

Abstract

This work presents a systematic comparative assessment of four machine learning methods for compressing Flamelet Generated Manifold (FGM) libraries in methane combustion simulations: Multi-Layer Perceptron (MLP), Random Forest (RF), Linear Regression (LR), and Support Vector Machine (SVR). Scalar dissipation rate and mixture fraction are used jointly as continuous input features, enabling each model to learn the full thermochemical response across the operating space. Models are trained on seven FlameMaster libraries and evaluated on fifteen unseen scalar dissipation rate conditions to assess genuine generalization capability. Hyperparameter optimization via cross-validated grid search identifies the optimal MLP configuration: four hidden layers (10-10-20-10 neurons), Tanh activation, and Adam solver. This model achieves a mean R^2 of 0.9747 across all species on the unseen test set (Temperature $R^2=0.9909$, OH $R^2=0.9647$). Random Forest achieves mean $R^2=0.9109$, while Linear Regression performs poorly (mean $R^2=0.43$), confirming the necessity of nonlinear methods for this thermochemical regression task. The results demonstrate that MLP-based FGM compression can achieve high accuracy with substantially reduced memory requirements compared to full look-up table storage.

Keywords: Flamelet methods, Artificial Neural Network, Random Forest, Linear Regression, Support Vector Machine.

1. Introduction

Using machine learning (ML), many combustion challenges can be addressed, including reduced-order modeling, data processing, optimization, and control. Due to the increasing prevalence of data-driven methods, fluid mechanics can benefit from learning algorithms and present challenges that may encourage these algorithms to complement human understanding and engineering intuition [1,2].

Numerical modeling of combustion systems that incorporate intricate kinetic mechanisms presents a significant level of complexity. The quantity of species pertaining to light hydrocarbon fuels can extend to several hundred, while the chemical kinetic mechanism associated with certain fuels such as kerosene encompasses a few hundred to a few thousand reactions [3]. The evolution of various species is represented by partial differential equations, which are coupled together. Additionally, the presence of turbulent flames introduces a diverse set of temporal and spatial dimensions, thereby adding complexity to the computational simulations. [4]. The temporal times of chemical processes involving intermediate species, specifically H , CH_2 , and C_2H_2 , are comparatively briefer in comparison to the time scales associated with CO_2 and H_2O [5]. In spite of significant advances in supercomputers, simplification of the combustion process is still not feasible. Therefore, it is of great interest to develop more advanced models [6].

1.1. ML in combustion and fluid dynamics

Recently, Brunton et al. [7] reviewed the history, current developments, and emerging opportunities of machine learning for fluid mechanics, and Duraisamy et al. [8] reviewed data-driven models constructed using machine learning to model turbulence. Numerous noteworthy studies have focused on the intersection of combustion and machine learning, offering a fresh

outlook on the field of combustion research. As an early milestone in the field, Kalogirou et al.[9] reviewed artificial intelligence techniques in combustion control and modeling, establishing the foundation for subsequent data-driven approaches. That early work emphasized traditional methodologies, including expert systems, genetic algorithms, and neural networks [9,10].

The recent advancements in ML have led to the emergence of various innovative and resilient data-driven models. Consequently, it is imperative to conduct a comprehensive examination of the most recent combustion research, considering the utilization of state-of-the-art machine learning models. Ding et al.[11] used machine learning to identify turbulent combustion thermochemistry in equilibrated low swirl burners using methane, and Seltz et al.[12] defined a new modeling framework to resolve the transport equation. To identify the critical role of mixture fraction in the process and to identify essential radicals involved in branching reactions and soot formation, Zdybal et al.[13] analyzed data obtained in a 3D temporally evolving DNS of an n-heptane turbulent jet in the air. To predict soot production, Kessler et al.[14] compared neural networks, graph networks, and multivariate equations. Using ANN, Nguyen et al.[15] calculated combustion chemistry, temperature, radicals, intermediate species, and all stages of evolution (mixing, combustion, equilibrium), including radicals, OH, O, and H₂O.

1.2. Flamelet Model

A flamelet-based method, flamelet generated manifold (FGM), describes turbulent flames as ensembles of representative one-dimensional flamelets. In the FGM model, the flamelets are pre-computed and subsequently stored. A low-dimensional manifold is used to calculate the diffusion and convection effects of 1-D laminar flames [16]. Control variables (CVs) space is created by transforming from the space-time domain (x-t) to mixture fraction and progress variable [17]. The connection between the tabulated data and the flow field is established by incorporating either all

the control variables or a selected subset of them. The implementation of this preprocessing technique leads to a reduction in computational expenses without compromising the accuracy of the results. Large mechanisms can be adopted by employing rigid formulations that are not reliant on fluid dynamic calculations.

1.3. ML in Flamelet

Flamelet-based models use ML techniques to model complex non-linear data. ML methods require less memory compared to the FGM model because they only need to store the architecture characteristics and parameters [19]. Zhang et al.[20] effectively utilized ANN to model the flame of a methane-air mixture burner using FGM model. The efficacy of this methodology was evaluated by conducting a comparative analysis between the outcomes of numerical simulations and empirical observations. The utilization of ANNs in conjunction with flamelet-based models has been found to yield notable reductions in memory usage. Nonetheless, it is important to note that these models possess limited representation capabilities and necessitate extensive training durations, as mentioned in the study of Hansinger et al.[21]. In research conducted by Ihme et al.[22] a computational fluid dynamics (CFD) investigation was carried out on burner fueled by combination of methane and hydrogen. Instead of utilizing flamelet-progress variable (FPV) look-up tables, the researchers opted for the implementation of classical ANN. To enhance the efficiency of the neural network architecture, the researchers proposed the utilization of an optimization technique to determine the optimal number of layers and neurons.

Several optimization methods were used to optimize engine control parameters, including least squares support vector machines (LS-SVM), particle swarm optimization (PSO), and genetic algorithms (GA). A comparison of the GA-SVR model's performance with standard SVR models,

random forest models, and back propagation neural network models was performed by Guo et al.[33] Based on the results, it can be concluded that GA-SVR is more accurate and generalizable.

Advantageous in terms of memory performance compared to the complete replacement of look-up tables. However, the integration of ML models with look-up tables can pose challenges in both numerical implementation and practical usage. Bissantz et al.[34] employed a dense neural network (DNN) to substitute table-based outcomes and a sparse principal component analysis (SPCA) to ascertain control variables (CVs) in their computational fluid dynamics (CFD) simulation of flame quenching in premixed methane-air systems. The FGM-DNN model incorporated a progress variable and temperature as control variables (CVs). Although the runtime and prediction accuracy exhibit comparable performance, a significant reduction in memory usage of up to 98% can be achieved. The performance of the model is significantly impacted by both computer hardware and DNN architecture. In their study, Mousemi et al. [35] employed a decision tree (DT) and an ANN model in conjunction with a tabulated chemistry approach to replicate premixed methane-air flames in a Bladed Ring Clearance burner. The utilization of DT and ANN models resulted in a reduction of memory requirements by 40% and 92% correspondingly. In addition, the decision tree model exhibited superior computational efficiency. However, it should be noted that DT models are susceptible to the issues of overfitting and entrapment.

RF and DT models are ML techniques that have the potential to reduce training time and reveal intricate relationships among parameters in combustion systems [36]. Based on prior research, it has been observed that DT exhibit superior performance compared to RF owing to their greater flexibility in parameter adjustment [37]. Recent research has also examined their application in combustion modeling. With RF and ANN, Ren et al. [38] measured flame structure properties in which the RF model outperformed ANN in three DNS cases with different turbulence intensities.

Natural gas engine output parameters were forecast using RF and ANN models by Liu et al.[29]. Both methods perform comparably, but ANN models can be challenging to tune because of their complex parameters. In their first attempt to predict non-linearity between combustion parameters, Yao et al.[30] employed ensemble models to investigate the relationship between input variables, such as species concentrations, pressure, and temperature, and output variables, namely species compositions at subsequent times. The study revealed that DT exhibited superior performance compared RF in predicting species compositions and temperatures, on par with ANN.

According to research on machine learning in fluid mechanics and combustion, much of the research has focused on using data extracted from classical and traditional methods. ML models each have advantages and disadvantages. ANN models, for example, have longer training and more parameters to tune [31]. Overfitting and local optimum can happen with DTs [32]. Models such as RF and LR exhibit a relatively lower level of complexity during the training process, albeit at the cost of reduced accuracy and increased memory consumption [33]. In this study, four different ML methods are compared in FGM simulations. The systematic benchmark is conducted: (1) four ML methods (MLP, RF, LR, SVR) are compared under identical training conditions with scalar dissipation rate used as a continuous input alongside mixture fraction; (2) all models are evaluated on unseen scalar dissipation rate values to assess genuine generalization; and (3) cross-validated hyperparameter optimization identifies the optimal configuration.

2. Governing Equations

Combustion is a mass and energy conversion process in which chemical energy is converted into thermal energy. Fuel combines with atmospheric oxygen to generate carbon dioxide and water vapor, which have a lower enthalpy of production. Combustion is the primary energy source for

transportation, heating, and electricity generation. The main goals of combustion research are to reduce emissions and increase the efficiency of systems since our fossil fuel resources are limited. To simulate the temperature and combustion species, it is necessary to solve the combustion equations. In this section, the definitions of thermodynamic variables and then the governing equations are presented. The governing equations of the Flamelet model and machine learning are also given briefly.

2.1. Flamelet Equations

Non-premixed combustion occurs when the fuel and the oxidizer are injected separately into the combustor and experience simultaneous mixing and burning. The Flamelet methodology utilized in non-premixed combustion is founded upon the concept of characterizing the turbulent flame as an assemblage of laminar flame constituents that are integrated within a turbulent flow and engage in mutual interactions. The flame's local structure at every point along the flame front is expected to exhibit similarities to a laminar Flamelet, with the influence of turbulence limited to the evolution of the front.

The prevailing consensus is that the Flamelet concept holds true within the high Damköhler number (Eq. (1)), where the chemical time scale τ_c is compared to the flow time scale τ_t :

$$Da = \frac{\tau_t}{\tau_c} \quad (1)$$

In contrast to the turbulent time scale, a high Damköhler number (Da) denotes rapid chemical reactions. In this situation, the chemical reactions exhibit a rapid response to alterations in flow, thereby rendering unsteady effects negligible. Another requirement is that the thickness of the flame should be sufficiently small compared to the turbulent length scales in order to prevent

vortices from disturbing the internal structure of the flame. Despite the rapid nature of the chemistry involved, specifically the presence of a thin reaction layer, the diffusive flame thickness is comparatively larger and can potentially be influenced by the flow conditions.

One possible approach involves performing a conversion from the physical spatial-temporal domain to a new domain where the mixture fraction is introduced as an additional independent variable. Consequently, the newly established coordinate system is closely associated with an iso-surface of the mixture fraction, specifically the stoichiometric mixture fraction denoted as Z_{st} . The coordinates Z_τ and Z_r are situated within this iso-surface. The application of a Crocco-type coordinate transformation is carried out, wherein the transformation rules are methodically employed to the governing equations pertaining to species and energy. At this juncture, it is postulated that the flame front exhibits a unidimensional behavior. The terms associated with flame front gradient, specifically Z_τ and Z_r gradients, are disregarded to the normal flame gradients. The flamelet equations can be expressed as follows (Eqs. (5) and (6)) by considering $Le_i = \infty$ and negligible Soret effect.

$$\rho \frac{\partial Y_i}{\partial \tau} = \frac{\rho \chi}{\tau Le_i} \frac{\partial^2 Y_i}{\partial Z^2} + \frac{1}{\tau} \left(\frac{\lambda}{Le_i} - 1 \right) \left[\frac{\partial \rho \chi}{\partial Z} + \rho \chi \frac{c_p}{\lambda} \frac{\partial}{\partial Z} \left(\frac{\lambda}{c_p} \right) \right] \frac{\partial Y_i}{\partial Z} + \dot{\omega}_i \quad (2)$$

$$\rho \frac{\partial T}{\partial \tau} = \frac{\rho \chi}{\tau} \frac{\partial^2 T}{\partial Z^2} + \frac{\rho \chi}{\tau c_p} \frac{\partial c_p}{\partial Z} \frac{\partial T}{\partial Z} + \frac{\rho \chi}{\tau c_p} \sum_{i=1}^N \left(\frac{c_{p_i}}{Le_i} \frac{\partial Y_i}{\partial Z} \right) \frac{\partial T}{\partial Z} - \frac{1}{c_p} \sum_{i=1}^N h_i \dot{\omega}_i + \frac{Q_R}{c_p} \quad (3)$$

where τ , T , Q_R , c_{p_i} , h_i , $\dot{\omega}_i$, Y_i , $Le_i = \lambda / \rho D_{im} c_p$, and D_{im} are time, temperature, radiating heat loss, i -th specific heat, enthalpy, net rate of production, mass fraction, Lewis number, and multicomponent ordinary diffusion coefficients, respectively. In these equations, χ is the instantaneous scalar dissipation rate defined by:

$$\chi = \chi D_z (\nabla Z) \quad (4)$$

Eqs. (4) and (5) are commonly simplified by assuming that certain terms can be neglected. Specifically, in Eq. (4), the second term on the right-hand side is considered negligible. Similarly, in Eq. (5), both the second and third terms on the right-hand side are assumed to have negligible impact. Given these simplifications, the equations can be expressed in the following manner:

$$\rho \frac{\partial Y_i}{\partial \tau} = \frac{\rho \chi}{\chi Le_i} \frac{\partial Y_i}{\partial Z} + \dot{\omega}_k \quad (5)$$

$$\rho \frac{\partial T}{\partial \tau} = \frac{\rho \chi}{\chi} \frac{\partial T}{\partial Z} - \frac{1}{c_p} \sum_{i=1}^N h_i \dot{\omega}_i + \frac{Q_R}{c_p} \quad (6)$$

For the energy Eq. (6), although the term that involves the Z-derivative of the heat capacity is sometimes retained.

By utilizing a chemical mechanism, establishing suitable boundary conditions, and incorporating scalar dissipation rate profiles $\chi(Z)$, the flamelet equations (Eqs. (4) and (5) for combustion, and Eqs. (6) and (7) for species) can be numerically solved to derive equations for the determination of species mass fractions (Eq. (5)) and temperature distributions (Eq. (6)):

$$Y_i = Y_i(Z, \chi) \quad (7)$$

$$T = T(Z, \chi) \quad (8)$$

The fundamental principle of the Flamelet approach is predicated upon the utilization of a variable transformation, denoted as $(x, t) \rightarrow (Z, \chi)$. In the event that the dynamics and flame structure become decoupled, the comprehensive depiction of the temperature and species profiles'

spatial and temporal evolutions can be achieved solely through the spatial and temporal evolution of Z .

3. Materials and Methods

3.1. Machine Learning Equations

As ML methods mature and become more available, the best fit for specific problems can be assigned to supervised learning algorithms. Algorithms are chosen based on the size and nature of the data. Depending on the quantity of available training data, various principles of thumb for choosing proper ML method have been presented in Table 1. The most common machine learning method in combustion is supervised learning. In supervised learning are inferred as machine learning models from labeled data sets that map inputs to outputs. Fitting thermodynamic response functions can be done with regression techniques on continuous outputs. Using supervised learning, model parameters are learned from data by minimizing a loss function Eq. (9).

In this study, the input feature vector comprises two continuous variables: mixture fraction Z and scalar dissipation rate χ . This formulation allows each ML model to learn the joint thermochemical response $T(Z, \chi)$ and $Y_i(Z, \chi)$ across the full operating space, rather than treating χ as a discrete library index. All libraries are combined into a single training dataset, with $[Z, \chi]$ as inputs and thermochemical quantities as outputs.

$$\arg_{\theta \in \mathcal{P}} \max E(Y, f(X, \theta)) \quad (9)$$

where E is error, X is the input, Y the output, $\mathcal{P}$ all model parameters, and the implicit dependence of f function is investigated on model parameters θ .

Table 1. Model Selection Guidelines for ML Methods.[33,34]

Rule	Description
Start with simplicity	In general, simpler models tend to generalize better than complex ones. Before moving on to more complex models, start with simple models such as linear regression.
Consider the size of the dataset	In order to prevent overfitting, choose models with low complexity, if your dataset is small. In contrast, a large dataset can allow you to explore more complex models.
Understand the problem and the data	Gain a deep understanding of the problem and data you are analyzing. You can use this knowledge to select models appropriate for the particular problem domain and data type (for example, classification, regression, structured, or unstructured).
Evaluate model assumptions	A model's assumptions about the underlying data distribution and relationships need to be evaluated. Ensure that your chosen model is aligned with your dataset's assumptions. The linear regression method, for instance, assumes a linear relationship between the features and the target variable.
Accuracy or interpretability	It is important to provide more transparent explanations of predictions, such as linear models or decision trees, if interpretability and understanding are important. The interpretation of complex models like neural networks may be more challenging.
Computational resources	To train and deploy some models, significant computational resources (e.g., memory, processing power) are required. Ensure the models you choose can be implemented on the available computing resources.

Model architecture, feature set, and optimization method for selecting model parameters differ between supervised learning algorithms. Throughout this article, four different methods are used to recreate Flamelet libraries, which are briefly discussed below.

3.1.1. Linear Regression using Stochastic Gradient Descent

The stochastic gradient descent (SGD) method is a simple, yet efficient, approach to fitting linear classifiers and regressors with convex loss functions, such as support vector machines (linear) and logistic regressions. While linear regression (LR) (Figure 1 **Error! Reference source not found.**) has existed in the machine learning community for a long time, it has recently gained significant attention in large-scale learning. An SGD is not a machine learning model, and it is only a method for training a model. It is merely a technique for optimization of Linear models [36].

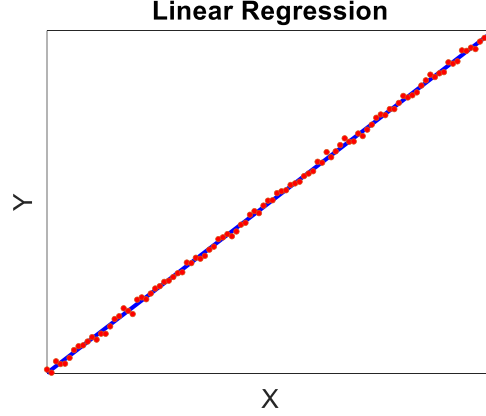


Figure 1. Linear Regression prediction

A Linear Regression learning routine is implemented, which supports a variety of loss functions and penalties for fitting linear regression equations. The model parameters θ are determined by minimizing an error function (Eq. (10)).

$$E = \frac{1}{n} \sum_{i=1}^n (Y_i - f(X_i, \theta))^2 \quad (10)$$

3.1.2. Decision Trees using Random Forest Regressor

The decision tree is a model of possible outcomes that uses a flowchart-like structure to facilitate decision-making. Decision-tree algorithms are supervised learning algorithms. Categorical and continuous output variables can both be calculated using this method. Every decision tree has high variance, but when we combine them in parallel, the resultant variance is low because each decision tree is perfectly trained on the sample data. As a result, the output depends on more than one decision tree. For regression problems, the final output is the mean of all the outputs called Aggregation step. To form sample datasets for each model, Bootstrap is used to randomly sample rows and features from the dataset [30].

Using Bootstrap and Aggregation, also known as bagging, Random Forest is a method of performing regression and classification tasks. This involves combining multiple decision trees instead of relying solely on one decision tree to determine the final output (Figure ۲). A Random Forest consists of multiple decision trees as its base learning model.

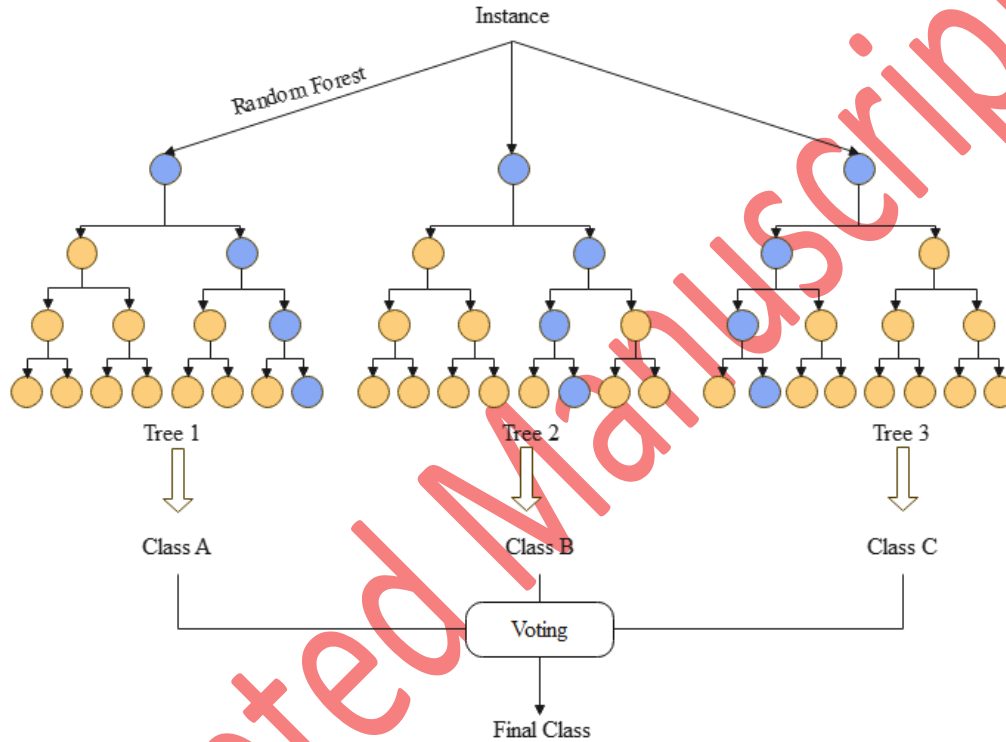


Figure 2. Schematic of a Random Forest algorithm with 3 decision trees

۳.۱.۳. Support Vector Machine

The SVM is a deterministic algorithm that is used to create decision boundaries in datasets that are linearly separable. As depicted in **Error! Reference source not found.**, the decision boundaries (Eq. (۱۱)) are established by determining the utmost distance between points and the decision boundary, also known as support vector points. SVMs employ hyperplanes as decision boundaries for the purpose of data classification and regression [۳۷].

$$\sum_{i=1}^M (w_i X_i + b_i) = \cdot \quad (11)$$

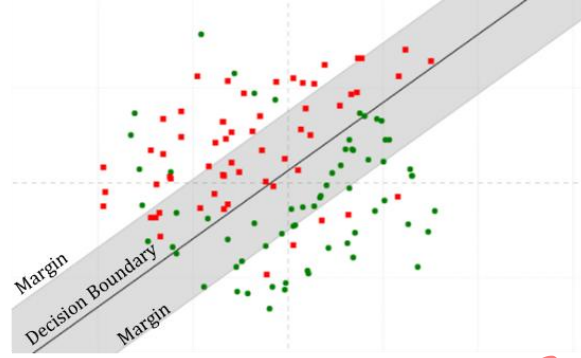


Figure 3. SVM For a binary classification

where $w \in \mathbb{R}^M$ representing the weights of the normal vector of hyperplane, and b is the bias coefficient. The margins, which refer to the hyperplanes can be mathematically represented as Eq. (12):

$$\sum_{i=1}^M (w_i X_i + b_i) = \pm 1 \quad (12)$$

3.1.4. Neural Networks using Multi-Layer Perceptron (MLP)

Logistic regression cannot replicate non-linear decision boundaries, which limits its applicability to problems requiring complex physics-based classification. It is advisable to employ more dynamic algorithms for the applications. MLP or fully connected feedforward neural network is characterized by a configuration of neurons, also known as individual reasoning regression units, organized in a hierarchical arrangement of layers [34]. This structure is depicted in Figure 4.

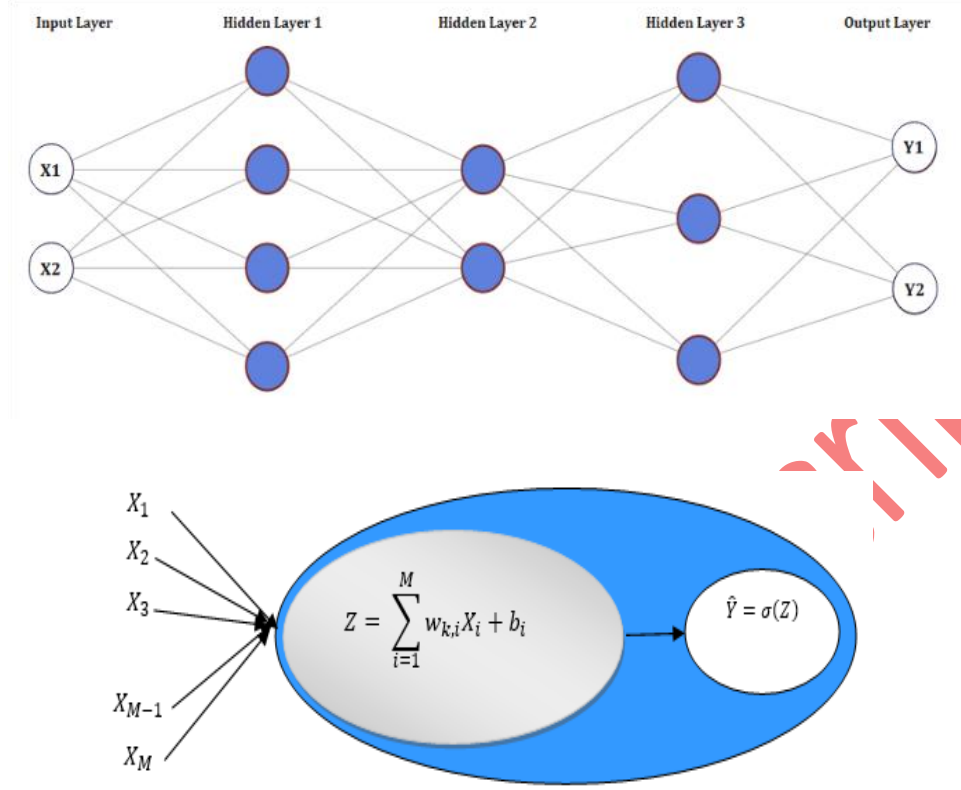


Figure 4. Schematic of an MLP. (a) Network architecture, (b) Neuron and its operations.

$$\hat{Y} = \sigma(Z) \quad \text{while} \quad Z = \sum_{i=1}^M w_{k,i} X_i + b_k \quad (13)$$

σ is the neuron activation function. In MLP, the outputs of the previous layer are utilized as features for each neuron in the subsequent layer. MLP establishes a mapping from input to output. Researchers have developed specialized neural networks to address needs specific applications.

3.2. Comparison of applied methods

In this study, the aforementioned four methods have been employed for the purpose of predicting temperature and species mass fraction. Based on the investigations, each method has its own positive and negative points, which are mentioned in Table 2.

Table 2. Advantages and disadvantages of applied methods

Model	Advantages	Disadvantages
Linear regression	• High accuracy • Ease of implementation	• Need for hyperparameters such as setting and repetition parameters • Sensitive to feature scaling
Random forest	• High versatility • High accuracy and flexibility • Effective way to estimate missing data	• Time-consuming to calculate the data for each tree • It requires a lot of computing and storage resources • The possibility of overfitting in a small number of trees
Support vector machine	• Effective in large spaces • Effective in cases where the number of dimensions is greater than the number of samples • Versatile due to kernel functions • Ability to specify custom cores in addition to common cores	• They do not directly provide probability estimates. • Computationally expensive • Not as easily interpretable as some other algorithms
Multilayer perceptron	• Ability to learn nonlinear models • Ability to learn models in real time (online learning)	• The initial architecture needs to be set. • Sensitive to feature scaling

In order to obtain high-accuracy data through the machine learning model for the present study, it is necessary to choose the most optimal method, then use different architectures to increase the model's accuracy so that the data satisfies the conditions of the problem well.

4. Numerical Method

In this research, the primary data are obtained by solving the Flamelet equations using FlameMaster. Training data for the ML models were generated at $20 \times$ grid points per library; a resolution check confirmed temperature and key species mass fractions differ by less than 1% between $20 \times$ and $40 \times$ grid points across all scalar dissipation rate values, establishing $20 \times$ points as converged. Note that this training data resolution study is distinct from the counterflow validation grid study in Section 3.1. The mixture fraction (Z-flamelet) approach uses FlameMaster

(an open-source C++ program package for 0D combustion and 1D laminar flame calculations) [39]. In both methods, a structured grid with focus on flame are discretized with central method. The hardware used for this research comprised an Intel Core i7-8850K (3.3 GHz, 8 cores, 16 threads), 16 GB DDR4 RAM, and 16 MB cache.

3.1. Algorithm configuration

The output data format extracted, is “.fla”, which cannot be read in Python. For this reason, a code in C++ was developed to read this data and present the file as a data set in “.csv” format with respect to their scalar dissipation rate. After transferring the data to Python, all libraries were combined into a single dataset. The scalar dissipation rate χ is included as a continuous input feature alongside the mixture fraction Z , yielding a two-dimensional input vector $[Z, \chi]$ for each data point. This formulation enables the ML models to interpolate between trained conditions and to learn the functional dependence of thermochemical quantities on both control variables simultaneously. Different machine-learning models were then evaluated on this unified dataset. After checking the output data by each model, the accuracy and error of each model are calculated; so that the optimal initial model can be selected. The method with maximum accuracy in predicting temperature and different species is selected as the main method for optimization.

After choosing the optimal method, the output data from the method were compared with other data solved by Flamelet equations to validate the reproduced data, and maximum errors will be computed. In cases with low accuracy, the model's architecture is changed to satisfy the desired conditions for prediction. After satisfying all the conditions, the optimal model and its reports are extracted for further investigation (schematic shown in Figure 9).

2. Results and Discussion

To use machine learning in developing the flamelet combustion model library, first need to create a library using this method; so that the necessary data for inputting into the machine learning algorithm is provided. In creating the Flamelet model library, the conventional steady counterflow flame configuration is commonly used, and the library takes shape by changing the distances between fuel and oxidizer. Obviously, in the first step, the accuracy of the generated data needs to be verified. Therefore, in this section, the accuracy of the non-premixed flame is first validated, and then the library is formed by changing the scalar dissipation rates. Finally, these data are used and reproduced by various machine learning methods.

2.1. Validation

2.1.1. Problem Definition

It is necessary to evaluate the accuracy of the modeling after examining the number of mesh grids required to form a Flamelet library. There are experimental and numerical results presented in for the cross-flow of methane fuel injected at a distance of 1.476 cm with a velocity of 76.8 cm/s at a temperature of 300.15 K and air with a velocity of 73.4 cm/s at a temperature of 300 K. Consequently, a one-dimensional cross-flow was modeled. (Figure 6)

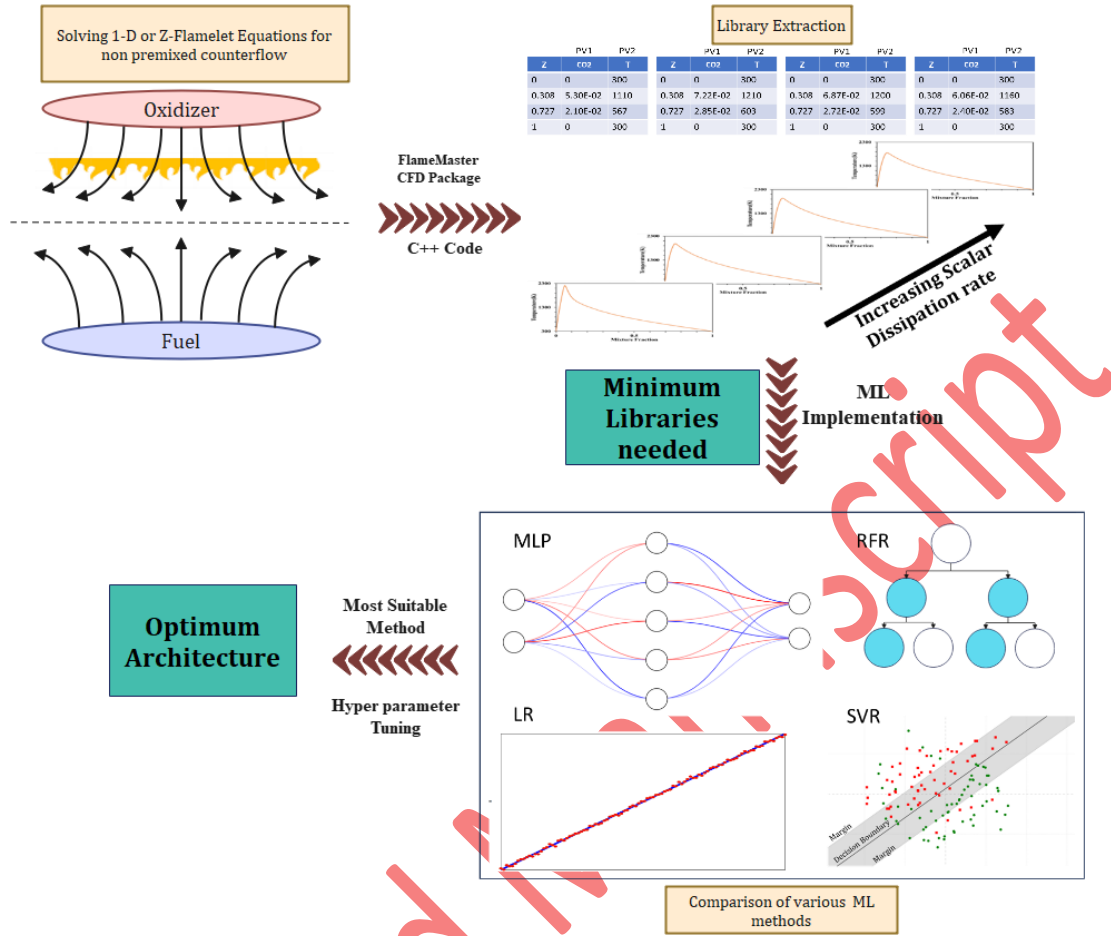


Figure 5. Procedure flowchart of the present study.

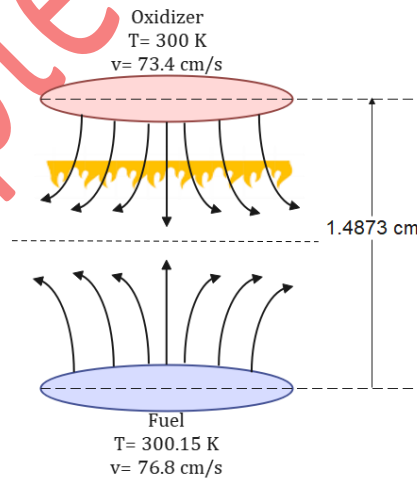
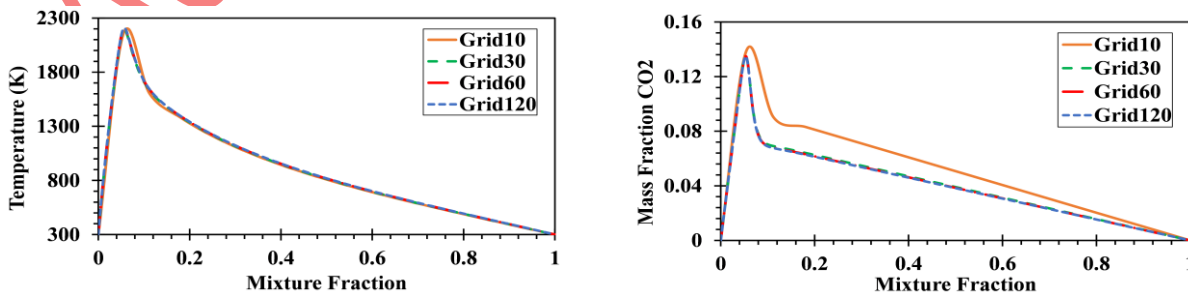


Figure 6. Schematic of computational domain and boundary conditions

3.1.2. Mesh Study

To validate the physical fidelity of the FlameMaster solutions used as training targets, the present modeling was compared with the experimental and numerical results of Smooke et al. [39] for the counterflow methane combustion. Note that this grid independence study applies to the counterflow validation flame configuration and is distinct from the training data discretization study described in Section 4. This experiment reported the temperature and mass fractions of species such as CO_2 and OH at a pressure of 1 atm. First, the simulation results were validated by comparing them with results in Smooke et al.'s [39] study. Then the required library for the Flamelet method was prepared, solving Flamelet equations for a non-premixed counterflow with different lengths was done. However, before forming the library, it is necessary to investigate how many grids are needed in one-dimensional space. Four networks of 10, 30, 60, and 120 grid points were investigated, and the temperature and mass fractions of CO_2 and OH in these four networks were compared. As shown in Figure 5(a), the results of the four networks do not have a significant temperature difference, and network independence is achieved. However, for the results of the mass fraction of CO_2 and OH shown in Figure 5 (b) and (c), the results of the three networks of 30, 60, and 120 grid points are consistent with each other; but the results of the network containing 10 points are different from them. Therefore, a 30-point network was used in the formation of the library.



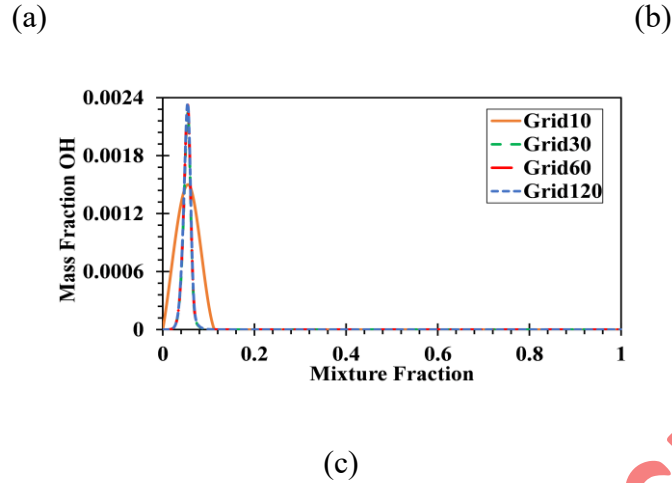


Figure 7. Grid independency (a) Temperature (b) CO₂ (c) OH

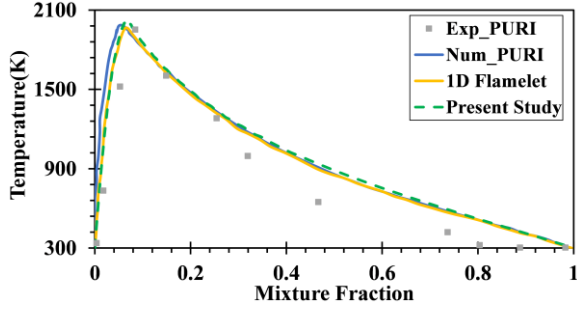
3.1.3. Comparison of flamelet equation and experimental data

As shown in Figure 4, the temperature results are consistent with the modeling results in [39]; however, they deviate slightly from the experimental results. In the experimental results, the maximum temperature value in the fuel fraction is approximately 2110, whereas, in the modeling results, it is obtained as 2115.

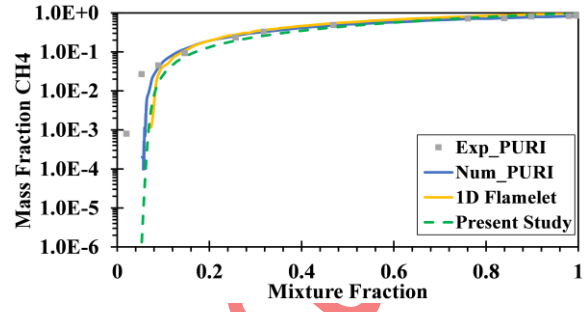
The results of species mass fractions show a more significant deviation from experimental results with respect to temperature; but are more consistent with simulation results (Figure 4). There is a slight difference in experimental results and our study in the mixture fraction of 0.1. Regarding the experimental results and numerical results for the N₂ species, there is a significant overlap between experimental and numerical results with little difference observed.

Based on Figure 4, it can be seen that the results of species mass fractions for combustion products such as CO and CO₂ are predicted with less accuracy compared to the primary combustion species (methane and oxygen). Agreement is also observed for H₂O and H₂ species with numerical results. However, for H₂O species, results deviate from experimental data for mixture fractions greater than 0.26. This deviation of H₂O starts at mixture fraction of 0.19 for

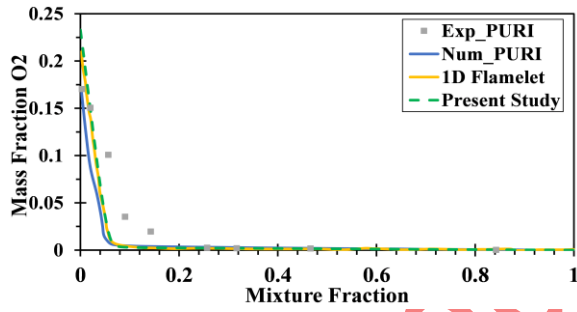
0.26 and eventually reaches 0.267 for 0.28. This deviation occurs for H_2 species at a mixture fraction of 0.27, equal to 0.066. Therefore, the present approach computes temperature and species accurately.



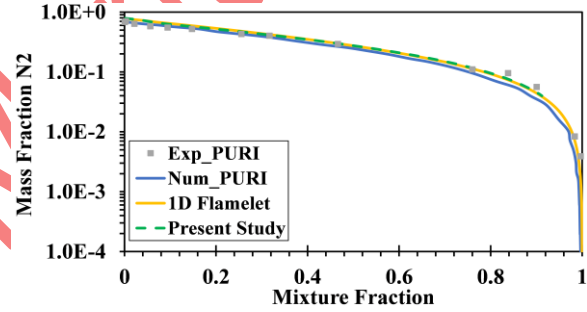
(a)



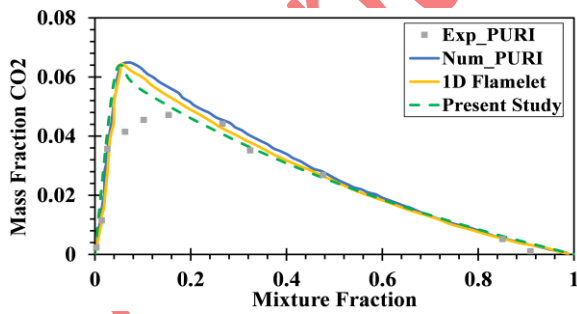
(b)



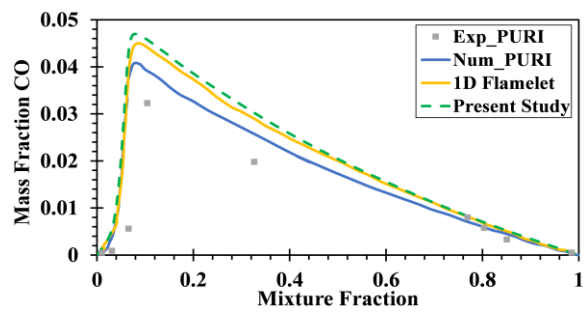
(c)



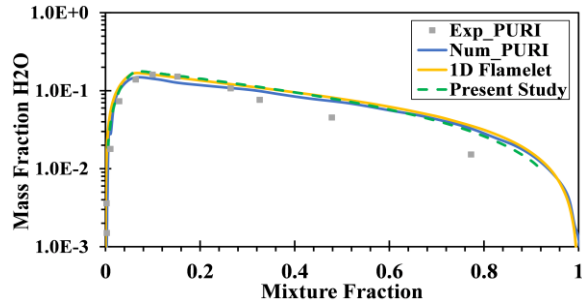
(d)



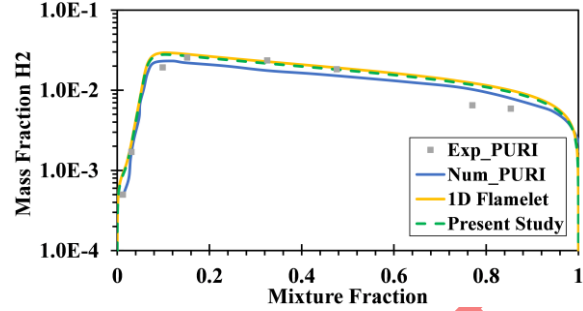
(e)



(f)



(g)

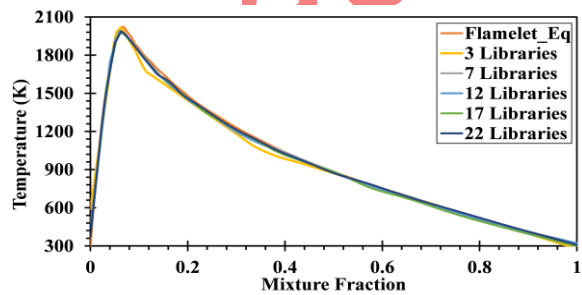


(h)

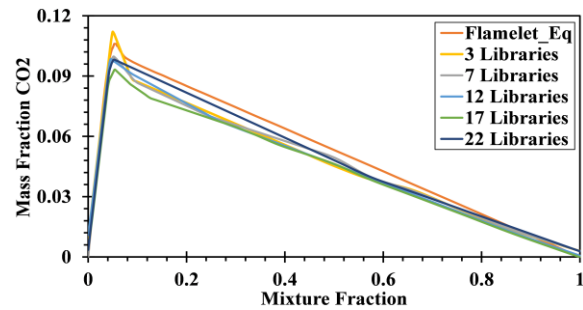
Figure 8. Comparison of numerical and experimental results of, 1D Flamelet equation and this study (a) Temperature (b) CH4 (c) O2 (d) N2 (e) CO2 (f) CO (g) H2O (h) H2.

3.2. Minimum number of required libraries

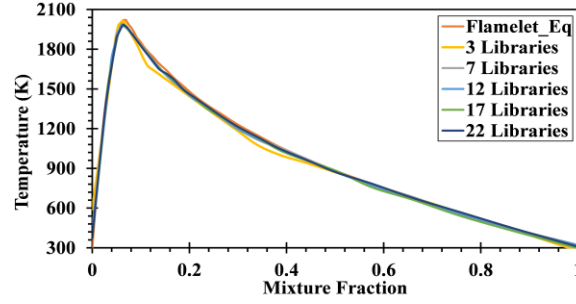
The second step investigated in this paper is the independence of a machine-learning model from the number of libraries used for training. To build a dataset for training a machine learning model, there are various ways and conditions to determine the minimum required number of libraries that will reduce computational costs. For these models to be reliable, their accuracy with the least number of libraries should be acceptable. Therefore, as shown in **Error! Reference source not found.**, various models were created using 3, 7, 12, 17, 22, and 27 libraries to compare temperature, CO, and OH species, each library containing 100 data.



(a)



(b)



(c)

Figure 9. Minimum library needed for forming dataset required for training machine learning models

The results for temperature, CO_v , and OH mass fractions for the different numbers of libraries are presented in Figure 9, where the scalar dissipation rate is 0.1 (1/s) . The results show that all models have an average error of less than 4% for temperature and less than 10% for CO_v compared to the results extracted from the Flamelet equation according to Table 3. Moreover, as the number of libraries increases, the results better conform to those extracted from the Flamelet equation. Significant accuracy can be obtained for temperature predictions even with only three libraries. However, for better predictions, a higher number of libraries is required. Considering all the conditions, including errors and the trend of changes in Table 3, 22 libraries was selected. Subsequently, all methods are trained using 22 Flamelet libraries with different scalar dissipation rates (χ) between 0.001 to 29.0 (1/s) , as sorted in Table 4 and shown in Figure 10. Errors in Table 3 are reported as percentage deviation from FlameMaster reference values and are used solely for library selection. Full R^2 , RMSE, and MAE metrics for the final trained models are reported in Table 6, Table 10, and Table 11.

Table 3. Maximum, minimum and average error of each model formed by different number of libraries

Number of libraries	Maximum error (%)		Minimum error (%)		Mean error (%)	
	Temperature	CO_v	Temperature	CO_v	Temperature	CO_v

3	49,6	20,06	0,11	1,13	4	7,17
7	32,19	14,24	0,04	3,09	2,30	8,82
12	28,84	19,49	0,07	2,26	2,23	9,01
17	18,69	24,94	1,43	9,10	2,06	10,08
22	20,19	14,04	0,02	1,62	1,69	6,00

Table 4. Seven libraries and scalar dissipation rates

Scalar Dissipation Rate	χ_1	χ_2	χ_3	χ_4	χ_5	χ_6	χ_7
Value	0,01	0,0	10	14,0	20,0	20	29,0

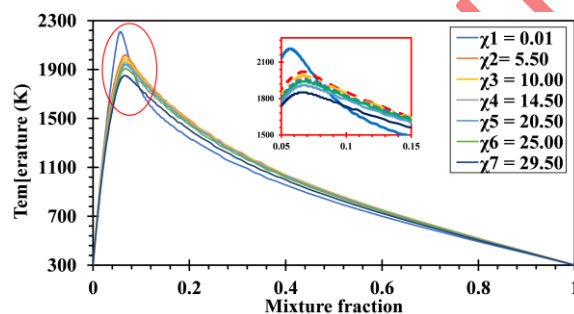


Figure 10. Temperature difference for different scalar dissipation rates

5.3. Comparison of different ML algorithms

Data from the Flamelet library is reproduced by using various algorithms. To reproduce the data, four different methods were used with different approaches. The MLP method represent a neural network, a linear model by the LR method, a decision tree by the RFR method, and a hyperplane by the SVR method. Different models were found to have different accuracies when examining the initial data. The SVR was excluded from the quantitative comparison. With an RBF kernel, SVR training scales as $O(n^2)$ in memory and $O(n^3)$ in computation, making it impractical for multi-output thermochemical regression. Without extensive regularization tuning it produces non-

physical predictions including negative mass fractions. A fully tuned single-species SVR is suggested for future work. Results for the remaining three methods are shown in Figure 11.

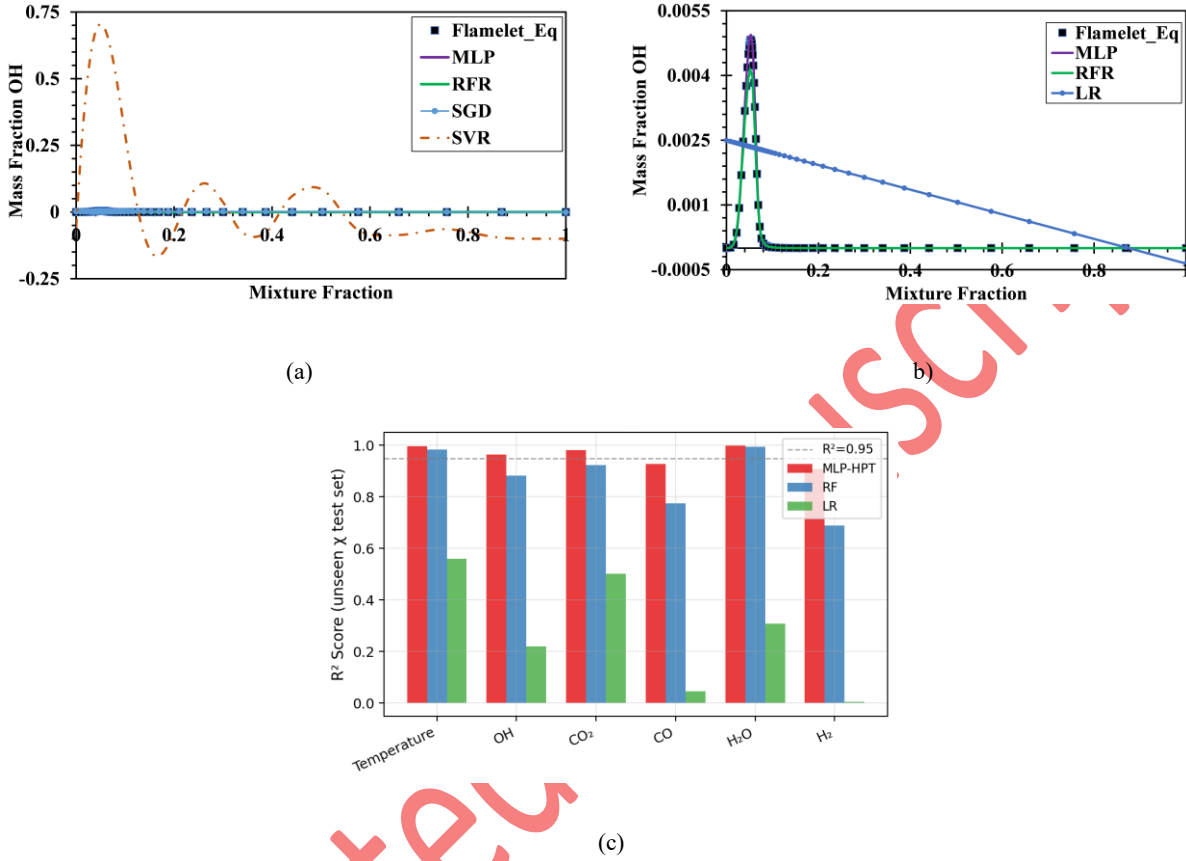


Figure 11. Comparison of ML methods on unseen scalar dissipation rate conditions ($\chi = 4.0 \text{ s}^{-1}$): (a) considering the SVR algorithm (b) OH mass fraction predictions by MLP-HPT, RF, and LR versus FlameMaster reference (SVR excluded), (c) R^2 scores by species and model across the full unseen χ test set.

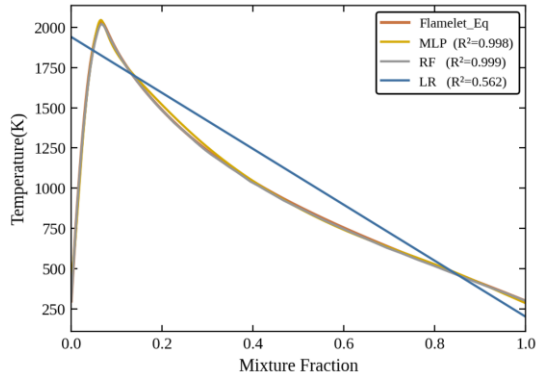
In order to anticipate the OH species mass fraction, the three models and their results are compared to the Flamelet equations as shown in **Error! Reference source not found.** Additionally, the MLP algorithm provides more precise predictions than prior algorithms in this species. MLP is accurate in predicting values as well as trends. On unseen scalar dissipation rate conditions, the MLP, RF, and LR methods achieve OH species R^2 values of 0.9647, 0.8826, and 0.2210 respectively, with corresponding RMSE values of 3.07×10^{-4} , 6.02×10^{-4} , and 1.68×10^{-3} .

The accuracy and behavior of the chart relating to the LR model shows that this model does not act appropriately in species.

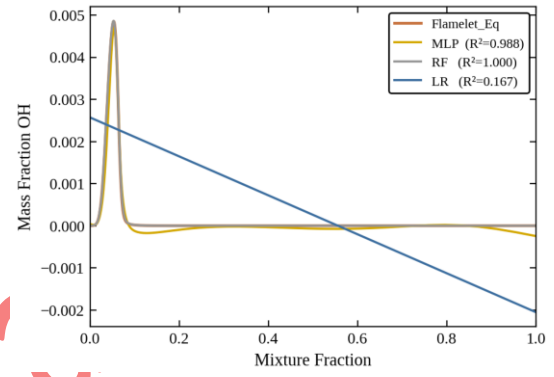
Various models demonstrate different performance in predicting other species, such as temperature and OH mass fraction. MLP, RFR, and LR predict CO mass fraction with R^2 values of 0.9269, 0.9742, and 0.0472 respectively on unseen conditions, while for CO₂ species, they demonstrate higher levels of accuracy at 97.96%, 91.01%, and 80.90%, respectively. Irrespective trend prediction in LR for different species, rules out this method from consideration. This method has the lowest accuracy among 7 other methods and predicts positive parameters at negative rates.

The MLP and RFR models demonstrate a high degree of unity with the numerical data obtained by calculating Flamelet equations in their ability to predict H₂O species. The predictive performance of MLP and RFR, and LR for H₂O species identification is reported as 97.80%, 98.78%, and 76.22%, respectively. However, it should be noted that the LR model demonstrates unsuitable results. The average accuracy of the RFR approach has been enhanced due to its superior precision when dealing with lower mixtures fractions. The performance of the MLP model exhibits superior accuracy and trend consistency compared to the other models in the H₂ species. **Error! Reference source not found.** depicting the H₂ species reveals distinct variations among the models. The LR method exhibits the least suitability, and the RFR model displays a considerable margin of error for mixture fractions comprising less than 0.1. Apart from the flaw mentioned above, the RFR model exhibits a degree of noise for mixture fractions exceeding 0.1, thereby making it unsuitable as the optimal option. The MLP model accurately predicted the H₂ mass fraction behavior. Hence, based on the obtained results, the MLP model exhibits the highest accuracy of 97.30%, rendering it the most favorable option.

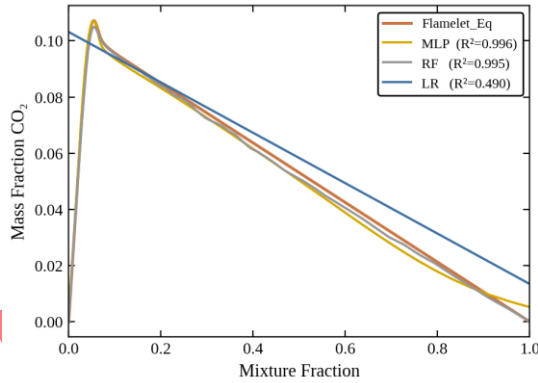
RF achieves a mean R^2 of 0.9109 and ranks second; it performs well on major species but shows larger errors for H_2 ($R^2=0.7881$) and CO ($R^2=0.7753$). LR achieves a mean R^2 of only 0.47 and fails on most minor species, confirming that linear models are inadequate for this nonlinear thermochemical regression task. Based on these results, the MLP model is selected as the primary method for further optimization.



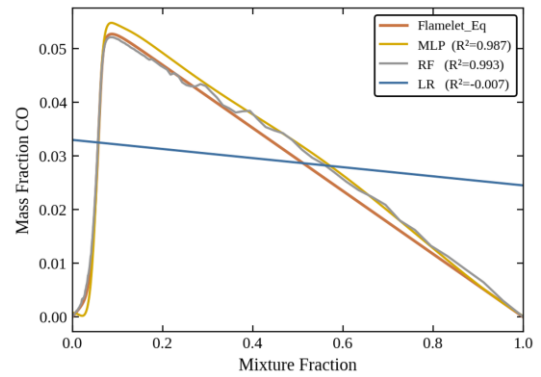
(a)



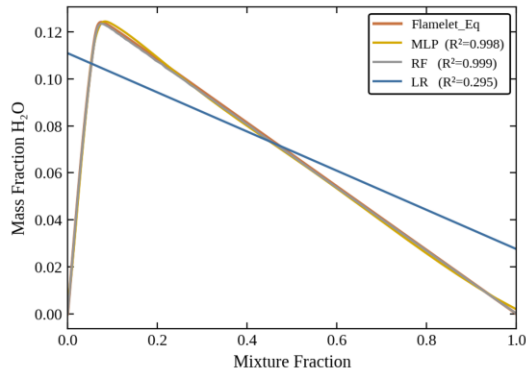
(b)



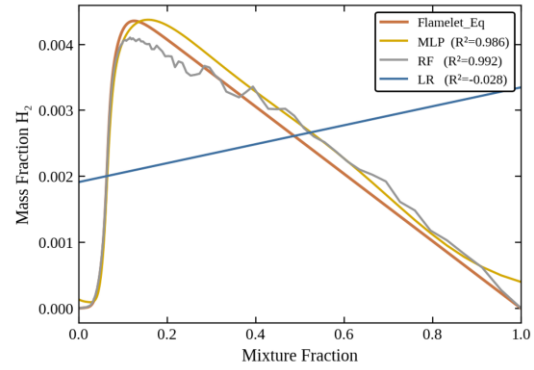
(c)



(d)



(e)



(f)

Figure 12. Comparison of MLP-HPT, RF, and LR predictions against FlameMaster reference solutions at an unseen scalar dissipation rate ($\chi = 4.0 \text{ s}^{-1}$): (a) temperature, (b) OH, (c) CO_2 , (d) CO, (e) H_2O , and (f) H_2 mass fractions as functions of mixture fraction Z . R^2 values shown in legend are computed on the full unseen χ test set.

4.4. Different architectures of MLP

Machine learning models provide different accuracies in predicting different phenomena, and this accuracy can vary depending on the nature of the data. In other words, one model may justify and predict a phenomenon in a way that another model cannot achieve the same level of accuracy. In addition to the overall difference in accuracy, the accuracy in each predicted parameter is also not the same. Suppose a model can achieve a general accuracy ($\sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}}$) of 96%. The level of precision can be subject to fluctuations contingent upon the dataset's characteristics and the extent of numerical values involved.

Table 9 presents five distinct models utilizing varying architectures of the MLP algorithm, each exhibiting differing levels of accuracy. Each of the models mentioned above exhibits varying levels of accuracy when it comes to predicting temperature and species. Table 9 aims to determine the optimal level of prediction accuracy, thereby preventing researchers from further enhancing the precision of data for justification and reproducibility in subsequent studies.

Table 5. Five different models with different prediction accuracies

Model	Architecture	(R ² %) of the model	(R ² %) compared to the data extracted from solving Flamelet equations in $\chi = 0$ (1/s)		
			Temperature	CO ₂	OH
1	MLP 3-10 ^a	96,77	90,81	92,48	93,60
2	MLP 0-10	91,32	91,00	84,07	89,98
3	MLP 2-10	86,93	84,16	76,43	88,12
4	MLP 1-20	72,28	70,40	00,70	02,00
5	MLP 0-0	44,90	00,21	40,18	30,72

a: MLP (Multi-Layer Perceptron), 3 is number of hidden layers, and 10 is number of neurons in each layer

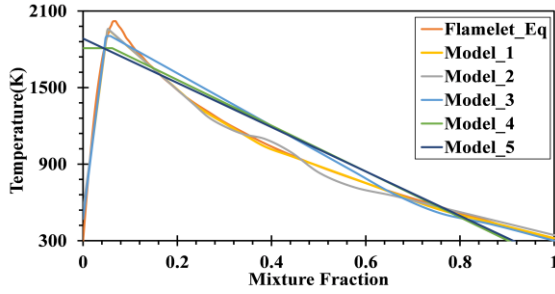
Figure 12 displays the output plots for the temperature, CO₂, and OH species models presented in Table 5. The figure indicates that models 4 and 5 have deviated from the expected plot shape. The models exhibit a decreasing trend in the mixture fraction that is less than 0.1, despite the expectation for an increase in the parameters. Models 4 and 5 reproduced unacceptable results in forecasting OH mass fraction.

Upon analysis of the graphical representations of the five models mentioned in Table 5 it can be inferred that Table 6 is valuable in comprehending the presented data. Table 6 was compiled by examining the temperature and CO₂ species in nine different mass fractions.

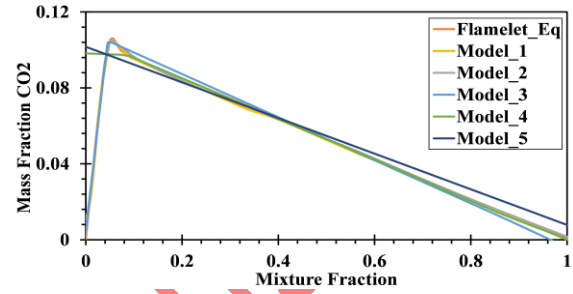
Table 6. Comparison of temperature (K) and CO₂ mass fraction at 9 mixture fractions — FlameMaster vs MLP architectures.

Mixture Fraction	Temperature						CO ₂ Mass Fraction					
	Flamelet Eq	Model 1	Model 2	Model 3	Model 4	Model 5	Flamelet Eq	Model 1	Model 2	Model 3	Model 4	Model 5
0.106	1820	1787	1787	1800	1729	1700	0.920	0.946	0.900	0.980	0.901	0.917
0.21	1400	1402	1449	1091	1042	1019	0.84	0.826	0.826	0.862	0.840	0.820
0.30	1230	1220	1196	1406	1381	1363	0.740	0.727	0.730	0.709	0.743	0.730
0.387	1060	1030	1096	1227	1220	1212	0.603	0.646	0.648	0.661	0.600	0.603
0.503	879	881	828	989	1016	1011	0.029	0.023	0.028	0.029	0.026	0.040

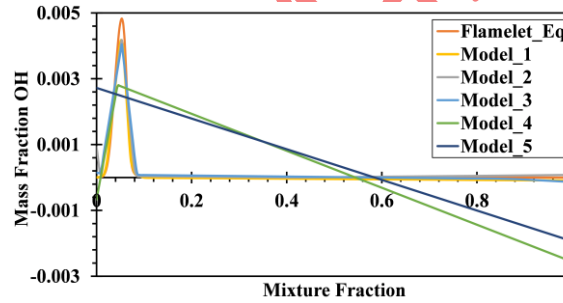
0.709	778	773	701	778	737	740	0.373	0.370	0.374	0.302	0.309	0.398
0.707	067	043	063	026	061	070	0.209	0.208	0.209	0.240	0.201	0.307
0.879	443	440	472	426	370	370	0.139	0.142	0.144	0.113	0.131	0.201
1.0	300	319	340	300	120	148	0	0.009	0.016	0.004	0.004	0.078



(a)



(b)



(c)

Figure 13. Effect of MLP architecture on prediction accuracy at an unseen scalar dissipation rate ($\chi = 4.0 \text{ s}^{-1}$): (a) temperature, (b) CO_2 , and (c) OH , mass fractions for the optimal HPT configuration (4 hidden layers: 10-15-20-15 neurons), 2 medium architectures,

3.4. Uniform structure models

After analyzing the specified methods in previous sections, it was determined that the MLP method exhibited better prediction accuracy and data generation capabilities, thereby establishing it as the optimal model for data generation. This model's notable characteristic is its capacity to make highly accurate predictions for nonlinear phenomena, which may be enhanced through architectural modifications. Changing the architecture of the neural networks are mostly referred

to changes in the hidden layers and neurons. In order to select the optimal model with reliable performance, modifications were made to its architecture by increasing the number of hidden layers from 1 to 5 and the number of neurons (5, 10, 15, 20, 25). All presented models demonstrate uniform architectures, ensuring the number of neurons remains consistent across all hidden layers; since non-uniform architectures are more complex, they are more difficult to design and fine-tune.

The models can be evaluated using accuracy and error classification, as demonstrated in Table V. The MLP₅₋₅ model is deemed inadequate for user modeling. The model's accuracy is below 50%, and its error exceeds 0.5, shows it unreliable for utilization in modeling. Conversely, models exhibiting an accuracy exceeding 90% may be given precedence.

Moreover, in the case that the MSE error is below 0.005, the model may be deemed as the optimal model. Only MLP₅₋₂₀, MLP₅₋₁₅ and MLP₄₋₂₀ models meet both criteria (colored blue). The dissimilarity between the models mentioned above lies in their respective durations for training and prediction time. Thus, the MLP₅₋₁₅ model is a viable option for presenting the findings. Additionally, Table V compares the accuracy of other models used in this study with MLP models.

Table V. Comparison of MLP uniform architectures — R², RMSE, and computation time.

Architecture			R ² (%)	RMSE	Training time (s)	Prediction time(s)
Number of hidden layers	Neurons	Name				
5	25	MLP ₅₋₂₅ ^a	98,74	0,0000	14,20	1,96
	20	MLP ₅₋₂₀	97,94	0,0011	10,13	2,43
	15	MLP ₅₋₁₅	99,34	0,0003	10,71	1,89
	10	MLP ₅₋₁₀	91,32	0,0038	12,02	1,07
	5	MLP ₅₋₅	44,90	0,0301	2,70	0,37
4	25	MLP ₄₋₂₅	98,30	0,0009	21,27	1,74
	20	MLP ₄₋₂₀	98,77	0,0000	12,70	2,01
	15	MLP ₄₋₁₅	97,40	0,0010	10,39	1,86

3	10	MLP _{ε-10}	94,49	0,0021	23,17	1,73
	0	MLP _{ε-20}	73,04	0,0102	3,78	0,27
	20	MLP ₂₀₋₂₀	98,20	0,0010	17,60	1,24
	20	MLP ₂₀₋₂₀	97,94	0,0009	11,33	1,78
	10	MLP ₂₀₋₁₀	97,77	0,0013	12,10	1,60
	10	MLP ₂₀₋₁₀	90,36	0,0039	31,00	1,17
2	0	MLP ₂₀₋₂₀	74,76	0,0097	3,10	0,20
	20	MLP ₂₀₋₂₀	94,01	0,0012	10,24	1,34
	20	MLP ₂₀₋₂₀	96,31	0,0014	16,01	1,29
	10	MLP ₂₀₋₁₀	98,16	0,0007	14,41	1,10
	10	MLP ₂₀₋₁₀	86,93	0,0061	17,40	0,77
	0	MLP ₂₀₋₂₀	88,90	0,0001	13,09	0,01
1	20	MLP ₁₀₋₂₀	91,80	0,0019	24,20	0,07
	20	MLP ₁₀₋₂₀	93,08	0,0030	27,13	0,66
	10	MLP ₁₀₋₁₀	94,29	0,0020	28,37	0,08
	10	MLP ₁₀₋₁₀	69,00	0,0206	16,30	0,40
	0	MLP ₁₀₋₂₀	62,28	0,0240	19,47	0,31
	100 (Default)	MLP _{def}	80,90	0,0162	3,16	1,40
Other Models		RFR	96,00	0,0016	14,06	1,82
		LR	0,019	0,0222	19,41	0,90

a: MLP (Multi-Layer Perceptron), ϵ is number of hidden layers, and 20 is number of neurons in each layer

0,0. Hyperparameter tuning

As demonstrated in the preceding section, the inclusion of additional hidden layers and neurons has significantly improved the accuracy of the prediction results. Nevertheless, it is imperative to acknowledge that these parameters are not the only indicators of effectiveness, thus necessitating the investigation and assessment of additional parameters for MLP. This section examines hyperparameters, including the number of hidden layers and neurons, which significantly impact the accuracy of machine learning models in predicting data. By considering all effective factors in

MLP algorithm accuracy, we can choose an optimal model based on hidden layers, neurons, activation function, solver, alpha coefficient, learning rate, and tolerance.

- **Number of hidden layers:** In the initial form of MLP, a single hidden layer is considered. However, this study used this number from 1 to 5 to find the optimal hyperparameter.
- **Number of neurons in each layer:** This number is set to 100 in the initial state. 5, 10, 15, 20, and 25 are considered values for this parameter in this study.
- **Activation function:** An activation function in neural networks produces a value in the output using a node's input values. It maps the weighted sum of the node inputs to a specific domain (depending on the activation function). The final value is transferred to the next layer by this function. In this study, three different activation functions (ReLU, Tanh, and Sigmoid) were investigated (Table 4).
- **Solver:** Various solvers can be used to adjust the weight and adapt target and output, and in this study, three solvers, SGD and Adam, have been used.
 - SGD (Stochastic Gradient Descent).
 - AdaDelta: limits the window of gradients to some fixed size.
 - Adams (Gradient-based Optimizer).

The default solver, "Adam," works well in terms of training time and validation score on relatively large datasets (with thousands of training samples or more)

- **The Alpha coefficient** represents the power of data regularization. The loss function is calculated by dividing L_2 regularization by sample size. 0.01, 0.05, 0.1, and 0.5 are considered values for this parameter in this study.

- **Tolerance:** When a loss or score does not improve over consecutive iterations, convergence is achieved, and training stops. Four tolerance values were used in this study: 10^{-1} , 10^{-2} , 10^{-3} , 10^{-4} , and 10^{-5} .

Table 8. Activation functions transfer domain and equations.

Activation function	Transfer domain	Equation
ReLU	$(-\infty, \infty)$	$\varphi(x) = \max(0, x)$
Tanh	$(-1, 1)$	$\varphi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
Sigmoid	$(0, 1)$	$\varphi(x) = \frac{1}{1 + e^{-x}}$

A structured grid search was conducted over all hyperparameter combinations shown in Table 9.

Table 9. All the hyperparameters.

Number of hidden Layers	Number of neurons in each layer	Activation Function	Solver	Alpha coefficient	Tolerance
1	5	Sigmoid	SGD	10^{-1}	10^{-1}
2	10	ReLU	Adam	10^{-2}	10^{-2}
3	15				10^{-3}
4	20	Tanh	AdaDelta	10^{-3}	10^{-4}
5	25			10^{-4}	10^{-5}

The optimal model is a neural network with four hidden layers, 10 neurons in the first layer, 10 neurons in the second layer, 20 neurons in the third layer, and 15 neurons in the fourth hidden layer. The Tanh activation function outperformed ReLU and Sigmoid for this smooth thermochemical data, consistent with its ability to represent bounded outputs. Adam was selected as the optimizer, with $\alpha=10^{-3}$ and $\text{tolerance}=10^{-4}$. This configuration achieves a 5-fold cross-validated R^2 of 0.91 ± 0.01 , confirming robustness across data splits. Layer depth was found to be the most influential hyperparameter, with shallow networks (1-2 layers) consistently underperforming regardless of neuron count. The top five MLP configurations are presented in Table 10.

Table 10. Top five MLP configurations from cross-validated hyperparameter search

Fittest models	No. Hidden layers	Neurons Configuration	Activation Function	Solver	Alpha	Tol	cross-validated R ²
1 st	2	10-10-20-10	Tanh	Adam	0.001	0.00001	99.81
2 nd	2	10-10-10-10					99.76
3 rd	0	10-10-10-10-10					99.34
4 th	0	10-10-10-10-10					99.32
5 th	2	10-10-20-10					99.21

As a result of the examinations, Figure 14 presents results of the optimal model using hyperparameter tuning (named HPT) (Table 10), the weakest model MLP₀₋₀ (Table 9), and the best model MLP₀₋₁₀ (Table 9.) presents the results based on a scalar dissipation rate of $\phi(1/s)$.

By comparing the data of these three different models with the numerical solution of the flamelet equations, it is concluded that these models can accurately predict different species' temperatures and behavior. According to the model evaluation and the accuracy of the models, there is a little difference between the two models in terms of species (Table 11). Hence, the results of MLP₀₋₁₀ and HPT are similar to those of the flamelet equations. In addition, HPT's accuracy on predicting other species is acceptable to seen in Table 12.

Table 11. Performance comparison of HPT, MLP₀₋₁₀, and MLP₀₋₀ on $\chi=0(1/s)$ test set

Model	Temp R ² %	Temp RMSE (K)	CO ₂ R ² %	CO ₂ RMSE	OH R ² %	OH RMSE
HPT	0.9909	27.31	0.9823	3.29×10^{-3}	0.9647	3.07×10^{-4}
MLP ₀₋₁₀	0.9941	33.18	0.9741	3.88×10^{-3}	0.9012	4.01×10^{-4}
MLP ₀₋₀	0.7834	194.2	0.0912	1.04×10^{-2}	0.4231	1.12×10^{-3}

Table 12. MLP-HPT prediction R² % for all 00 species on $\chi=0(1/s)$ test set.

H ₂	97.70	CH ₂	97.37	C ₂ H ₂	94.90	N ₂ O	94.40
H	93.73	CH ₃	98.70	C ₂ H ₄	97.78	HNO	93.09
O	97.08	CO	99.36	HCCO	93.91	CN	94.08
O ₂	97.00	HCO	97.10	CH ₃ CO	94.14	HCN	97.13
OH	92.77	CH ₂ O	97.43	HCCOH	92.28	H ₂ CN	98.32

H ₂ O	97,02	CH ₃ OH	92,72	N	92,86	HCNN	92,63
HO ₂	94,36	CH ₂ O	94,28	NH	98,63	HCNO	93,10
H ₂ O ₂	96,83	CH ₃ OH	97,91	NH ₂	97,63	HOCN	96,39
C	97,90	C ₂ H	92,90	NH ₂	97,61	HNCO	96,72
CH	90,34	C ₂ H ₂	92,00	NNH	99,13	NCO	90,02
CH ₂	97,70	C ₂ H ₂	96,37	NO	94,90	CH ₃ CHO	94,40
CH ₂ (s)	93,73	C ₂ H ₂	98,70	NO ₂	96,68	N ₂	93,09

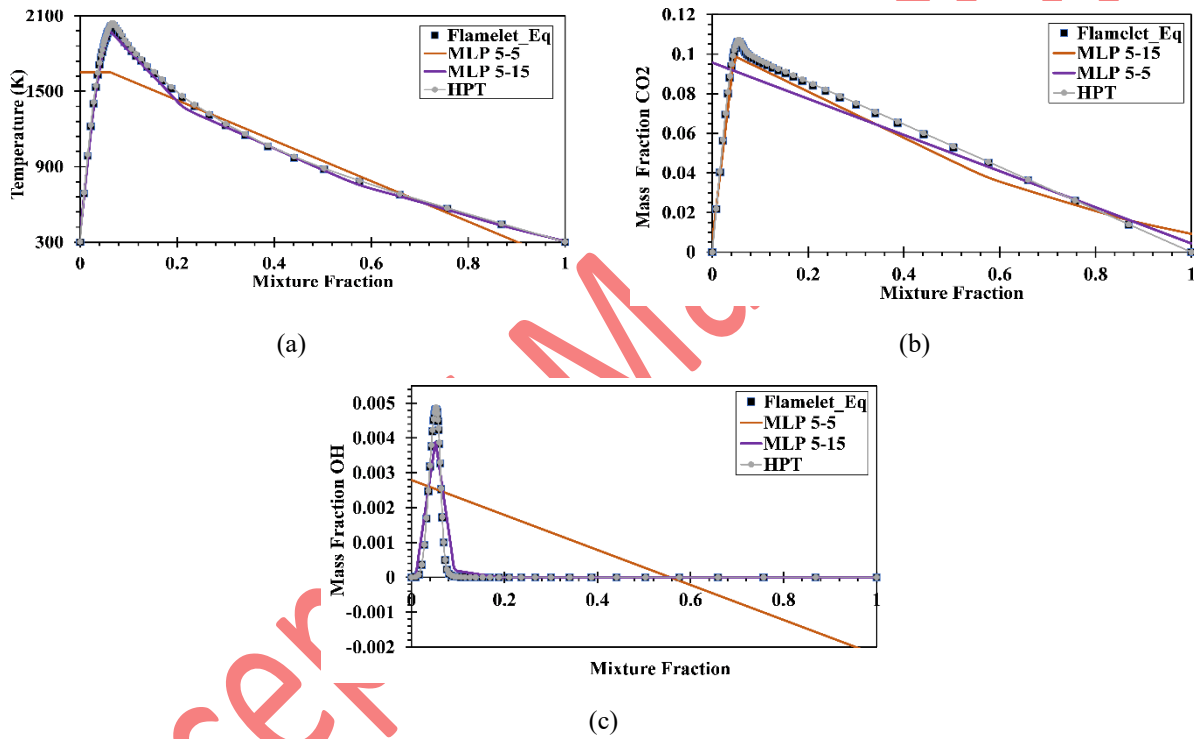


Figure 14. Comparison of the optimal HPT model, MLP 5L-15, and MLP 5L-5 (weakest architecture) at an unseen scalar dissipation rate ($\chi = 5.0 \text{ s}^{-1}$): (a) temperature, (b) OH, and (c) CO₂ mass fractions versus FlameMaster reference.

To assess the genuine predictive capability of the optimal MLP-HPT model beyond the training data, all models were evaluated on fifteen scalar dissipation rate values that were deliberately withheld from training ($\chi = 0.1, 1.0, 2.0, 4.0, 7.0, 8.0, 11.0, 13.0, 16.0, 17.0, 19.0, 22.0, 23.0, 26.0$, and 28.0 s^{-1}). This constitutes a genuine generalization test, as opposed to a reconstruction

of training data. The scalar dissipation rate is included as a continuous input feature alongside the mixture fraction, enabling the model to interpolate across the full χ operating range rather than treating each library as a discrete case.

Table 13 presents the complete set of statistical metrics for the MLP-HPT model on this unseen test set, reporting the coefficient of determination (R^2), root mean square error (RMSE) in physical units, mean absolute error (MAE), and normalized RMSE (NRMSE) for temperature and eight key species. The mean R^2 across all species is 0.9747, confirming strong generalization capability. Species with smooth, monotonic profiles such as N_2 , CH_4 , H_2O , and O_2 achieve R^2 values above 0.998, reflecting near-perfect reproduction of the thermochemical state. Temperature prediction achieves $R^2=0.9909$ with an RMSE of 27.21 K, which is within acceptable tolerance for CFD applications given the temperature range of approximately 1,800 K in the training data.

Species with sharper concentration peaks require closer attention. OH and H_2 show lower generalization R^2 values of 0.9647 and 0.9091, respectively, with NRMSE values of 7.37% and 0.66%. These species exhibit narrow peaks near the stoichiometric mixture fraction that are sensitive to the precise location of the flame front, making them inherently more challenging for interpolation between training χ values. CO similarly achieves $R^2=0.9269$ with NRMSE=0.77%. Despite these lower values relative to the smooth-profile species, all nine species maintain R^2 above 0.90, which is considered acceptable for FGM compression applications where the primary goal is memory reduction while preserving thermochemical fidelity. A parity plot for temperature, OH, and CO_2 across the full unseen test set is presented in Fig. 10, confirming the absence of systematic bias in the model predictions.

For comparison, Random Forest achieves a mean R^2 of 0.9109 on the same unseen conditions, performing acceptably on major species but showing larger deviations for H_2 ($R^2=0.7881$) and CO

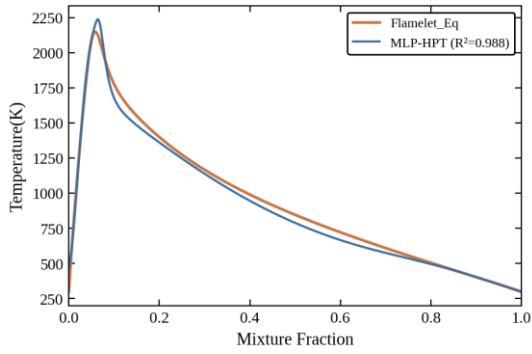
($R^2=0.9742$). Linear Regression fails on most minor species with a mean R^2 of 0.42 , confirming that the nonlinearity of thermochemical profiles across χ cannot be captured by a linear model. These results collectively confirm the selection of MLP-HPT as the optimal method for FGM library compression in the present configuration.

Table 13. MLP-HPT Comprehensive Metrics on Unseen Scalar Dissipation Rate Test Set

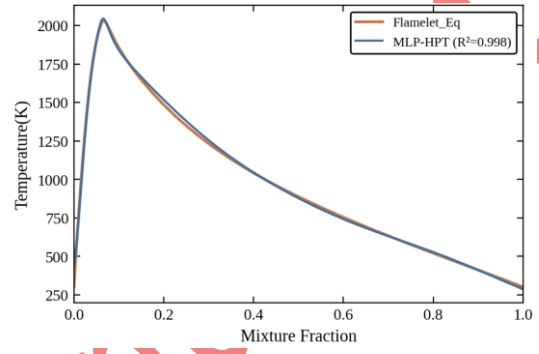
Species	R^2	RMSE	MAE	NRMSE (%)	Quality	Species	R^2
Temperature	0.9909	27.31 K	17.00 K	1.48	Excellent	Temperature	0.9909
H ₂ O	0.9979	1.26×10^{-3}	8.28×10^{-4}	0.98	Excellent	H ₂ O	0.9979
CH ₄	0.9980	7.23×10^{-3}	5.37×10^{-3}	0.72	Excellent	CH ₄	0.9980
O ₂	0.9981	1.83×10^{-3}	1.34×10^{-3}	0.78	Excellent	O ₂	0.9981
N ₂	0.9990	4.38×10^{-3}	3.09×10^{-3}	0.07	Excellent	N ₂	0.9990
CO ₂	0.9823	3.29×10^{-3}	2.17×10^{-3}	2.09	Very Good	CO ₂	0.9823
OH	0.9647	3.07×10^{-4}	1.83×10^{-4}	7.37	Good	OH	0.9647
CO	0.9269	4.12×10^{-3}	1.97×10^{-3}	5.77	Good	CO	0.9269
H ₂	0.9091	4.03×10^{-4}	1.03×10^{-4}	5.66	Good	H ₂	0.9091

To further demonstrate the model's generalization capability across the full scalar dissipation rate range, Figure 10 presents the MLP-HPT temperature predictions against FlameMaster reference solutions at six unseen χ values spanning the entire operating range from $\chi = 0.1 \text{ s}^{-1}$ to $\chi = 28.0 \text{ s}^{-1}$. The model maintains close agreement with the reference solutions at all six conditions, including both the low- χ regime ($\chi = 0.1 \text{ s}^{-1}$, Figure 10a) where the flame structure is broad and diffusion-dominated, and the high- χ regime ($\chi = 28.0 \text{ s}^{-1}$, Figure 10f) where the flame approaches extinction and the temperature profile is significantly suppressed. The intermediate conditions at $\chi = 8.0$ and 16.0 s^{-1} (Figure 10c and d) similarly show strong agreement, with R^2 values consistent with the mean reported in Table 13. These results confirm that the inclusion of χ as a continuous

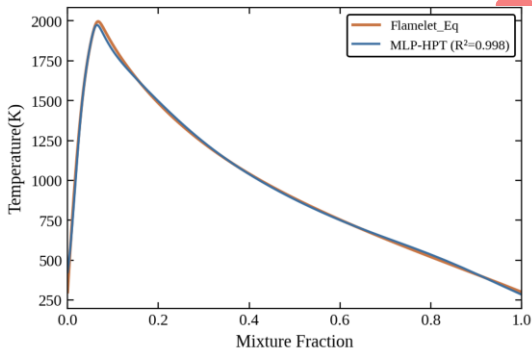
input feature enables the MLP-HPT model to interpolate smoothly between trained conditions rather than extrapolating from the nearest discrete library, which is the essential requirement for deployment in a CFD solver where the local scalar dissipation rate varies continuously across the computational domain.



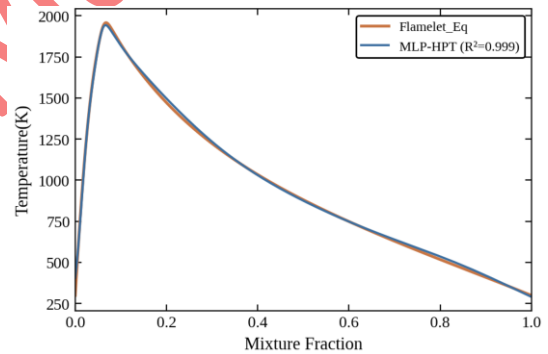
(a)



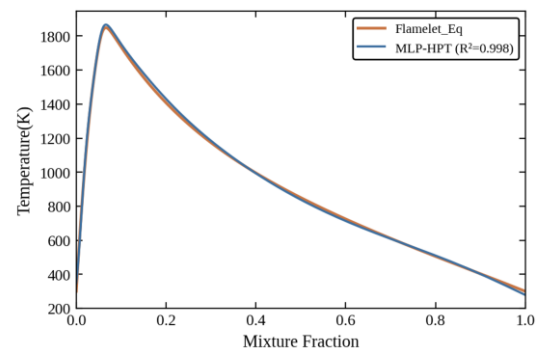
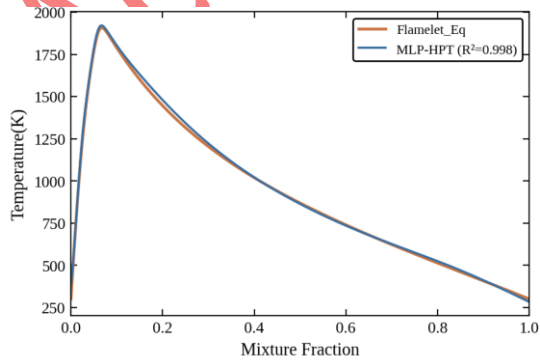
(b)



(c)



(d)



(e)

(f)

Figure 15. MLP-HPT temperature predictions versus FlameMaster reference across six unseen scalar dissipation rate values: (a) $\chi = 0.1$, (b) $\chi = 4.0$, (c) $\chi = 8.5$, (d) $\chi = 16.0$, (e) $\chi = 22.0$, and (f) $\chi = 28.0 \text{ s}^{-1}$. All χ values shown were withheld from training, demonstrating the model's generalization capability across the full operating range.

6. Conclusion

This study presents a systematic benchmark of four machine learning methods for FGM library compression in methane combustion simulations. Scalar dissipation rate and mixture fraction are used as joint continuous inputs, and all models are evaluated on unseen scalar dissipation rate conditions to provide a genuine test of generalization capability. Using an initial library of cross-flow solutions with scalar dissipation rates between 0.1 and 28.0 s^{-1} for training, the following conclusions are drawn:

- Seven libraries are suitable for creating a machine learning database with an error of 2.3% .
- After evaluating all models on unseen scalar dissipation rate conditions, the MLP model achieves the highest mean R^2 of 0.9747 (Temperature $R^2=0.9909$), outperforming RF (mean $R^2=0.9109$) and LR (mean $R^2=0.82$).
- Species with smooth profiles (N_2 , H_2O , CH_4) are predicted with $R^2 > 0.998$, while species with sharper concentration peaks (H_2 , CO) yield lower but acceptable R^2 values of 0.909 and 0.927 , respectively, on unseen conditions.
- Cross-validated hyperparameter optimization identifies the optimal MLP configuration (4 hidden layers: $10-10-20-10$ neurons, Tanh activation, Adam solver), achieving a cross-validated R^2 of 0.981 ± 0.008 . Full CFD deployment testing remains a direction for future work; the present evaluation on unseen scalar dissipation rate values provides a meaningful assessment of interpolation capability within the FGM framework.

Nomenclature and Acronym

ρ	Density (kgm^{-3})	Da	Damköhler number	LR	Linear Regression
τ	Time	D_z	Diffusion coefficient	ML	Machine Learning
h	Enthalpy	Le	Lewis number	MLP	Multilayer Perceptron
$\dot{\omega}_F$	Reaction rate ($\text{kgm}^{-3}\text{s}^{-1}$)	Z	Mixture fraction	PCA	Principle Component Analysis
χ	Scalar dissipation rate (s^{-1})	Acronyms		RF	Random Forest
T	Temperature	ANN	Artificial Neural Network	SGD	Stochastic Gradient Descent
Q_R	Radiation heat loss	DNS	Direct Numerical Simulation	SVM	Support Vector Machine
c_p	Specific heat capacity ($\text{Jkg}^{-1}\text{K}^{-1}$)	DT	Decision Tree		

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used Claude (Anthropic) to improve the writing style of the manuscript. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

- [1] P. Compais, J. Arroyo, M. Á. Castán-Lascorz, J. Barrio, and A. Gil, "Detection of slight variations in combustion conditions with machine learning and computer vision," *Eng. Appl. Artif. Intell.*, vol. 126, p. 106772, Nov. 2023, doi: [10.1016/j.engappai.2023.106772](https://doi.org/10.1016/j.engappai.2023.106772).
- [2] N. Amiri, M. Honarmand, M. Dizani, A. Moosavi, and S. Kazemzadeh Hannani, "Shear-thinning droplet formation inside a microfluidic T-junction under an electric field," *Acta Mech.*, vol. 232, no. 7, pp. 2030–2054, Jul. 2021, doi: [10.1007/s00707-021-02960-y](https://doi.org/10.1007/s00707-021-02960-y).
- [3] I. El Helou, J. M. Foale, R. S. Pathania, R. Ciardiello, A. W. Skiba, and E. Mastorakos, "A comparison between fossil and synthetic kerosene flames from the perspective of soot emissions in a swirl spray RQL burner," *Fuel*, vol. 331, p. 120608, Jan. 2023, doi: [10.1016/j.fuel.2022.120608](https://doi.org/10.1016/j.fuel.2022.120608).
- [4] S. Mehrdad Bathaei, M. Parsa Ghofrani Maab, G. Zarabian Ghaeini, M. Kim, J. Abolfazli Esfahani, and K. Chun Kim, "Combustion regime identification and characteristics of nitrogen-diluted ammonia with hot air oxidizer: Non-premixed counterflow flame," *Therm. Sci. Eng. Prog.*, vol. 39, p. 101722, Mar. 2023, doi: [10.1016/j.tsep.2023.101722](https://doi.org/10.1016/j.tsep.2023.101722).
- [5] D. Pasik, T. Golin Almeida, E. Ahongshangbam, S. Iyer, and N. Myllys, "Monoterpene oxidation pathways initiated by acyl peroxy radical addition," *Atmospheric Chem. Phys.*, vol. 20, no. 7, pp. 4313–4331, 2020, doi: [10.5194/acp-20-4313-2020](https://doi.org/10.5194/acp-20-4313-2020).

- [٦] A. Jafari and A. Afsharfard, "Conceptual design of dynamic vibration absorber for mitigation of subway tracks vibration," in *12th International Conference on Acoustics and Vibration*, ٢٠٢٢. <https://civilica.com/doc/١٦١١١٢٦/>.
- [٧] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine Learning for Fluid Mechanics," *Annual Review of Fluid Mechanics*, vol. ٥٢, no. Volume ٥٢, ٢٠٢٠. Annual Reviews, pp. ٤٧٧–٥٠٨, ٢٠٢٠. doi: [10.1146/annurev-fluid-٠١٠٧١٩-٠٦٠٢١٤](https://doi.org/10.1146/annurev-fluid-٠١٠٧١٩-٠٦٠٢١٤).
- [٨] K. Duraisamy, G. Iaccarino, and H. Xiao, "Turbulence Modeling in the Age of Data," *Annual Review of Fluid Mechanics*, vol. ٥١, no. Volume ٥١, ٢٠١٩. Annual Reviews, pp. ٣٥٧–٣٧٧, ٢٠١٩. doi: [10.1146/annurev-fluid-٠١٠٥١٨-٠٤٠٥٤٧](https://doi.org/10.1146/annurev-fluid-٠١٠٥١٨-٠٤٠٥٤٧).
- [٩] S. A. Kalogirou, "Artificial intelligence for the modeling and control of combustion processes: a review," *Prog. Energy Combust. Sci.*, vol. ٢٩, no. ٦, pp. ٥١٥–٥٦٦, Jan. ٢٠٠٣, doi: [10.1016/S0360-5310\(03\)00088-3](https://doi.org/10.1016/S0360-5310(03)00088-3).
- [١٠] N. Amiri, J. M. Prisaznuk, P. Huang, P. R. Chiarot, and X. Yong, "Deep-learning-enhanced modeling of electrosprayed particle assembly on non-spherical droplet surfaces," *Soft Matter*, vol. ٢١, no. ٤, pp. ٦١٣–٦٢٥, ٢٠٢٥, doi: [10.1039/D4SM01160K](https://doi.org/10.1039/D4SM01160K).
- [١١] A. Afsharfard, A. Jafari, Y. A. Rad, H. Tehrani, and K. C. Kim, "Modifying Vibratory Behavior of the Car Seat to Decrease the Neck Injury," *J. Vib. Eng. Technol.*, vol. ١١, no. ٣, pp. ١١١٥–١١٢٦, Apr. ٢٠٢٣, doi: [10.1007/s42447-022-00627-4](https://doi.org/10.1007/s42447-022-00627-4).
- [١٢] T. Ding, T. Readshaw, S. Rigopoulos, and W. P. Jones, "Machine learning tabulation of thermochemistry in turbulent combustion: An approach based on hybrid flamelet/random data and multiple multilayer perceptrons," *Combust. Flame*, vol. ٢٣١, p. ١١١٤٩٣, Sep. ٢٠٢١, doi: [10.1016/j.combustflame.2021.111493](https://doi.org/10.1016/j.combustflame.2021.111493).
- [١٣] A. Seltz, P. Domingo, L. Vervisch, and Z. M. Nikolaou, "Direct mapping from LES resolved scales to filtered-flame generated manifolds using convolutional neural networks," *Combust. Flame*, vol. ٢١٠, pp. ٧١–٨٢, Dec. ٢٠١٩, doi: [10.1016/j.combustflame.2019.08.014](https://doi.org/10.1016/j.combustflame.2019.08.014).
- [١٤] K. Zdybał, G. D'Alessio, A. Attili, A. Coussement, J. C. Sutherland, and A. Parente, "Local manifold learning and its link to domain-based physics knowledge," *Appl. Energy Combust. Sci.*, vol. ١٤, p. ١٠٠١٣١, Jun. ٢٠٢٣, doi: [10.1016/j.jaecs.2023.100131](https://doi.org/10.1016/j.jaecs.2023.100131).
- [١٥] T. Kessler, P. C. St. John, J. Zhu, C. S. McEnally, L. D. Pfefferle, and J. H. Mack, "A comparison of computational models for predicting yield sooting index," *Proc. Combust. Inst.*, vol. ٣٨, no. ١, pp. ١٣٨٥–١٣٩٣, Jan. ٢٠٢١, doi: [10.1016/j.proci.2020.07.009](https://doi.org/10.1016/j.proci.2020.07.009).
- [١٦] H.-T. Nguyen, P. Domingo, L. Vervisch, and P.-D. Nguyen, "Machine learning for integrating combustion chemistry in numerical simulations," *Energy AI*, vol. ٥, p. ١٠٠٠٨٢, Sep. ٢٠٢١, doi: [10.1016/j.egyai.2021.100082](https://doi.org/10.1016/j.egyai.2021.100082).
- [١٧] E. S. Joe and D. Arumuga Perumal, "Combustion modelling of sequential combustion in steam-methane reformation (SMR) furnace using adiabatic flamelet generated manifold," *Therm. Sci. Eng. Prog.*, vol. ٤٠, p. ١٠١٧٩٥, May ٢٠٢٣, doi: [10.1016/j.tsep.2023.101795](https://doi.org/10.1016/j.tsep.2023.101795).
- [١٨] H. Bao *et al.*, "Large-eddy simulation of dual-fuel spray ignition at varying levels of methane diluted ambient oxidizer using FGM," *Fuel*, vol. ٣٥١, p. ١٢٨٩٠١, Nov. ٢٠٢٣, doi: [10.1016/j.fuel.2023.128901](https://doi.org/10.1016/j.fuel.2023.128901).

- [19] M. Ihme, W. T. Chung, and A. A. Mishra, "Combustion machine learning: Principles, progress and prospects," *Prog. Energy Combust. Sci.*, vol. 91, p. 101010, Jul. 2022, doi: [10.1016/j.pecs.2022.101010](https://doi.org/10.1016/j.pecs.2022.101010).
- [20] J. Zhang *et al.*, "Artificial Neural Networks for Chemistry Representation in Numerical Simulation of the Flamelet-Based Models for Turbulent Combustion," *IEEE Access*, vol. 8, pp. 80020–80029, 2020, doi: [10.1109/ACCESS.2020.2999943](https://doi.org/10.1109/ACCESS.2020.2999943).
- [21] M. Hansinger, Ge, Yipeng, and M. and Pfitzner, "Deep Residual Networks for Flamelet/progress Variable Tabulation with Application to a Piloted Flame with Inhomogeneous Inlet," *Combust. Sci. Technol.*, vol. 194, no. 8, pp. 1087–1113, Jun. 2022, doi: [10.1080/00102202.2020.1822826](https://doi.org/10.1080/00102202.2020.1822826).
- [22] M. Ihme, C. Schmitt, and H. Pitsch, "Optimal artificial neural networks and tabulation methods for chemistry representation in LES of a bluff-body swirl-stabilized flame," *Proc. Combust. Inst.*, vol. 32, no. 1, pp. 1027–1030, Jan. 2009, doi: [10.1016/j.proci.2008.06.100](https://doi.org/10.1016/j.proci.2008.06.100).
- [23] Q. Guo, Ren, Wanxing, and W. and Lu, "A Method for Predicting Coal Temperature Using CO with GA-SVR Model for Early Warning of the Spontaneous Combustion of Coal," *Combust. Sci. Technol.*, vol. 194, no. 3, pp. 023–038, Feb. 2022, doi: [10.1080/00102202.2020.1772767](https://doi.org/10.1080/00102202.2020.1772767).
- [24] J. Bissantz *et al.*, "Application of dense neural networks for manifold-based modeling of flame-wall interactions," *Appl. Energy Combust. Sci.*, vol. 13, p. 10113, Mar. 2023, doi: [10.1016/j.jaecs.2023.10113](https://doi.org/10.1016/j.jaecs.2023.10113).
- [25] A. Mousemi, Jadidi, Mehdi, Dworkin, Seth B., and W. K. and Bushe, "Application of machine learning in low-order manifold representation of chemistry in turbulent flames," *Combust. Theory Model.*, vol. 27, no. 1, pp. 83–102, Jan. 2023, doi: [10.1080/13647830.2022.2103740](https://doi.org/10.1080/13647830.2022.2103740).
- [26] K. Li, P. Rahnama, R. Novella, and B. Somers, "Combining flamelet-generated manifold and machine learning models in simulation of a non-premixed diffusion flame," *Energy AI*, vol. 14, p. 100666, Oct. 2023, doi: [10.1016/j.egyai.2023.100666](https://doi.org/10.1016/j.egyai.2023.100666).
- [27] A. Saravanan, S. Parida, M. Murugan, M. S. Reddy, P. V. Elumalai, and S. Kumar Dash, "Thermal performance prediction of a solar air heater with a C-shape finned absorber plate using RF, LR and KNN models of Machine learning," *Therm. Sci. Eng. Prog.*, vol. 38, p. 101630, Feb. 2022, doi: [10.1016/j.tsep.2022.101630](https://doi.org/10.1016/j.tsep.2022.101630).
- [28] J. (任嘉豪) Ren, H. (王海鸥) Wang, J. (邢江宽) Xing, K. (罗坤) Luo, and J. (樊建人) Fan, "A lower-dimensional approximation model of turbulent flame stretch and its related quantities with machine learning approaches," *Phys. Fluids*, vol. 32, no. 11, p. 110113, Nov. 2020, doi: [10.1063/1.5188877](https://doi.org/10.1063/1.5188877).
- [29] J. Liu, H. Sun, Y. Zhang, J. Hu, and T. Zou, "Steady-state sequence optimization with incremental input constraints in two-layer model predictive control," *ISA Trans.*, vol. 128, pp. 144–158, Sep. 2022, doi: [10.1016/j.isatra.2021.11.039](https://doi.org/10.1016/j.isatra.2021.11.039).
- [30] S. Yao, A. Kronenburg, A. Shamooni, O. T. Stein, and W. Zhang, "Gradient boosted decision trees for combustion chemistry integration," *Appl. Energy Combust. Sci.*, vol. 11, p. 100077, Sep. 2022, doi: [10.1016/j.jaecs.2022.100077](https://doi.org/10.1016/j.jaecs.2022.100077).

- [۳۱] F. Kartal and U. Özveren, "Prediction of activation energy for combustion and pyrolysis by means of machine learning," *Therm. Sci. Eng. Prog.*, vol. ۳۳, p. ۱۰۱۳۴۶, Aug. ۲۰۲۲, doi: [10.1016/j.tsep.2022.101346](https://doi.org/10.1016/j.tsep.2022.101346).
- [۳۲] O. Owoyele, P. Kundu, M. M. Ameen, T. Echekeki, and S. Som, "Application of deep artificial neural networks to multi-dimensional flamelet libraries and spray flames," *Int. J. Engine Res.*, vol. ۲۱, no. ۱, pp. ۱۵۱–۱۶۸, Jan. ۲۰۲۰, doi: [10.1177/1468087419837770](https://doi.org/10.1177/1468087419837770).
- [۳۳] R. Yang, T. Xie, and Z. Liu, "The Application of Machine Learning Methods to Predict the Power Output of Internal Combustion Engines," *Energies*, vol. ۱۵, no. ۹, ۲۰۲۲, doi: [10.3390/en15093242](https://doi.org/10.3390/en15093242).
- [۳۴] J. Xie, M. Sage, and Y. F. Zhao, "Feature selection and feature learning in machine learning applications for gas turbines: A review," *Eng. Appl. Artif. Intell.*, vol. ۱۱۷, p. ۱۰۵۵۹۱, Jan. ۲۰۲۳, doi: [10.1016/j.engappai.2022.105591](https://doi.org/10.1016/j.engappai.2022.105591).
- [۳۵] H. Ma, T. Peng, C. Zhang, C. Ji, Y. Li, and M. S. Nazir, "Developing an evolutionary deep learning framework with random forest feature selection and improved flow direction algorithm for NOx concentration prediction," *Eng. Appl. Artif. Intell.*, vol. ۱۲۳, p. ۱۰۶۳۶۷, Aug. ۲۰۲۳, doi: [10.1016/j.engappai.2023.106367](https://doi.org/10.1016/j.engappai.2023.106367).
- [۳۶] G. P. Chander and S. Das, "Hesitant t-spherical fuzzy linear regression model based decision making approach using gradient descent method," *Eng. Appl. Artif. Intell.*, vol. ۱۲۲, p. ۱۰۶۰۷۴, Jun. ۲۰۲۳, doi: [10.1016/j.engappai.2023.106074](https://doi.org/10.1016/j.engappai.2023.106074).
- [۳۷] S. K. Manoharan, R. K. Megalingam, A. H. Kota, V. K. T. P., and K. S. Sankardas, "Hybrid fuzzy support vector machine approach for Coconut tree classification using image measurement," *Eng. Appl. Artif. Intell.*, vol. ۱۲۶, p. ۱۰۶۸۰۶, Nov. ۲۰۲۳, doi: [10.1016/j.engappai.2023.106806](https://doi.org/10.1016/j.engappai.2023.106806).
- [۳۸] M. Hemmat, D. Toghraie, and F. Amoozad, "Prediction of viscosity of MWCNT-Al₂O₃ (۲۰:۸۰)/SAE۴۰ nano-lubricant using multi-layer artificial neural network (MLP-ANN) modeling," *Eng. Appl. Artif. Intell.*, vol. ۱۲۱, p. ۱۰۵۹۴۸, May ۲۰۲۳, doi: [10.1016/j.engappai.2023.105948](https://doi.org/10.1016/j.engappai.2023.105948).
- [۳۹] M. D. Smooke, I. K. Puri, and K. Seshadri, "A comparison between numerical calculations and experimental measurements of the structure of a counterflow diffusion flame burning diluted methane in diluted air," *Twenty-First Symp. Int. Combust.*, vol. ۲۱, no. ۱, pp. ۱۷۸۳–۱۷۹۲, Jan. ۱۹۸۸, doi: [10.1016/S0082-0784\(88\)80412-0](https://doi.org/10.1016/S0082-0784(88)80412-0).