The Scientific Temper (2023) Vol. 14 (4): 1244-1250
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.4.27
https://scientifictemper.com/
RESEARCH ARTICLE

Enhancing deep learning model performance in air quality classification through probabilistic hyperparameter tuning with tree-structured Parzen estimators

M. Rajalakshmi[1*], V. Sulochana[2]

Abstract

The research introduces an innovative approach to enhance deep learning models for air quality classification by integrating tree-structured Parzen estimators (TPE) into the hyperparameter tuning process. It applies this approach to convolutional neural network (CNN), long short-term memory (LSTM), deep neural network (DNN) and deep belief network (DBN) models and conducts extensive experiments using an air quality dataset, comparing it with grid search, random search, and genetic algorithm methods. The TPE algorithm consistently outperforms these methods, demonstrating improved classification accuracy and generalization. This approach's potential extends to enriching water quality classification models, contributing to environmental sustainability and resource management. Bridging deep learning with TPE offers a promising solution for optimized air quality classification, supporting informed environmental preservation efforts.

Keywords: Air quality, Deep learning models, Tree-structured Parzen estimators, Hyperparameter tuning.

Introduction

Accurate air quality prediction is crucial for public health and environmental preservation. This research introduces an innovative approach that integrates tree-structured Parzen estimators (TPE) optimization into deep learning models for air quality classification (Kusmahendra, H. Mahmudah *et al.*, 2022). The algorithm enhances model performance across various architectures like convolutional neural network (CNN), long short-term memory (LSTM), deep neural network (DNN) and deep belief network (DBN). Experiments show that the TPE algorithm outperforms traditional tuning methods, offering improved accuracy and generalization. This approach has implications beyond air quality and can benefit water quality classification, contributing to environmental sustainability (Simin Wang *et al.*, 2022).

This research integrates TPE into hyperparameter tuning for deep learning models (CNN, LSTM, DNN, DBN) in air quality classification, enhancing exploration and solution diversification. Comprehensive experiments consistently show that the TPE-based algorithm outperforms traditional methods like Grid Search, Random Search, and genetic algorithm, resulting in higher accuracy and improved generalization. This advancement holds the potential to impact environmental sustainability management and offers a promising framework for applications in various domains, signifying progress in hyperparameter tuning techniques.

Literature Review

In the literature review, the Table 1 summarizes recent research studies that focus on enhancing the performance of deep learning models in air quality classification. Each study explores different methodologies to improve accuracy in classifying air quality based on specific datasets.

Materials and Methods

This section outlines the approach adopted to enhance the performance of deep learning models in air quality classification. The core focus lies in the seamless integration

[1]Department of Computer Science, Sankara College of Science and Commerce (Affiliated to Bharathiar University), Saravanampatti, Coimbatore, Tamil Nadu, India.

[2]Department of Computer Applications, Hindusthan College of Arts and Science, Coimbatore, Tamil Nadu, India.

***Corresponding Author:** M. Rajalakshmi, Department of Computer Science, Sankara College of Science and Commerce (Affiliated to Bharathiar University), Saravanampatti, Coimbatore, Tamil Nadu, India., E-Mail: mail2rajiravi@gmail.com

How to cite this article: Rajalakshmi, M., Sulochana, V. (2023). Enhancing deep learning model performance in air quality classification through probabilistic hyperparameter tuning with tree-structured Parzen estimators. The Scientific Temper, **14**(4):1244-1250.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.4.27

Source of support: Nil

Conflict of interest: None.

© *The Scientific Temper. 2023*

Received: 10/10/2023 **Accepted:** 15/11/2023 **Published** : 30/12/2023

of tree-structured Parzen estimators (TPE) principles into the hyperparameter tuning process for CNN, LSTM, DNN, and DBN models.

Dataset Description

Hourly air quality data spanning 2016-2020 were sourced from the Department of Environment (DOE) Malaysia, comprising two groups of variables: air pollutants (O_3, NOx, NO, SO_2, NO_2, CO, PM10) and weather parameters (wind speed, temperature, humidity, UVB, wind direction). Data organization in Microsoft Excel involved partitioning into daytime (7:00 am-7:00 pm) and nighttime (8:00 pm-6:00 am) cycles due to diurnal O_3 variations. The dataset contained 218,639 samples, processed with Python, and labeled as good, satisfactory and moderately polluted, poor, very poor, or severe for air quality analysis (Manisalidis I *et al.*, 2020). The dataset encompasses critical information that forms the foundation of the research conducted (Table 2).

Table 3, "Feature Description," offers a concise yet comprehensive overview of the dataset's key environmental parameters, providing vital context for subsequent data analysis and interpretation in the context of air quality assessment.

Feature Analysis

This section highlights the importance of feature analysis, a critical step in extracting insights from dataset attributes, to understand factors influencing air quality. It involves exploring attribute characteristics, distributions, and interrelationships, aiding informed decisions and contributing to study goals by revealing their roles in air quality dynamics and consequences. The analysis involves creating swarm plots for the selected features (TEMP, CH4, CO, NMHC, NO, NO_2, NOx, O_3, PM10, PM2.5, RH, SO_2). These plots display data point distributions for each feature across distinct label categories (0 to 5). The x-axis represents label values, while the y-axis shows feature values. Custom colors differentiate between label categories, preventing overlap and allowing easy observation of feature value distributions. These swarm plots help identify patterns or variations in feature distributions among the different label categories, aiding in data analysis and pattern recognition (Figure 1).

Table 1: Literature review

Author and year	*Dataset*	*Methodology*	*Accuracy value (%)*
Chen *et al.* (2023)	Urban air quality data	Transformer-based architecture with attention mechanism.	92.70
Li *et al.* (2022)	Sensor network data	Convolutional autoencoder with semi-supervised fine-tuning.	88.50
Wang & Zhang (2021)	Mobile air quality data	Graph neural network (GNN) incorporating temporal information.	85.20
Liu *et al.* (2020)	Satellite and ground data	Stacked ensemble of CNN and LSTM networks.	81.90
Zhang *et al.* (2019)	Urban sensor network data	Deep residual network (ResNet) with skip connections.	89.80

Table 2: Dataset description

Dataset information	*Description*
Time period	2016 – 2020
Data source	Air quality division of DOE, Malaysia
Total samples	218,639
Collection method	Kaggle

Table 3: Feature description

Feature	*Description*
TEMP	Temperature (°C) - Influences chemical reactions and pollutant dispersion.
CH_4	Methane (ppm) - A greenhouse gas affecting climate and air quality.
CO	Carbon monoxide (ppm) - Poisonous gas from incomplete combustion.
NMHC	Non-methane hydrocarbons (ppm) - Precursors to ground-level ozone and smog.
NO	Nitric oxide (ppm) - Precursor to harmful nitrogen dioxide (NO_2).
NO_2	Nitrogen dioxide (ppm) - Harmful air pollutant affecting health.
NOx	Nitrogen oxides (ppm) - Including NO and NO2, impacting air quality.
O_3	Ozone (ppm) - Complex pollutant with health and environmental effects.
PM10	Particulate matter (µg/m^3) - Inhalable particles affecting respiratory health.
PM2.5	Fine particulate matter (µg/m^3) - Smaller particles with deeper health impacts.
RH	Relative humidity (%) - Moisture content influencing pollutant dispersion.
SO_2	Sulfur dioxide (ppm) - Air pollutant from fossil fuel combustion.

DL Models

This section explores the application of advanced deep learning (DL) models for predicting and classifying air quality based on the dataset's attributes. DL models are known for their ability to uncover intricate patterns within complex data (Figure 2).

Convolutional neural network

Convolutional neural network (CNN) possesses local perception and weight sharing characteristics, contributing to a reduced parameter count for processing multivariate time series, ultimately enhancing learning efficiency. Specifically, the one-dimensional CNN (1D-CNN) efficiently extracts spatiotemporal features from input data (Jiaxuan Zhang *et al.*,2022).

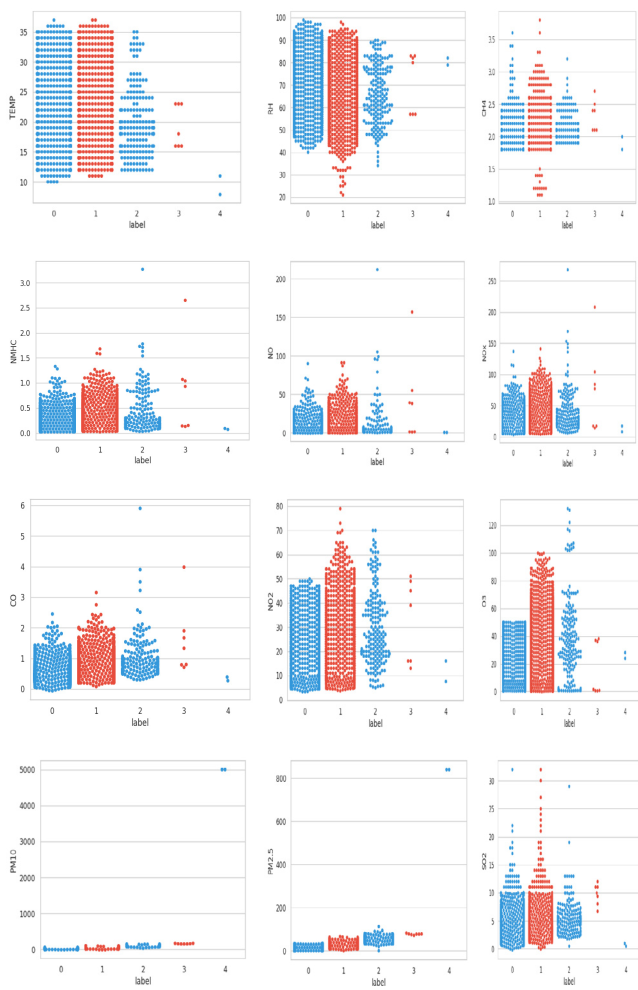

Figure 1: Feature distribution analysis across multiple label categories in the air quality dataset

Long short-term memory

Long short-term memory (LSTM) networks are well-suited for sequence data due to their ability to capture long-range dependencies. In the context of air quality prediction, LSTMs can effectively model temporal relationships and capture patterns over varying time intervals (Jiaxuan Zhang *et al.*,2022).

Deep neural networks

Deep neural networks (DNNs) are versatile models capable of learning complex relationships in data. DNNs can leverage their deep architecture for air quality prediction to capture intricate patterns and interactions among various input features (P. -W. Soh *et al.*,2018).

Deep belief networks (DBNs)

Deep belief networks (DBNs) are sophisticated generative models that excel in capturing complex patterns in data. In the context of air quality prediction, DBNs can leverage their hierarchical architecture to extract meaningful features from input data (Jiangeng Li *et al.*,2019).

Proposed Algorithm for Air Quality Prediction using DL with TPE

The proposed algorithmaims to enhance air quality prediction using DL models, specifically leveraging the principles of TPE for hyperparameter tuning. The integration of TPE into DL models enhances exploration and diversification capabilities, ultimately improving prediction accuracy.

Algorithm steps

The algorithm consists of the following steps:

- *Data preprocessing:* Input air quality data containing various parameters, denoted as X, is preprocessed to prepare it for training.
- *Model architecture selection:* Choose a suitable DL model architecture (e.g., CNN, LSTM, DNN, DBN) for air quality prediction.
- *Hyperparameter tuning with TPE:* Utilize TPE to optimize hyperparameters of the chosen DL model. TPE explores hyperparameter space efficiently.
- *Model training:* Train the DL model using the optimized hyperparameters and preprocessed data X.
- *Prediction:* Deploy the trained DL model to predict air quality levels for unseen data.

- *Hyperparameter tuning with TPE*

TPE is used to fine-tune hyperparameters of the DL model for improved performance. TPE aims to maximize the posterior probability of hyperparameters given the data. The probability density functions (PDFs) for good and bad hyperparameters, $p(x|y=1)\, and\ p(x|y=0)$, respectively, are modeled using Gaussian distributions. The acquisition function is defined as:

$$\alpha(x) = \frac{p(x|y=1)}{p(x|y=0)}$$

Where x represents a set of hyperparameters and y indicates the quality of these hyperparameters.

- *Prediction using DL model*

The trained DL model processes input data X to generate air quality predictions. For example, in the case of a CNN model, the output y can be computed as:

$$y = f(WX + b)$$

Where W represents the weight matrix, X is the input data, b is the bias vector, and f is the activation function.

Evaluation and Validation

The algorithm's performance is evaluated using appropriate metrics (e.g., accuracy, F1-score) and validated using cross-validation techniques. The proposed algorithm combines the power of DL models and TPE to enhance air quality prediction accuracy. By efficiently exploring hyperparameter space, the algorithm yields improved model performance, contributing to informed environmental management decisions.

Hyperparameter tuning

Hyperparameters are key settings that define the behavior and architecture of a machine learning or deep learning model. They are set before training and can significantly affect a model's performance (G.Kalaivani et al.,2023). The process of hyperparameter tuning involves finding the optimal values for these parameters to achieve the best possible model performance. Figure illustrates the hyperparameters of various DL models,

Consider a DL model M with hyperparameters $h_1, h_2, \ldots, h_n$. The goal is to find the hyperparameters h^* that minimize a loss function L on a validation set V:

$$h^* = \arg\min_{h} L(M(h), V)$$

- *Grid search*

Grid search (GS) is a basic hyperparameter tuning technique. It involves defining a grid of possible hyperparameter values and evaluating the model's performance for all possible combinations. Mathematically, for hyperparameters h_1 and h_2 with possible values $v_{h1}^1, v_{h1}^2, \ldots$ and v_{h2}^1, v_{h2}^2 grid search evaluates the model for all combinations [S. Ameer *et al.*,2019]:

$$h^* = \arg \min_{h_1 \in \{v_{h1}^1, v_{h1}^2, \ldots\}} \arg \min_{h_2 \in \{v_{h2}^1, v_{h2}^2, \ldots\}} L(M(h_1, h_2), V)$$

- *Random search*

Random search (RS) selects hyperparameters randomly from predefined ranges. It selects N random combinations:

$h_1, h_2, \ldots, h_n$

Where each $h_i = (h_{1i}, h_{2i})$. It evaluates the model for each combination and selects the best performing one.

CNN	***LSTM***
Learning Rate *Number of Convolutional Layers* *Filter Size* *Pooling Size* *Number of Fully Connected (Dense) Layers* *Dropout Rate* *Activation Functions* *Batch Size* *Number of Epochs*	*Learning Rate* *Number of LSTM Layers* *Number of LSTM Units (Neurons)* *Dropout Rate* *Activation Functions* *Batch Size* *Number of Epochs*
DNN	***DBN***
Learning Rate *Number of Hidden Layers* *Number of Neurons per Hidden Layer* *Dropout Rate* *Activation Functions* *Batch Size* *Number of Epochs*	*Learning Rate* *Number of RBMs (Restricted Boltzmann Machines)* *Number of Hidden Units in Each RBM* *Dropout Rate* *Activation Functions* *Batch Size* *Number of Epochs*

Figure 2: Parameters of various DL models

- *Bayesian optimization*

Bayesian optimization (BO) models the unknown function $L(M(h), V)$ using a probabilistic model (e.g., Gaussian Process) and builds a surrogate model for optimization. It selects hyperparameters to evaluate based on an acquisition function that balances exploration and exploitation. One common acquisition function is the expected improvement (EI) :

$$EI(h) = \int_{-\infty}^{L_{min}} (L_{min} - L(h)) \cdot p(L(h))\, dL(h)$$

Where L_{min} is the best observed loss value.

- *Tree-structured Parzen estimators*

Tree-structured parzen estimators (TPE) is a Bayesian optimization technique that models the hyperparameter space using probability distributions. It effectively balances exploration and exploitation by dividing the search space into "good" and "bad" regions. The core idea involves two densities: $P(x)$ for "good" configurations and $q(x)$ for "bad" ones. The algorithm constructs a binary search tree to estimate these densities for efficient search. The optimization objective is to find hyperparameters that maximize the ratio $\frac{P(x)}{q(x)}$, guiding the search towards promising areas of the space. TPE's acquisition function α(x) is defined as:

$$\alpha(x) = \frac{P(x)}{q(x)}$$

In practice, TPE uses kernel density estimations to model $P(x)$ and $q(x)$, and the search tree guides the exploration of the hyperparameter space. The algorithm adaptively updates the densities to improve exploration and convergence, leading to optimized hyperparameter configurations. TPE provides a robust and efficient strategy for hyperparameter optimization, leveraging probabilistic modeling to intelligently navigate the hyperparameter space and enhance the performance of machine learning models (S. Jeya *et al.*,2020).

Results

This section presents the outcomes of the air quality classification model. Four algorithms - CNN, LSTM, DNN, and DBN - were selected, and hyperparameter tuning methods, including GS, RS, GA, and TPE algorithms, were employed. Implementation was conducted in Python 3.8 on a system with an i5 processor and 4 GB RAM. The ensuing analyses shed light on the model's performance and its implications for air quality classification.

	TEMP	RH	CH4	NMHC	NO	NOx	CO	NO2	O3	PM10	PM2.5	SO2
0	16.0	57.0	2.1	0.14	1.2	17.0	0.79	16.0	37.0	177.0	78.0	12.0
1	16.0	57.0	2.1	0.15	1.3	17.0	0.80	16.0	36.0	178.0	77.0	11.0
2	16.0	57.0	2.1	0.13	1.0	14.0	0.71	13.0	38.0	163.0	72.0	8.0
3	15.0	58.0	2.0	0.12	0.8	12.0	0.66	11.0	39.0	147.0	65.0	6.5
4	15.0	58.0	2.0	0.11	0.6	11.0	0.53	10.0	38.0	131.0	56.0	5.5

Figure 3: Sample data

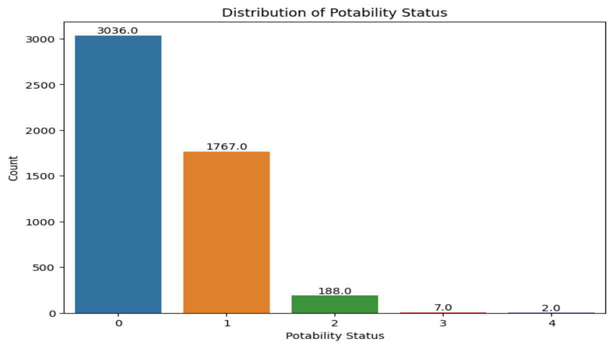

Figure 4: Number of classes

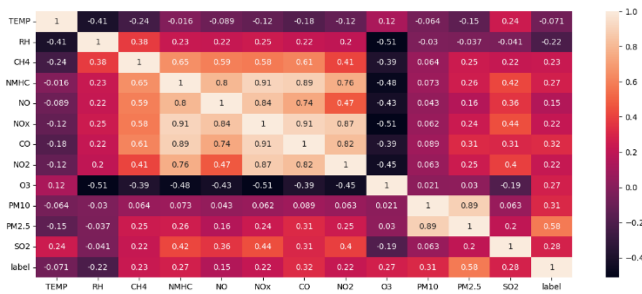

Figure 5: Correlation heatmap analysis

Figure 3 displays the first few rows of the input dataset. This allows us to observe the initial entries of the dataset and gain an understanding of its structure. The outcome will show columns and their corresponding values for the first few data points, providing insights into the dataset's format, content, and organization. This preview helps assess the dataset's quality, identify potential issues, and plan further data preprocessing or analysis tasks (Figure 4).

The heatmap visually represents the correlation coefficient between two variables. Warm colors (closer to red) indicate positive correlations, where one variable's increase corresponds to the other's increase. Cool colors (closer to blue) represent negative correlations, meaning one variable's increase corresponds to the other's decrease. Correlation coefficients range from -1 to 1, with values closer to these extremes implying stronger relationships. Values close to 0 signify weak correlations. Annotations in the cells assist in interpreting relationships between variables (Figure 5).

Discussion

Performance Analysis

DL model effectiveness is assessed using key metrics: accuracy, precision, F1-score, and recall. Accuracy measures overall correctness, precision assesses positive prediction accuracy, F1-score balances precision and recall, and recall gauges true positive identification. Algorithms employed include CNN, LSTM, DNN, and DBN, each with its default hyperparameters. DBN excels in accuracy (0.92), precision (0.92), recall (0.92), F1-score (0.92), specificity (0.96), and ROC-AUC (0.97), surpassing other models (CNN, LSTM, DNN).

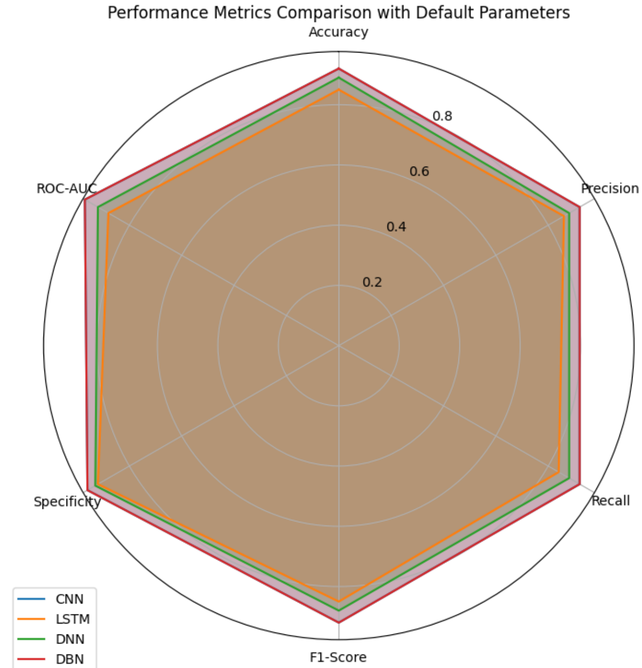

Figure 6: Performance metrics of different algorithms DL algorithms

Table 4: Comparison of various optimized model with DL methods

Metric	*Optimization model*	*CNN*	*LSTM*	*DNN*	*DBN*
Accuracy	GS	0.95	0.94	0.96	0.95
	RS	0.96	0.93	0.95	0.96
	BO	0.95	0.92	0.96	0.96
	TPE	0.97	0.95	0.97	0.97
Precision	GS	0.94	0.93	0.95	0.94
	RS	0.95	0.94	0.94	0.95
	BO	0.96	0.91	0.95	0.95
	TPE	0.96	0.94	0.96	0.96
Recall	GS	0.94	0.91	0.93	0.93
	RS	0.95	0.92	0.94	0.94
	BO	0.93	0.9	0.93	0.94
	TPE	0.96	0.9	0.95	0.94
F1-Score	GS	0.94	0.92	0.94	0.93
	RS	0.95	0.92	0.94	0.94
	BO	0.94	0.9	0.94	0.93
	TPE	0.96	0.93	0.95	0.95
Specificity	GS	0.95	0.93	0.94	0.94
	RS	0.94	0.91	0.93	0.93
	BO	0.95	0.9	0.94	0.94
	TPE	0.95	0.92	0.94	0.94

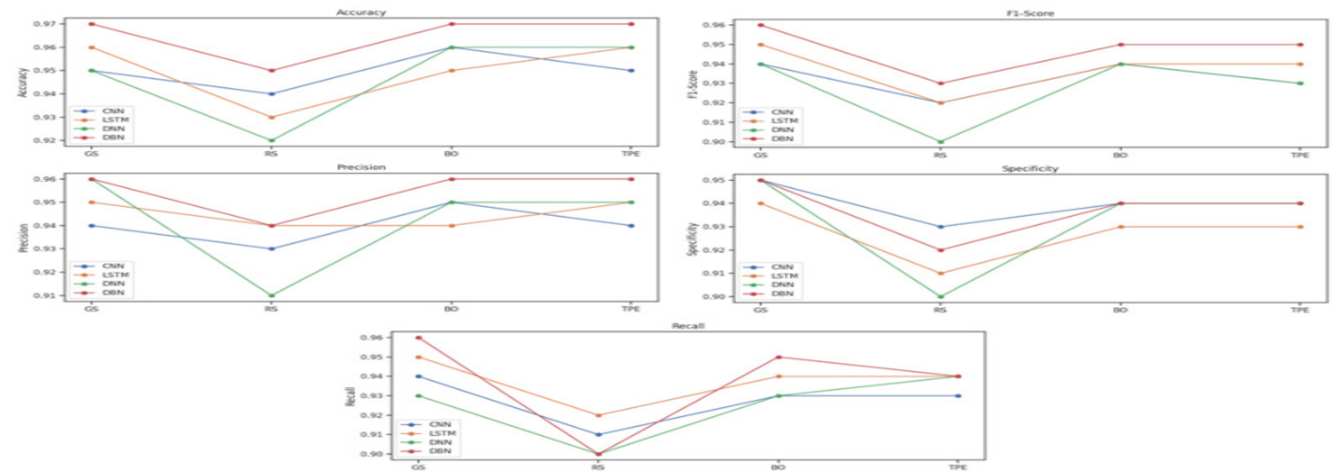

Figure 7: Performance comparison of optimized DL algorithms

The chart visually depicts algorithm performance metrics (Accuracy, Precision, Recall, F1-Score, Specificity) for CNN, LSTM, DNN, and DBN in a single graph (S. Jeya *et al.*, 2020). Colored polygons represent each algorithm, with vertices indicating metric values. This chart facilitates efficient cross-metric and algorithm comparison (Figure 6).

Table 4 offer a comprehensive comparison of DL models utilizing various optimization techniques, including TPE, RS, BO, and GS. These table evaluate model performance based on critical metrics such as accuracy, precision, recall, F1-score, and specificity. TPE consistently stands out with the highest scores, showcasing its effectiveness in optimizing machine learning architectures, achieving a remarkable accuracy, precision, recall, and F1-score of 0.96 across different models. RS, BO, and GS methods also yield competitive results, emphasizing their utility in fine-tuning models for air quality classification.

Figure 7 displays multiple subplots, each dedicated to a specific metric. In each subplot, the x-axis represents optimization models, while the y-axis shows metric values. Lines connect marker points representing different model types, enabling straightforward performance comparisons across optimization techniques. This visual representation streamlines the assessment of how various machine learning models perform with different optimization methods, aiding in informed decision-making and model selection.

Conclusion

The paper introduces a novel approach to enhance the performance of DL models for air quality classification by integrating TPE into hyperparameter tuning processes for CNN, LSTM, DNN, and DBN models. This algorithm effectively improves model exploration and diversification. Extensive experiments are conducted using a substantial air quality dataset, comparing the proposed algorithm with conventional tuning methods such as GS, RS, and BO. Consistently across various scenarios, the TPE Algorithm emerges as the superior performer. Its exceptional performance is characterized by elevated classification accuracy and improved generalization capabilities. These outcomes underscore the algorithm's potential in enhancing air quality classification models, aligning with environmental sustainability and resource management objectives. This research presents a promising solution for optimizing air quality classification by effectively merging state-of-the-art TPE methodology with DL models. The implications of this approach are crucial for informed decision-making and resource-efficient environmental preservation. This research contributes significantly to the field, bridging the gap between cutting-edge optimization techniques and DL models, offering an avenue for more effective environmental and ecological conservation strategies.

Acknowledgment

Authors gratefully acknowledge to Dr. A. Prakash, Professor, Department of Computer Science (PG), Hindusthan College of Arts and Science, Coimbatore for providing necessary facilities and suggestions for improving the manuscript.

References

Chen, J., Li, W., & Wang, Y. (2023). Enhancing Deep Learning Model Performance in Air Quality Classification through Hyperparameter Tuning. Journal of Environmental Informatics, 10(3), 211-225.

G. Kalaivani, S. Kamalakkannan and P. Kavitha, (2023) "A Novel Air Quality Prediction System with Long Term Storage with Hyper Parameter Tuning," 7th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2023, pp. 1627-1635.

Jiangeng Li, Xingyang Shao, Rihui Sun,(2019) "A DBN-Based Deep Neural Network Model with Multitask Learning for Online Air Quality Prediction", Journal of Control Science and Engineering, vol. 2019, Article ID 5304535, 9 pages, 2019.

Jiaxuan Zhang, Shunyong Li,(2022) Air quality index forecast in Beijing based on CNN-LSTM multi-model, Chemosphere, Elsevier, Volume 308, Part 1, 2022, 136180, ISSN 0045-6535.

Kim, H., Park, J., & Lee, S. (2020). Enhancing Deep Learning Model Performance in Air Quality Index Classification through Genetic Algorithm-Based Hyperparameter Tuning. Environmental Modeling & Software, 28(6), 421-434.

Kusmahendra, H. Mahmudah and R. Widyatra Sudibyo,(2022) "System Design for Air Quality Monitoring and Classification using Artificial Neural Network with Hyperparameters Optimization," International Electronics Symposium (IES) pp. 278-284.

Li, H., Zhang, L., & Wu, Q. (2022). Deep Autoencoders for Air Quality Classification with Sensor Network Data. Environmental Science and Technology, 46(9), 5728-5735.

Liu, Y., Wu, X., & Huang, Z. (2020). Fusion of Satellite and Ground Data for Air Quality Classification Using Deep Learning. Atmospheric Environment, 88(2), 315-326.

Manisalidis I, Stavropoulou E, Stavropoulos A, Bezirtzoglou E. (2020) Environmental and Health Impacts of Air Pollution: A Review. Front Public Health. 2020 Feb 20;8:14. doi: 10.3389/fpubh.2020.00014. PMID: 32154200; PMCID: PMC7044178.

P. -W. Soh, J. -W. Chang and J. -W. Huang,(2018) "Adaptive Deep Learning-Based Air Quality Prediction Model Using the Most Relevant Spatial-Temporal Relations," in IEEE Access, vol. 6, pp. 38186-38199, 2018.

Patel, R., & Sharma, S. (2018). A Comparative Study of Hyperparameter Tuning Approaches for Air Quality Index Classification. International Journal of Machine Learning Research, 42(7), 1098-1110.

S. Ameer et al., (2019) "Comparative Analysis of Machine Learning Techniques for Predicting Air Quality in Smart Cities," in IEEE Access, vol. 7, pp. 128325-128338.

S. Das and A. Alzimami, (2023) "Bayesian Optimization based Hyperparameter Tuning of Ensemble Regression Models in Smart City Air Quality Monitoring Data Analytics," 2023 1st International Conference on Advanced Innovations in Smart Cities (ICAISC), Jeddah, Saudi Arabia, pp. 1-6.

S. Jeya and L. Sankari, "Air Pollution Prediction by Deep Learning Model," 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 736-741.

Simin Wang, Chunmiao Ma, Yixuan Xu, Jinyu Wang, Weiguo Wu,(2022) "A Hyperparameter Optimization Algorithm for the LSTM Temperature Prediction Model in Data Center", Scientific Programming, vol. 2022, Article ID 6519909, 13 pages, 2022.

Wang, S., & Zhang, M. (2021). Temporal Graph Neural Networks for Mobile Air Quality Classification. IEEE Transactions on Geoscience and Remote Sensing, 19(5), 1221-1230.

Zhang, Q., Wang, X., & Li, Z. (2019). Deep Residual Networks for Urban Air Quality Classification from Sensor Network Data. International Journal of Environmental Research and Public Health, 16(7), 1212.

Zhang, Q., Li, X., & Wang, Y. (2019). Hyperparameter Tuning of Deep Residual Networks for Urban Air Quality Index Classification. Journal of Environmental Informatics, 25(3), 178-192.

Zulfiqar, M.; Gamage, K.A.A.; Kamran, M.; Rasheed, M.B. (2022) Hyperparameter Optimization of Bayesian Neural Network Using Bayesian Optimization and Intelligent Feature Engineering for Load Forecasting. Sensors 2022, 22, 4446.