widespread adoption of machine learning techniques across various fields. These methods were soon applied to forest fire prediction, leveraging cartographic models generated from GIS data. Among the earliest studies to incorporate machine learning, [26] evaluated two approaches: Gaussian Discriminant Analysis and ANNs, a form of deep learning (with deep learning categorized under the broader domain of machine learning). ANNs were an early attempt by the scientific community to mimic the structural and functional principles of neurons in the human brain. However, it is important to acknowledge that current ANNs are far from accurately replicating the complex workings of the billions of neurons in the brain. As highlighted in [27], ANNs do not precisely emulate biological neurons. Instead, artificial neurons are simplified representations that loosely approximate the behavior of their biological counterparts. Despite this, ANN models consistently outperformed Gaussian Discriminant Analysis in this study [26]. Other studies like [28] incorporated many other ML techniques like different types of decision trees, decision forests, regression techniques, etc.

Recent wildfire studies have gone beyond traditional tabular classifiers by incorporating machine learning with remotely sensed vegetation, weather, topographic, and historical fire variables. Reference [29] introduced a large-scale remote-sensing dataset for next-day wildfire-spread prediction and benchmarked neural models, Logistic Regression, and Random Forest. Deep learning approaches have also been used for wildfire detection, burned-area mapping, severity estimation and spread prediction from satellite imagery and spatially structured inputs [30]. These approaches demonstrate the benefit of using CNN, recurrent and attention-based architectures when image sequences or dense spatiotemporal observations are available.

However, ensemble-tree methods remain strong reference models for heterogeneous tabular data. Modern gradient-boosting frameworks such as LightGBM and CatBoost generalize traditional boosting with computationally efficient tree construction and better handling of categorical or ordered information [31], [32]. Recent benchmark studies also show that deep neural architectures do not outperform tree-based methods in general for small and medium-sized tabular datasets [33], [34]. Alternative architectures such as residual multilayer perceptrons or the FT-Transformer can be competitive, but the relative advantage is dataset size, feature structure and tuning protocol dependent.

Within this broader modelling landscape, Neurochaos Learning offers an alternative nonlinear feature-transformation paradigm based on chaotic neuronal dynamics, randomness and heterogeneity in the composition of neurons. For the first time in [10], authors proposed the novel Neurochaos Learning architecture, where the chaotic nature of brain is considered for developing the input layer of classification model. Later in [35], the 1D GLS map used as neurons are replaced by 1D logistic map to analyse the classification performance. Two more properties of brain namely randomness and heterogeneity are considered and RHNL architecture is proposed in [12]. RHNL gives better classification performance for various dataset considered and even outperforms DL in the low training sample regime.

D. EVALUATION UNDER TEMPORAL AND SPATIAL DEPENDENCE

When the application involves prediction at later times or at previously unobserved locations, evaluation protocols must account for dependence among observations. Random partitioning may yield temporally adjacent or spatially neighboring samples to be distributed across both training and test sets, leading to potentially optimistic estimates of predictive performance. Therefore, structure-aware cross-validation strategies have been recommended [36] for datasets with temporal, spatial, hierarchical or related dependence structures.

Such consideration is particularly relevant for wildfire datasets where observations may have seasonal, geographic, or environmental similarities. So leakage-resistant evaluation

should be used whenever the needed temporal or spatial metadata are available. The principle, in the present study, leads to a chronological evaluation for AFF and a spatially grouped evaluation for PFF. CFF, however, is limited to repeated stratified random evaluation since its processed representation does not contain explicit temporal or spatial identifiers.

Although physics-based modelling, remote sensing, GIS-based susceptibility assessment and machine-learning-based wildfire prediction have made significant advances, direct comparison across studies is still difficult because the underlying datasets differ substantially in size, feature representation, geographic coverage and data modality. In particular, image and sequence-based deep-learning approaches are not directly comparable to the relatively small tabular datasets considered in this work.

Previous studies on Neurochaos Learning have shown the advantages of chaos-based feature transformations and heterogeneous neuron composition for classification under data constraints [9], [11], [12]. However, their behavior in structurally different wildfire datasets has not been systematically studied in the context of high-sample and low-sample settings, or under protocols that explicitly evaluate temporal and spatial generalization. In this paper, we address this issue by providing a cross-dataset empirical evaluation of ChaosNet and RHNL on AFF, CFF and PFF, supported by repeated-seed analysis and dataset-specific evaluation protocols.

Table 2 summarizes representative developments in physical fire modelling, remote-sensing-based wildfire analysis, machine learning, structure-aware evaluation, and Neurochaos Learning.

III. DATASET DESCRIPTION

A. ALGERIAN FOREST FIRE DATASET (AFF)

According to [37], Algeria is the most impacted nation among MENA countries and ranks fourth among all countries monitored by EFFIS. The dataset considered in this study, sourced from [13], consists of observations gathered from two separate areas of Algeria: Béjaia and Sidi Bel-Abbés from the north-east and north-west regions, respectively. Data acquisition was carried out during the period from June to September 2012, corresponding to the months identified as having the highest frequency of fire occurrences between 2007 and 2018. The dataset categorizes each observation into one of two classes — “fire” or “not fire” as it has been designed as a classification task. The total number of samples comes up to **244**. The exact class distribution is provided in Table 3.

The finalized AFF dataset was validated for column consistency, numeric integrity, missing values, and duplicate observations. No missing-value imputation, duplicate removal, or synthetic resampling was required.

The AFF model input comprises 11 predictors: four meteorological variables, six variables from the Fire Weather

Index system, and a binary regional indicator. The day, month, and year attributes are not supplied to the classifiers; month and day are retained only to establish chronological ordering for the time-aware evaluation. Table 4 summarizes the AFF feature schema and the role of each variable group.

The FWI system also called the Canadian Forest Fires Weather Index system or the Forest Weather Index consists of 6 components [38]. **Fine Fuel Moisture Code (FFMC)** is a numerical value that indicates the moisture content of cured fine fuels, such as litter, and weights approximately 0.05 pounds per square foot. **Duff Moisture Code (DMC)** is the amount of moisture in decomposed and loosely packed organic material which is specifically 2 to 4 inches deep and weighs around $1lb/ft^2$ in a dry state. **Drought Code (DC)** accounts for the deep layer of compact and organic matter weighing around $10lb/ft^2$ in a dry state. **Initial Spread Index (ISI)** is derived from a combination of the Fine Fuel Moisture Code (FFMC) and wind speed (Ws). It depicts the spread rate while not considering the impact of fuel quantities. **Buildup Index (BUI)** is the combination of DMC and DC. **Forest Weather Index (FWI)** is the combination of BUI and ISI.

The six FWI- system predictors provide complementary measures of fuel moisture and potential fire behaviour [38]. FFMC characterizes fine-fuel moisture, while DMC and DC represent moisture conditions in progressively deeper organic layers. ISI combines FFMC and wind speed to estimate potential spread rate, BUI integrates DMC and DC, and FWI combines ISI and BUI into an overall fire-danger index.

B. CANADIAN FOREST FIRE DATASET (CFF)

An experiment was developed to examine the constructed dataset with the aim of forecasting wildfire events in a designated forest region of Canada between 2013 and 2014. Fire zone information was sourced from the Canadian Wildland Fire Information System (CWFIS). In this study [15], data conversion was carried out using tools like GDAL (Geospatial Data Abstraction Library) and HEG (HDF-EOS to GeoTIFF Conversion Tool). Since the data was extracted from satellite images, challenges such as geometric distortions and variations due to cloud cover and atmospheric conditions need to be considered. Atmospheric correction was applied, as the raw data had already undergone preprocessing for other aspects, making any other additional techniques like geo-referencing, ortho-rectification or radiometric correction unnecessary.

During construction of the original dataset, the satellite-derived observations were spatially clipped and processed using the interpolation and extrapolation procedures reported by the dataset authors [15]. The present study uses the resulting processed tabular dataset and does not independently repeat the satellite-level atmospheric correction, interpolation, or extrapolation stages. The experimental variables were verified to contain no missing values; therefore, no additional missing-value imputation was performed.

TABLE 2. Summary of representative and relevant prior work.

Study	Data or Task	Approach	Main Contribution	Limitation or Relevance to This Study
Fons; Rothermel; Anderson [16], [18], [19]	Controlled fuel beds and surface-fire behaviour	Physical and semi-empirical spread equations; standardized fuel models	Established the physical basis for estimating fire spread and fuel-dependent fire behaviour	Uses simplified fuel and environmental assumptions and does not address data-driven fire classification
Chuvieco and Congalton [22]	Forest-fire hazard mapping on the Mediterranean coast of Spain	Remote sensing and GIS-based spatial-layer integration	Demonstrated the use of satellite imagery and GIS for spatial fire-hazard assessment	Region-specific and primarily focused on static hazard mapping rather than predictive classification
Stojanova et al. [28]	Forest-fire occurrence prediction using tabular environmental records	Decision trees, ensembles, and other data-mining methods	Provided an early comparison of data-driven methods for forest-fire prediction	Predates modern boosting, uncertainty reporting, and structure-aware evaluation protocols
Sayad et al. [15]	MODIS-derived NDVI, land-surface temperature, and thermal-anomaly data	Remote sensing, data mining, ANN, and SVM	Introduced the Canadian wildfire dataset and demonstrated satellite-based occurrence prediction	The processed representation used in the present study does not retain explicit temporal or spatial identifiers
Gargiulo et al.; Huot et al.; Ghali and Akhloufi [24], [29], [30]	Active-fire detection, burned-area analysis, and next-day spread prediction from satellite observations	CNN-based super-resolution and deep spatial or spatiotemporal models	Demonstrated the utility of deep learning when dense imagery, raster sequences, or multitemporal observations are available	Requires image-based or sequential inputs and is not directly comparable to the small tabular datasets considered here
Iban and Sekertekin [25]	MODIS active-fire pixels and environmental conditioning factors in Turkey	LR, SVM, LDA, RF, GB, XGBoost, and AdaBoost with GIS	Integrated remotely sensed fire observations, GIS variables, and ensemble learning for susceptibility mapping	Evaluates a regional spatial-mapping task rather than cross-dataset tabular classification
Ke et al.; Prokhorenkova et al.; Gorishniy et al.; Grinsztajn et al. [31]–[34]	General heterogeneous tabular classification	LightGBM, CatBoost, residual MLPs, and FT-Transformer	Established strong modern baselines and showed that tree ensembles remain competitive on typical tabular datasets	Not wildfire-specific, but motivates balanced comparison between tree-based and neural models
Roberts et al. [36]	Temporally, spatially, or hierarchically dependent observations	Structure-aware and blocked cross-validation	Demonstrated why random validation can provide optimistic estimates for dependent data	Provides methodological guidance rather than a wildfire-classification model; motivates the AFF and PFF protocols
Harikrishnan and Nagaraj; Balakrishnan et al. [9], [10]	General classification under limited-data conditions	GLS-map chaotic neurons with cosine-similarity classification	Introduced the foundational Neurochaos Learning and ChaosNet architectures	Did not provide a systematic cross-dataset evaluation for wildfire classification
Sethi et al. [11]	General tabular classification	Neurochaos feature transformation followed by conventional ML classifiers	Extended Neurochaos representations to hybrid NL–ML classification pipelines	Temporal and spatial generalization on wildfire datasets was not examined
AS and Nagaraj [12]	Classification using heterogeneous chaotic-neuron populations	Random mixtures of GLS- and logistic-map neurons	Introduced RHNL to incorporate neuronal randomness and heterogeneity	Cross-dataset wildfire performance under HSTR, LSTR, temporal, and spatial evaluation remained unexplored

TABLE 3. AFF dataset class distribution.

Class	Occurences
"not fire" $\rightarrow$ 0	106
"fire" $\rightarrow$ 1	138

The processed CFF dataset contains three predictor variables, as summarized in Table 5: the Normalized Difference Vegetation Index (NDVI), land-surface temperature (LST), and burned area. NDVI represents vegetation condition, while LST characterizes the thermal state of the land surface. The CLASS attribute is retained as the binary fire/no-fire target. Two input-feature representations were evaluated using the same processed dataset and unchanged target labels. In the normal setting, BURNED_AREA was supplied directly as the third input feature. In the log-transformed setting, it was replaced by $\log(1 + \text{BURNED_AREA})$, while NDVI, LST, and the target labels remained unchanged. This comparison assesses whether reducing the skewness of the burned-area feature materially affects classification performance.

TABLE 4. Feature schema and variable roles for the AFF dataset.

Category	Variables	Role
Geographic	region	Binary indicator distinguishing the two sampled Algerian regions; supplied as a classifier input.
Meteorological	Temperature, RH, Ws, Rain	Ambient temperature, relative humidity, wind speed, and rainfall; supplied as classifier inputs.
Fuel-moisture indices	FFMC, DMC, DC	Characterize moisture conditions in fine surface fuels, loosely compacted organic material, and deeper drought-affected layers.
Fire-behaviour indices	ISI, BUI, FWI	Represent potential fire spread, combined fuel-moisture conditions, and overall fire-weather danger.
Temporal metadata	day, month, year	Excluded from classifier inputs; month and day are used only to establish chronological ordering for time-aware evaluation.
Target	Classes	Binary classification label: fire (1) or not fire (0).

TABLE 5. Parameters of the CFF dataset.

Parameter	Unabbreviated	Aspect
NDVI	Normalized Difference Vegetation Index	Crop health
LST	Land Surface Temperature	Soil temperature
Burned_Area	Overall Burned Area	Burned-area extent

C. PORTUGAL FOREST FIRE DATASET (PFF)

The Montesinho Natural Park in Portugal's Tras-os-Montes Northeast region is where the Portugal Forest Fire data is sourced from January 2000 to December 2003. An inspector collected part of the data by registering several features such as time, date, and spatial location every time a forest fire occurred within a 9x9 grid. Along with the different elements of the FWI system, the type of vegetation involved and the total burned area were also recorded. A polytechnic institute in Bragança gathered the remaining portion of the database, which included many weather observations. A meteorological station in the heart of Moesinho Park captured these throughout a 30 minute period. Inserted within a supra-Mediterranean climate, the average annual temperature range was from 8°C – 12°C [14].

The PFF dataset contains meteorological, environmental, temporal, and spatial attributes collected from the Montesinho Natural Park region in Portugal. The predictor variables used for model training include the spatial coordinates X and Y , fire-weather indices FFMCI, DMC, DC, and ISI, and meteorological variables including temperature, relative humidity, wind, and rain as summarised in Table 6. Temporal information is incorporated through month and day attributes, which are transformed using cyclical sine and cosine encoding to generate $Month_Sin$, $Month_Cos$, Day_Sin , and Day_Cos . This encoding preserves the cyclic nature of seasonal information and avoids treating temporal variables as simple ordinal quantities.

Following cyclical encoding of the month and day attributes, the numerical predictors were mapped to the interval $[0, 1]$ using Min–Max normalization. This provides a common bounded representation across variables with different numerical ranges and ensures compatibility with the Neurochaos feature extractor. The $spatial_group$ identifier and the binary target were excluded from this transformation.

The $area$ attribute is not provided to the classifiers as an input feature and is only used for target-label construction. The distribution of burned area ($area$ attribute) is highly right-skewed and heavy-tailed, so it is first transformed by $\log(1 + area)$. Then the median transformed value of 0.4187 is used as the binary classification threshold, resulting in two almost balanced classes with 259 and 258 samples respectively. This data-driven threshold avoids imposing an arbitrary raw burned-area cut-off, and provides a balanced classification setting. However, the binary formulation does not capture differences in fire magnitude within each class;

TABLE 6. Revised feature construction for the PFF dataset.

Category	Variables	Role
Spatial features	X, Y	Input features
Fire-weather indices	FFMC, DMC, DC, ISI	Input features
Meteorological features	Temperature, relative humidity, wind, rain	Input features
Temporal features	$Month_Sin, Month_Cos, Day_Sin, Day_Cos$	Cyclically encoded input features
area	$\log(1 + area)$	Used for binary label construction
Target variable	Median-thresholded log-area label	Binary classification label
Spatial grouping variable	$spatial_group = X_Y$	Used only for grouped splitting and validation

therefore, the results should be interpreted as classification of relative burned-area levels, rather than prediction of exact fire severity.

To account for spatial dependence, a spatial grouping variable is constructed using the coordinate pair as $spatial_group = X_Y$. The coordinate pairs define 36 unique spatial groups and this grouping variable is used only for spatially grouped splitting and validation and is not supplied to the classifiers as an input feature. In the spatially grouped evaluation protocol, samples belonging to the same spatial cell are not allowed to appear simultaneously in the training and testing sets. This reduces geographic leakage and provides a more realistic estimate of model generalization across unseen spatial locations.

IV. CLASSIFIERS

Deep Learning models like DNN, CNN, LSTM and other deep architectures are important baselines in wildfire studies but their suitability is strongly dependent on the data modality, sample size and sampling structure. CNNs are particularly appropriate for dense image or raster inputs, whereas LSTMs and related recurrent models are mainly intended for sequential observations with meaningful temporal dependence. In addition, high-capacity deep architectures typically require substantially larger training sets and can be prone to overfitting when applied to small tabular datasets without sufficient regularization or external data. The three datasets considered in this work are processed tabular datasets rather than image sequences or dense spatiotemporal tensors. AFF contains a small number of meteorological and Fire Weather Index observations, CFF is available in this study as a processed tabular representation that includes NDVI, land-surface temperature and burned-area variables, and PFF is evaluated using tabular meteorological, fire-weather, spatial and cyclically encoded temporal predictors. Adding CNN or LSTM baselines would therefore require constructing additional tensor-based representations beyond the native form of the datasets. For this reason, the present comparison

TABLE 7. References corresponding to the tuned hyperparameters for each algorithm.

Algorithm	Hyperparameters tuned
SVM	C , $kernel$, $gamma$, $degree$
RF	$min_samples_leaf$, max_depth , $n_estimators$, $min_samples_split$
DT	max_depth , $min_samples_split$, $min_samples_leaf$
DT with AdaboostM1	$learning_rate$, $n_estimators$
DT with Bagging	$max_samples$, $n_estimators$, $max_features$
XGBoost	$learning_rate$, $colsample_bytree$, $n_estimators$, max_depth , reg_alpha , $subsample$, reg_lambda
NL	$initial\ neural\ activity(q)$, $discrimination\ threshold(b)$, $noise\ intensity(\epsilon)$

is limited to strong tabular ML baselines such as Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), AdaBoost, Bagging DT, and XGBoost, together with evaluated Neurochaos-based models such as ChaosNet and RHNL variants. Deep spatial and sequential models are therefore discussed as related work, while direct empirical comparison is left for future evaluations on larger image-based, gridded or explicitly sequential wildfire datasets. For NL models [9], [11], [12], the three hyperparameters we need to tune are discrimination threshold (b), initial neural activity (q) and noise intensity (ϵ). Table 7 provides an overview of the set of hyperparameters tuned for all the ML and NL algorithms.

A. SUPPORT VECTOR MACHINE (SVM)

The hyperparameters for SVM [39], a support vector machine for classification based tasks are $kernel$, C , $degree$, and $gamma$. C , also known as the regularization parameter, controls the bias-variance trade-off of the algorithm. $kernel$ is the feature mapping choice which defines the decision boundary shape. $Gamma$ influences the radius of a single training point and $degree$ is the polynomial degree for the poly kernel. Table 8 gives the values of various parameters considered for SVM.

B. RANDOM FOREST (RF)

The hyperparameters for the RF model [40], include max_depth , $min_samples_leaf$, $min_samples_split$, $n_estimators$. $n_estimators$ defines the number of trees which are being grown in a forest. max_depth is the maximum allowed depth each tree in the forest is allowed. $min_samples_leaf$ declares the minimum samples required in a leaf and $min_samples_split$ defines the minimum samples to split an internal node. The various parameters used for configuring the Random Forest classifier are summarized in Table 9.

C. DECISION TREE (DT)

The hyperparameters for the DT model [41], include max_depth , which is used to define the maximum depth to which a tree can grow, $min_samples_leaf$, which declares the minimum number of samples required for a leaf (node), and $min_samples_split$, which defines the minimum samples to

split in an internal node. Table 10 summarizes the parameters employed in configuring the DT classifier.

D. DT WITH ADABOOST M1

The hyperparameters for the boosted model [42], include $learning_rate$, which helps shrink each weak learner's contribution and $n_estimators$, which accounts for the number of boosting rounds. The parameters utilized for configuring the DT with Adaboost M1 classifier are summarized in Table 11.

E. DT WITH BAGGING

The hyperparameters for the bagging model [43] include $max_features$ which accounts for the number of features sampled per base estimator, $max_samples$ which is the fraction of training samples drawn per base estimator, and $n_estimators$, which is the number of bootstrapped models to aggregate. The configuration parameters for the Decision Tree (DT) with the Bagging M1 classifier are detailed in Table 12.

F. XGBoost

The XGBoost classifier [44] was utilized for binary classification, with parameter tuning focused on mitigating class imbalance and enhancing model performance. The hyperparameters for XGBoost in classification tasks include $n_estimators$ which is the number of boosting rounds (trees), max_depth or the maximum tree depth, $learning_rate$ which is the step size shrinkage per boosting step, $colsample_bytree$ which defines the fraction of features sampled for each tree, $subsample$ which is the fraction of rows sampled per tree, reg_alpha for L1 regularization, and reg_lambda for L2 regularization. The configuration parameters for the XGBoost classifier are detailed in Table 13.

G. ChaosNet

Neurochaos Learning (NL) architecture [9] represents a chaos-driven neuronal framework that emulates the intrinsic chaotic behavior of the human brain within its neural structure, a characteristic typically absent in traditional machine learning models. ChaosNet is the basic Neurochaos Learning (NL) architecture as shown in Figure 3 that consists of a layer of chaotic neurons, modeled as 1D Generalized Lüroth Series (GLS) maps. The Generalized Lüroth Series

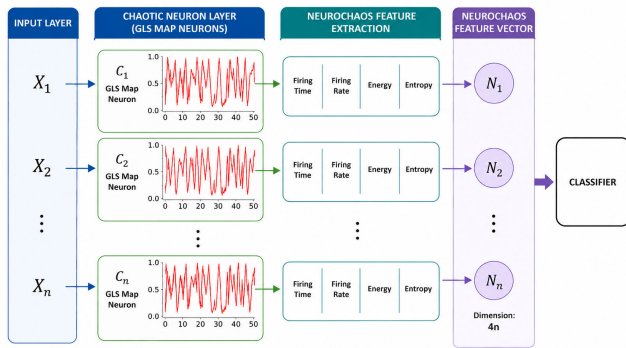


FIGURE 3. Neurochaos Learning (NL) Architecture.

TABLE 8. Parameter grid for SVM.

Hyperparameter	Grid
C	[0.1, 1, 10, 100]
$kernel$	['rbf', 'sigmoid', 'linear', 'poly']
$gamma$	[1, 0.1, 0.01, 0.001] if $kernel$ is ['rbf', 'sigmoid', 'poly'] else ['scale']
$degree$	[2, 3, 4] if $kernel == 'poly'$ else [3]

(GLS) map, GLS : $[0, 1) \rightarrow [0, 1)$, is defined as follows:

$$GLS(x) = \begin{cases} \frac{x}{b}, & 0 \leq x < b, \\ \frac{1-x}{1-b}, & b \leq x < 1, \end{cases} \quad (1)$$

where $x \in [0, 1)$ and $0 < b < 1$.

The hyperparameters defined for ChaosNet are initial neural activity (q), discrimination threshold (b) and ϵ , the noise intensity. The input features (in this study: the environmental variables) are not assumed to themselves follow non-linear dynamics. Instead, each normalized input feature ($x_j \in [0, 1]$), such as temperature, relative humidity, wind speed, or a fire-weather index, acts as an input stimulus to the corresponding chaotic neuron. Initial neural activity is considered as the point from which the neuron starts firing when an input stimulus triggers it. The chaotic firing of each GLS neuron ceases once its activity value evolving from the initial neural state q enters the ϵ neighborhood of the specific input stimulus. Hence, different neurons may stop firing at different times. The range of ChaosNet hyperparameters considered are given in Table 14.

For each input variable, the resulting chaotic neural trajectory is represented using four ChaosFEX descriptors: firing rate, firing time, energy, and entropy. [11]

H. RANDOM HETEROGENOUS NEUROCHAOS LEARNING (RHNL)

Random Heterogenous Neurochaos Learning (RHNL) architecture is an updated structure of NL where randomness and heterogeneity properties of the human brain are considered along with chaotic behavior [12]. In RHNL, GLS and

TABLE 9. Parameter grid for RF.

Hyperparameter	Grid
max_depth	[50, 40, 30, 20, 10]
$min_samples_split$	[10, 5, 2]
$min_samples_leaf$	[4, 2, 1]
$n_estimators$	[250, 200, 150, 100, 50]

TABLE 10. Parameter grid for DT.

Hyperparameter	Grid
max_depth	[10, 20, 30, 40, 50]
$min_samples_split$	[2, 5, 10]
$min_samples_leaf$	[1, 2, 4]

TABLE 11. Parameter grid for DT with Adaboost M1.

Hyperparameter	Grid
$learning_rate$	[.01, .1, .5, 1.0]
$n_estimators$	[50, 100, 150, 200]

TABLE 12. Parameter grid for bagging model.

Hyperparameter	Grid
$max_samples$	[.6, .8, 1.0]
$max_features$	[.6, .8, 1.0]
$n_estimators$	[50, 100, 150, 200]

TABLE 13. Parameter grid for XGBoost.

Hyperparameter	Grid
$learning_rate$	[.01, .1, .2]
$n_estimators$	[50, 100, 150, 200]
max_depth	[3, 5, 7, 10]
$colsample_bytree$	[.6, .8, 1.0]
$subsample$	[.6, .8, 1.0]
reg_alpha	[0, .1, 1]
reg_lambda	[1, 1.5, 2]

TABLE 14. Parameter grid for ChaosNet.

Hyperparameter	Grid Range
q	[.001 – 1.0]
b	[.01 – 0.5]
ϵ	[.001 – .5]

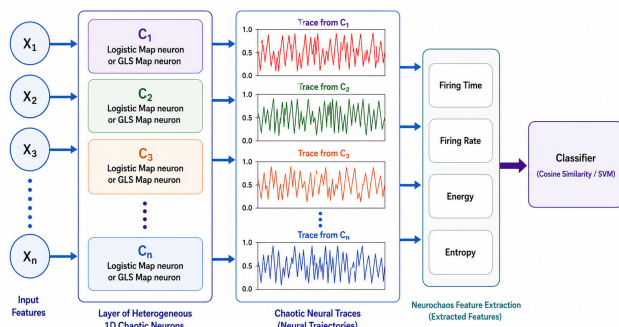
logistic map neurons are placed in the input layer at random locations.

There are three variations of RHNL structure namely $RHNL_{25L75G}$, $RHNL_{50L50G}$ and $RHNL_{75L25G}$. $RHNL_{25L75G}$ consist of 25% locations with logistic map neurons and remaining with GLS map neurons. In $RHNL_{50L50G}$, 50% of locations are placed with logistic map neurons and remaining with GLS map neurons. 75% locations in $RHNL_{75L25G}$ are randomly assigned with logistic map function and the remaining with GLS function.

TABLE 15. Parameter grid for RHNL.

Hyperparameter	Grid
q	[.001 – .5]
b	[.01 – .5]
ϵ	[.001 – .3]

In RHNL [12], the features generated from input samples namely Firing Rate, Firing Time, Energy and Entropy are fed to cosine similarity classifier. The structure of RHNL where cosine similarity classifier in the output layer is replaced by SVM are termed as $RHNL_{RH25L75G}+SVM$, $RHNL_{RH50L50G}+SVM$ and $RHNL_{RH75L25G}+SVM$ as shown in Figure 4. The range of values used for RHNL hyperparameters tuning are given in Table 15.

**FIGURE 4.** Random Heterogeneous Neurochaos Learning (RHNL) Architecture.

Previous work has also studied Neurochaos Learning under the lens of causality-preserving feature transformation. Harikrishnan et al. [45] evaluated ChaosNet on controlled cause-effect classification tasks involving coupled autoregressive processes, coupled one-dimensional chaotic maps, and a real-world prey–predator system. Their study reported that the features of ChaosFEX preserved causal directionality as measured using Granger Causality for autoregressive processes and Compression-Complexity Causality for nonlinear chaotic and prey–predator systems. However, these results were derived in explicitly causal time-series settings. The present work does not do causal discovery, intervention analysis or empirical causality preservation testing on the forest-fire datasets. Therefore, in this work Neurochaos Learning is evaluated as a nonlinear chaos-based feature-transformation and classification framework and any causal-preservation properties are referred only as pertinent prior evidence, not as a claim established by the present experiments.

V. EXPERIMENTAL SETUP AND EVALUATION STRATEGY

For each dataset, the classification problem is defined over a set of labelled observations $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbb{R}^d$ denotes the input-feature vector and $y_i \in \{0, 1\}$ denotes the corresponding binary class label. The objective is to learn a classifier f such that $\hat{y}_i = f(\mathbf{x}_i)$ minimizes prediction error on

unseen observations. At the general problem level, this can be expressed through the empirical classification risk

$$\hat{R}(f) = \frac{1}{N} \sum_{i=1}^N \mathcal{L}(y_i, f(\mathbf{x}_i)), \quad (2)$$

where $\mathcal{L}$ denotes the classifier-specific loss or decision criterion. Because the evaluated methods include tree ensembles, SVM, XGBoost, and Neurochaos-based classifiers, their internal fitting objectives and optimization procedures are model-specific rather than governed by a single common training loss. Section IV summarizes the evaluated classifiers and their tuned hyperparameters.

For a consistent model-selection procedure across classifiers, hyperparameters were optimized with Optuna using five-fold cross-validation on the training partition, with mean macro-F1 used as the study-level optimization objective. The dataset-specific evaluation protocols, cross-validation schemes, Optuna trial budgets, and repeated seeds are summarized in Table 16.

A. DATA PREPROCESSING AND CLASS-IMBALANCE HANDLING

For AFF, the finalized dataset was validated to contain no missing values or duplicate observations; consequently, no missing-value imputation, duplicate removal, or synthetic resampling was required. SVM inputs were standardized using `StandardScaler`. ChaosNet and all RHNL variants used Min–Max normalization, with the normalized values clipped to the interval $[0, 1]$ to satisfy the requirements of the chaotic feature extractor. The remaining conventional tree-based classifiers operated on the original numerical feature representation.

Class imbalance in AFF was handled using balanced class weighting for AdaBoost, Bagging DT, Decision Tree, Random Forest, and SVM. XGBoost used the corresponding `scale_pos_weight` value derived from the class distribution. ChaosNet and the RHNL variants did not use class weighting or resampling.

For CFF, two feature representations were evaluated using the same binary target: the original burned-area feature and its $\log(1 + \text{BURNED_AREA})$ transformation. No additional missing-value imputation or synthetic resampling was performed. SVM inputs were standardized using `StandardScaler`, fitted independently within each cross-validation training fold and refitted on the complete training partition before held-out evaluation. ChaosNet and all RHNL variants used training-fitted Min–Max normalization, with held-out observations transformed using only the corresponding training-derived parameters and clipped to the interval $[0, 1]$. The conventional tree-based classifiers operated on the original numerical feature representation.

Class imbalance in CFF was handled using balanced class weighting for AdaBoost, Bagging DT, Decision Tree, Random Forest, and SVM. XGBoost used the documented `scale_pos_weight` setting, whereas ChaosNet and the

RHNL variants did not use class weighting or resampling. All model-selection experiments used shuffled stratified five-fold cross-validation.

For PFF, cyclical sine-cosine encoding was applied to the month and day attributes during construction of the common preprocessed dataset. The tree-based classifiers operated directly on the resulting prepared representation. For SVM, `StandardScaler` was fitted independently within each cross-validation training fold and refitted on the complete training partition before evaluation on the held-out partition. ChaosNet and all RHNL variants operated on Min-Max-normalized inputs, with values clipped to $[0, 1]$ to satisfy the input requirements of the chaotic feature extractor. The median-based target construction produced nearly balanced classes containing 259 and 258 samples. Consequently, no synthetic resampling was performed.

B. EVALUATION PROTOCOLS

For the AFF dataset, two evaluation protocols were considered. First, an 80:20 stratified random split was used as the baseline evaluation strategy. Second, a seed-dependent chronological split was used to evaluate temporal generalization. Samples were first ordered using the month and day attributes. Starting from an 80% training cut-off, a deterministic seed-dependent forward shift was applied to generate multiple temporal hold-outs while ensuring that at least 10% of the observations remained in the test partition. Under every seed, all training observations preceded all test observations, preventing temporal leakage. The reported variation across seeds therefore reflects both model variation and changes in the chronological evaluation cut-off.

During hyperparameter optimization, shuffled five-fold `KFold` cross-validation was used for the random-split protocol. For the time-aware protocol, five-fold `TimeSeriesSplit` was used with a deterministic seed-dependent gap of zero to two observations between the training and validation partitions. With an Optuna budget of 25 trials per model, experiments were conducted using seeds 42, 52, 62, 72, and 82.

For the CFF dataset, only an 80:20 stratified random split was used because the processed dataset does not retain explicit temporal or spatial identifiers. Shuffled stratified five-fold cross-validation was used during hyperparameter optimization to preserve the class distribution across folds. Two input-feature representations were evaluated using the same processed dataset and unchanged binary `CLASS` labels: (i) the original representation using `BURNED_AREA` directly and (ii) a log-transformed representation using $\log(1 + \text{BURNED_AREA})$. Experiments were repeated across 10 seeds, namely 42, 52, 62, 72, 82, 92, 102, 112, 122, and 132, with an Optuna budget of 50 trials per model.

For the PFF dataset, two evaluation protocols were considered. First, an 80:20 stratified random train-test split was used as the baseline. Second, a spatially grouped split was used to evaluate geographic generalization, with approximately 20% of the spatial groups assigned to the

test partition. Samples belonging to the same spatial cell, represented by `spatial_group = X_Y`, were prevented from appearing in both the training and test partitions. This reduces geographic leakage and provides a stricter estimate of model performance on unseen spatial locations. During hyperparameter optimization, shuffled stratified five-fold cross-validation was used for the random-split protocol, whereas five-fold `GroupKFold` cross-validation was used for the spatial protocol. Experiments were repeated across 10 seeds, namely 42, 52, 62, 72, 82, 92, 102, 112, 122, and 132, with an Optuna budget of 50 trials per model.

The low-sample training regime was evaluated using the same dataset-specific protocols described above. For each dataset, experiments were repeated across multiple seed values, and results were reported using mean and standard deviation. To keep the LSTR comparison tractable, only the best-performing RHNL configuration identified from the corresponding HSTR analysis was retained for each dataset. This avoids an excessive expansion of the low-sample comparison space while still evaluating the strongest RHNL representative against other classifiers under data-scarce conditions. Runtime analysis was also performed to compare training and inference costs across the considered ML and NL-based classifiers.

C. STATISTICAL ANALYSIS

Statistical significance was evaluated for the HSTR experiments using the seed-level macro-F1 scores. Within each dataset setting, classifiers evaluated under the same seed used identical held-out test observations and were therefore treated as matched measurements. The non-parametric Friedman test was used to assess the omnibus null hypothesis of equivalent classifier performance ranks. Tied ranks were corrected using the standard Friedman procedure. Given the small number of matched seed blocks, significance was estimated using 100,000 within-seed classifier-label permutations with the add-one correction, and Kendall's coefficient of concordance W was reported as an omnibus effect-size measure. Statistical significance was assessed at $\alpha = 0.05$.

The analysis was applied to AFF under random-split evaluation, CFF under the normal and log-transformed feature settings, and PFF under random and spatially grouped evaluation. AFF time-aware evaluation was excluded because its five chronological test sets are nested, substantially overlapping suffixes of the same ordered dataset and thus treating them as repeated blocks would overstate their independence. All analyzed settings contained $k = 13$ classifiers, giving 12 degrees of freedom for the Friedman test.

VI. RESULTS

A. ALGERIAN FOREST FIRE

The AFF results are summarized in terms of HSTR performance in Table 17 and LSTR behavior in Figure 6 and Table 18. The dataset has a small number of samples but relatively strong class separability, so the HSTR setting

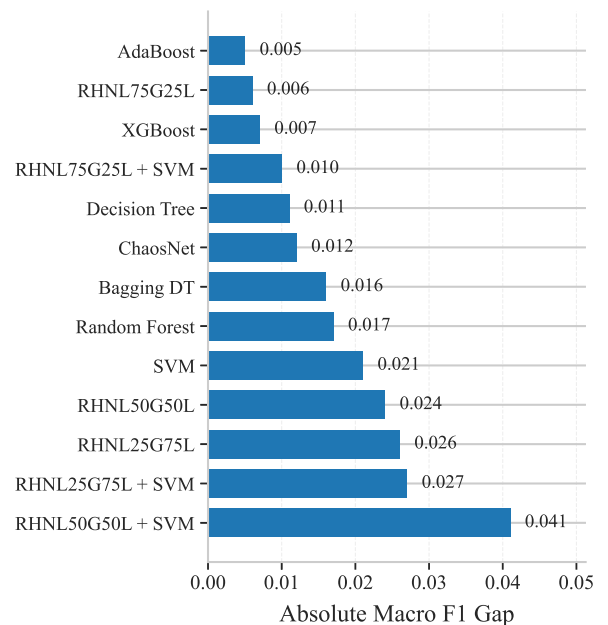
TABLE 16. Evaluation protocols used for the AFF, CFF, and PFF datasets.

Dataset	Evaluation protocol	Cross-validation during tuning	Optuna trials	Seeds	Purpose
AFF	80:20 stratified random split and seed-dependent chronological split with an 80% baseline training cut-off and at least 10% held out for testing	Shuffled five-fold KFold for the random split; five-fold TimeSeriesSplit with a seed-dependent gap of 0–2 observations for the time-aware split	25	42, 52, 62, 72, 82	Baseline comparison and temporal-generalization assessment across multiple chronological cut-offs
CFF	80:20 stratified random split for both the original and log-transformed burned-area input-feature representations	Shuffled stratified five-fold cross-validation	50	42, 52, 62, 72, 82, 92, 102, 112, 122, 132	Repeated-seed evaluation and assessment of sensitivity to the burned-area feature representation
PFF	80:20 stratified random split and spatially grouped split with approximately 20% of the spatial groups held out using <code>spatial_group = X_Y</code>	Shuffled stratified five-fold cross-validation for the random split; five-fold GroupKFold for the spatial split	50	42, 52, 62, 72, 82, 92, 102, 112, 122, 132	Baseline comparison and geographic-generalization assessment with reduced spatial leakage

leads to near-ceiling performance for several classifiers. Under the time-aware split, Random Forest and SVM achieve macro-F1 scores of 1.000 ± 0.000 , while Bagging DT and *RHNL*_{50G50L} + SVM also remain highly competitive with macro-F1 scores of 0.995 ± 0.009 and 0.995 ± 0.010 , respectively. Under the random split, we see similar behavior, with XGBoost, Random Forest, Bagging DT and SVM having macro-F1 scores greater than 0.979. These results suggest that in the high-sample setting, AFF is sufficiently separable for both conventional classifiers and RHNL-SVM hybrids to achieve good performance.

The comparison between random and time-aware evaluation further indicates that the performance of most high-performing models remains stable across the two protocols. This behavior is also reflected in Figure 5, which reports the temporal generalization gap, i.e., the absolute difference in macro-F1 between the random and time-aware settings. Several conventional ensemble models, SVM, and selected RHNL-SVM variants have small gaps indicating the temporal ordering present in AFF does not induce a severe distribution shift for the strongest performing models. The HSTR results suggest a competitive performance between classical ensemble methods and NL variants, not a clear dominance of any one model family. The bootstrap confidence intervals lend additional support to this interpretation, since the strongest models have narrow confidence ranges that are close to the upper bound of performance.

The LSTR results provide a more informative comparison under data scarcity. As shown in Figure 6(a), ChaosNet performs best in the earliest low-sample region under random-split evaluation, indicating that the chaos-based representation is effective when only a very small fraction of the training data is available. Under the time-aware protocol shown in Figure 6(b), the selected representative RHNL-SVM model, *RHNL*_{50G50L} + SVM, and ChaosNet

**FIGURE 5.** Temporal generalization gap.

produce the strongest early low-sample performance. Tree-based ensemble methods first need more training data before they achieve competitive performance. Further, the representative values in Table 18 show that Bagging DT, Random Forest and XGBoost become increasingly competitive with an increase in training fraction and dominate at higher training fractions. Hence, the main benefit of Neurochaos-based representations on AFF is most evident in the extreme low-sample regime rather than in the high-sample regime, where several classifiers already approach ceiling-level performance.

TABLE 17. AFF-HSTR test performance under random and time-aware evaluation protocols. Results are reported as mean $\pm$ standard deviation across multiple seed runs.

Model	Split	F1	Accuracy	Precision	Recall	95% CI (F1)
AdaBoost	Random	0.975 $\pm$ 0.016	0.976 $\pm$ 0.015	0.978 $\pm$ 0.015	0.974 $\pm$ 0.016	[0.954, 0.992]
	Time-aware	0.980 $\pm$ 0.018	0.981 $\pm$ 0.018	0.983 $\pm$ 0.015	0.979 $\pm$ 0.019	[0.957, 0.995]
Bagging DT	Random	0.979 $\pm$ 0.019	0.980 $\pm$ 0.018	0.983 $\pm$ 0.015	0.976 $\pm$ 0.021	[0.959, 0.996]
	Time-aware	0.995 $\pm$ 0.009	0.996 $\pm$ 0.009	0.996 $\pm$ 0.008	0.995 $\pm$ 0.010	[0.984, 1.000]
ChaosNet	Random	0.912 $\pm$ 0.024	0.914 $\pm$ 0.024	0.915 $\pm$ 0.023	0.913 $\pm$ 0.025	[0.875, 0.950]
	Time-aware	0.900 $\pm$ 0.048	0.905 $\pm$ 0.044	0.924 $\pm$ 0.031	0.896 $\pm$ 0.050	[0.855, 0.937]
Decision Tree	Random	0.966 $\pm$ 0.022	0.967 $\pm$ 0.021	0.972 $\pm$ 0.018	0.964 $\pm$ 0.024	[0.942, 0.987]
	Time-aware	0.977 $\pm$ 0.020	0.977 $\pm$ 0.019	0.981 $\pm$ 0.016	0.975 $\pm$ 0.022	[0.950, 0.995]
Random Forest	Random	0.983 $\pm$ 0.016	0.984 $\pm$ 0.015	0.986 $\pm$ 0.013	0.981 $\pm$ 0.018	[0.966, 0.996]
	Time-aware	1.000 $\pm$ 0.000	1.000 $\pm$ 0.000	1.000 $\pm$ 0.000	1.000 $\pm$ 0.000	[1.000, 1.000]
SVM	Random	0.979 $\pm$ 0.019	0.980 $\pm$ 0.018	0.980 $\pm$ 0.019	0.979 $\pm$ 0.019	[0.958, 0.996]
	Time-aware	1.000 $\pm$ 0.000	1.000 $\pm$ 0.000	1.000 $\pm$ 0.000	1.000 $\pm$ 0.000	[1.000, 1.000]
XGBoost	Random	0.987 $\pm$ 0.017	0.988 $\pm$ 0.016	0.990 $\pm$ 0.013	0.986 $\pm$ 0.019	[0.974, 1.000]
	Time-aware	0.994 $\pm$ 0.013	0.994 $\pm$ 0.013	0.994 $\pm$ 0.012	0.994 $\pm$ 0.012	[0.985, 1.000]
RHNL25G75L	Random	0.896 $\pm$ 0.022	0.898 $\pm$ 0.022	0.902 $\pm$ 0.025	0.899 $\pm$ 0.018	[0.855, 0.934]
	Time-aware	0.870 $\pm$ 0.056	0.876 $\pm$ 0.052	0.898 $\pm$ 0.043	0.867 $\pm$ 0.057	[0.823, 0.916]
RHNL25G75L + SVM	Random	0.954 $\pm$ 0.036	0.955 $\pm$ 0.035	0.955 $\pm$ 0.036	0.954 $\pm$ 0.035	[0.926, 0.979]
	Time-aware	0.981 $\pm$ 0.039	0.981 $\pm$ 0.039	0.983 $\pm$ 0.033	0.981 $\pm$ 0.038	[0.964, 1.000]
RHNL50G50L	Random	0.880 $\pm$ 0.031	0.882 $\pm$ 0.033	0.892 $\pm$ 0.038	0.882 $\pm$ 0.021	[0.837, 0.922]
	Time-aware	0.904 $\pm$ 0.036	0.907 $\pm$ 0.035	0.918 $\pm$ 0.031	0.900 $\pm$ 0.039	[0.864, 0.945]
RHNL50G50L + SVM	Random	0.954 $\pm$ 0.025	0.955 $\pm$ 0.024	0.957 $\pm$ 0.023	0.952 $\pm$ 0.026	[0.926, 0.975]
	Time-aware	0.995 $\pm$ 0.010	0.995 $\pm$ 0.010	0.995 $\pm$ 0.011	0.995 $\pm$ 0.010	[0.984, 1.000]
RHNL75G25L	Random	0.892 $\pm$ 0.029	0.894 $\pm$ 0.030	0.899 $\pm$ 0.033	0.892 $\pm$ 0.025	[0.852, 0.929]
	Time-aware	0.898 $\pm$ 0.043	0.902 $\pm$ 0.039	0.921 $\pm$ 0.029	0.894 $\pm$ 0.045	[0.854, 0.939]
RHNL75G25L + SVM	Random	0.967 $\pm$ 0.028	0.967 $\pm$ 0.028	0.967 $\pm$ 0.028	0.967 $\pm$ 0.028	[0.942, 0.988]
	Time-aware	0.957 $\pm$ 0.033	0.957 $\pm$ 0.033	0.958 $\pm$ 0.033	0.958 $\pm$ 0.032	[0.928, 0.985]

A computational efficiency analysis was also carried out for the low-sample training regime to compare training and inference costs across the considered classifiers. ChaosNet had the shortest training time in both random and time-aware setups, with training times in the microsecond range. However, ensemble methods like Bagging DT and Random Forest took considerably longer training time. RHNL_{50G50L} + SVM had a moderate training cost, but a higher inference latency than ChaosNet since the transformed features are passed to an additional SVM classifier. Representative runtimes are reported in Table 29 in the appendix.

B. CANADIAN FOREST FIRES

The CFF results are summarized in Table 19, Figures 7–8, and Table 20. Unlike AFF and PFF, the processed CFF dataset does not preserve explicit temporal or spatial identifiers in the final processed tabular representation. Thus, 10 independent seeds with stratified random splitting are used. Along with that, two feature formulations are evaluated: the original

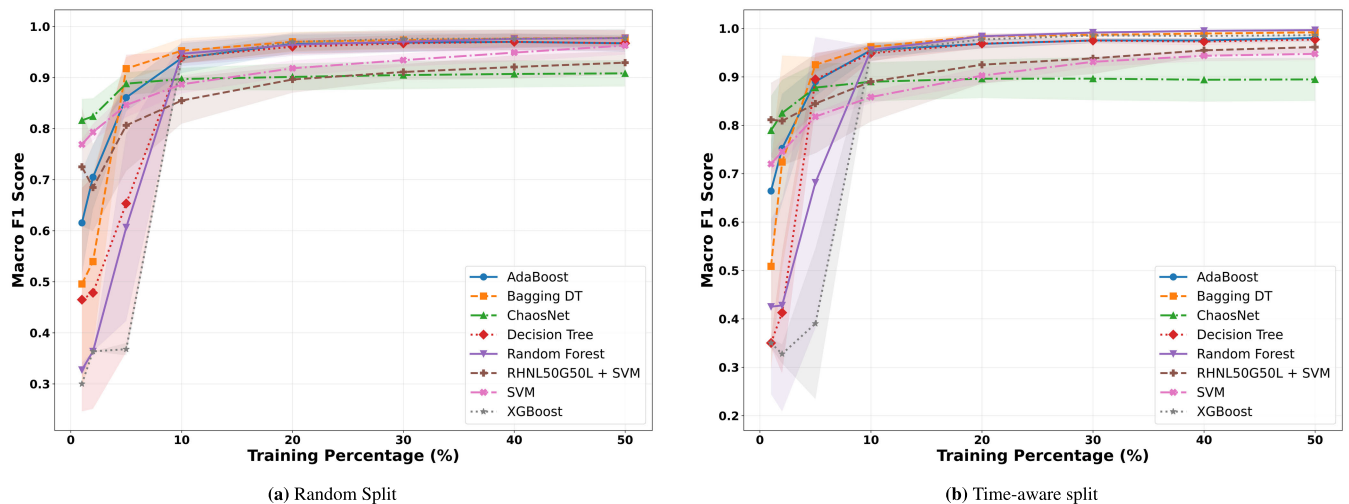
burned-area representation and a representation based on $\log(1 + \text{Burned_Area})$, a log-transformed representation.

CFF is substantially less separable than AFF under HSTR. The best models are traditional tree-based ensembles. Random Forest achieves the best macro-F1 of 0.740 under both normal and log-transformed settings. Bagging DT and XGBoost are close behind with AdaBoost and Decision Tree also competitive but a little weaker. The nearly identical performance between the normal and log-transformed formulations suggests that log-scaling of the burned-area feature does not materially alter the HSTR performance pattern for CFF.

The NL-based models show a more nuanced behavior. ChaosNet and the RHNL cosine-similarity variants provide moderately balanced performance, but their macro-F1 scores remain below those of the leading ensemble methods. In contrast, the RHNL-SVM variants yield comparatively high precision but lower recall, indicating a conservative classification tendency. This behavior is important because it suggests that the RHNL-SVM decision boundary reduces

TABLE 18. AFF-LSTR macro-F1 performance at representative training fractions under random and time-aware evaluation. Results are reported as mean $\pm$ standard deviation across multiple seed runs.

Model	Split	1%	5%	10%	50%
AdaBoost	Random	0.615 $\pm$ 0.10	0.861 $\pm$ 0.03	0.938 $\pm$ 0.03	0.967 $\pm$ 0.02
	Time-aware	0.664 $\pm$ 0.10	0.890 $\pm$ 0.03	0.955 $\pm$ 0.01	0.980 $\pm$ 0.01
Bagging DT	Random	0.496 $\pm$ 0.20	0.917 $\pm$ 0.02	0.953 $\pm$ 0.02	0.978 $\pm$ 0.01
	Time-aware	0.509 $\pm$ 0.21	0.925 $\pm$ 0.02	0.962 $\pm$ 0.01	0.992 $\pm$ 0.00
ChaosNet	Random	0.816 $\pm$ 0.04	0.888 $\pm$ 0.02	0.897 $\pm$ 0.02	0.908 $\pm$ 0.03
	Time-aware	0.789 $\pm$ 0.07	0.878 $\pm$ 0.05	0.890 $\pm$ 0.04	0.895 $\pm$ 0.04
Decision Tree	Random	0.465 $\pm$ 0.22	0.653 $\pm$ 0.29	0.940 $\pm$ 0.01	0.967 $\pm$ 0.01
	Time-aware	0.350 $\pm$ 0.01	0.895 $\pm$ 0.05	0.949 $\pm$ 0.02	0.977 $\pm$ 0.00
Random Forest	Random	0.327 $\pm$ 0.01	0.607 $\pm$ 0.18	0.947 $\pm$ 0.02	0.977 $\pm$ 0.02
	Time-aware	0.425 $\pm$ 0.18	0.682 $\pm$ 0.30	0.956 $\pm$ 0.01	0.997 $\pm$ 0.00
<i>RHNL</i> _{50G50L} + SVM	Random	0.725 $\pm$ 0.12	0.806 $\pm$ 0.09	0.855 $\pm$ 0.05	0.929 $\pm$ 0.02
	Time-aware	0.811 $\pm$ 0.08	0.845 $\pm$ 0.10	0.890 $\pm$ 0.08	0.962 $\pm$ 0.01
SVM	Random	0.769 $\pm$ 0.01	0.846 $\pm$ 0.02	0.887 $\pm$ 0.02	0.963 $\pm$ 0.02
	Time-aware	0.720 $\pm$ 0.02	0.818 $\pm$ 0.01	0.858 $\pm$ 0.01	0.948 $\pm$ 0.01
XGBoost	Random	0.300 $\pm$ 0.00	0.368 $\pm$ 0.01	0.938 $\pm$ 0.01	0.977 $\pm$ 0.02
	Time-aware	0.350 $\pm$ 0.01	0.390 $\pm$ 0.16	0.951 $\pm$ 0.02	0.986 $\pm$ 0.00

**FIGURE 6.** AFF-LSTR macro-F1 performance under random and time-aware evaluation. Shaded regions indicate variation across repeated seed-based evaluations.

false-positive predictions but misses a larger fraction of positive fire cases. Thus, for CFF-HSTR, the empirical evidence supports ensemble-based dominance rather than broad NL superiority.

The LSTR analysis reveals a complementary pattern. ChaosNet performs competitively in the extreme

low-sample regime, particularly at the earliest training fractions, suggesting that chaos-based transformations can be useful when only a very small training subset is available. However, as the training fraction increases, Random Forest becomes the dominant model in both the normal and log-transformed settings. This transition indicates that, for CFF,

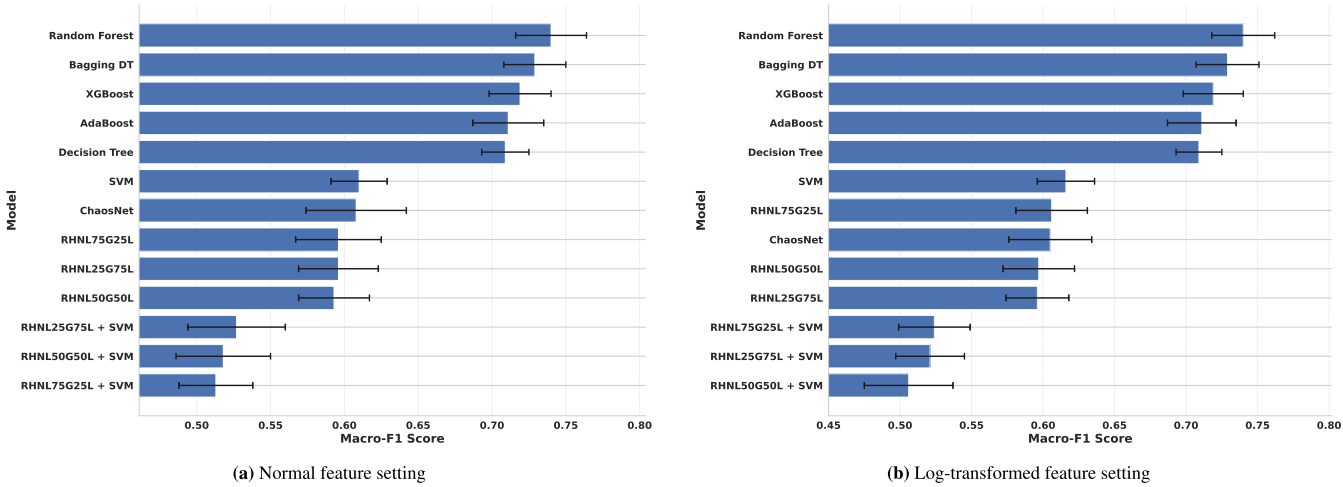


FIGURE 7. CFF-HSTR macro-F1 mean and standard deviation under normal and log-transformed burned-area feature settings.

TABLE 19. CFF-HSTR test performance under normal and log-transformed feature settings. Results are reported as mean $\pm$ standard deviation across multiple seed runs.

Model	Setting	F1	Accuracy	Precision	Recall	95% CI (F1)
Random Forest	Normal	0.740 $\pm$ 0.024	0.831 $\pm$ 0.016	0.763 $\pm$ 0.026	0.724 $\pm$ 0.023	[0.722, 0.757]
	Log-transformed	0.740 $\pm$ 0.022	0.831 $\pm$ 0.015	0.763 $\pm$ 0.024	0.724 $\pm$ 0.022	[0.721, 0.757]
Bagging DT	Normal	0.729 $\pm$ 0.021	0.835 $\pm$ 0.011	0.778 $\pm$ 0.021	0.705 $\pm$ 0.020	[0.711, 0.747]
	Log-transformed	0.729 $\pm$ 0.022	0.835 $\pm$ 0.011	0.779 $\pm$ 0.020	0.705 $\pm$ 0.022	[0.712, 0.747]
XGBoost	Normal	0.719 $\pm$ 0.021	0.823 $\pm$ 0.014	0.751 $\pm$ 0.026	0.701 $\pm$ 0.019	[0.700, 0.736]
	Log-transformed	0.719 $\pm$ 0.021	0.823 $\pm$ 0.014	0.751 $\pm$ 0.026	0.701 $\pm$ 0.019	[0.700, 0.736]
AdaBoost	Normal	0.711 $\pm$ 0.024	0.799 $\pm$ 0.017	0.712 $\pm$ 0.024	0.711 $\pm$ 0.024	[0.694, 0.730]
	Log-transformed	0.711 $\pm$ 0.024	0.799 $\pm$ 0.017	0.712 $\pm$ 0.024	0.711 $\pm$ 0.024	[0.694, 0.730]
Decision Tree	Normal	0.709 $\pm$ 0.016	0.795 $\pm$ 0.013	0.708 $\pm$ 0.017	0.712 $\pm$ 0.020	[0.691, 0.727]
	Log-transformed	0.709 $\pm$ 0.016	0.795 $\pm$ 0.013	0.708 $\pm$ 0.017	0.712 $\pm$ 0.020	[0.691, 0.727]
SVM	Normal	0.610 $\pm$ 0.019	0.658 $\pm$ 0.023	0.620 $\pm$ 0.014	0.668 $\pm$ 0.020	[0.593, 0.627]
	Log-transformed	0.616 $\pm$ 0.020	0.665 $\pm$ 0.022	0.624 $\pm$ 0.017	0.673 $\pm$ 0.025	[0.599, 0.633]
ChaosNet	Normal	0.608 $\pm$ 0.034	0.689 $\pm$ 0.033	0.606 $\pm$ 0.030	0.633 $\pm$ 0.040	[0.594, 0.627]
	Log-transformed	0.605 $\pm$ 0.029	0.691 $\pm$ 0.028	0.603 $\pm$ 0.027	0.626 $\pm$ 0.036	[0.590, 0.624]
RHNL25G75L	Normal	0.596 $\pm$ 0.027	0.687 $\pm$ 0.036	0.596 $\pm$ 0.026	0.613 $\pm$ 0.031	[0.580, 0.615]
	Log-transformed	0.596 $\pm$ 0.022	0.688 $\pm$ 0.024	0.594 $\pm$ 0.022	0.613 $\pm$ 0.028	[0.582, 0.615]
RHNL50G50L	Normal	0.593 $\pm$ 0.024	0.685 $\pm$ 0.032	0.592 $\pm$ 0.023	0.609 $\pm$ 0.024	[0.578, 0.612]
	Log-transformed	0.597 $\pm$ 0.025	0.699 $\pm$ 0.026	0.594 $\pm$ 0.025	0.605 $\pm$ 0.025	[0.580, 0.615]
RHNL75G25L	Normal	0.596 $\pm$ 0.029	0.680 $\pm$ 0.041	0.597 $\pm$ 0.026	0.618 $\pm$ 0.028	[0.580, 0.614]
	Log-transformed	0.606 $\pm$ 0.025	0.699 $\pm$ 0.026	0.603 $\pm$ 0.023	0.621 $\pm$ 0.031	[0.590, 0.625]
RHNL25G75L + SVM	Normal	0.527 $\pm$ 0.033	0.790 $\pm$ 0.009	0.756 $\pm$ 0.057	0.545 $\pm$ 0.019	[0.511, 0.545]
	Log-transformed	0.521 $\pm$ 0.024	0.788 $\pm$ 0.009	0.757 $\pm$ 0.081	0.541 $\pm$ 0.014	[0.505, 0.540]
RHNL50G50L + SVM	Normal	0.518 $\pm$ 0.032	0.791 $\pm$ 0.007	0.819 $\pm$ 0.045	0.541 $\pm$ 0.017	[0.502, 0.537]
	Log-transformed	0.506 $\pm$ 0.031	0.786 $\pm$ 0.006	0.780 $\pm$ 0.063	0.534 $\pm$ 0.017	[0.492, 0.526]
RHNL75G25L + SVM	Normal	0.513 $\pm$ 0.025	0.789 $\pm$ 0.006	0.798 $\pm$ 0.067	0.538 $\pm$ 0.014	[0.497, 0.531]
	Log-transformed	0.524 $\pm$ 0.025	0.790 $\pm$ 0.006	0.784 $\pm$ 0.067	0.543 $\pm$ 0.014	[0.506, 0.542]

tree ensembles benefit more strongly from additional data, whereas ChaosNet provides its main advantage under severe sample scarcity.

Overall, CFF highlights a different operating regime from AFF. While AFF is highly separable and allows several classifiers to approach ceiling-level performance,

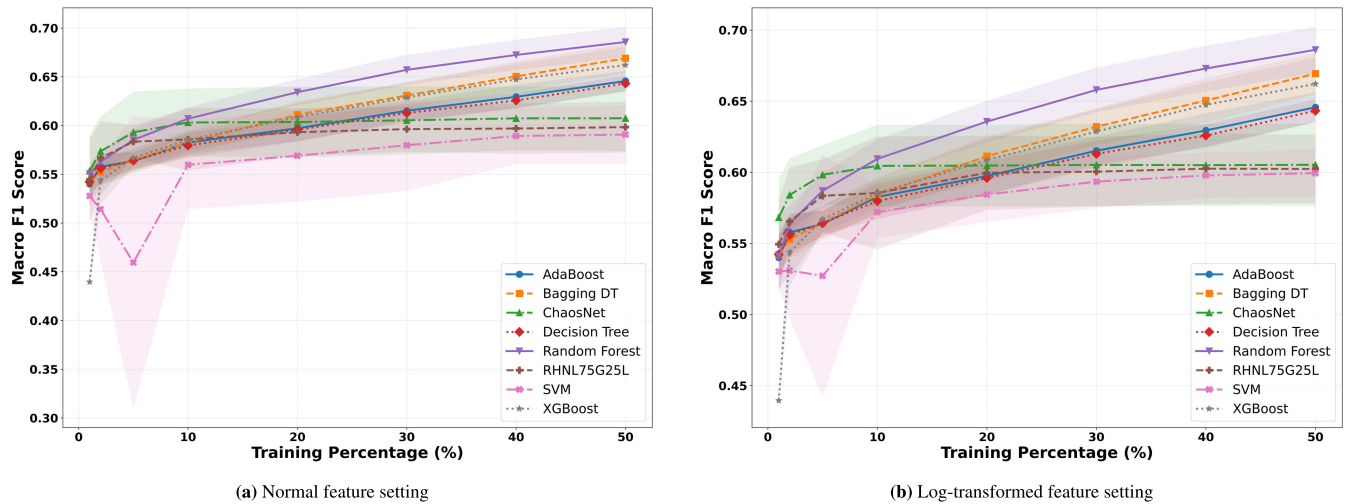


FIGURE 8. CFF-LSTR macro-F1 performance under normal and log-transformed burned-area feature settings. Shaded regions indicate variation across repeated seed-based evaluations.

TABLE 20. CFF-LSTR macro-F1 performance at representative training fractions under normal and log-transformed feature settings. Results are reported as mean $\pm$ standard deviation across repeated runs.

Model	Setting	1%	5%	10%	50%
AdaBoost	Normal	0.540 ± 0.02	0.564 ± 0.01	0.583 ± 0.01	0.646 ± 0.01
	Log	0.540 ± 0.02	0.564 ± 0.01	0.583 ± 0.01	0.646 ± 0.01
Bagging DT	Normal	0.543 ± 0.01	0.565 ± 0.01	0.583 ± 0.01	0.669 ± 0.02
	Log	0.542 ± 0.01	0.564 ± 0.01	0.584 ± 0.01	0.669 ± 0.01
ChaosNet	Normal	0.554 ± 0.04	0.593 ± 0.04	0.603 ± 0.03	0.607 ± 0.03
	Log	0.568 ± 0.03	0.598 ± 0.02	0.605 ± 0.03	0.605 ± 0.03
Decision Tree	Normal	0.542 ± 0.01	0.564 ± 0.01	0.580 ± 0.01	0.643 ± 0.01
	Log	0.542 ± 0.01	0.564 ± 0.01	0.580 ± 0.01	0.643 ± 0.01
Random Forest	Normal	0.550 ± 0.01	0.585 ± 0.01	0.607 ± 0.01	0.686 ± 0.02
	Log	0.541 ± 0.03	0.587 ± 0.01	0.609 ± 0.01	0.686 ± 0.02
<i>RHNL</i> _{75G25L}	Normal	0.544 ± 0.04	0.583 ± 0.02	0.586 ± 0.03	0.598 ± 0.03
	Log	0.549 ± 0.03	0.583 ± 0.03	0.585 ± 0.04	0.602 ± 0.02
SVM	Normal	0.528 ± 0.02	0.460 ± 0.15	0.560 ± 0.05	0.591 ± 0.03
	Log	0.530 ± 0.01	0.527 ± 0.08	0.572 ± 0.02	0.599 ± 0.02
XGBoost	Normal	0.440 ± 0.01	0.567 ± 0.01	0.585 ± 0.01	0.662 ± 0.02
	Log	0.440 ± 0.01	0.567 ± 0.01	0.585 ± 0.01	0.662 ± 0.02

CFF remains more challenging and exposes clearer differences between model families. The log-transformed formulation acts primarily as a robustness check rather than a source of improved performance, and the main

inference is that Neurochaos-based models are most relevant for the low-sample setting, whereas ensemble models remain preferable when sufficient CFF training data are available.

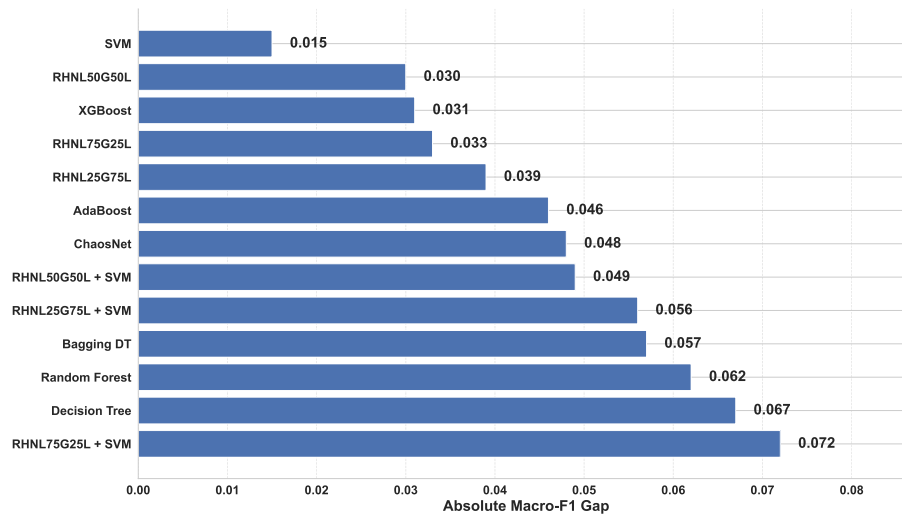


FIGURE 9. PFF-HSTR spatial generalization gap.

TABLE 21. PFF-HSTR performance under random and spatially grouped evaluation protocols. Results are reported as mean $\pm$ standard deviation across multiple seed runs.

Model	Setting	F1	Accuracy	Precision	Recall	95% CI (F1)
Random Forest	Random	0.576 ± 0.051	0.578 ± 0.051	0.579 ± 0.052	0.578 ± 0.051	[0.548, 0.606]
	Spatial	0.514 ± 0.040	0.518 ± 0.040	0.517 ± 0.042	0.517 ± 0.041	[0.490, 0.544]
Bagging DT	Random	0.566 ± 0.031	0.568 ± 0.030	0.570 ± 0.030	0.568 ± 0.030	[0.539, 0.601]
	Spatial	0.509 ± 0.032	0.514 ± 0.034	0.513 ± 0.036	0.512 ± 0.034	[0.485, 0.539]
<i>RHNL</i> _{25G75L} + SVM	Random	0.553 ± 0.043	0.556 ± 0.042	0.557 ± 0.043	0.556 ± 0.042	[0.525, 0.586]
	Spatial	0.497 ± 0.045	0.505 ± 0.043	0.506 ± 0.055	0.505 ± 0.050	[0.477, 0.532]
Decision Tree	Random	0.547 ± 0.040	0.550 ± 0.039	0.552 ± 0.040	0.550 ± 0.039	[0.520, 0.580]
	Spatial	0.480 ± 0.048	0.484 ± 0.045	0.486 ± 0.047	0.486 ± 0.047	[0.460, 0.518]
XGBoost	Random	0.540 ± 0.049	0.542 ± 0.048	0.543 ± 0.050	0.542 ± 0.048	[0.512, 0.572]
	Spatial	0.509 ± 0.040	0.512 ± 0.040	0.515 ± 0.043	0.514 ± 0.041	[0.486, 0.542]
AdaBoost	Random	0.535 ± 0.044	0.537 ± 0.044	0.537 ± 0.044	0.537 ± 0.044	[0.507, 0.566]
	Spatial	0.489 ± 0.052	0.493 ± 0.049	0.492 ± 0.052	0.492 ± 0.052	[0.474, 0.529]
ChaosNet	Random	0.534 ± 0.043	0.536 ± 0.043	0.536 ± 0.044	0.536 ± 0.043	[0.508, 0.565]
	Spatial	0.486 ± 0.048	0.492 ± 0.046	0.491 ± 0.054	0.491 ± 0.051	[0.461, 0.518]
<i>RHNL</i> _{50G50L} + SVM	Random	0.532 ± 0.051	0.534 ± 0.052	0.535 ± 0.053	0.534 ± 0.052	[0.505, 0.564]
	Spatial	0.483 ± 0.052	0.493 ± 0.051	0.495 ± 0.053	0.495 ± 0.052	[0.466, 0.520]
<i>RHNL</i> _{75G25L} + SVM	Random	0.532 ± 0.036	0.533 ± 0.037	0.533 ± 0.037	0.533 ± 0.037	[0.505, 0.562]
	Spatial	0.460 ± 0.046	0.465 ± 0.047	0.466 ± 0.054	0.466 ± 0.051	[0.435, 0.491]
SVM	Random	0.529 ± 0.043	0.530 ± 0.042	0.530 ± 0.043	0.530 ± 0.042	[0.498, 0.559]
	Spatial	0.514 ± 0.028	0.518 ± 0.026	0.522 ± 0.031	0.521 ± 0.031	[0.494, 0.552]
<i>RHNL</i> _{25G75L}	Random	0.520 ± 0.046	0.522 ± 0.047	0.523 ± 0.049	0.522 ± 0.047	[0.492, 0.551]
	Spatial	0.481 ± 0.053	0.486 ± 0.051	0.488 ± 0.054	0.489 ± 0.052	[0.452, 0.510]
<i>RHNL</i> _{75G25L}	Random	0.516 ± 0.055	0.517 ± 0.055	0.518 ± 0.056	0.517 ± 0.055	[0.487, 0.546]
	Spatial	0.483 ± 0.054	0.489 ± 0.053	0.491 ± 0.053	0.492 ± 0.051	[0.458, 0.513]
<i>RHNL</i> _{50G50L}	Random	0.507 ± 0.042	0.512 ± 0.042	0.512 ± 0.043	0.512 ± 0.042	[0.482, 0.542]
	Spatial	0.477 ± 0.042	0.482 ± 0.045	0.485 ± 0.043	0.484 ± 0.040	[0.450, 0.506]

C. PORTUGAL FOREST FIRE

The PFF results are summarized in terms of HSTR performance in Table 21, spatial generalization behavior in Figure 9, and LSTR behavior in Figure 11 and Table 22.

In contrast to AFF and CFF, the PFF dataset retains explicit spatial coordinates through the X and Y grid locations. Therefore, two evaluation protocols are considered: a conventional random split and a spatially grouped split. In the

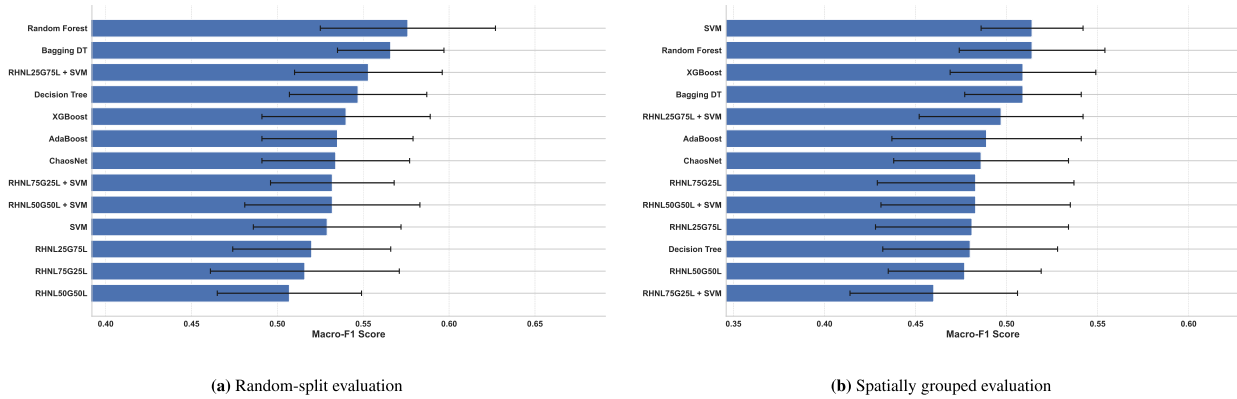


FIGURE 10. PFF-HSTR macro-F1 mean and standard deviation under random and spatially grouped evaluation.

TABLE 22. PFF-LSTR macro-F1 performance at representative training fractions under random and spatially grouped evaluation. Results are reported as mean $\pm$ standard deviation across multiple seed runs.

Model	Setting	1%	5%	10%	50%
AdaBoost	Random	0.474 $\pm$ 0.01	0.501 $\pm$ 0.01	0.505 $\pm$ 0.01	0.528 $\pm$ 0.01
	Spatial	0.463 $\pm$ 0.01	0.486 $\pm$ 0.01	0.492 $\pm$ 0.01	0.504 $\pm$ 0.02
Bagging DT	Random	0.477 $\pm$ 0.01	0.500 $\pm$ 0.01	0.510 $\pm$ 0.01	0.545 $\pm$ 0.02
	Spatial	0.413 $\pm$ 0.07	0.478 $\pm$ 0.01	0.489 $\pm$ 0.01	0.502 $\pm$ 0.02
ChaosNet	Random	0.489 $\pm$ 0.01	0.500 $\pm$ 0.01	0.510 $\pm$ 0.02	0.525 $\pm$ 0.03
	Spatial	0.469 $\pm$ 0.02	0.481 $\pm$ 0.02	0.484 $\pm$ 0.02	0.486 $\pm$ 0.04
Decision Tree	Random	0.378 $\pm$ 0.07	0.494 $\pm$ 0.01	0.501 $\pm$ 0.01	0.523 $\pm$ 0.01
	Spatial	0.364 $\pm$ 0.07	0.480 $\pm$ 0.01	0.489 $\pm$ 0.01	0.503 $\pm$ 0.02
Random Forest	Random	0.363 $\pm$ 0.06	0.501 $\pm$ 0.01	0.506 $\pm$ 0.01	0.543 $\pm$ 0.02
	Spatial	0.342 $\pm$ 0.05	0.475 $\pm$ 0.02	0.483 $\pm$ 0.01	0.502 $\pm$ 0.02
<i>RHNL</i> _{50G50L} + SVM	Random	0.485 $\pm$ 0.01	0.498 $\pm$ 0.01	0.500 $\pm$ 0.01	0.517 $\pm$ 0.02
	Spatial	0.471 $\pm$ 0.01	0.491 $\pm$ 0.01	0.493 $\pm$ 0.02	0.492 $\pm$ 0.03
SVM	Random	0.465 $\pm$ 0.02	0.496 $\pm$ 0.01	0.503 $\pm$ 0.01	0.519 $\pm$ 0.02
	Spatial	0.455 $\pm$ 0.01	0.475 $\pm$ 0.02	0.486 $\pm$ 0.02	0.494 $\pm$ 0.02
XGBoost	Random	0.333 $\pm$ 0.00	0.497 $\pm$ 0.01	0.503 $\pm$ 0.01	0.533 $\pm$ 0.02
	Spatial	0.324 $\pm$ 0.02	0.477 $\pm$ 0.02	0.481 $\pm$ 0.02	0.498 $\pm$ 0.02

TABLE 23. Friedman analysis of HSTR macro-F1 across matched seed runs.

Dataset	Setting	<i>n</i>	χ^2_F	p_{perm}	Kendall's <i>W</i>
AFF	Random	5	44.890	1/100001	0.748
CFF	Normal	10	107.997	1/100001	0.900
CFF	Log-trans.	10	108.604	1/100001	0.905
PFF	Random	10	19.998	0.0587	0.167
PFF	Spatial	10	18.626	0.0884	0.155

spatial setting, samples from the same spatial cell are prevented from appearing simultaneously in the training and testing partitions, providing a stricter estimate of geographic generalization.

Under HSTR, all models achieve substantially lower macro-F1 scores on PFF than on AFF, indicating that PFF is a more difficult and less separable classification problem. Under the random split, Random Forest achieves the highest macro-F1 score of 0.576, followed by Bagging DT (0.566), *RHNL*_{25G75L} + SVM (0.553), Decision Tree (0.547), and XGBoost (0.540). The remaining models lie in a narrow performance band between approximately 0.507 and 0.535, indicating that the random-split setting provides only modest separation between model families.

Performance decreases under spatially grouped evaluation for nearly all models. Random Forest and SVM obtain the highest spatial macro-F1 scores of 0.514, followed closely by Bagging DT and XGBoost, both with macro-F1 scores

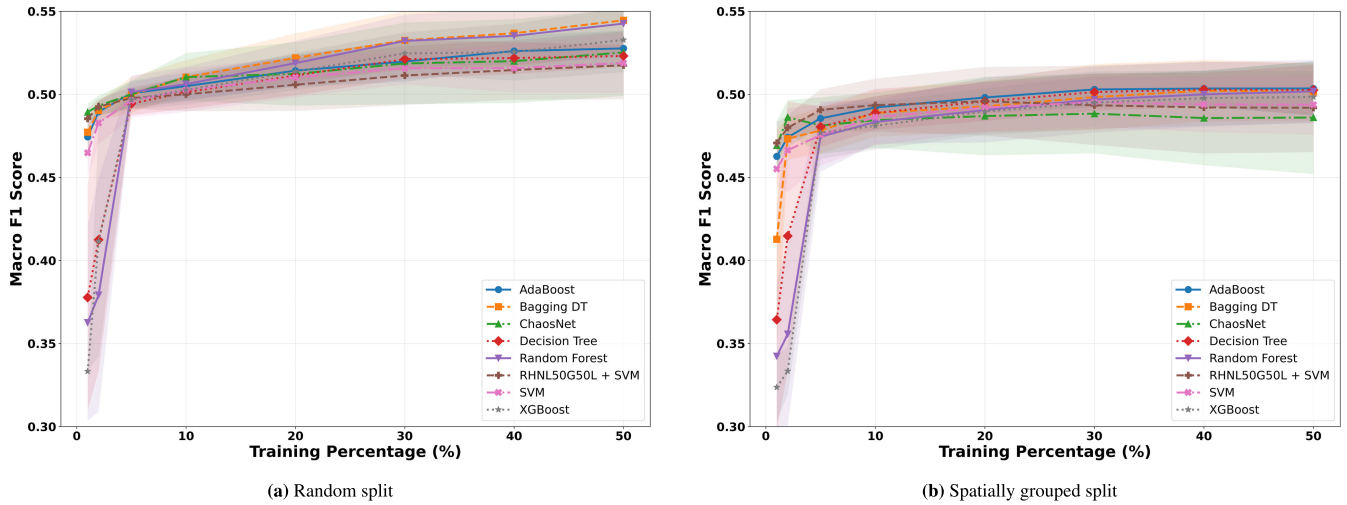


FIGURE 11. PFF-LSTR macro-F1 performance under random and spatially grouped evaluation. Shaded regions indicate variability across repeated seed-based evaluations.

TABLE 24. AFF-HSTR seed-wise tuned hyperparameters under random split evaluation.

Model	Params	Seed 42	Seed 52	Seed 62	Seed 72	Seed 82
AdaBoost	(n_{est}, lr)	(150, 0.01)	(200, 0.01)	(150, 1.0)	(100, 1.0)	(50, 0.01)
Bagging DT	(mf, ms, n_{est})	(0.6, 0.6, 200)	(1.0, 1.0, 150)	(0.8, 0.8, 150)	(1.0, 1.0, 100)	(1.0, 0.6, 200)
ChaosNet	(q, b, ϵ)	(0.4650, 0.3424, 0.2635)	(0.4682, 0.1398, 0.2607)	(0.4825, 0.3815, 0.3525)	(0.9941, 0.4978, 0.2166)	(0.6497, 0.1272, 0.3640)
Decision Tree	(d, msl, mss)	(20, 2, 10)	(40, 1, 2)	(50, 4, 10)	(10, 1, 2)	(20, 1, 5)
Random Forest	(d, msl, mss, n_{est})	(30, 1, 5, 100)	(40, 1, 10, 50)	(40, 4, 5, 100)	(40, 4, 2, 250)	(20, 4, 2, 50)
SVM	$(C, kernel)$	(1, linear)	(1, linear)	(10, linear)	(100, linear)	(100, linear)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.6, 0.2, 10, 150, 0.1, 1, 0.8)	(1.0, 0.2, 3, 150, 1, 2, 0.6)	(1.0, 0.01, 10, 50, 1, 1, 1.0)	(0.6, 0.1, 7, 100, 0.2, 0.8)	(1.0, 0.1, 7, 100, 0.1, 1.5, 0.6)
RHNL25G75L	(q, b, ϵ)	(0.40115, 0.06504, 0.25835)	(0.49967, 0.12014, 0.29850)	(0.31199, 0.39045, 0.16446)	(0.43309, 0.44360, 0.29044)	(0.41710, 0.17728, 0.29460)
RHNL25G75L + SVM	(q, b, ϵ)	(0.39029, 0.24168, 0.10985)	(0.38482, 0.21466, 0.23295)	(0.34210, 0.42898, 0.19255)	(0.49617, 0.32886, 0.23624)	(0.42582, 0.41738, 0.11636)
RHNL50G50L	(q, b, ϵ)	(0.40810, 0.06098, 0.25149)	(0.43647, 0.29854, 0.25393)	(0.33854, 0.38223, 0.14835)	(0.43309, 0.44360, 0.29044)	(0.41717, 0.17728, 0.29460)
RHNL50G50L + SVM	(q, b, ϵ)	(0.38933, 0.20599, 0.14264)	(0.38482, 0.21466, 0.23295)	(0.44326, 0.26994, 0.17035)	(0.49631, 0.48703, 0.11499)	(0.44864, 0.13468, 0.26997)
RHNL75G25L	(q, b, ϵ)	(0.36534, 0.25357, 0.26902)	(0.37740, 0.12400, 0.26734)	(0.36904, 0.43607, 0.14482)	(0.43309, 0.44360, 0.29044)	(0.40820, 0.17135, 0.17135)
RHNL75G25L + SVM	(q, b, ϵ)	(0.49768, 0.25215, 0.19885)	(0.38482, 0.21466, 0.23295)	(0.32365, 0.29354, 0.17210)	(0.31226, 0.31760, 0.11726)	(0.48526, 0.16164, 0.23768)

TABLE 25. AFF-HSTR seed-wise tuned hyperparameters under time-aware evaluation.

Model	Params	Seed 42	Seed 52	Seed 62	Seed 72	Seed 82
AdaBoost	(n_{est}, lr)	(150, 1.0)	(200, 0.01)	(150, 0.01)	(150, 0.01)	(50, 0.1)
Bagging DT	(mf, ms, n_{est})	(0.6, 1.0, 100)	(1.0, 1.0, 150)	(0.8, 0.8, 150)	(1.0, 0.6, 150)	(0.8, 0.8, 150)
ChaosNet	(q, b, ϵ)	(0.4904, 0.4447, 0.3448)	(0.5295, 0.3736, 0.3931)	(0.5874, 0.1922, 0.1469)	(0.5132, 0.1234, 0.2371)	(0.6447, 0.4943, 0.4518)
Decision Tree	(d, msl, mss)	(10, 2, 5)	(30, 2, 2)	(30, 2, 5)	(20, 2, 5)	(30, 2, 2)
Random Forest	(d, msl, mss, n_{est})	(20, 2, 5, 100)	(30, 2, 5, 50)	(30, 1, 2, 250)	(30, 2, 10, 150)	(20, 1, 10, 200)
SVM	$(C, kernel)$	(100, linear)	(10, linear)	(10, linear)	(10, linear)	(10, linear)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.8, 0.01, 3, 100, 1, 1.5, 1.0)	(0.8, 0.01, 3, 100, 1, 2, 0.8)	(0.6, 0.2, 7, 200, 0.1, 1.5, 0.6)	(0.8, 0.01, 10, 100, 1, 1, 0.8)	(0.8, 0.2, 7, 200, 0.1, 1.5, 0.6)
RHNL25G75L	(q, b, ϵ)	(0.07222, 0.06271, 0.16930)	(0.44814, 0.34951, 0.24770)	(0.44959, 0.28175, 0.20627)	(0.26038, 0.42469, 0.10462)	(0.45393, 0.09468, 0.27070)
RHNL25G75L + SVM	(q, b, ϵ)	(0.13516, 0.25227, 0.23725)	(0.46394, 0.18408, 0.29034)	(0.40235, 0.33860, 0.24695)	(0.43309, 0.44360, 0.29044)	(0.38922, 0.17954, 0.23936)
RHNL50G50L	(q, b, ϵ)	(0.01505, 0.22469, 0.12248)	(0.49418, 0.13909, 0.28579)	(0.32389, 0.42751, 0.17623)	(0.41082, 0.33435, 0.26671)	(0.49526, 0.18259, 0.24131)
RHNL50G50L + SVM	(q, b, ϵ)	(0.38824, 0.49530, 0.24724)	(0.44451, 0.12940, 0.29636)	(0.37846, 0.41273, 0.21604)	(0.46965, 0.39486, 0.10978)	(0.40000, 0.15206, 0.24714)
RHNL75G25L	(q, b, ϵ)	(0.01505, 0.22469, 0.12248)	(0.49418, 0.13909, 0.28578)	(0.21691, 0.44135, 0.20538)	(0.49834, 0.49568, 0.29847)	(0.45167, 0.21399, 0.26801)
RHNL75G25L + SVM	(q, b, ϵ)	(0.23347, 0.39473, 0.06790)	(0.45755, 0.22782, 0.29954)	(0.37958, 0.40785, 0.23854)	(0.01278, 0.40966, 0.12690)	(0.40063, 0.21986, 0.23813)

of 0.509. This reduction confirms that random evaluation overestimates generalization when spatially neighboring samples can be shared across training and testing partitions. The spatial results also show that no model achieves strong geographic transfer on PFF, suggesting substantial spatial heterogeneity in the relationship between meteorological, temporal, and fire-weather variables and the binary burned-area target.

The spatial generalization gap in Figure 9 further supports this interpretation. Although the relative ranking of the strongest models is broadly preserved, the absolute performance of most classifiers shifts downward under spatial evaluation. This behavior indicates that the primary

effect of spatially grouped validation is not to reverse the model ranking, but to expose the reduced magnitude of generalization performance under geographically disjoint testing. Consequently, the PFF-HSTR results should be interpreted as evidence of moderate predictive structure under random splitting but limited geographic robustness under spatially grouped evaluation.

The LSTR results provide additional insight into data efficiency. Under the random split, ChaosNet performs best at the earliest training fractions, performs the best at 1% achieving a macro-F1 of 0.489 ± 0.01 , while *RHNL50G50L* + *SVM* remains competitive with 0.485 ± 0.01 . As the available training data increases, conventional classifiers improve, with

TABLE 26. AFF-HSTR training cross-validation macro-F1 during hyperparameter tuning.

Model	Random Split CV Macro-F1	Time-aware CV Macro-F1
AdaBoost	0.969 ± 0.009	0.905 ± 0.048
Bagging DT	0.984 ± 0.000	0.931 ± 0.028
ChaosNet	0.929 ± 0.018	0.893 ± 0.031
Decision Tree	0.978 ± 0.005	0.944 ± 0.014
Random Forest	0.983 ± 0.005	0.917 ± 0.038
SVM	0.965 ± 0.005	0.914 ± 0.010
XGBoost	0.986 ± 0.003	0.913 ± 0.038
<i>RHNL</i> _{25G75L}	0.913 ± 0.007	0.856 ± 0.055
<i>RHNL</i> _{25G75L} + SVM	0.958 ± 0.016	0.915 ± 0.024
<i>RHNL</i> _{50G50L}	0.909 ± 0.012	0.873 ± 0.045
<i>RHNL</i> _{50G50L} + SVM	0.955 ± 0.010	0.921 ± 0.025
<i>RHNL</i> _{75G25L}	0.913 ± 0.011	0.874 ± 0.058
<i>RHNL</i> _{75G25L} + SVM	0.964 ± 0.017	0.913 ± 0.027

TABLE 27. AFF-HSTR seed-wise test macro-F1 under random split evaluation.

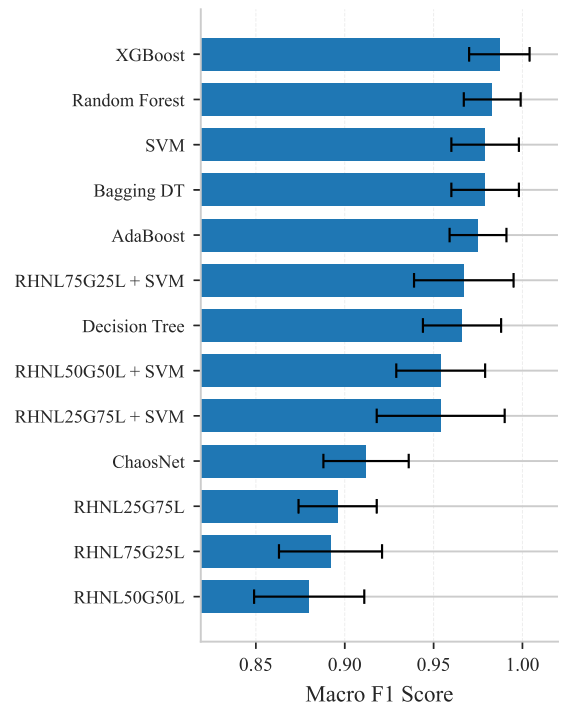
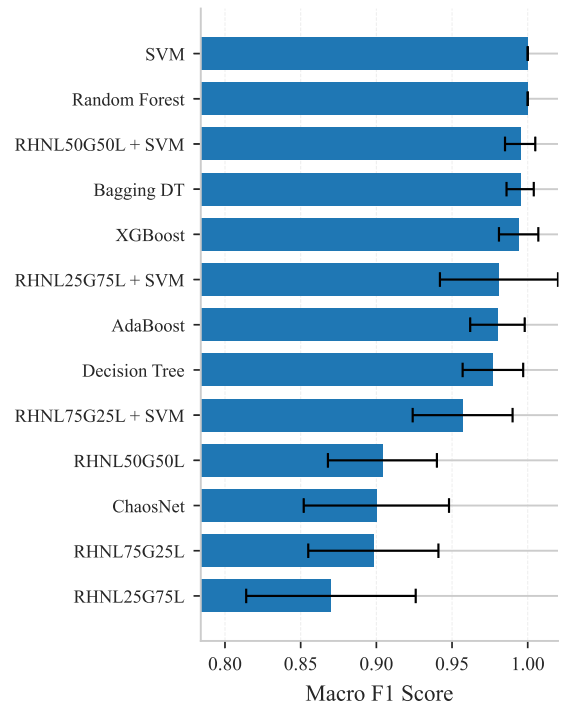
Model	42	52	62	72	82
AdaBoost	0.958	0.959	0.979	0.979	1.000
Bagging DT	0.958	1.000	0.958	0.979	1.000
ChaosNet	0.897	0.896	0.917	0.958	0.894
Decision Tree	0.936	0.959	0.958	0.979	1.000
Random Forest	0.979	1.000	0.958	0.979	1.000
SVM	0.958	0.958	1.000	0.979	1.000
XGBoost	1.000	1.000	0.958	0.979	1.000
<i>RHNL</i> _{25G75L}	0.897	0.918	0.895	0.857	0.914
<i>RHNL</i> _{25G75L} + SVM	0.895	0.938	0.979	0.958	1.000
<i>RHNL</i> _{50G50L}	0.837	0.876	0.914	0.857	0.914
<i>RHNL</i> _{50G50L} + SVM	0.916	0.938	0.979	0.958	0.979
<i>RHNL</i> _{75G25L}	0.873	0.876	0.937	0.857	0.914
<i>RHNL</i> _{75G25L} + SVM	0.938	0.958	1.000	0.937	1.000

TABLE 28. AFF-HSTR seed-wise test macro-F1 under time-aware evaluation.

Model	42	52	62	72	82
AdaBoost	0.968	0.954	1.000	0.979	1.000
Bagging DT	1.000	0.977	1.000	1.000	1.000
ChaosNet	0.968	0.885	0.942	0.868	0.838
Decision Tree	1.000	0.954	0.971	0.958	1.000
Random Forest	1.000	1.000	1.000	1.000	1.000
SVM	1.000	1.000	1.000	1.000	1.000
XGBoost	0.968	1.000	1.000	1.000	1.000
<i>RHNL</i> _{25G75L}	0.832	0.833	0.885	0.828	0.974
<i>RHNL</i> _{25G75L} + SVM	0.903	1.000	1.000	1.000	1.000
<i>RHNL</i> _{50G50L}	0.903	0.861	0.971	0.891	0.896
<i>RHNL</i> _{50G50L} + SVM	1.000	1.000	1.000	1.000	0.974
<i>RHNL</i> _{75G25L}	0.903	0.883	0.885	0.844	0.974
<i>RHNL</i> _{75G25L} + SVM	0.903	0.955	1.000	0.979	0.949

Bagging DT obtaining the strongest random split result at the 50% fraction by reaching 0.545 ± 0.02 . Under the spatially grouped evaluation, *RHNL*_{50G50L} + SVM performs the best at 1% training fraction with 0.471 ± 0.01 , whereas AdaBoost becomes strongest at the 50% fraction with 0.504 ± 0.02 .

Overall, PFF provides the strongest evidence that leakage-resistant evaluation materially changes the interpretation of model performance. Unlike AFF, where several models approach ceiling-level performance, and unlike CFF,

**FIGURE 12.** AFF-HSTR mean and standard deviation of macro-F1 under random split evaluation.**FIGURE 13.** AFF-HSTR mean and standard deviation of macro-F1 under time-aware evaluation.

where random evaluation is the only defensible protocol due to missing spatial and temporal identifiers, PFF directly

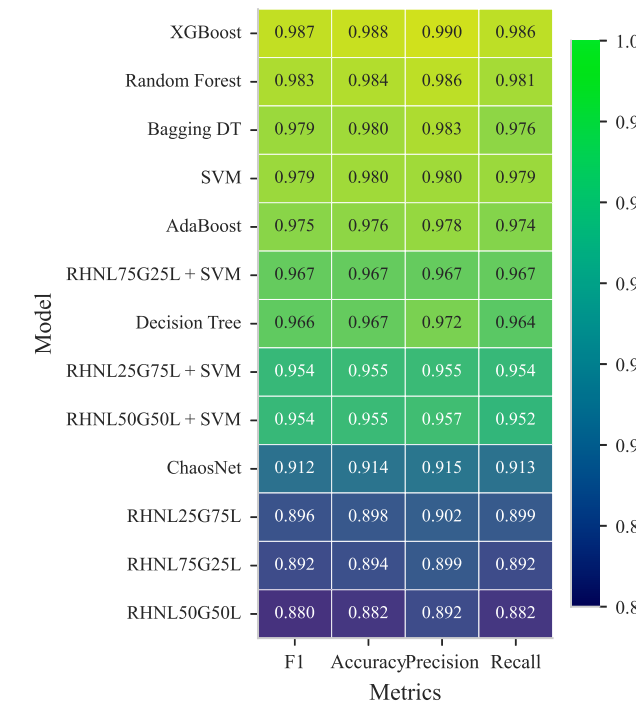


FIGURE 14. AFF-HSTR metric heatmap under random split evaluation.

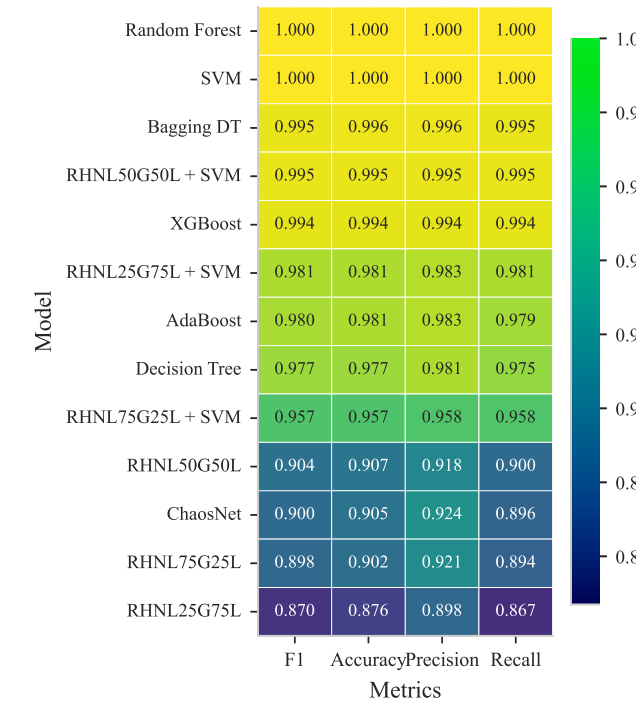


FIGURE 15. AFF-HSTR metric heatmap under time-aware evaluation.

demonstrates the difficulty of geographic transfer. The results therefore support a more conservative conclusion: Neurochaos-based models are competitive in selected

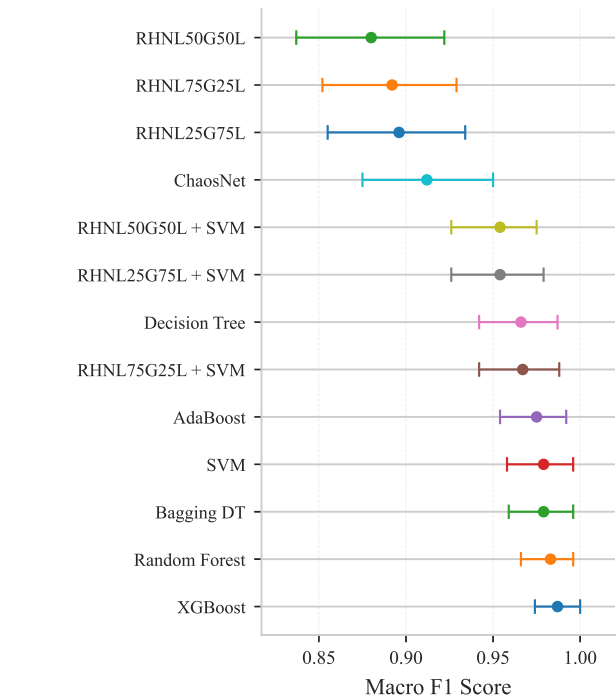


FIGURE 16. AFF-HSTR bootstrap confidence intervals under random split evaluation.

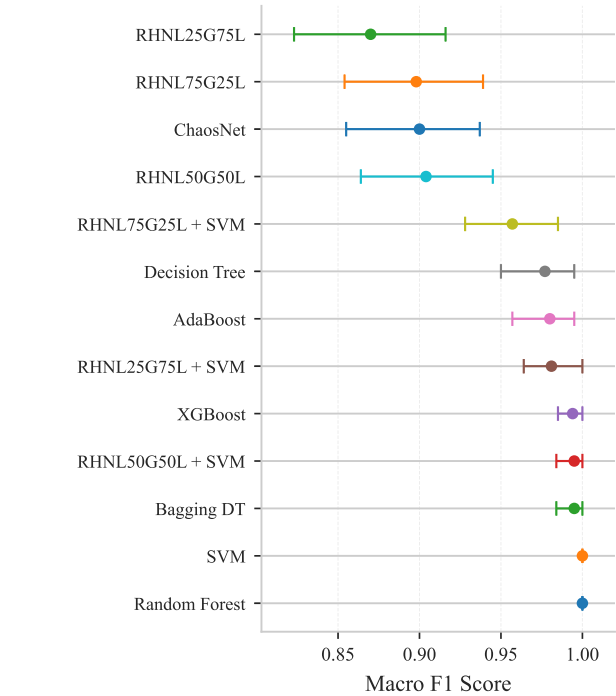
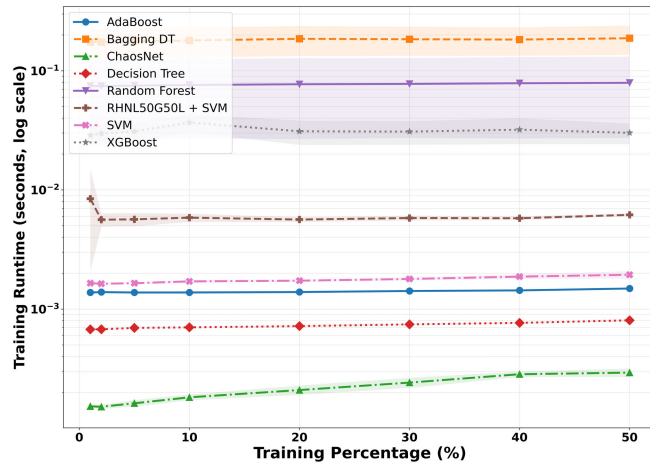
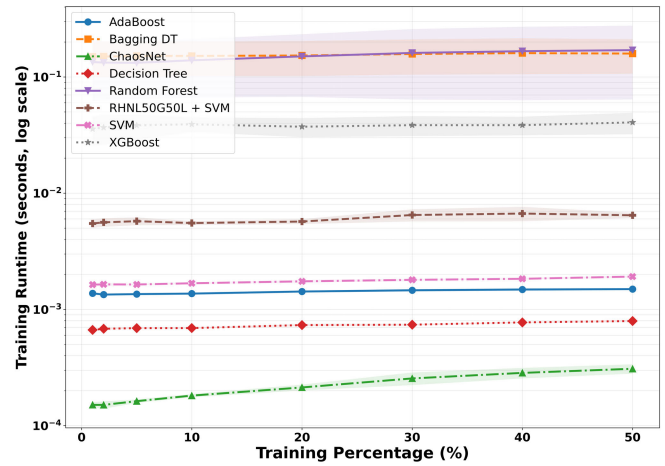


FIGURE 17. AFF-HSTR bootstrap confidence intervals under time-aware evaluation.

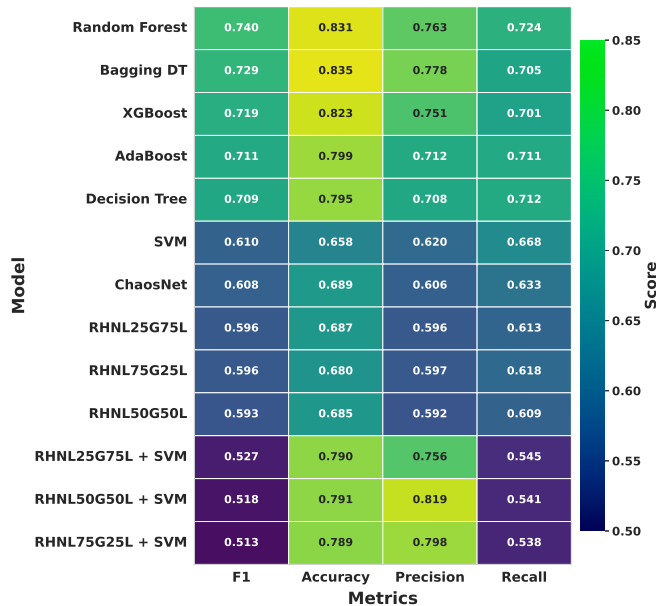
low-sample settings, but the strongest HSTR performance on PFF is obtained by conventional ensemble methods, and



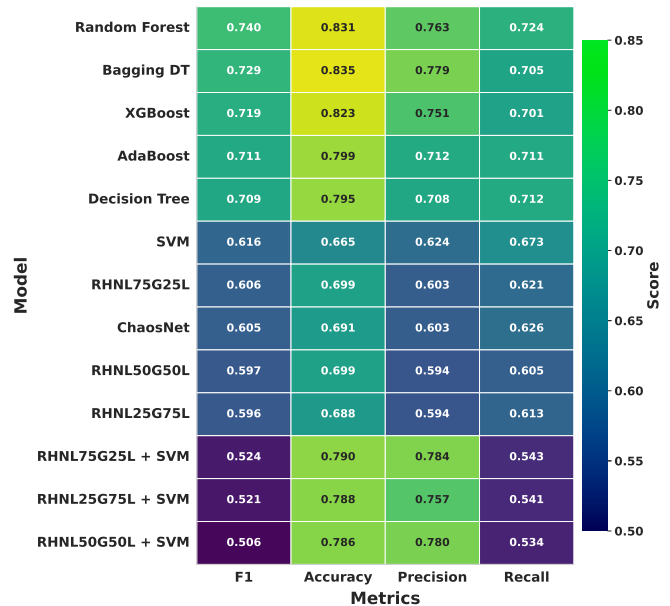
(a) Random split



(b) Time-aware split

FIGURE 18. AFF-LSTR runtime training plots.

(a) Normal feature setting



(b) Log-transformed feature setting

FIGURE 19. CFF-HSTR metric heatmaps.

all model families remain limited under spatially grouped generalization.

D. STATISTICAL SIGNIFICANCE ANALYSIS

The HSTR Friedman results are summarized in Table 23. Significant differences among classifier ranks were observed for AFF under random-split evaluation and for CFF under both the normal and log-transformed feature settings ($p_{\text{perm}} = 1/100001$ in all three cases), with Kendall's W ranging from 0.748 to 0.905. In contrast, the omnibus null hypothesis was not rejected for PFF under either random-split ($p_{\text{perm}} = 0.0587$) or spatially grouped ($p_{\text{perm}} = 0.0884$) evaluation.

Because the Friedman test is an omnibus procedure, these findings establish whether systematic differences exist among classifier ranks but do not identify significant differences between individual classifier pairs. Accordingly, no claim of pairwise statistical superiority is made.

VII. CONCLUSION AND FUTURE WORK

This work examined whether Neurochaos-based learning can provide a practical alternative to conventional machine-learning classifiers for forest-fire classification under realistic constraints of small sample size, heterogeneous tabular features, and protocol-dependent

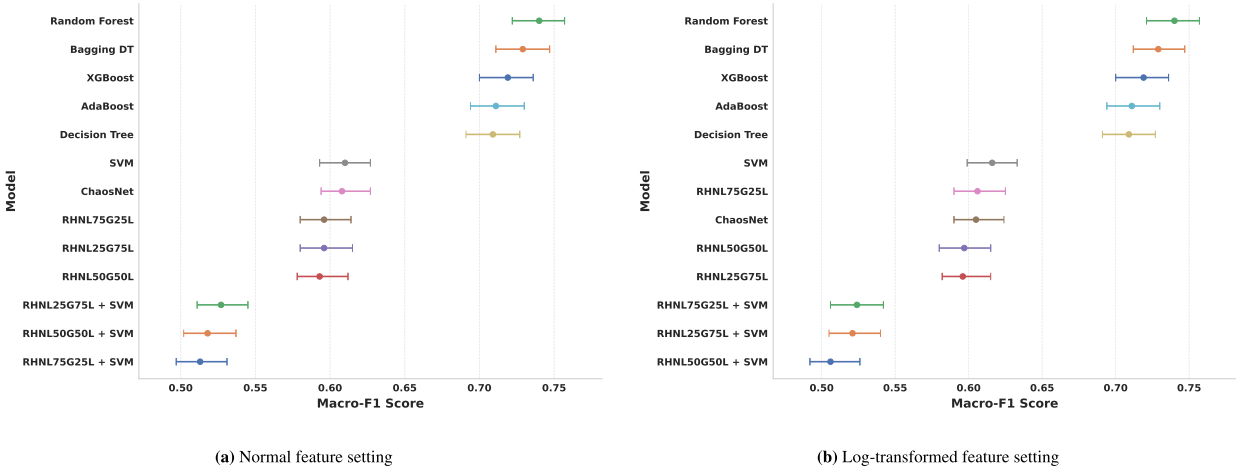


FIGURE 20. CFF-HSTR bootstrap confidence intervals for macro-F1.

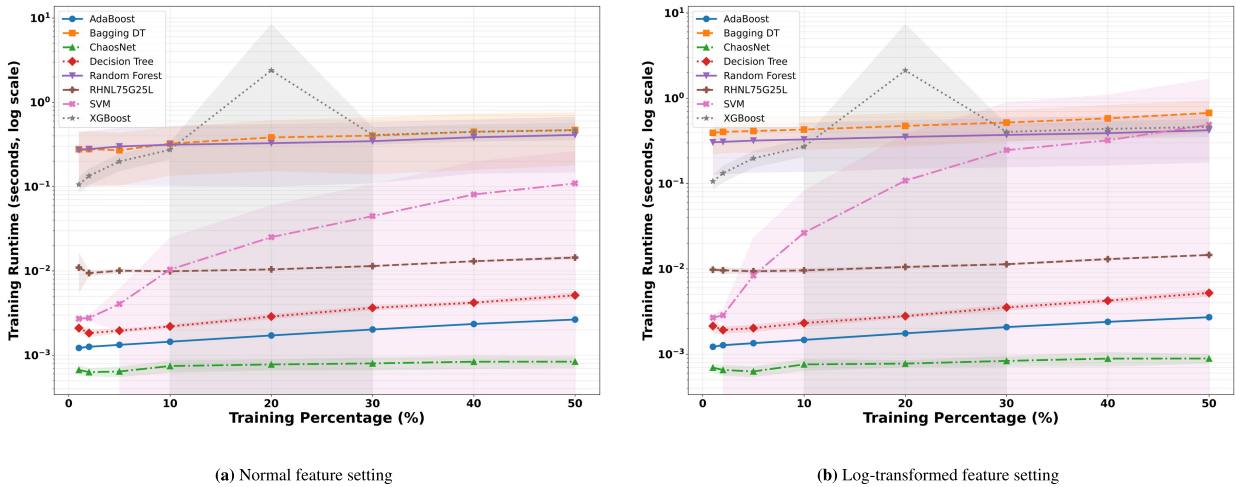


FIGURE 21. CFF-LSTR runtime training plots.

TABLE 29. AFF-LSTR representative training and inference runtimes.

Model	Split	Train 1%	Infer 1%	Train 50%	Infer 50%
AdaBoost	Random	1.379 ms	139.3 μ s	1.489 ms	138.1 μ s
	Time-aware	1.376 ms	141.3 μ s	1.497 ms	138.4 μ s
Bagging DT	Random	173.9 ms	6.855 ms	187.6 ms	6.918 ms
	Time-aware	149.3 ms	5.886 ms	158.9 ms	5.869 ms
ChaosNet	Random	152.5 μ s	631.9 μ s	292.8 μ s	560.7 μ s
	Time-aware	150.8 μ s	6.985 ms	308.7 μ s	568.3 μ s
Decision Tree	Random	675.8 μ s	84.5 μ s	805.1 μ s	90.9 μ s
	Time-aware	665.9 μ s	82.2 μ s	795.2 μ s	88.1 μ s
Random Forest	Random	74.5 ms	2.205 ms	79.2 ms	2.262 ms
	Time-aware	134.5 ms	4.168 ms	170.2 ms	5.382 ms
RHN _{L50G50L} + SVM	Random	8.425 ms	5.161 ms	6.169 ms	4.971 ms
	Time-aware	5.477 ms	18.01 ms	6.447 ms	4.990 ms
SVM	Random	1.651 ms	257.8 μ s	1.944 ms	285.4 μ s
	Time-aware	1.635 ms	262.7 μ s	1.919 ms	277.0 μ s
XGBoost	Random	28.6 ms	1.007 ms	30.1 ms	999.4 μ s
	Time-aware	35.6 ms	1.022 ms	40.6 ms	1.060 ms

generalization. Rather than limiting to a single benchmark problem for forest-fire prediction, the study evaluated

three structurally different datasets with dataset-appropriate validation settings, i.e., random and chronological evaluation for AFF, repeated stratified evaluation for CFF, and random as well as spatially grouped evaluation for PFF. This design permits the comparison to differentiate between four issues often confounded in wildfire-classification studies: predictive accuracy, data efficiency, computational cost, and robustness to temporal or spatial dependence.

The results show that the model superiority is highly regime dependent. In the high sample setting, traditional classifiers still remain highly competitive and in many cases superior. AFF is almost separable under HSTR, with Random Forest and SVM achieving ceiling-level time-aware performance. Bagging DT and RHN_{L50G50L} + SVM are close to the best performing models. CFF exhibits a different behavior: For both original and log-transformed feature settings, Random Forest shows the best high-sample performance, suggesting the dominant decision structure is more sufficiently captured by ensemble trees than the

TABLE 30. CFF-HSTR seed-wise tuned hyperparameters under the normal feature setting for seeds 42–82.

Model	Params	42	52	62	72	82
AdaBoost	(n_{est}, lr)	(100, 1.0)	(50, 1.0)	(150, 0.01)	(100, 0.5)	(200, 0.5)
Bagging DT	(mf, ms, n_{est})	(1.0, 1.0, 100)	(1.0, 1.0, 200)	(1.0, 1.0, 100)	(1.0, 1.0, 200)	(1.0, 0.8, 100)
ChaosNet	(q, b, ϵ)	(0.215, 0.125, 0.499)	(0.249, 0.146, 0.456)	(0.249, 0.469, 0.400)	(0.416, 0.124, 0.202)	(0.142, 0.461, 0.472)
Decision Tree	(d, msl, mss)	(30, 1, 2)	(20, 1, 2)	(40, 1, 2)	(50, 1, 2)	(20, 1, 2)
Random Forest	(d, msl, mss, n_{est})	(50, 2, 2, 50)	(20, 2, 2, 50)	(20, 2, 5, 200)	(20, 2, 5, 100)	(20, 1, 5, 250)
SVM	(C, deg, γ, ker)	(100, NA, 1, <i>rbf</i>)	(0.1, NA, 0.01, <i>sig</i>)	(1, NA, 1, <i>rbf</i>)	(1, NA, 0.1, <i>rbf</i>)	(0.1, NA, 0.1, <i>rbf</i>)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.8, 0.2, 5, 200, 0.1, 2, 0.6)	(1.0, 0.2, 7, 200, 0.1, 1.5, 0.6)	(0.8, 0.2, 10, 100, 0, 1, 0.8)	(1.0, 0.2, 7, 150, 0, 1, 0.6)	(1.0, 0.2, 7, 150, 1, 1, 1.0)
RHNL25G75L	(q, b, ϵ)	(0.452, 0.386, 0.259)	(0.424, 0.033, 0.274)	(0.346, 0.250, 0.295)	(0.428, 0.018, 0.200)	(0.408, 0.171, 0.298)
RHNL25G75L + SVM	(q, b, ϵ)	(0.044, 0.096, 0.289)	(0.293, 0.257, 0.237)	(0.099, 0.361, 0.237)	(0.172, 0.084, 0.179)	(0.077, 0.281, 0.214)
RHNL50G50L	(q, b, ϵ)	(0.460, 0.272, 0.247)	(0.411, 0.271, 0.261)	(0.407, 0.274, 0.299)	(0.429, 0.459, 0.279)	(0.400, 0.033, 0.225)
RHNL50G50L + SVM	(q, b, ϵ)	(0.126, 0.139, 0.230)	(0.170, 0.343, 0.292)	(0.186, 0.177, 0.149)	(0.144, 0.128, 0.150)	(0.206, 0.497, 0.256)
RHNL75G25L	(q, b, ϵ)	(0.388, 0.187, 0.269)	(0.379, 0.237, 0.294)	(0.347, 0.360, 0.298)	(0.419, 0.403, 0.291)	(0.408, 0.033, 0.288)
RHNL75G25L + SVM	(q, b, ϵ)	(0.146, 0.361, 0.193)	(0.171, 0.308, 0.295)	(0.174, 0.345, 0.161)	(0.217, 0.269, 0.126)	(0.243, 0.210, 0.209)

TABLE 31. CFF-HSTR seed-wise tuned hyperparameters under the normal feature setting for seeds 92–132.

Model	Params	92	102	112	122	132
AdaBoost	(n_{est}, lr)	(50, 0.1)	(200, 0.1)	(150, 0.1)	(100, 0.1)	(150, 0.5)
Bagging DT	(mf, ms, n_{est})	(1.0, 1.0, 200)	(1.0, 1.0, 200)	(1.0, 0.8, 100)	(1.0, 1.0, 150)	(1.0, 0.8, 50)
ChaosNet	(q, b, ϵ)	(0.259, 0.400, 0.440)	(0.165, 0.476, 0.471)	(0.142, 0.461, 0.498)	(0.053, 0.328, 0.153)	(0.423, 0.097, 0.194)
Decision Tree	(d, msl, mss)	(40, 2, 2)	(40, 1, 2)	(30, 1, 2)	(50, 1, 2)	(40, 1, 2)
Random Forest	(d, msl, mss, n_{est})	(50, 1, 5, 50)	(20, 2, 2, 150)	(30, 2, 2, 250)	(20, 1, 5, 250)	(20, 1, 5, 50)
SVM	(C, deg, γ, ker)	(10, 3, 1, <i>poly</i>)	(0.1, NA, 0.1, <i>rbf</i>)	(10, NA, 1, <i>rbf</i>)	(10, 3, 1, <i>poly</i>)	(1, NA, 1, <i>rbf</i>)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(1.0, 0.2, 10, 200, 0, 2, 0.6)	(1.0, 0.1, 10, 150, 0.1, 1.5, 0.8)	(1.0, 0.2, 10, 150, 0.1, 2, 0.6)	(0.8, 0.2, 10, 200, 0.1, 1, 0.8)	(0.8, 0.2, 10, 100, 0, 1, 0.6)
RHNL25G75L	(q, b, ϵ)	(0.419, 0.483, 0.282)	(0.401, 0.130, 0.269)	(0.456, 0.492, 0.243)	(0.408, 0.149, 0.224)	(0.436, 0.237, 0.277)
RHNL25G75L + SVM	(q, b, ϵ)	(0.392, 0.337, 0.056)	(0.099, 0.187, 0.245)	(0.078, 0.135, 0.262)	(0.223, 0.154, 0.107)	(0.221, 0.077, 0.096)
RHNL50G50L	(q, b, ϵ)	(0.401, 0.264, 0.279)	(0.380, 0.134, 0.282)	(0.435, 0.144, 0.237)	(0.408, 0.149, 0.224)	(0.425, 0.210, 0.285)
RHNL50G50L + SVM	(q, b, ϵ)	(0.459, 0.280, 0.125)	(0.129, 0.162, 0.211)	(0.147, 0.437, 0.192)	(0.223, 0.154, 0.107)	(0.263, 0.137, 0.135)
RHNL75G25L	(q, b, ϵ)	(0.416, 0.314, 0.284)	(0.397, 0.020, 0.251)	(0.433, 0.334, 0.237)	(0.372, 0.224, 0.258)	(0.436, 0.237, 0.277)
RHNL75G25L + SVM	(q, b, ϵ)	(0.186, 0.354, 0.277)	(0.129, 0.162, 0.211)	(0.060, 0.218, 0.280)	(0.136, 0.212, 0.253)	(0.183, 0.090, 0.159)

TABLE 32. CFF-HSTR seed-wise tuned hyperparameters under the log-transformed feature setting for seeds 42–82.

Model	Params	42	52	62	72	82
AdaBoost	(n_{est}, lr)	(100, 1.0)	(50, 1.0)	(150, 0.01)	(100, 0.5)	(200, 0.5)
Bagging DT	(mf, ms, n_{est})	(1.0, 1.0, 100)	(1.0, 1.0, 200)	(1.0, 1.0, 100)	(1.0, 1.0, 200)	(1.0, 1.0, 150)
ChaosNet	(q, b, ϵ)	(0.257, 0.431, 0.457)	(0.220, 0.381, 0.476)	(0.224, 0.024, 0.442)	(0.183, 0.093, 0.444)	(0.435, 0.141, 0.260)
Decision Tree	(d, msl, mss)	(30, 1, 2)	(20, 1, 2)	(40, 1, 2)	(40, 1, 2)	(20, 1, 2)
Random Forest	(d, msl, mss, n_{est})	(50, 2, 2, 50)	(20, 2, 2, 50)	(40, 2, 5, 200)	(50, 1, 10, 250)	(30, 2, 5, 250)
SVM	(C, deg, γ, ker)	(100, NA, 1, <i>rbf</i>)	(0.1, NA, 0.01, <i>sig</i>)	(1, NA, 1, <i>rbf</i>)	(1, NA, 0.1, <i>rbf</i>)	(0.1, NA, 0.1, <i>rbf</i>)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.8, 0.2, 5, 200, 0.1, 2, 0.6)	(1.0, 0.2, 7, 200, 0.1, 1.5, 0.6)	(0.8, 0.2, 10, 100, 0, 1, 0.8)	(1.0, 0.2, 7, 150, 0, 1, 0.6)	(1.0, 0.2, 7, 150, 1, 1, 1.0)
RHNL25G75L	(q, b, ϵ)	(0.436, 0.093, 0.275)	(0.433, 0.342, 0.265)	(0.407, 0.164, 0.297)	(0.374, 0.429, 0.275)	(0.400, 0.033, 0.225)
RHNL25G75L + SVM	(q, b, ϵ)	(0.285, 0.270, 0.104)	(0.191, 0.431, 0.120)	(0.182, 0.286, 0.265)	(0.283, 0.469, 0.161)	(0.389, 0.180, 0.174)
RHNL50G50L	(q, b, ϵ)	(0.414, 0.196, 0.298)	(0.405, 0.285, 0.282)	(0.407, 0.164, 0.297)	(0.430, 0.465, 0.285)	(0.412, 0.081, 0.250)
RHNL50G50L + SVM	(q, b, ϵ)	(0.475, 0.459, 0.181)	(0.283, 0.482, 0.291)	(0.298, 0.198, 0.095)	(0.185, 0.318, 0.261)	(0.314, 0.436, 0.255)
RHNL75G25L	(q, b, ϵ)	(0.414, 0.196, 0.298)	(0.388, 0.204, 0.286)	(0.346, 0.136, 0.298)	(0.408, 0.387, 0.299)	(0.365, 0.079, 0.299)
RHNL75G25L + SVM	(q, b, ϵ)	(0.238, 0.497, 0.228)	(0.289, 0.278, 0.265)	(0.183, 0.409, 0.263)	(0.214, 0.098, 0.250)	(0.404, 0.201, 0.172)

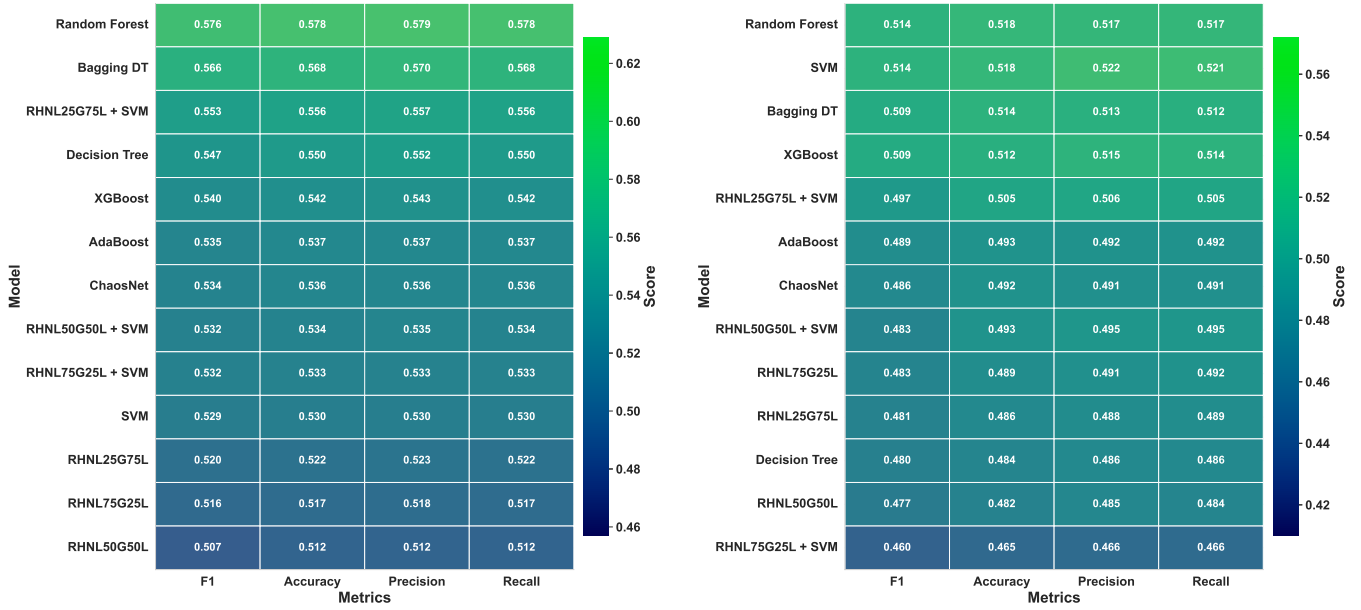
TABLE 33. CFF-HSTR seed-wise tuned hyperparameters under the log-transformed feature setting for seeds 92–132.

Model	Params	92	102	112	122	132
AdaBoost	(n_{est}, lr)	(50, 0.1)	(200, 0.1)	(150, 0.1)	(100, 0.1)	(150, 0.5)
Bagging DT	(mf, ms, n_{est})	(1.0, 1.0, 200)	(1.0, 0.8, 100)	(1.0, 0.8, 100)	(1.0, 1.0, 150)	(1.0, 0.8, 50)
ChaosNet	(q, b, ϵ)	(0.259, 0.400, 0.440)	(0.171, 0.473, 0.420)	(0.075, 0.465, 0.496)	(0.356, 0.336, 0.322)	(0.118, 0.420, 0.492)
Decision Tree	(d, msl, mss)	(40, 2, 2)	(40, 1, 2)	(30, 1, 2)	(50, 1, 2)	(40, 1, 2)
Random Forest	(d, msl, mss, n_{est})	(50, 1, 5, 50)	(20, 2, 2, 150)	(20, 2, 2, 250)	(20, 1, 5, 250)	(20, 1, 5, 50)
SVM	(C, deg, γ, ker)	(10, 3, 1, <i>poly</i>)	(0.1, NA, 0.1, <i>rbf</i>)	(10, NA, 1, <i>rbf</i>)	(10, 3, 1, <i>poly</i>)	(1, NA, 1, <i>rbf</i>)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(1.0, 0.2, 10, 200, 0, 2, 0.6)	(1.0, 0.1, 10, 150, 0.1, 1.5, 0.8)	(1.0, 0.2, 10, 150, 0.1, 2, 0.6)	(0.8, 0.2, 10, 200, 0.1, 1, 0.8)	(0.8, 0.2, 10, 100, 0, 1, 0.6)
RHNL25G75L	(q, b, ϵ)	(0.375, 0.285, 0.297)	(0.400, 0.488, 0.295)	(0.437, 0.208, 0.237)	(0.414, 0.467, 0.266)	(0.457, 0.013, 0.258)
RHNL25G75L + SVM	(q, b, ϵ)	(0.210, 0.182, 0.238)	(0.129, 0.162, 0.211)	(0.346, 0.323, 0.110)	(0.354, 0.241, 0.151)	(0.215, 0.442, 0.176)
RHNL50G50L	(q, b, ϵ)	(0.458, 0.011, 0.245)	(0.439, 0.135, 0.258)	(0.446, 0.286, 0.235)	(0.423, 0.390, 0.266)	(0.419, 0.143, 0.285)
RHNL50G50L + SVM	(q, b, ϵ)	(0.389, 0.461, 0.112)	(0.099, 0.187, 0.245)	(0.186, 0.303, 0.280)	(0.354, 0.241, 0.151)	(0.299, 0.279, 0.174)
RHNL75G25L	(q, b, ϵ)	(0.415, 0.032, 0.300)	(0.397, 0.020, 0.251)	(0.415, 0.202, 0.298)	(0.405, 0.315, 0.267)	(0.423, 0.375, 0.291)
RHNL75G25L + SVM	(q, b, ϵ)	(0.312, 0.156, 0.132)	(0.175, 0.285, 0.270)	(0.155, 0.266, 0.298)	(0.354, 0.241, 0.151)	(0.298, 0.101, 0.148)

evaluated Neurochaos variants when there are enough labelled data. PFF also supports that high-sample random-split performance does not indicate operational generalization, as all models show performance decay with spatially grouped testing.

The main contribution of Neurochaos Learning therefore seems to be most pronounced in low-sample and

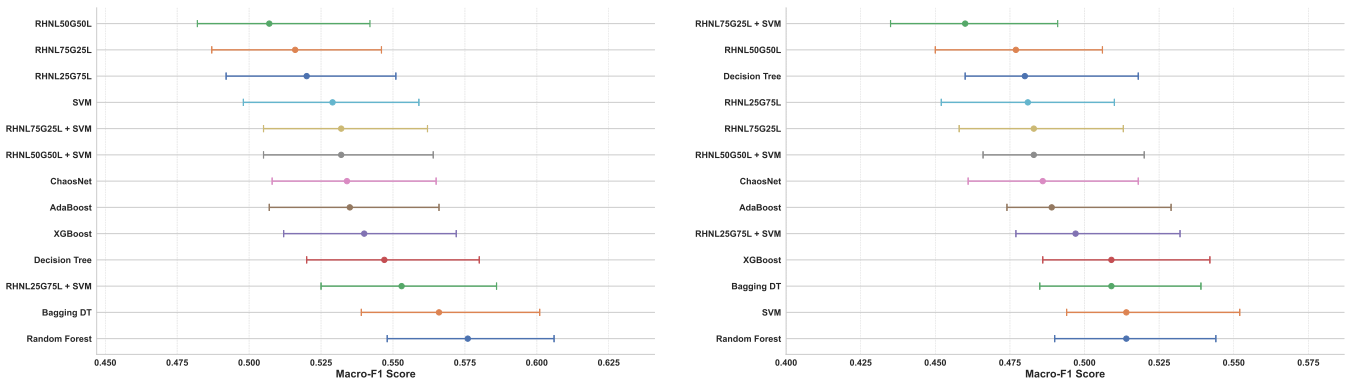
low-cost regimes. ChaosNet is strong when a very small fraction of the training data is available, especially on AFF and CFF, and selected RHNL-SVM configurations remain competitive under chronologically or spatially constrained low-sample evaluation. This behaviour is important since wildfire datasets are often constrained by sparse labelled events, regional heterogeneity and changing environmental



(a) Random-split evaluation

(b) Spatially grouped evaluation

FIGURE 22. PFF-HSTR metric heatmaps under random and spatially grouped evaluation.



(a) Random-split evaluation

(b) Spatially grouped evaluation

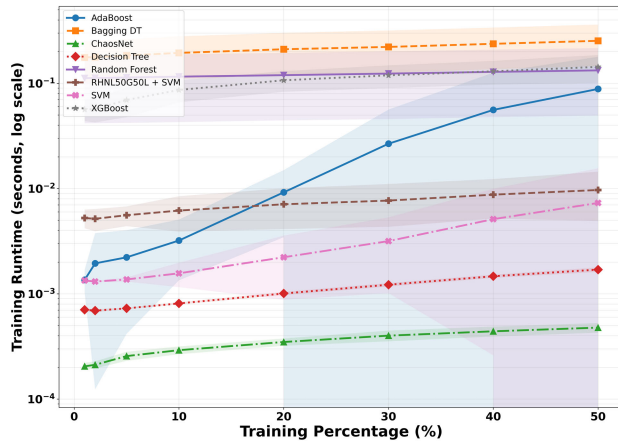
FIGURE 23. PFF-HSTR bootstrap confidence intervals for macro-F1 under random and spatially grouped evaluation.

conditions. The runtime results also support these findings. For the three datasets, ChaosNet trains in sub-millisecond time at 1% training fraction, whereas ensemble methods need a much larger training time. Thus the value of Neurochaos-based models is not that they replace established classifiers across the board, but that they offer a compact and computationally efficient representation mechanism for settings where labelled data and retraining time are at a premium.

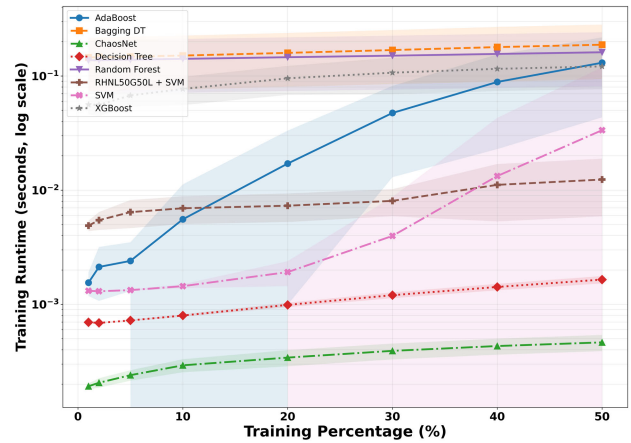
A second important finding from the methodology is that the PFF results show that random-split evaluation can overestimate the geographic generalization potential if spatially related samples are included in both the training and test partitions. The top PFF macro-F1 score under spatially grouped evaluation remains limited and no model

performs well on transfer to unseen spatial cells. This result is important for the paper as it demonstrates that the interpretation of model capability can differ depending on the validation protocol, even when using the same features and classifiers. For wildfire applications, where deployment often involves prediction in new locations or future conditions, leakage-resistant evaluation should be considered a core requirement, not an optional check on robustness.

The evidence as a whole suggests a balanced conclusion. When there is enough training data and the tabular decision structure can be learned, conventional ensemble methods are the most reliable choice. Particularly, neurochaos-based models such as ChaosNet and some RHNL-SVM hybrids are the most appealing in the case of data-scarce,



(a) Random Split setting



(b) Spatial-aware split setting

FIGURE 24. PFF-LSTR runtime training plots.

TABLE 34. CFF-HSTR training cross-validation macro-F1 during hyperparameter tuning.

Model	Normal	Log-transformed
AdaBoost	0.667 ± 0.018	0.667 ± 0.018
Bagging DT	0.697 ± 0.013	0.698 ± 0.013
ChaosNet	0.612 ± 0.013	0.615 ± 0.011
Decision Tree	0.671 ± 0.015	0.670 ± 0.015
Random Forest	0.719 ± 0.012	0.718 ± 0.012
SVM	0.645 ± 0.101	0.613 ± 0.008
XGBoost	0.698 ± 0.016	0.698 ± 0.016
<i>RHNL</i> _{25G75L}	0.607 ± 0.007	0.605 ± 0.006
<i>RHNL</i> _{25G75L} + SVM	0.532 ± 0.014	0.531 ± 0.014
<i>RHNL</i> _{50G50L}	0.606 ± 0.004	0.605 ± 0.006
<i>RHNL</i> _{50G50L} + SVM	0.526 ± 0.012	0.528 ± 0.016
<i>RHNL</i> _{75G25L}	0.605 ± 0.005	0.605 ± 0.006
<i>RHNL</i> _{75G25L} + SVM	0.529 ± 0.011	0.528 ± 0.016

computationally-constrained or fast retrainable problems. The contribution of this study is therefore twofold: (i) an empirical assessment of Neurochaos Learning for forest-fire classification across datasets, and (ii) a clarification of the operating conditions under which chaos-based representations are likely to be useful.

These findings can be built upon in future work, expanding Neurochaos-based wildfire classification toward broader evaluation oriented for deployment. The logical next step is to evaluate ChaosNet and RHNL-SVM hybrids on larger multi-region wildfire datasets where temporal ordering, spatial coordinates, vegetation structure, topography, meteorology, and satellite-derived fire indicators are jointly available. Such datasets would allow explicit evaluation under temporal forecasting, geographic transfer and spatiotemporal distribution-shift settings. Future work will investigate the spatial heterogeneity and regional covariate shift that drives the observed decline in geographic

transfer, evaluate mitigation via domain adaptation and spatially informed representations, incorporate appropriately regularized Deep Learning baselines, and study the effect of meteorological inputs on ChaosFEX descriptors without interpreting chaotic-map dynamics as physical models of meteorological processes. Another promising direction is to combine Neurochaos feature transformations with calibrated decision thresholds and uncertainty-aware warning levels so that the models can support operational fire-risk screening, not just binary classification. Finally, the current results show strong low-sample and low-training-cost behavior and future work should investigate the memory footprint, energy consumption, and deployment of Neurochaos-based models on edge devices for fast retraining in resource-constrained wildfire-monitoring systems.

APPENDIX A

ADDITIONAL RESULTS FOR THE ALGERIAN FOREST FIRE DATASET

A. SEED-WISE HYPERPARAMETER CONFIGURATIONS

Tables 24 and 25 report the seed-wise hyperparameters selected for AFF-HSTR under the random and time-aware evaluation protocols, respectively. The values in each seed column follow the parameter order specified in the “Params” column.

Here, lr denotes learning rate, mf denotes max features, ms denotes max samples, d denotes maximum depth, msl denotes minimum samples per leaf, mss denotes minimum samples split, cb denotes colsample by tree, α denotes L1 regularization, λ denotes L2 regularization, and ss denotes subsample.

B. EXPANDED HSTR RESULTS

This subsection reports supplementary AFF-HSTR results that were not included in the main manuscript to avoid overloading the Results section. The additional results

TABLE 35. CFF-HSTR seed-wise test macro-F1 under the normal feature setting.

Model	42	52	62	72	82	92	102	112	122	132
AdaBoost	0.685	0.687	0.701	0.687	0.720	0.769	0.716	0.724	0.717	0.707
Bagging DT	0.725	0.750	0.757	0.721	0.746	0.759	0.708	0.696	0.718	0.711
ChaosNet	0.572	0.597	0.647	0.587	0.638	0.600	0.600	0.664	0.547	0.628
Decision Tree	0.682	0.688	0.716	0.690	0.718	0.725	0.733	0.718	0.717	0.704
Random Forest	0.707	0.737	0.762	0.719	0.787	0.762	0.716	0.728	0.731	0.749
SVM	0.615	0.600	0.631	0.606	0.580	0.617	0.578	0.634	0.624	0.616
XGBoost	0.685	0.722	0.740	0.712	0.746	0.743	0.734	0.722	0.689	0.700
<i>RHNL</i> _{25G75L}	0.570	0.582	0.640	0.574	0.644	0.573	0.566	0.598	0.602	0.605
<i>RHNL</i> _{25G75L} + SVM	0.565	0.571	0.520	0.521	0.540	0.510	0.576	0.486	0.498	0.482
<i>RHNL</i> _{50G50L}	0.578	0.570	0.652	0.604	0.588	0.575	0.579	0.574	0.607	0.607
<i>RHNL</i> _{50G50L} + SVM	0.523	0.537	0.523	0.500	0.565	0.500	0.561	0.521	0.498	0.450
<i>RHNL</i> _{75G25L}	0.609	0.585	0.645	0.612	0.637	0.573	0.556	0.569	0.569	0.605
<i>RHNL</i> _{75G25L} + SVM	0.544	0.535	0.523	0.513	0.520	0.489	0.552	0.474	0.496	0.488

TABLE 36. CFF-HSTR seed-wise test macro-F1 under the log-transformed feature setting.

Model	42	52	62	72	82	92	102	112	122	132
AdaBoost	0.685	0.687	0.701	0.687	0.720	0.769	0.716	0.724	0.717	0.707
Bagging DT	0.725	0.750	0.757	0.721	0.749	0.759	0.704	0.696	0.718	0.711
ChaosNet	0.572	0.586	0.658	0.611	0.624	0.600	0.563	0.631	0.577	0.628
Decision Tree	0.682	0.688	0.716	0.690	0.718	0.725	0.733	0.718	0.717	0.704
Random Forest	0.707	0.737	0.765	0.728	0.780	0.762	0.716	0.722	0.731	0.749
SVM	0.619	0.609	0.648	0.623	0.585	0.611	0.581	0.634	0.629	0.616
XGBoost	0.685	0.722	0.740	0.712	0.746	0.743	0.734	0.722	0.689	0.700
<i>RHNL</i> _{25G75L}	0.574	0.580	0.652	0.609	0.591	0.612	0.590	0.584	0.577	0.596
<i>RHNL</i> _{25G75L} + SVM	0.529	0.535	0.535	0.523	0.555	0.535	0.534	0.484	0.474	0.510
<i>RHNL</i> _{50G50L}	0.568	0.576	0.652	0.618	0.620	0.583	0.576	0.575	0.590	0.606
<i>RHNL</i> _{50G50L} + SVM	0.516	0.546	0.510	0.523	0.488	0.488	0.569	0.475	0.486	0.464
<i>RHNL</i> _{75G25L}	0.568	0.585	0.647	0.610	0.636	0.610	0.579	0.627	0.587	0.607
<i>RHNL</i> _{75G25L} + SVM	0.540	0.537	0.523	0.513	0.531	0.563	0.540	0.474	0.486	0.531

TABLE 37. PFF-HSTR seed-wise tuned hyperparameters under random-split evaluation for seeds 42–82.

Model	Params	42	52	62	72	82
AdaBoost	(<i>n_{est}</i> , <i>lr</i>)	(150, 0.1)	(50, 0.01)	(50, 1.0)	(100, 0.5)	(50, 1.0)
Bagging DT	(<i>mf</i> , <i>ms</i> , <i>n_{est}</i>)	(0.6, 0.8, 50)	(0.6, 1.0, 200)	(0.8, 0.6, 200)	(1.0, 1.0, 100)	(0.8, 0.8, 50)
ChaosNet	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.002, 0.213, 0.055)	(0.366, 0.281, 0.991)	(0.033, 0.496, 0.206)	(0.460, 0.114, 0.345)	(0.320, 0.312, 0.275)
Decision Tree	(<i>d</i> , <i>msl</i> , <i>mss</i>)	(10, 2, 10)	(40, 4, 10)	(50, 2, 10)	(10, 1, 2)	(10, 2, 2)
Random Forest	(<i>d</i> , <i>msl</i> , <i>mss</i> , <i>n_{est}</i>)	(50, 4, 2, 50)	(10, 4, 2, 50)	(50, 2, 2, 50)	(30, 2, 10, 150)	(20, 4, 2, 50)
SVM	(<i>C</i> , <i>deg</i> , <i>γ</i> , <i>ker</i>)	(10, NA, 0.1, rbf)	(10, 4, 1, poly)	(100, 2, 0.1, poly)	(1, 2, 0.1, poly)	(10, NA, NA, linear)
XGBoost	(<i>cb</i> , <i>lr</i> , <i>d</i> , <i>n_{est}</i> , <i>α</i> , <i>λ</i> , <i>ss</i>)	(0.6, 0.2, 10, 100, 0.1, 0.8)	(0.6, 0.01, 3, 150, 0.1, 2, 1.0)	(0.6, 0.2, 7, 200, 0.1, 1.5, 0.6)	(0.6, 0.1, 3, 200, 0.1, 2, 0.6)	(0.6, 0.01, 3, 150, 1, 2, 1.0)
<i>RHNL</i> _{25G75L}	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.031, 0.280, 0.485)	(0.271, 0.242, 0.407)	(0.074, 0.162, 0.064)	(0.013, 0.209, 0.460)	(0.336, 0.096, 0.373)
<i>RHNL</i> _{25G75L} + SVM	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.304, 0.231, 0.370)	(0.428, 0.161, 0.300)	(0.010, 0.197, 0.341)	(0.154, 0.092, 0.288)	(0.431, 0.276, 0.088)
<i>RHNL</i> _{50G50L}	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.200, 0.282, 0.364)	(0.215, 0.233, 0.385)	(0.055, 0.243, 0.136)	(0.181, 0.233, 0.384)	(0.052, 0.103, 0.212)
<i>RHNL</i> _{50G50L} + SVM	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.045, 0.288, 0.285)	(0.193, 0.295, 0.465)	(0.283, 0.159, 0.327)	(0.419, 0.216, 0.269)	(0.127, 0.046, 0.273)
<i>RHNL</i> _{75G25L}	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.030, 0.242, 0.375)	(0.215, 0.233, 0.385)	(0.386, 0.020, 0.181)	(0.083, 0.244, 0.380)	(0.012, 0.048, 0.140)
<i>RHNL</i> _{75G25L} + SVM	(<i>b</i> , <i>ϵ</i> , <i>q</i>)	(0.488, 0.219, 0.199)	(0.223, 0.300, 0.483)	(0.347, 0.240, 0.374)	(0.035, 0.155, 0.391)	(0.201, 0.202, 0.163)

include training cross-validation performance in Table 26 obtained during hyperparameter optimization, seed-wise test-set macro-F1 values in Table 27 and supplementary metric-level plots in Figures 12–17. These results provide

TABLE 38. PFF-HSTR seed-wise tuned hyperparameters under random-split evaluation for seeds 92–132.

Model	Params	92	102	112	122	132
AdaBoost	(n_{est}, lr)	(100, 0.1)	(50, 0.1)	(50, 0.01)	(50, 1.0)	(200, 0.1)
Bagging DT	(mf, ms, n_{est})	(0.8, 0.8, 150)	(0.8, 1.0, 100)	(1.0, 1.0, 100)	(0.6, 0.6, 200)	(1.0, 0.8, 200)
ChaosNet	(b, ϵ, q)	(0.243, 0.221, 0.353)	(0.411, 0.189, 0.317)	(0.342, 0.270, 0.946)	(0.254, 0.291, 0.305)	(0.377, 0.439, 0.999)
Decision Tree	(d, msl, mss)	(10, 4, 5)	(40, 1, 2)	(30, 1, 5)	(30, 1, 10)	(10, 1, 10)
Random Forest	(d, msl, mss, n_{est})	(40, 2, 5, 200)	(10, 1, 10, 100)	(20, 1, 2, 50)	(20, 4, 10, 250)	(20, 2, 10, 150)
SVM	(C, deg, γ, ker)	(100, 4, 1, poly)	(10, NA, 0.01, rbf)	(100, 4, 0.1, poly)	(1, NA, 0.1, rbf)	(1, NA, 0.1, rbf)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.6, 0.2, 7, 150, 0.1, 1.5, 1.0)	(0.6, 0.1, 7, 150, 0.1, 1.5, 1.0)	(0.6, 0.2, 10, 150, 1, 1.5, 0.6)	(0.8, 0.01, 10, 100, 0, 2, 1.0)	(0.8, 0.1, 10, 200, 0.1, 1.5, 1.0)
$RHNL_{25G75L}$	(b, ϵ, q)	(0.261, 0.218, 0.467)	(0.339, 0.276, 0.412)	(0.096, 0.121, 0.191)	(0.011, 0.010, 0.094)	(0.049, 0.290, 0.282)
$RHNL_{25G75L} + SVM$	(b, ϵ, q)	(0.045, 0.011, 0.085)	(0.239, 0.235, 0.448)	(0.146, 0.188, 0.380)	(0.121, 0.060, 0.250)	(0.334, 0.242, 0.388)
$RHNL_{50G50L}$	(b, ϵ, q)	(0.176, 0.283, 0.151)	(0.175, 0.123, 0.049)	(0.012, 0.192, 0.486)	(0.086, 0.017, 0.156)	(0.346, 0.116, 0.393)
$RHNL_{50G50L} + SVM$	(b, ϵ, q)	(0.161, 0.081, 0.391)	(0.312, 0.270, 0.403)	(0.097, 0.176, 0.330)	(0.047, 0.279, 0.297)	(0.091, 0.196, 0.350)
$RHNL_{75G25L}$	(b, ϵ, q)	(0.461, 0.112, 0.389)	(0.013, 0.131, 0.158)	(0.365, 0.247, 0.424)	(0.353, 0.234, 0.448)	(0.439, 0.174, 0.365)
$RHNL_{75G25L} + SVM$	(b, ϵ, q)	(0.271, 0.027, 0.460)	(0.455, 0.273, 0.441)	(0.108, 0.074, 0.245)	(0.091, 0.189, 0.342)	(0.474, 0.252, 0.315)

TABLE 39. PFF-HSTR seed-wise tuned hyperparameters under spatially grouped evaluation for seeds 42–82.

Model	Params	42	52	62	72	82
AdaBoost	(n_{est}, lr)	(50, 0.5)	(50, 0.01)	(200, 0.01)	(50, 1.0)	(100, 0.01)
Bagging DT	(mf, ms, n_{est})	(0.6, 0.6, 50)	(0.6, 0.6, 200)	(1.0, 0.6, 150)	(1.0, 0.6, 150)	(0.6, 0.8, 50)
ChaosNet	(b, ϵ, q)	(0.176, 0.188, 0.205)	(0.261, 0.467, 0.075)	(0.086, 0.421, 0.146)	(0.013, 0.019, 0.965)	(0.050, 0.107, 0.903)
Decision Tree	(d, msl, mss)	(30, 1, 2)	(40, 1, 2)	(10, 2, 5)	(20, 2, 2)	(10, 4, 5)
Random Forest	(d, msl, mss, n_{est})	(40, 2, 10, 50)	(10, 4, 10, 150)	(30, 2, 10, 250)	(50, 2, 10, 200)	(10, 1, 10, 200)
SVM	(C, deg, γ, ker)	(100, 2, 0.01, poly)	(100, NA, 0.01, rbf)	(1, NA, 0.1, rbf)	(1, 4, 0.1, poly)	(1, 2, 1, poly)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.8, 0.1, 3, 200, 0.1, 2, 0.8)	(1.0, 0.1, 3, 200, 0.1, 1.5, 0.6)	(0.8, 0.1, 7, 200, 0.1, 1, 0.8)	(0.6, 0.1, 7, 100, 0, 2, 0.8)	(0.6, 0.01, 3, 100, 1, 2, 1.0)
$RHNL_{25G75L}$	(b, ϵ, q)	(0.043, 0.170, 0.048)	(0.395, 0.125, 0.406)	(0.482, 0.187, 0.446)	(0.075, 0.145, 0.242)	(0.270, 0.209, 0.455)
$RHNL_{25G75L} + SVM$	(b, ϵ, q)	(0.372, 0.055, 0.016)	(0.133, 0.265, 0.277)	(0.321, 0.299, 0.355)	(0.095, 0.029, 0.361)	(0.016, 0.138, 0.190)
$RHNL_{50G50L}$	(b, ϵ, q)	(0.295, 0.258, 0.020)	(0.395, 0.125, 0.406)	(0.482, 0.187, 0.446)	(0.102, 0.026, 0.217)	(0.016, 0.194, 0.436)
$RHNL_{50G50L} + SVM$	(b, ϵ, q)	(0.169, 0.049, 0.033)	(0.121, 0.296, 0.129)	(0.326, 0.015, 0.477)	(0.196, 0.297, 0.395)	(0.196, 0.285, 0.092)
$RHNL_{75G25L}$	(b, ϵ, q)	(0.348, 0.245, 0.123)	(0.395, 0.125, 0.406)	(0.482, 0.187, 0.446)	(0.166, 0.030, 0.334)	(0.283, 0.250, 0.447)
$RHNL_{75G25L} + SVM$	(b, ϵ, q)	(0.347, 0.279, 0.499)	(0.012, 0.137, 0.403)	(0.288, 0.292, 0.326)	(0.387, 0.140, 0.268)	(0.159, 0.139, 0.165)

TABLE 40. PFF-HSTR seed-wise tuned hyperparameters under spatially grouped evaluation for seeds 92–132.

Model	Params	92	102	112	122	132
AdaBoost	(n_{est}, lr)	(50, 0.01)	(200, 1.0)	(50, 1.0)	(50, 0.1)	(150, 0.1)
Bagging DT	(mf, ms, n_{est})	(0.8, 0.8, 100)	(0.6, 0.6, 50)	(0.8, 1.0, 100)	(0.6, 0.8, 50)	(0.6, 0.6, 100)
ChaosNet	(b, ϵ, q)	(0.243, 0.221, 0.353)	(0.134, 0.329, 0.926)	(0.202, 0.382, 0.269)	(0.480, 0.364, 0.213)	(0.006, 0.343, 0.942)
Decision Tree	(d, msl, mss)	(40, 4, 2)	(20, 1, 10)	(50, 2, 10)	(40, 2, 10)	(20, 1, 5)
Random Forest	(d, msl, mss, n_{est})	(20, 4, 2, 100)	(20, 2, 10, 150)	(40, 2, 5, 150)	(50, 1, 2, 50)	(30, 4, 5, 50)
SVM	(C, deg, γ, ker)	(1, NA, 0.1, rbf)	(100, 2, 0.01, poly)	(100, 3, 1, poly)	(10, 2, 1, poly)	(100, NA, 0.01, rbf)
XGBoost	$(cb, lr, d, n_{est}, \alpha, \lambda, ss)$	(0.6, 0.01, 7, 200, 0.1, 1, 1.0)	(1.0, 0.1, 10, 200, 0.1, 2, 0.8)	(0.8, 0.1, 3, 100, 0, 2, 1.0)	(1.0, 0.01, 7, 200, 1, 2, 0.6)	(1.0, 0.2, 10, 150, 1, 1.5, 0.8)
$RHNL_{25G75L}$	(b, ϵ, q)	(0.174, 0.128, 0.392)	(0.014, 0.107, 0.364)	(0.035, 0.164, 0.460)	(0.166, 0.224, 0.378)	(0.044, 0.052, 0.418)
$RHNL_{25G75L} + SVM$	(b, ϵ, q)	(0.497, 0.210, 0.349)	(0.054, 0.112, 0.405)	(0.280, 0.099, 0.284)	(0.176, 0.052, 0.059)	(0.211, 0.282, 0.390)
$RHNL_{50G50L}$	(b, ϵ, q)	(0.396, 0.133, 0.444)	(0.087, 0.089, 0.224)	(0.247, 0.299, 0.419)	(0.067, 0.088, 0.430)	(0.183, 0.073, 0.437)
$RHNL_{50G50L} + SVM$	(b, ϵ, q)	(0.480, 0.190, 0.355)	(0.465, 0.033, 0.193)	(0.297, 0.045, 0.429)	(0.218, 0.042, 0.246)	(0.188, 0.250, 0.386)
$RHNL_{75G25L}$	(b, ϵ, q)	(0.365, 0.091, 0.450)	(0.068, 0.115, 0.408)	(0.297, 0.130, 0.213)	(0.159, 0.223, 0.017)	(0.289, 0.180, 0.342)
$RHNL_{75G25L} + SVM$	(b, ϵ, q)	(0.067, 0.272, 0.302)	(0.405, 0.032, 0.277)	(0.246, 0.189, 0.492)	(0.044, 0.298, 0.402)	(0.031, 0.015, 0.454)

TABLE 41. PFF-HSTR training cross-validation macro-F1 during hyperparameter optimization.

Model	Random	Spatial
AdaBoost	0.546 ± 0.025	0.531 ± 0.018
Bagging DT	0.567 ± 0.014	0.539 ± 0.021
ChaosNet	0.558 ± 0.008	0.555 ± 0.027
Decision Tree	0.547 ± 0.025	0.533 ± 0.031
Random Forest	0.568 ± 0.017	0.538 ± 0.023
SVM	0.545 ± 0.010	0.543 ± 0.013
XGBoost	0.569 ± 0.020	0.545 ± 0.026
$RHNL_{25G75L}$	0.563 ± 0.014	0.558 ± 0.018
$RHNL_{25G75L} + SVM$	0.573 ± 0.014	0.561 ± 0.022
$RHNL_{50G50L}$	0.558 ± 0.017	0.565 ± 0.019
$RHNL_{50G50L} + SVM$	0.580 ± 0.018	0.570 ± 0.025
$RHNL_{75G25L}$	0.562 ± 0.019	0.554 ± 0.018
$RHNL_{75G25L} + SVM$	0.575 ± 0.016	0.562 ± 0.023

TABLE 42. CFF-LSTR representative training and inference runtimes.

Model	Setting	Train 1%	Infer 1%	Train 50%	Infer 50%
AdaBoost	Normal	1.201 ms	132.0 μ s	2.571 ms	162.0 μ s
	Log	1.248 ms	136.0 μ s	2.642 ms	164.0 μ s
Bagging DT	Normal	188.753 ms	9.178 ms	311.735 ms	11.282 ms
	Log	184.305 ms	8.876 ms	305.683 ms	10.932 ms
ChaosNet	Normal	175.0 μ s	782.0 μ s	349.0 μ s	668.0 μ s
	Log	160.0 μ s	610.0 μ s	345.0 μ s	681.0 μ s
Decision Tree	Normal	721.0 μ s	100.0 μ s	2.224 ms	134.0 μ s
	Log	713.0 μ s	99.0 μ s	2.215 ms	136.0 μ s
Random Forest	Normal	98.480 ms	3.523 ms	154.313 ms	6.219 ms
	Log	135.125 ms	5.041 ms	225.509 ms	9.331 ms
$RHNL_{75G25L}$	Normal	5.385 ms	5.261 ms	5.525 ms	5.331 ms
	Log	4.422 ms	5.500 ms	5.546 ms	5.327 ms
SVM	Normal	2.046 ms	750.0 μ s	75.802 ms	14.825 ms
	Log	1.612 ms	607.0 μ s	309.606 ms	12.021 ms
XGBoost	Normal	59.756 ms	1.456 ms	287.060 ms	2.690 ms
	Log	87.465 ms	1.776 ms	355.491 ms	3.258 ms

additional evidence for the stability and reproducibility of the AFF-HSTR results under both random and time-aware evaluation protocols.

C. EXPANDED LSTR RESULTS AND RUNTIME ANALYSIS

This subsection provides supplementary AFF-LSTR results and runtime measurements that were not included in the

TABLE 43. PFF-HSTR seed-wise test macro-F1 under random-split evaluation.

Model	42	52	62	72	82	92	102	112	122	132
AdaBoost	0.558	0.516	0.528	0.548	0.480	0.480	0.604	0.585	0.479	0.577
Bagging DT	0.567	0.576	0.528	0.546	0.552	0.624	0.596	0.571	0.510	0.586
ChaosNet	0.596	0.499	0.490	0.499	0.566	0.622	0.525	0.509	0.500	0.528
Decision Tree	0.555	0.548	0.610	0.461	0.513	0.558	0.577	0.534	0.528	0.585
Random Forest	0.605	0.555	0.596	0.527	0.603	0.630	0.623	0.615	0.458	0.548
SVM	0.558	0.596	0.471	0.519	0.528	0.488	0.510	0.528	0.485	0.606
XGBoost	0.567	0.558	0.460	0.538	0.592	0.583	0.565	0.546	0.435	0.558
<i>RHNL</i> _{25G75L}	0.597	0.567	0.516	0.433	0.489	0.566	0.528	0.478	0.499	0.528
<i>RHNL</i> _{25G75L} + SVM	0.538	0.604	0.532	0.474	0.495	0.577	0.603	0.604	0.558	0.548
<i>RHNL</i> _{50G50L}	0.596	0.548	0.532	0.479	0.515	0.471	0.508	0.507	0.474	0.441
<i>RHNL</i> _{50G50L} + SVM	0.566	0.509	0.519	0.508	0.587	0.423	0.545	0.600	0.490	0.577
<i>RHNL</i> _{75G25L}	0.452	0.567	0.548	0.460	0.567	0.612	0.504	0.529	0.442	0.479
<i>RHNL</i> _{75G25L} + SVM	0.510	0.498	0.615	0.546	0.500	0.500	0.500	0.548	0.538	0.566

TABLE 44. PFF-HSTR seed-wise test macro-F1 under spatially grouped evaluation.

Model	42	52	62	72	82	92	102	112	122	132
AdaBoost	0.601	0.520	0.435	0.482	0.510	0.487	0.453	0.483	0.402	0.521
Bagging DT	0.528	0.504	0.501	0.480	0.466	0.507	0.469	0.556	0.511	0.567
ChaosNet	0.508	0.424	0.472	0.513	0.440	0.577	0.515	0.497	0.406	0.506
Decision Tree	0.550	0.519	0.403	0.433	0.488	0.427	0.482	0.476	0.552	0.475
Random Forest	0.510	0.459	0.488	0.500	0.479	0.521	0.504	0.577	0.506	0.597
SVM	0.560	0.475	0.503	0.505	0.521	0.478	0.560	0.517	0.499	0.523
XGBoost	0.522	0.460	0.504	0.509	0.451	0.521	0.576	0.543	0.457	0.544
<i>RHNL</i> _{25G75L}	0.475	0.456	0.402	0.536	0.427	0.581	0.464	0.541	0.485	0.441
<i>RHNL</i> _{25G75L} + SVM	0.546	0.433	0.492	0.490	0.449	0.553	0.480	0.545	0.441	0.546
<i>RHNL</i> _{50G50L}	0.447	0.474	0.402	0.436	0.458	0.541	0.538	0.497	0.494	0.487
<i>RHNL</i> _{50G50L} + SVM	0.530	0.482	0.538	0.505	0.470	0.460	0.434	0.487	0.369	0.555
<i>RHNL</i> _{75G25L}	0.463	0.434	0.412	0.518	0.439	0.580	0.480	0.518	0.432	0.556
<i>RHNL</i> _{75G25L} + SVM	0.477	0.416	0.426	0.498	0.375	0.490	0.453	0.510	0.428	0.530

main Results section. Since the LSTR setting evaluates model behavior under progressively increasing training fractions, it provides a direct view of data efficiency and low-sample robustness. The representative runtime results reported in Table 29 and Figure 18 summarize the training and inference costs of the evaluated classifiers at the lowest and highest reported training fractions. These results complement the LSTR performance analysis by showing that ChaosNet achieves strong early low-sample performance with very low training cost, whereas ensemble methods generally require higher computational time despite becoming increasingly competitive as the available training data increases.

APPENDIX B
ADDITIONAL RESULTS FOR THE CANADIAN FOREST FIRE DATASET

A. SEED-WISE HYPERPARAMETER CONFIGURATIONS

Tables 30–33 report the seed-wise hyperparameters selected for CFF-HSTR under the normal and log-transformed burned-area feature settings. To preserve readability in the two-column IEEE format, each setting is split across two tables according to seed range. The values in each seed column follow the parameter order specified in the “Params” column.

In the hyperparameter tables, n_{est} denotes the number of estimators, lr denotes learning rate, mf denotes maximum

TABLE 45. PFF-LSTR training and inference runtimes under random-split evaluation. All values are reported as mean $\pm$ standard deviation in milliseconds.

Model	Train 1%	Infer 1%	Train 10%	Infer 10%	Train 50%	Infer 50%
AdaBoost	1.359 $\pm$ 0.35	0.171 $\pm$ 0.06	3.211 $\pm$ 1.97	0.296 $\pm$ 0.17	88.446 $\pm$ 95.08	3.269 $\pm$ 3.61
Bagging DT	176.329 $\pm$ 79.16	11.317 $\pm$ 5.53	194.642 $\pm$ 89.59	12.635 $\pm$ 6.19	253.589 $\pm$ 113.82	13.096 $\pm$ 6.45
ChaosNet	0.205 $\pm$ 0.02	1.052 $\pm$ 0.36	0.292 $\pm$ 0.03	0.931 $\pm$ 0.04	0.478 $\pm$ 0.05	0.935 $\pm$ 0.04
Decision Tree	0.705 $\pm$ 0.02	0.126 $\pm$ 0.00	0.811 $\pm$ 0.02	0.127 $\pm$ 0.00	1.699 $\pm$ 0.09	0.135 $\pm$ 0.00
Random Forest	112.076 $\pm$ 74.16	4.690 $\pm$ 2.92	115.801 $\pm$ 76.94	4.968 $\pm$ 3.12	132.792 $\pm$ 87.91	5.337 $\pm$ 3.34
RHNL _{50G50L} +SVM	5.257 $\pm$ 1.12	4.826 $\pm$ 0.92	6.185 $\pm$ 2.40	4.663 $\pm$ 0.64	9.697 $\pm$ 5.03	5.149 $\pm$ 1.04
SVM	1.340 $\pm$ 0.03	0.290 $\pm$ 0.02	1.567 $\pm$ 0.43	0.459 $\pm$ 0.14	7.307 $\pm$ 8.71	1.208 $\pm$ 0.69
XGBoost	56.377 $\pm$ 13.27	1.607 $\pm$ 0.26	86.516 $\pm$ 21.54	1.873 $\pm$ 0.09	143.341 $\pm$ 44.67	2.244 $\pm$ 0.28

TABLE 46. PFF-LSTR training and inference runtimes under spatially grouped evaluation. All values are reported as mean $\pm$ standard deviation in milliseconds.

Model	Train 1%	Infer 1%	Train 10%	Infer 10%	Train 50%	Infer 50%
AdaBoost	1.548 $\pm$ 0.38	0.208 $\pm$ 0.07	5.555 $\pm$ 5.99	0.518 $\pm$ 0.48	130.557 $\pm$ 91.89	5.647 $\pm$ 4.02
Bagging DT	145.235 $\pm$ 75.79	10.194 $\pm$ 5.09	150.861 $\pm$ 78.83	10.388 $\pm$ 5.19	187.807 $\pm$ 98.98	10.779 $\pm$ 5.34
ChaosNet	0.192 $\pm$ 0.01	0.932 $\pm$ 0.07	0.293 $\pm$ 0.04	0.965 $\pm$ 0.09	0.465 $\pm$ 0.08	0.977 $\pm$ 0.11
Decision Tree	0.697 $\pm$ 0.02	0.126 $\pm$ 0.01	0.799 $\pm$ 0.02	0.127 $\pm$ 0.00	1.648 $\pm$ 0.13	0.136 $\pm$ 0.00
Random Forest	137.184 $\pm$ 71.35	5.738 $\pm$ 2.84	141.162 $\pm$ 73.60	6.054 $\pm$ 3.00	161.202 $\pm$ 84.61	6.540 $\pm$ 3.24
RHNL _{50G50L} +SVM	4.891 $\pm$ 0.54	4.991 $\pm$ 0.62	6.943 $\pm$ 2.00	4.908 $\pm$ 0.66	12.386 $\pm$ 6.84	5.321 $\pm$ 0.75
SVM	1.315 $\pm$ 0.02	0.296 $\pm$ 0.02	1.443 $\pm$ 0.06	0.451 $\pm$ 0.10	33.496 $\pm$ 90.47	1.184 $\pm$ 0.55
XGBoost	56.079 $\pm$ 10.46	1.671 $\pm$ 0.05	76.949 $\pm$ 22.55	1.870 $\pm$ 0.10	121.329 $\pm$ 47.16	2.217 $\pm$ 0.31

features, ms denotes maximum samples, d denotes maximum depth, $mssl$ denotes minimum samples per leaf, mss denotes minimum samples for split, cb denotes column subsampling, ss denotes row subsampling, and ker denotes the SVM kernel. For ChaosNet and RHNL variants, (q, b, ϵ) denotes initial neural activity, discrimination threshold, and epsilon, respectively.

B. EXPANDED HSTR RESULTS

This subsection reports supplementary CFF-HSTR results that were not included in the main Results section to preserve readability. The additional results include training cross-validation performance in Table 34 obtained during hyperparameter optimization, seed-wise test-set macro-F1 values in Tables 35 – 36.

Figures 19–20 provide complementary visual summaries of the CFF-HSTR results. The bootstrap confidence-interval plots provide an additional view of uncertainty in macro-F1 whereas the heatmaps summarize the complete metric profile. The close correspondence between the normal and log-transformed settings confirms that the burned-area transformation does not materially alter the principal CFF-HSTR conclusions.

C. EXPANDED LSTR RESULTS AND RUNTIME ANALYSIS

This subsection provides supplementary computational-efficiency results for CFF-LSTR under the normal and log-transformed burned-area feature settings. The main Results section presents the LSTR performance curves and representative macro-F1 values, whereas Table 42 and Figure 21 summarize the corresponding training and inference costs at representative training fractions. These results complement the predictive-performance analysis by

characterizing computational scaling as the available training data increase.

APPENDIX C

ADDITIONAL RESULTS FOR THE PORTUGAL FOREST FIRE DATASET

A. SEED-WISE HYPERPARAMETER CONFIGURATIONS

Tables 37–40 report the seed-wise hyperparameter configurations selected for PFF-HSTR under random and spatially grouped evaluation. To maintain readability in the two-column format, the results for each evaluation protocol are divided into two tables covering seeds 42–82 and 92–132, respectively. The values in each seed column follow the parameter order specified in the “Params” column.

In the hyperparameter tables, n_{est} denotes the number of estimators, lr denotes learning rate, mf denotes maximum features, ms denotes maximum samples, d denotes maximum depth, $mssl$ denotes minimum samples per leaf, mss denotes minimum samples for split, cb denotes column subsampling, ss denotes row subsampling, and ker denotes the SVM kernel. For ChaosNet and RHNL variants, (b, ϵ, q) denotes the discrimination threshold, noise intensity, and initial neural activity, respectively.

B. EXPANDED HSTR RESULTS

This subsection reports supplementary PFF-HSTR results that were not included in the main Results section to preserve readability. The additional results include cross-validation performance obtained during hyperparameter optimization, seed-wise test-set macro-F1 values, metric-level heatmaps, mean and standard deviation plots, bootstrap confidence intervals, and the absolute spatial generalization gap. These analyses provide additional evidence regarding model

stability and the reduction in predictive performance under geographically disjoint evaluation.

Figures 22–23 provide supplementary visual analyses of PFF-HSTR performance. The metric heatmaps provide a joint comparison of accuracy, macro-F1, precision, and recall while the bootstrap confidence intervals further characterize uncertainty in macro-F1.

C. EXPANDED LSTR AND RUNTIME ANALYSIS

This subsection provides supplementary PFF-LSTR results and computational-efficiency measurements under the random and spatially grouped evaluation protocols. The main Results section reports the principal low-sample performance trends, whereas the supplementary runtime analysis compares model training time and inference latency at representative training fractions in Table 45, 46 and Figure 24. These measurements distinguish predictive performance from computational cost and show how the evaluated classifiers scale as the available training data increase. Training and inference times were measured using the same experimental pipeline and hardware configuration across all models. The reported values correspond to repeated evaluations and are presented separately for the random and spatially grouped protocols.

ACKNOWLEDGMENT

The authors gratefully acknowledge Sri Mata Amritanandamayi Devi (Amma), Chancellor, Amrita Vishwa Vidyapeetham, for her inspiration and for providing financial support for the Article Processing Charges (APC) of this publication.

REFERENCES

- [1] M. D. Flannigan, B. J. Stocks, and B. M. Wotton, "Climate change and forest fires," *Sci. Total Environ.*, vol. 262, no. 3, pp. 221–229, 2000.
- [2] A. Rowell and P. F. Moore, *Global Review of Forest Fires*. Gland, Switzerland: WWF and IUCN, 2000.
- [3] J. Cotter, "Forest fires: Influences of climate change and human activity," Greenpeace Research Laboratories, Exeter, U.K., Tech. Note 05/2009, Jul. 2009.
- [4] P. S. A. Parsa and K. Zehra, "Forest fires and climate change: Causes, effects and management," *Disaster Develop.*, vol. 11, no. 2, pp. 107–123, Jul./Dec. 2022.
- [5] *Forest Fire Activities*, Forest Survey of India, Uttarakhand, India, 2025, pp. 45–50.
- [6] H. Ritchie and M. Roser. (2024). *Annual Number of Wildfires Map*. [Online]. Available: <https://ourworldindata.org/grapher/annual-number-of-fires>
- [7] (2024). *The Annual Area Burnt By Wildfires*. [Online]. Available: <https://ourworldindata.org/wildfires>
- [8] N. B. Harikrishnan and N. Nagaraj, "When noise meets chaos: Stochastic resonance in neurochaos learning," *Neural Netw.*, vol. 143, pp. 425–435, Nov. 2021.
- [9] N. B. Harikrishnan and N. Nagaraj, "A novel chaos theory inspired neuronal architecture," in *Proc. Global Conf. Advancement Technol. (GCAT)*, Oct. 2019, pp. 1–6.
- [10] H. N. Balakrishnan, A. Kathpalia, S. Saha, and N. Nagaraj, "ChaosNet: A chaos based artificial neural network architecture for classification," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 29, no. 11, Nov. 2019.
- [11] D. Sethi, N. Nagaraj, and H. N. B., "Neurochaos feature transformation for machine learning," *Integration*, vol. 90, pp. 157–162, May 2023.
- [12] R. Ajai A S and N. Nagaraj, "Random heterogeneous neurochaos learning architecture for data classification," *Chaos Theory Appl.*, vol. 7, no. 1, pp. 10–30, Mar. 2025.
- [13] F. Abid and N. Izeboudjen, "Predicting forest fire in Algeria using data mining techniques: Case study of the decision tree algorithm," in *Advanced Intelligent Systems for Sustainable Development*. Cham, Switzerland: Springer, 2020, pp. 363–370.
- [14] P. Cortez and A. Morais, "A data mining approach to predict forest fires using meteorological data," in *Proc. New Trends Artif. Intell.*, 2007, pp. 512–523.
- [15] Y. O. Sayad, H. Mousannif, and H. Al Moatassime, "Predictive modeling of wildfires: A new dataset and machine learning approach," *Fire Saf. J.*, vol. 104, pp. 130–146, Mar. 2019.
- [16] W. L. Fons, "Analysis of fire spread in light forest fuels¹," *J. Agricul. Res.*, vol. 72, no. 3, p. 93, 1946.
- [17] C. S. Tarifa and A. M. Torralbo, "Flame propagation along the interface between a gas and a reacting medium," in *Proc. Symp. Combustion*, Jan. 1967, pp. 533–544.
- [18] R. C. Rothermel, "Mathematical model for predicting fire spread in wildland fuels," U.S. Dept. of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, Ogden, UT, USA, Res. Pap. INT-115, 1972.
- [19] H. E. Anderson, "Aids to determining fuel models for estimating fire behavior," U.S. Dept. of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, Ogden, UT, USA, Gen. Tech. Rep. INT-GTR-122, 1982.
- [20] A. L. Sullivan, "Wildland surface fire spread modelling, 1990–2007. 1: Physical and quasi-physical models," *Int. J. Wildland Fire*, vol. 18, no. 4, pp. 349–368, Jun. 2009.
- [21] D. Morvan, G. Accary, S. Meradji, and N. Frangieh, "Fifty years of progress in wildland fire modelling: From empirical to fully physical CFD models," *Comp. Rendus. Mécanique*, vol. 350, no. S1, pp. 1–9, Nov. 2024.
- [22] E. Chuvieco and R. G. Congalton, "Application of remote sensing and geographic information systems to forest fire hazard mapping," *Remote Sens. Environ.*, vol. 29, no. 2, pp. 147–159, Aug. 1989.
- [23] K. Coskuner, "Assessing the performance of MODIS and VIIRS active fire products in the monitoring of wildfires: A case study in Turkey," *iForest-Biogeosciences Forestry*, vol. 15, no. 2, pp. 85–94, Apr. 2022.
- [24] M. Gargiulo, D. A. G. Dell' Aglio, A. Iodice, D. Riccio, and G. Ruello, "A CNN-based super-resolution technique for active fire detection on Sentinel-2 data," in *Proc. Photon. Electromagn. Res. Symp. (PIERS-Spring)*, Jun. 2019, pp. 418–426.
- [25] M. C. Iban and A. Sekertekin, "Machine learning based wildfire susceptibility mapping using remotely sensed fire data and GIS: A case study of adana and mersin provinces, Turkey," *Ecological Informat.*, vol. 69, Jul. 2022, Art. no. 101647.
- [26] J. A. Blackard and D. J. Dean, "Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables," *Comput. Electron. Agricult.*, vol. 24, no. 3, pp. 131–151, Dec. 1999.
- [27] A. Baba, "Neural networks from biological to artificial and vice versa," *BioSystems*, vol. 235, Jan. 2024, Art. no. 105110.
- [28] D. Stojanova, P. Panov, A. Kobler, S. Džeroski, and K. Taškova, "Learning to predict forest fires with different data mining techniques," in *Proc. Conf. Data Mining Data Warehouses (SiKDD)*, 2006, pp. 255–258.
- [29] F. Huot, R. L. Hu, N. Goyal, T. Sankar, M. Ihme, and Y.-F. Chen, "Next day wildfire spread: A machine learning dataset to predict wildfire spreading from remote-sensing data," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4412513.
- [30] R. Ghali and M. A. Akhloufi, "Deep learning approaches for wildland fires using satellite remote sensing data: Detection, mapping, and prediction," *Fire*, vol. 6, no. 5, p. 192, May 2023.
- [31] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.
- [32] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018.

- [33] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 18932–18943.
- [34] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?" in *Proc. Adv. Neural Inf. Process. Syst.* 35, 2022, pp. 507–520.
- [35] R. A. A. S., H. N. B., and N. Nagaraj, "Analysis of logistic map based neurons in neurochaos learning architectures for data classification," *Chaos, Solitons Fractals*, vol. 170, May 2023, Art. no. 113347.
- [36] D. R. Roberts, V. Bahn, S. Ciuti, M. S. Boyce, J. Elith, G. Guillera-Arroita, S. Hauenstein, J. J. Lahoz-Monfort, B. Schröder, W. Thuiller, D. I. Warton, B. A. Wintle, F. Hartig, and C. F. Dormann, "Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure," *Ecography*, vol. 40, no. 8, pp. 913–929, Aug. 2017.
- [37] J. San-Miguel-Ayanz, T. Durrant, R. Boca, G. Libertà, A. Branco, D. de Rigo, D. Ferrari, P. Maianti, T. Artés Vivancos, H. Costa, and F. Lana, "Advance EFFIS report on forest fires in Europe, Middle East and North Africa 2017," European Commission, Joint Research Centre/Publications Office of the European Union, Luxembourg, Rep. EUR 29316 EN (JRC111456), 2018.
- [38] C. E. Van Wagner et al., "Structure of the Canadian forest fire weather index," Canadian Forestry Service, Dept. of the Environment, Ottawa, ON, Canada, Publ. No. 1333, 1974.
- [39] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annu. Workshop Comput. Learn. theory*, Jul. 1992, pp. 144–152.
- [40] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [41] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, Mar. 1986.
- [42] R. E. Schapire, "Explaining AdaBoost," in *Empirical Inference*, 2013, pp. 37–52.
- [43] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, Aug. 1996.
- [44] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.
- [45] N. B. Harikrishnan, A. Kathpalia, and N. Nagaraj, "Causality preserving chaotic transformation and classification using neurochaos learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 2046–2058. [Online]. Available: <https://proceedings.neurips.cc/paperfiles/paper/2022/file/0d9057d84a9fc37523bf826232ea6820-Paper-Conference.pdf>



KUNAL KUMAR PANT is currently pursuing the bachelor's degree in electronics and communication engineering with Amrita Vishwa Vidyapeetham, Amritapuri Campus. His research interests include data efficient machine learning, sustainable and ethical AI, applied AI, and philosophy in AI.



REMYA AJAI A S (Member, IEEE) is currently an Assistant Professor with the Department of Electronics and Communication Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri Campus. Her research interests include chaos theory, machine learning, biomedical image processing, VLSI signal processing, and embedded systems.



NITHIN NAGARAJ (Senior Member, IEEE) received the bachelor's degree in electrical and electronics engineering from the National Institute of Technology Karnataka (NITK), Surathkal, in 1999, the master's degree in electrical engineering from Rensselaer Polytechnic Institute (RPI), Troy, NY, USA, in 2001, and the Ph.D. degree from the National Institute of Advanced Studies (NIAS), Bengaluru, in 2010. He has previously held positions with GE Global Research, IISER Pune, and Amrita University. He is currently a Professor and the Head with the Complex Systems Programme, NIAS, and a member of the NIAS Consciousness Studies Programme. His research interests include brain-inspired artificial intelligence, chaos theory, information theory, complexity theories of causality, and consciousness. He is an Invited Member of the Advisory Council of METI: Messaging Extra Terrestrial Intelligence International, USA. He was conferred the "Distinguished Alumnus Award" for outstanding service and contribution towards research and education by the NITK Surathkal Alumni Association, in September 2024.

...