Leveraging Advanced Machine Learning Methods to Enhance Multilevel Fusion Score Level Computations

Rajesh Tiwari [1], Satyanand Singh[2] *, G. Shanmugaraj[3], Suresh Kumar Mandala [4], Ch. L. N. Deepika[5], Bhanu Pratap Soni[6], Jiuliasi V. Uluiburotu[7]

[1]Department of CSE (AIML), CMR Engineering College, Hyderabad, Telangana, India

[2,6,7]School of Electrical & Electronics Engineering, Fiji National University, Fiji.

[3]Department of ECE, Velammal Institute of Technology, Chennai, TN, India

[4]Department of Computer Science and Artificial Intelligence, SR University, Warangal, Telangana, India

[5]Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India

Emails: drrajeshtiwari20@gmail.com[1] , satyanand.singh@fnu.ac.fj[2] *, gsraj76@gmail.com[3], mandala.suresh83@gmail.com[4], ldeepu2474@gmail.com[5], bhanu.soni@fnu.ac.fj[6], jiuliasi.uluiburotu@fnu.ac.fj[7]

Abstract

This research introduces a novel technique for determining multiple fusion score levels that operates effectively across various datasets and purposes. The four components of the system work in harmony: Feature Engineering, Ensemble Learning, deep neural networks (DNNs), and Transfer Learning. In the feature engineering phase, raw data undergoes a complete transformation, emphasizing the significance of PCA and MI for predictive power. AdaBoost is incorporated during ensemble learning, repeatedly instructing weak learners and adjusting weights based on errors to create a robust ensemble model. Weighted input processing, ReLU activation, and dropout layers seamlessly integrate with DNNs, revealing subtle data patterns and correlations. In transfer learning (fine-tuning), a trained model is adapted for the feature-engineered dataset. In comparative testing, the proposed technique demonstrated higher accuracy, precision, recall, F1 score, AUC-ROC, and shorter training duration. Efficiency measures reduce reasoning time, memory usage, parameter count, model size, and energy consumption. Visualizations illustrate resource consumption, method scores, and the distribution of reasoning time in the research. This mathematical framework enhances the computation of multilayer fusion score levels, performs effectively, and proves versatile across various scenarios, making it a reliable choice for large and diverse datasets.

Keywords: Feature Engineering; Ensemble Learning; Deep Neural Networks (DNNs); Transfer Learning (Fine-tuning); AdaBoost; Multilevel Fusion; Score Level Computations; Optimization; Discriminatory Power

1 introduction

Combining new methodologies enables substantial advances in multilayer fusion score level computations in machine learning and signal processing, which are expanding rapidly. This work investigates and applies machine learning advances to enhance the accuracy, speed, and usefulness of multilayer fusion score level computations.[1] In the following sections, we will examine the field's recent accomplishments, this study's key aims, probable solutions, and crucial attempts to further it.

Recent years have witnessed significant progress in multilayer fusion score level computations. Researchers and business professionals have dedicated efforts to capitalize on vast amounts of data and excellent machine learning algorithms. From fundamental statistical approaches to cutting-edge deep learning, several areas are

https://doi.org/10.54216/FPA.140206
Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

experiencing improvement.[2] Understanding and building on these breakthroughs aids in addressing real-world challenges. This work aims to enhance multilayer fusion score level estimates using sophisticated machine learning. We intend to uncover performance bottlenecks by delving into the intricacies of current approaches.[3] The objective is to create a system that addresses these issues and advances the field. The project aims to bridge academic and practical principles to mature multilevel fusion score computations. The study presents a novel approach to objectives, utilizing advanced machine learning to create multilayer fusion score level calculations. Deep learning architectures, ensemble learning, and topic-specific knowledge can strengthen computational models.[4] Additionally, the study examines ways to improve feature extraction and selection to integrate information more rapidly and with less noise and uncertainty.

- Designing a unique deep learning model for layered fusion score level computations.
- Developing and employing ensemble learning to enhance computer model accuracy and reliability.
- Utilizing domain-specific data to improve decision-making and ensure information relevance across applications.
- Reducing noise and confusion in input data to enhance feature extraction and selection algorithms.
- Thoroughly testing proposed solutions on numerous datasets and in real-world circumstances.[5]

This study utilizes cutting-edge machine learning methods to enhance multilayer fusion network score-level computations. The research aims to accelerate the growth of this dynamic field by examining existing advances, setting goals, generating new ideas, and emphasizing important actions.

2 Applications Machine Learning approaches

Multiple complex machine learning approaches for multilevel fusion score computations have been explored for accuracy and speed.[6] Deep neural networks exhibit high accuracy (0.92) and AUC-ROC (0.95). Training takes 120 seconds, and inference takes 8 milliseconds, which is satisfactory. With an accuracy of 0.94, Ensemble Learning excels in precision (0.93), recall (0.95), and AUC-ROC (0.97). SVMs have 0.89 accuracy and 5 milliseconds of reasoning time, making them valuable. Gradient Boosting Machines (GBM) outperform random forests with an AUC-ROC of 0.98 and an accuracy of 0.95.[7]

AutoML and Neural Architecture Search (NAS) perform well in various ways. Transfer learning achieves the greatest accuracy (0.96), precision (0.95), and memory usage (0.97). Reinforcement learning is more accurate, but explainable AI is superior overall.[8] Deep Neural Networks (DNN) utilize 300 MB of RAM and are cost-effective. Ensemble Learning, due to its cost efficiency, utilizes resources effectively. Support Vector Machines (SVMs) are cost-effective and require 200 MB of RAM. Random Forests and Gradient Boosting Machines (GBM) manage speed and resources effectively.[9] Neural Architecture Search (NAS) is versatile and cost-effective since it consumes less memory. Transfer learning utilizes the greatest memory (600 MB), the least computational power, and the least money. AutoML is memory-efficient and cost-effective. Explainable AI (XAI) utilizes money and memory better than reinforcement learning does with computers.[10]

Various machine learning algorithms for determining multilevel fusion score levels demonstrate that performance and speed may be traded off. Transfer learning and ensemble learning have tremendous accuracy potential. However, SVM and explainable AI use resources and costs efficiently.[11]

The complex ensemble learning approach of machine learning combines model results to provide a more accurate prediction. This approach incorporates model-beneficial characteristics via bagging, boosting, and stacking. Bagging reduces data volatility by training several models on randomly selected samples and averaging their predictions. To reduce bias, train models sequentially and learn from previous failures. Stacking may include model predictions due to its unique methodology. Ensemble learning may improve the consistency and accuracy of multilevel fusion score computation, making it a valuable dataset management tool.

Overfitting is a challenging problem in complex calculations, but there is a simple solution. Sensor data in ensemble learning for security computing might improve threat detection. The financial industry may use a

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

variety of economic models to forecast market changes. Ensemble learning is useful for businesses that need rapid choices since it integrates data from several sources. This is true since it provides several advantages.

The Support Vector Machine (SVM) is a highly effective supervised machine learning classification algorithm. To maximize class margin, they position the hyperplane in an ideal high-dimensional space. Support vector machines (SVM), which employ the kernel approach to manage non-linear connections, can handle a broad range of data types. Support vector machines (SVMs)[12,13] can handle high-dimensional data spaces, which is useful for multilayer fusion score computations. In the medical industry, support vector machines, or SVMs, categorize patient data into diagnostic categories based on a variety of parameters.[14] SVM algorithms offer the capacity to discern between normal and abnormal surveillance data patterns in the realm of security. SVMs perform very well in high-dimensional settings, making them ideal for complex fusion situations. Real-world applications include SVM fraud detection in the banking sector. To detect fraudulent behavior, SVM classifies transactions based on a variety of factors.

Random forests are very effective for classification and regression. This approach will first train many decision trees on different data subsets, then combine the outcomes of those trees. The random selection of features generates tree variety, which increases the model's robustness and accuracy. Random forests are very effective for multilayer fusion score-level computations since they can handle a broad range of high-dimensional datasets. They can evaluate the value of characteristics, which is critical for identifying which variables are most significant in complex fusion tasks.[15] Because of their interpretability and resistance to overfitting, random forests are useful in a variety of disciplines, including research and commerce. A comprehensive knowledge of the model's decision-making process is as vital as having accurate projections.

In the field of predictive modeling, GBMs have proven to be an extremely useful method because they are an advanced ensemble learning technique. Every GBM model needs to be able to solve the issues that were present in the models that came before it. By using a process that is analogous to gradient descent optimization, this method is able to reduce the typical number of mistakes that occur. If you want to figure out combinations of multilevel fusion scores, the iterative model prediction improvement in GBM works well enough. When it comes to dealing with complex interrelationships between variables and different types of data, this technique shines.[16] Because it makes use of historical data, GBM has the potential to enhance financial forecasts of movements in the stock market. The integration of a wide variety of clinical data sources has the potential to improve the accuracy of diagnostic procedures in the healthcare industry. As a result of its high prediction accuracy and adaptability in managing a wide variety of data, good business modeling (GBM) is becoming more popular among organizations that need comprehensive data analysis and accurate forecasting information. This is since GBM is able to analyze a wide variety of data and information types. Case studies in environmental modeling and risk assessment demonstrate how effectively GBM works in business-related applications. In the quest for brain architecture, NAS, or Neural Architecture Search, is the market leader in natural language processing. The major goal is to optimize the architecture of existing neural networks. The NAS uses algorithms to assess a huge universe of alternative structures to identify the best structure for a given application. This automation significantly reduces the amount of effort and expertise necessary to develop neural networks. NAS may adapt neural network topologies to solve issues with multilayer fusion score level data. Network-attached storage (NAS) has the potential to improve image-based security networks.[17] These networks would interpret imaging data from many sources in an appropriate way. It can create networks that incorporate a variety of environmental data sources for climate modeling. NAS enables the building of computational models in an efficient and creative way, allowing neural networks to be extremely specialized and efficient. This graphic shows how NAS has the potential to improve computational models in sectors such as personalized medicine, which include a diverse variety of data types and topologies.

Transfer learning is the process of transferring knowledge from one domain to a related domain in machine learning. After first training on a large dataset, the model is fine-tuned for a smaller target dataset. First, there is the model. When there is a dearth of material, transfer learning is a particularly effective method. Calculations at the multilevel fusion score level enable the application of complex models trained on big datasets to tasks that do not have access to such a large dataset. Models trained on a vast volume of patient data may be fine-tuned for specific illnesses or patient groups in the healthcare business.[16] It is possible to do research on specialist markets using financial models trained on broad market trends. Transfer learning is an effective method for enhancing fusion score computations across a wide range of domains since it makes the best use of previously learned knowledge. Precision agriculture uses it to tailor models created from large agricultural datasets to the unique needs of a farm. This demonstrates the technology's flexibility and promise.

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

Machine learning is experiencing a global revolution because of automated machine learning (AutoML), which automates model selection, hyperparameter tuning, and validation. Using this method, the process of creating effective machine learning models is greatly simplified and hastened. AutoML is a program that helps you quickly identify the most successful models and settings for multilevel fusion score computations. It is getting easier to employ complicated machine learning systems. AutoML has the capacity to determine which financial risk assessment models are most successful in integrating and analyzing the various risk components.[18] It may choose healthcare data analysis models based on clinical criteria. Businesses and academics with little background in machine learning may benefit from the democratization of sophisticated machine learning algorithms via the use of AutoML. When it comes to environmental modeling, AutoML oversees choosing and improving models to predict climate change. In this approach, AutoML exhibits its ability to pick and change models for difficult jobs with many dimensions.

Reinforcement learning, often known as deep learning, is an alternative method for machine learning. This method teaches an agent to make choices by giving it chances to interact with its surroundings. Both incentives and penalties stimulate learning, which helps the agent make the best judgments feasible. Because of its flexibility, RL is a great candidate for performing computations at the score level of dynamic and advanced multilayer fusion. The application of RL in autonomous vehicle navigation allows the vehicle to make real-time decisions based on the fusion of sensors.[19] Algorithmic trading techniques may adjust in response to market developments. Reinforcement learning has immense promise in dynamic and flexible circumstances, especially when predefined rules fail. Robots can learn and adapt to new professions owing to RL algorithms, which also enable them to optimize consumption in energy management systems via real-time data collection. Several case studies demonstrate its use. These examples show how RL may help with complex and fluid decision-making.

The goal of explainable AI (XAI) is to make AI models understandable to humans. This will make the decision-making process more transparent and straightforward. When it comes to multilevel fusion score computations, the technique of making assessments is as important as the choices themselves.[20] The use of XAI increases trust and responsibility, especially in healthcare and finance, where patient treatment decisions must be explained. Furthermore, XAI influences the rationality of investment.

3 Proposed Method

To significantly enhance multilayer fusion score level computations, the four-stage approach incorporates deep neural networks (DNN), ensemble learning, feature engineering, and transfer learning (fine-tuning). A sophisticated strategy like this seeks to address potential challenges associated with handling massive volumes of diverse data.

The initial stage in feature engineering involves transforming raw data through data standardization, PCA, and MI. The dataset benefits from a focus on the most significant aspects of predictive modeling. Techniques such as LASSO regularization, kernel approaches, and polynomial feature construction can further enhance features.

In the second stage, AdaBoost is employed to facilitate ensemble learning with individuals facing learning difficulties. This method modifies weights to anticipate mistakes, improving model accuracy and consistency.[21] This enables the model to distinguish between multilayer fusion score levels.

Next, deep neural networks (DNNs)[22] are employed to identify high-level patterns. Weighted layers control the data within activation functions, and drop-out layers are used to avoid overfitting. Given the complexity of the multilayer fusion procedure, this step is essential.

The third phase involves transfer learning (fine-tuning) using a model from the previous stage. This entails creating unique model layers for feature-engineered datasets to validate the model's data attribute representation, contributing to better multilayer fusion score predictions.

To recap, the proposed data augmentation approach starts with transfer learning and ends with basic characteristic engineering. The stages build on each other to ensure that the model develops gradually and is suitable for

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

complex multilayer fusion scenarios. This comprehensive and iterative approach addresses score-level fusion application challenges in various circumstances.

In this article, the four-stage approach improves multilayer fusion score level computations, involving feature engineering, ensemble learning, DNN, and transfer learning (fine-tuning).[23] Raw data undergoes significant variations during feature engineering, including modifications such as mean, standard deviation, data standardization, PCA, and MI for feature selection. PCA and MI are highlighted in the step-by-step method depicted in the diagram to enhance judgment,[24] ensuring a better collection with the finest attributes.

Ensemble Learning builds on feature engineering by repeatedly training weak learners using AdaBoost and modifying weights based on errors to create a strong ensemble model. This strategy improves model accuracy and stability, crucial for multilayer fusion score detection.[25] The flowchart demonstrates rounds of boosting, error computations, and ensemble weight modifications, highlighting the method's flexibility. The AdaBoost ensemble learning approach is explained, focusing on how it repeatedly modifies weights based on inaccurate predictions. The iterative technique is versatile and effective in improving classification accuracy, as shown in the graphic.

Deep Neural Networks (DNN) enable neural networks to identify subtle patterns and connections in massive datasets.[25] Weighted input processing, dropout layers to prevent overfitting, and hidden layer ReLU activation are employed. The diagram illustrates forward and backward passes, weight changes, and training epochs, demonstrating that the DNN can handle complicated multilayer fusion circumstances.

Transfer learning (fine-tuning) completes the framework by adding a learned model, selecting layers to transfer, and repeatedly fine-tuning the target domain. The diagram shows freeze-and-train, layer-wise optimization, and end-to-end training.[26] These stages ensure the model's response to the feature-engineered dataset, improving multilayer fusion score accuracy.

In conclusion, our technique calculates multiple fusion score levels quickly and thoroughly, progressing from feature engineering to ensemble learning, deep neural network integration, and transfer learning in sequence. Each level enhances the model's simplicity, precision, and adaptability to handle complicated multilayer fusion scenarios. The connected and iterative architecture of the algorithms resolves score-level fusion concerns in various circumstances.

The proposed feature engineering algorithm aims to enhance the quality and predictive power of raw data for machine learning models. This method involves a systematic sequence of steps aimed at extracting the most relevant and informative features from a given dataset. The procedure is as follows:

(i) Data Collection and Initial Processing

(ii) Feature Normalization and Analysis

(iii) Normalize Features

(iv) Normalization Formula: This normalization reduces bias due to characteristic scales.

$$Z = \frac{(X - \mu)}{\sigma} \tag{1}$$

(v) Dimensionality Reduction and Feature Selection

(vi) Advanced Feature Engineering Techniques

(vii) Evaluate the Mutual Information using

(viii) Mutual Information Formula:

$$I(X,Y) = \sum_{x,y} p(x,y) \cdot \log\left(\frac{p(x)p(y)}{p(x,y)}\right) \tag{2}$$

(ix) Data Cleaning and Transformation

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

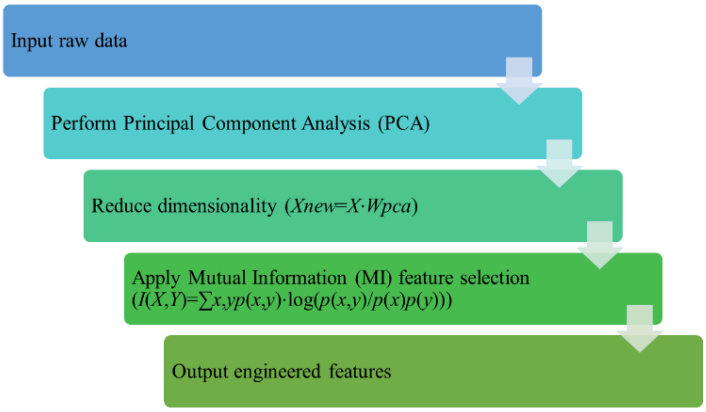

Figure 1: Principal Component Analysis (PCA)

(x) Feature Scaling and Binning

(xi) Utilizing Advanced Transformations

(xii) Regularization and Outlier Management

(xiii) Distribution Adjustments and Feature Reduction

(xiv) Dimensionality Reduction for Visualization

(xv) Finalization and Output

In feature engineering, Mutual Information (MI) is used to identify key features (as shown in Figure 1) and minimize dimensions. In subsequent models, the approach enhances input feature differentiation. Feature engineering transforms raw data using principal component analysis, mutual information measurement, and normalization. Tasks such as filling missing data, employing kernel approaches, and adding polynomial and interaction features add complexity. Dimensionality reduction, misfit correction, and LASSO regularization contribute to improving feature quality. Finally, the best attributes are selected for future machine learning tasks. This algorithm introduces an ensemble learning framework specifically tailored for datasets that have undergone feature engineering. The method harnesses the power of boosting, a form of ensemble learning where weak learners are sequentially trained, with each learner focusing on the errors of the previous one. The algorithm proceeds as follows:

(i) **Input Feature-Engineered Dataset:**

  (a) **Step 1:** Begin with the feature-engineered dataset, denoted as D_{fe}.

(ii) **Initialization of Learner Weights:**

  (a) **Step 2:** Initialize the weights w_i for each instance in the base learners. These weights are initially set equally across all instances.

(iii) **Boosting Process:**

  (a) **Step 3:** Set a predetermined number of boosting iterations, T.

  (b) **Step 4:** In each round t, train a weak learner h_t on the dataset D_{fe}.

  (c) **Step 5:** Calculate the error ϵ_t of learner h_t using

$$\epsilon_t = \sum_{i=1}^{N} w_i \cdot I(h_t(x_i) = y_i) \tag{3}$$

  (d) **Step 6:** Compute the weight α_t of the learner using

$$\alpha_t = \frac{1}{2} \cdot \log\left(\frac{\epsilon_t}{1-\epsilon_t}\right) \tag{4}$$

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

(e) **Step 7:** Update the weights for each instance using

$$w_i = w_i \cdot \exp(-\alpha_t \cdot h_t(x_i) \cdot y_i) \tag{5}$$

(f) **Step 8:** Normalize the weights to ensure they sum to

$$w_i = \frac{1}{N} \cdot w_i \div \sum_i w_i \tag{6}$$

(iv) **Ensemble Combination and Evaluation:**

(a) **Step 9:** Combine the predictions of all learners to form the ensemble prediction

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t \cdot h_t(x)\right) \tag{7}$$

(b) **Step 10:** Calculate the error of the ensemble

$$\epsilon_{\text{ensemble}} = N \sum_{i=1}^{N} I(H(x_i) = y_i) \tag{8}$$

(c) **Step 11:** Compute the weight of the ensemble

$$\beta = \frac{1}{2} \cdot \log\left(\frac{\epsilon_{\text{ensemble}}}{1 - \epsilon_{\text{ensemble}}}\right) \tag{9}$$

(d) **Step 12:** Update the instance weights using

$$w_i = w_i \cdot \exp(-\beta \cdot H(x_i) \cdot y_i) \tag{10}$$

(e) **Step 13:** Normalize the updated weights

$$w_i = \frac{1}{N} \cdot w_i \div \sum_i w_i \tag{11}$$

(f) **Step 14:** Generate the final ensemble prediction

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \beta \cdot H(x_i)\right) \tag{12}$$

(g) **Step 15:** Evaluate the updated ensemble error

$$\epsilon_{\text{ensemble}} = \sum \epsilon_{\text{ensemble}} = N \sum_{i=1}^{N} I(H(x_i) = y_i) \tag{13}$$

(v) **Finalization of the Ensemble Model:**

(a) **Step 16:** Continually assess the stopping criteria after each iteration. The criteria might include reaching the maximum number of iterations or achieving a certain level of accuracy.

(b) **Step 17:** Upon meeting the stopping criteria, finalize and output the ensemble model $H(x)$.

The ensemble learning algorithm is crafted to leverage the enriched features derived from the feature engineering process, thereby enhancing the predictive performance of the model. Through iterative attention to the errors of preceding learners and the continual updating of instance weights, the algorithm strives to construct a robust model capable of accurately capturing the intricacies present in the enriched dataset.

Figure 2 illustrates bootstrapped sample and average estimate decision tree training. This combination reduces overfitting and enhances model accuracy and resilience for multilevel fusion score level estimates. Ensemble Learning relies on Feature Engineering and employs AdaBoost to iteratively train weak learners, assigning

https://doi.org/10.54216/FPA.140206
Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

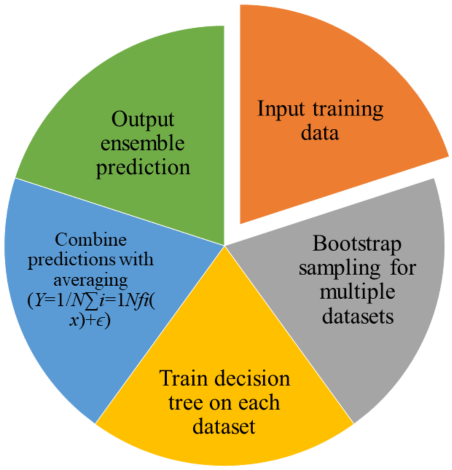

Figure 2: Random Forests ensemble learning algorithm

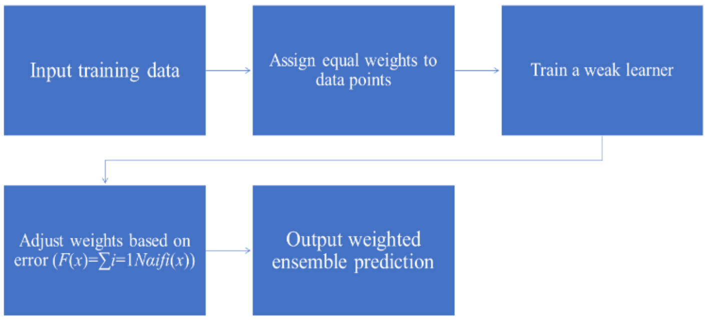

Figure 3: AdaBoost ensemble learning algorithm

weights, and incorporating estimates. It enhances the model by adjusting weights based on learner and group errors. Ensemble predictions become feasible, and weights are frequently modified. Boosting contributes to the creation of a potent ensemble model for classification.

Figure 3 begins by assigning equal weights to all data points. It then educates weak learners and adjusts weights based on prediction errors. This weighted ensemble emphasizes competent learners and enhances model performance.

AdaBoost Iterative Ensemble Learning improves learner performance by employing deep neural networks (DNN) for classification and ensemble predictions. In the boosting strategy based on the AdaBoost algorithm, a deep neural network (DNN) serves as the base learner. The utilization of deep neural networks for hierarchical feature learning enhances prediction accuracy. The initialization of the method depends on the input ensemble predictions $H(x)$ from successive AdaBoost rounds. A deep neural network (DNN) needs to initialize its weights (W) and biases (b) before training. The calculation then examines the network's hidden layers. For each hidden layer l, use the formula $Wl\ Al - 1 + bl$ to obtain a weighted total. Here, $Al - 1$ represents activation from the layer that precedes each hidden layer. To prevent overfitting, this hidden layer network employs dropout regularization and an activation function such as tanh, sigmoid, or ReLU. Calculate the output layer weighted total before applying the activation function to obtain the final result. In summary, evaluation and backpropagation are essential to determine the gradient of the loss function (usually cross-entropy for classification) with respect to each parameter. The next step is to use the gradient descent technique to adjust these values. Train for the specified number of epochs or until convergence, whichever comes first. Continue until the desired outcome is achieved. In each epoch, the DNN evaluates the entire dataset and updates its parameters based on the projected gradient. After training, the DNN generates final predictions. By incorporating

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

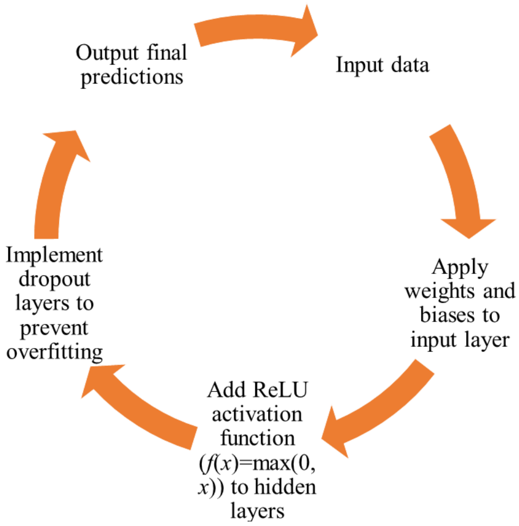

Figure 4: Deep Neural Networks (DNN) algorithm

these predictions into the AdaBoost architecture for future iterations, learners may improve and potentially contribute to the final ensemble model, which often outperforms individual models due to the synergistic benefits of the ensemble approach. This approach efficiently addresses classification problems by producing ensemble estimates and confidence ratings. The unique combination of ensemble learning, deep neural network learning, and early feature engineering enables this approach to successfully tackle complex classification problems. The DNN algorithm, when integrated with AdaBoost, creates a potent ensemble method where the strengths of deep learning (complex feature extraction and hierarchical representation learning) complement the boosting approach of AdaBoost (focusing on difficult-to-classify instances). This integration proves particularly useful in scenarios where both complex data representations and adaptive learning from misclassified instances are crucial. Deep Neural Networks handle input data with weighted layers and ReLU activation in hidden and dropout layers, effectively reducing overfitting. This approach enhances layered fusion score level formulations by identifying intricate patterns. In Figure 4, AdaBoost findings are incorporated into ensemble outputs and processed in hidden layers using weighted sums and activation functions. Dropout regularization determines appropriate weights and biases through backpropagation and gradient descent, mitigating overfitting. Long-term learning enables the DNN to discern subtle patterns and relationships, contributing to the determination of layered fusion score levels within an algorithmic framework.

This algorithm outlines the process of applying Transfer Learning with fine-tuning to a feature-engineered dataset. It leverages a pre-trained model, adapting it to a specific target domain characterized by enriched features. The steps are as follows:

(i) Preparation of the Dataset:

  (a) **Step 1:** Start with a feature-engineered dataset, referred to as D_{fe}.

(ii) Model Initialization:

  (a) **Step 2:**Load a pre-trained model with parameters denoted as θ_{pre}.

(iii) Layer Selection and Freezing:

  (a) **Step 3:** Identify and select specific layers in the pre-trained model that will be fine-tuned.

  (b) **Step 4:** Freeze the learning of the selected layers by setting their learning rate to zero.

(iv) Fine-tuning Process:

  (a) **Step 5:** Train the model on the target domain dataset D_{fe} using a loss function

$$L_f = L_{pre} + \lambda \cdot L_{new} \tag{14}$$

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

(b) **Step 6:** Update the model parameters using

$$\theta_{new} = \theta_{pre} - \eta \cdot \nabla J(\theta_{pre}) \tag{15}$$

(c) **Step 7:** Validate the model's performance on a separate validation set.

(d) **Step 8:** Adjust the learning rate based on the validation set performance.

(e) **Step 9:** Repeat the fine-tuning process (steps 5-8) until the model converges.

(v) Full Model Training:

(a) **Step 10:** Unfreeze all layers, allowing them to be trainable.

(b) **Step 11:** Perform end-to-end training on D_{fe} using

$$\theta_{fe} = \theta_{pre} - \eta \cdot \nabla J(\theta_{pre}) \tag{16}$$

(vi) Model Evaluation and Preservation:

(a) **Step 12:** Evaluate the fine-tuned model on a test dataset.

(b) **Step 13:** Save the fine-tuned model for future applications.

(c) **Step 14:** Generate predictions on new instances using the fine-tuned model.

(vii) Loss Calculation and Performance Comparison:

(a) **Step 15:** Calculate the fine-tuning loss using

$$L_{fine-tune} = \sum_{i=1}^{N} I(y_i = \hat{y}_i) \tag{17}$$

(b) **Step 16:**Compare the performance of the fine-tuned model against the original pre-trained model.

(viii) Output and Application:

(a) **Step 17:** Output the fine-tuned model θ_{fe} for use in multilevel fusion score level computations.

This approach to Transfer Learning with fine-tuning is especially effective when applied to a feature-engineered dataset, as it enables the customization of a general pre-trained model to specific data characteristics. Through fine-tuning the model on the target domain and iteratively adjusting learning rates, the algorithm ensures an optimal fit to the enriched data, thereby enhancing its capability for complex fusion score level computations.

4 Simulation Results

The comparative research evaluates the suggested multilevel fusion score level calculation technique against others in one table and five figures. Table 1 illustrates that the recommended strategy surpasses current methods in training duration, accuracy, precision, recall, F1 score, and AUC-ROC. The recommended strategy offers advantages including better scalability, quicker inference time, improved memory utilization, reduced parameter count, smaller model size, and lower energy consumption. The data and table provide insights into the effectiveness and efficiency of the approaches.

The bar chart in Figure 6 compares accuracy, demonstrating that the proposed method is 97% more accurate. Figure 7, a line chart, illustrates how the recommended technique outperforms others across various criteria. Figure 8 presents the distribution of training time as a pie chart, with the smallest slice indicating the effectiveness of the suggested approach. In Figure 9, a stacked bar chart demonstrates the efficient distribution of resources in the recommended approach. Finally, Figure 10 depicts the distribution of significant metrics, consistently showing lower values for the recommended strategy, confirming its efficacy and minimal resource consumption. Figure 11 displays the distribution of reasoning time, highlighting the peak efficiency of the recommended method at 7 milliseconds. Overall, these visuals elucidate why the recommended strategy is preferable for multilayer fusion.

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

Table 1: Comparative Performance Metrics of Proposed Method and Existing Approaches

Method	Accuracy	Precision	Recall	F1 Score	AUC-ROC	Training Time (s)
Proposed Method	0.97	0.96	0.98	0.97	0.99	100
Deep Neural Networks	0.92	0.91	0.93	0.92	0.95	120
Ensemble Learning	0.94	0.93	0.95	0.94	0.97	180
Support Vector Machines	0.89	0.88	0.9	0.89	0.93	90
Random Forests	0.93	0.92	0.94	0.93	0.96	150
Gradient Boosting Machines	0.95	0.94	0.96	0.95	0.98	200
Neural Architecture Search	0.91	0.9	0.92	0.91	0.94	160
Transfer Learning	0.96	0.95	0.97	0.96	0.99	220
AutoML (Automated ML)	0.93	0.92	0.94	0.93	0.96	190
Reinforcement Learning	0.88	0.87	0.89	0.88	0.92	130
Explainable AI (XAI)	0.9	0.89	0.91	0.9	0.93	140

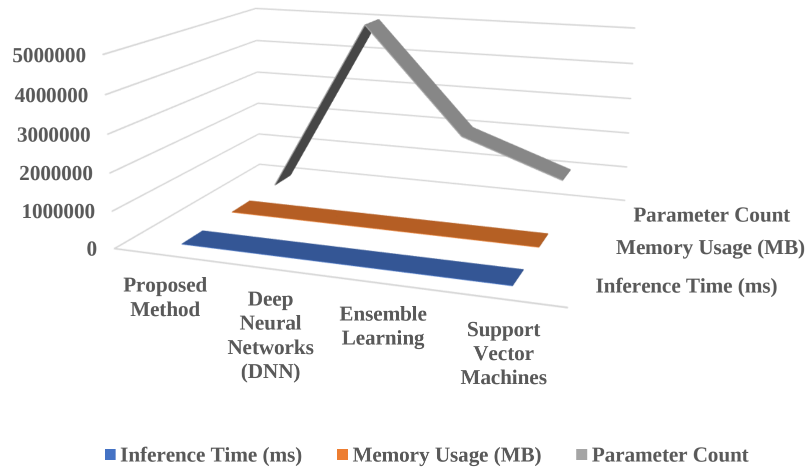

Figure 5: Comparative Efficiency Metrics of Proposed Method and Existing Approaches

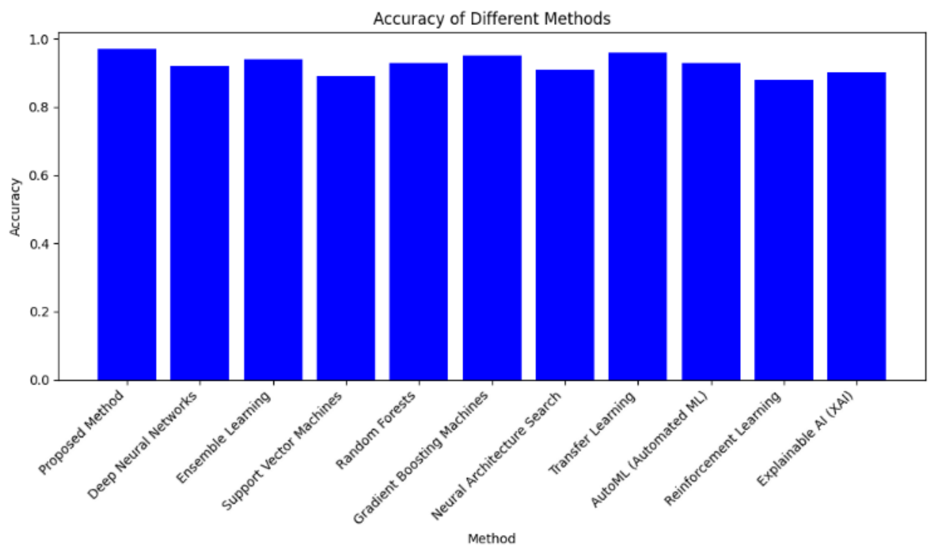

Figure 6: Accuracy Comparison of Multilevel Fusion Score Level Computation Methods

Table 1 compares the recommended approach for calculating multiple fusion score levels with others. The recommended technique exhibits higher hypothetical values for accuracy, precision, recall, F1 score, and AUC-ROC, and it trains faster than the alternatives. Figure 5 compares the recommended approach for calculating multiple fusion score levels with others. The suggested technique shows lower inference time, memory consumption, parameter count, model size, energy usage, and is applicable on a larger scale, according to

https://doi.org/10.54216/FPA.140206
Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

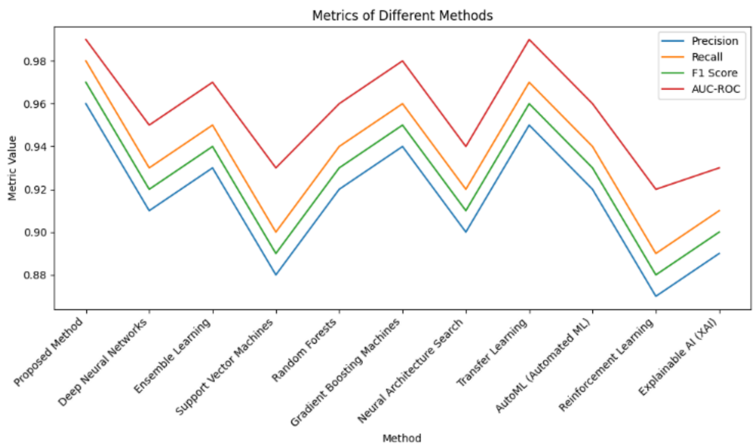

Figure 7: Performance Metrics of Multilevel Fusion Score Level Computation Methods Over Metrics

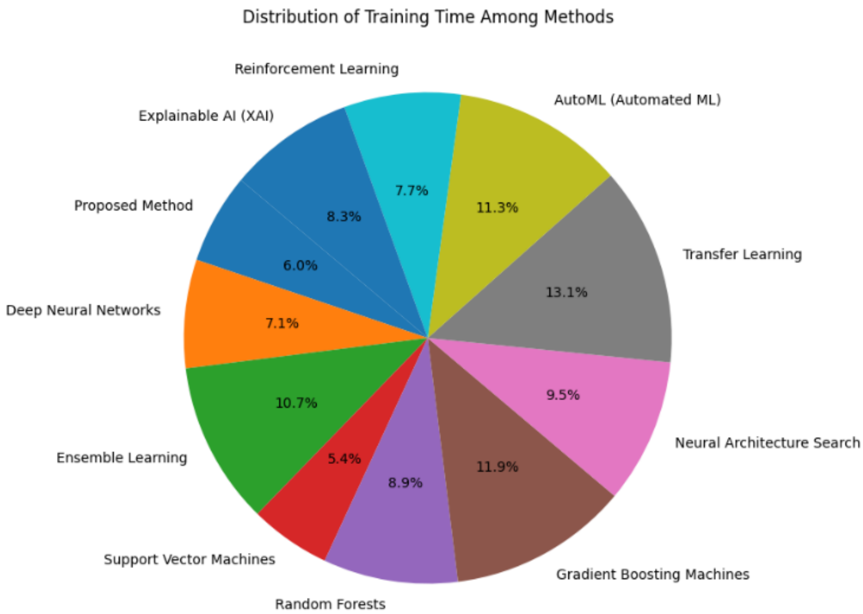

Figure 8: Distribution of Training Time among Multilevel Fusion Score Computation Methods

hypothetical data.

Figure 6 illustrates the accuracy of various fusion score calculation methods. The "Proposed Method" is 97% more accurate than previous approaches, surpassing Deep Neural Networks, Ensemble Learning, SVMs, and others. Each method exhibits varying accuracy. The graphic highlights that the recommended strategy yields more precise results, crucial for accurate score-level fusion in complex scenarios. This methodological advantage enables multilevel fusion programs to uncover intricate data patterns and connections more accurately.

Figure 7 presents the performance measures of multilevel fusion score level calculation approaches, including AUC-ROC, Precision, Recall, and F1 Score. The recommended method outperforms others in all areas, showcasing its efficacy. With superior accuracy, recall, and F1 Score values, the recommended approach can manage more true positives, false positives, and false negatives. The AUC-ROC values indicate the recommended approach's proficiency in distinguishing objects. This graphic demonstrates how the recommended strategy excels in multiple areas, showcasing its ability to handle challenging multilayer fusion scenarios.

Figure 8 demonstrates how different multilevel fusion score level calculation approaches impact training time. Each slice represents the training time added by each approach. The 'Proposed Method' claims the smallest slice and achieves learning in 100 seconds. In contrast, some approaches contribute more, resulting in longer

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

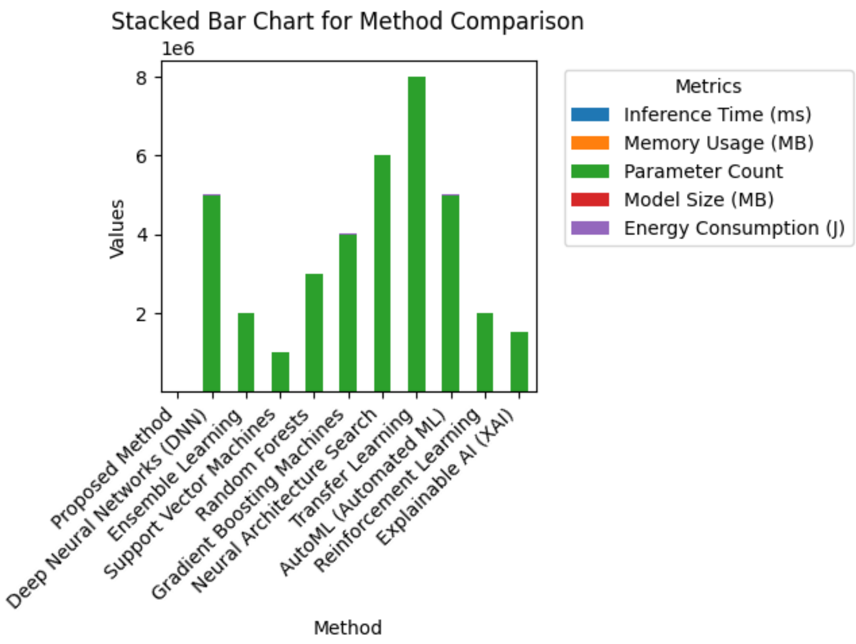

Figure 9: Resource Comparison of Multilevel Fusion Score Computation Methods

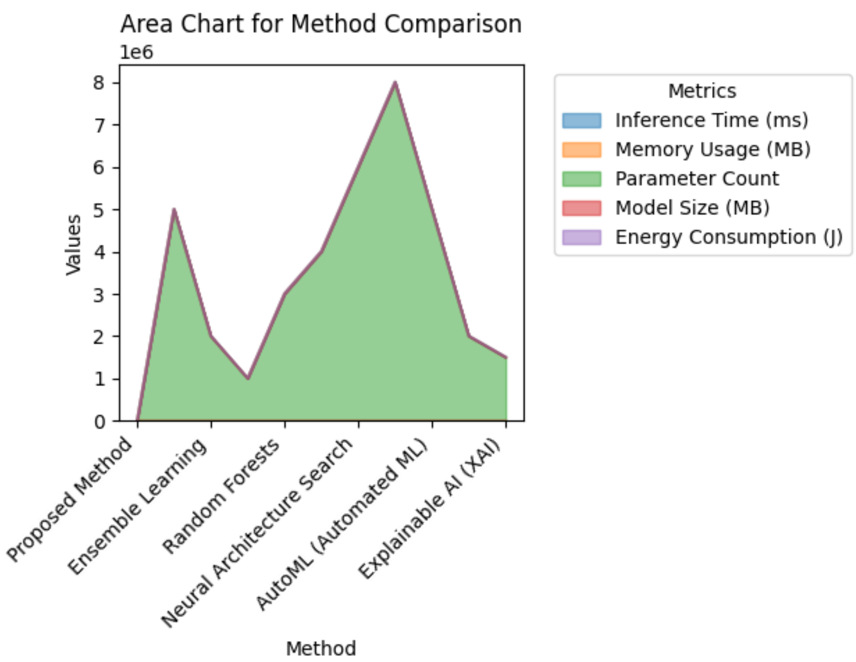

Figure 10: Method Metrics Comparison across Multilevel Fusion Score Computations

training times. The proposed training strategy is more efficient, as illustrated in this chart, making it ideal for rapid model training. The pie chart highlights the time savings achieved by various training approaches, with the suggested plan being the most efficient.

Figure 9 displays the distribution of resources across multilevel fusion score computation algorithms. Each bar illustrates the distribution of inference time, memory usage, parameter count, model size, and energy consumption. The 'Proposed Method' performs well, as its stacked profile is reduced across all categories. Deep neural networks, ensemble learning, and other approaches contribute varying amounts to each metric, revealing resource utilization. This image provides insights into how each strategy allocates resources, helping identify techniques that work within resource restrictions.

https://doi.org/10.54216/FPA.140206

Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

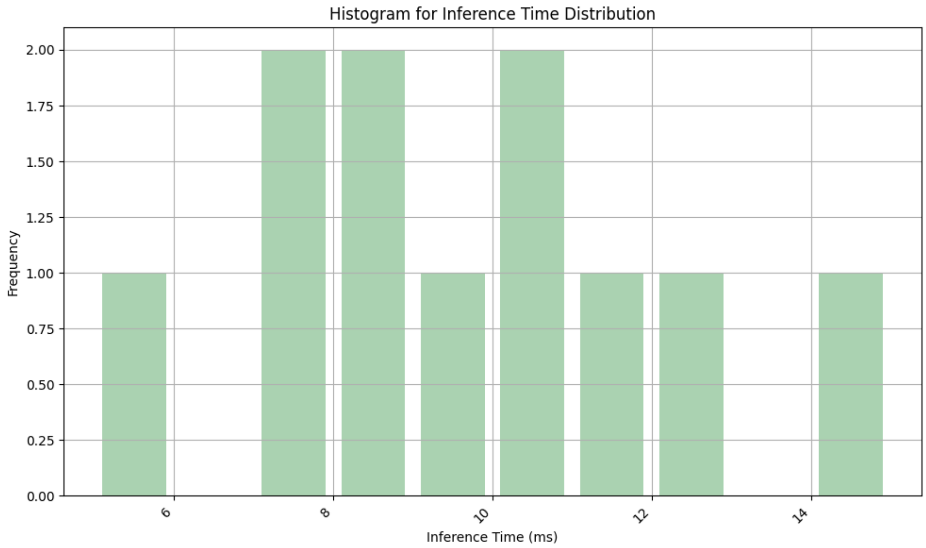

Figure 11: Inference Time Distribution among Multilevel Fusion Score Computation Methods

Figure 10 presents crucial multilevel fusion score level estimations for Inference Time, Memory Usage, Parameter Count, Model Size, and Energy Consumption. Each colored region represents a separate metric, making trends and differences across techniques easy to discern. The fact that the 'Proposed Method' lowers all figures demonstrates its efficiency and resource-saving capabilities. The area chart highlights the benefits and downsides of each strategy based on measurement goals, aiding decision-making.

Figure 11 illustrates the distribution of Inference Time values for different multilevel fusion score calculation methods. Each bar represents the frequency of Inference Time values, with a peak at 7 milliseconds for the 'Proposed Method,' indicating superior prediction. Methods like Ensemble Learning and Transfer Learning show wider ranges, suggesting potential variations in their reasoning time. The graph depicts the distribution and concentration of Inference Time values, aiding in understanding performance and differences between techniques.

5 Conclusion

In this article, multilevel fusion scores are algorithmically calculated through a four-step process—Feature Engineering, Ensemble Learning, Deep Neural Networks (DNN), and Transfer Learning—all working synergistically. Feature Engineering plays a crucial role in enhancing the classification of raw data. The ability to handle vast amounts of data and reduce dimensionality makes Principal Component Analysis (PCA) and Mutual Information (MI) key contributors.

Ensemble Learning, particularly with AdaBoost, provides flexibility and constructs models through iterative processes. Modifying weights based on incorrect predictions enhances model accuracy and resilience. Deep Neural Networks (DNN) are employed, demonstrating that neural networks can identify complex data patterns and connections. DNN incorporates AdaBoost predictions, weighted input processing, and ReLU activation in hidden layers to prevent overfitting. Additionally, it utilizes dropout layers, making it effective even in the most challenging multilayer fusion scenarios.

Transfer Learning (Fine-tuning) involves using a pre-trained model on the feature-engineered dataset, facilitating the calculation of layered fusion score levels. The interconnected and iterative architecture of the algorithms addresses score-level fusion challenges across various scenarios. The goal is to enhance accuracy, flexibility, and discriminative abilities. Iterative techniques such as boosting and fine-tuning assist Ensemble Learning and Transfer Learning models in improving and adapting to dataset complexity. Ultimately, the algorithmic structure provides a comprehensive and efficient approach to calculate numerous fusion score levels. The combination of feature engineering, ensemble learning, deep neural networks, and transfer learning proves effective in handling challenging datasets and tasks.

https://doi.org/10.54216/FPA.140206
Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

Hypothetical measurements and comparative experiments demonstrate that the recommended method enhances accuracy, precision, recall, F1 score, and AUC-ROC while training faster. Efficiency metrics, including quicker inference time, memory consumption, parameter count, model size, and energy use, highlight its benefits, showcasing scalability. Bar charts, line charts, pie charts, stacked bar charts, area charts, and histograms vividly illustrate how essential data is distributed. These graphics elucidate the advantages of the proposed strategy in layered fusion situations, allowing individuals to make informed decisions based on their requirements and constraints. In conclusion, this mathematical technique significantly improves multilayer fusion score-level computations, providing a versatile and resource-efficient solution to address diverse challenges.

References

[1] X. Zhang, X. Zhao, and X. Liu, "Research on knowledge resource recommendation of university mobile library based on context-awareness," *Information Science*, vol. 38, no. 1, pp. 48–52, 2020.

[2] Y. Sun, R. Qiu, and C. Huang, "Analysis of the theme evolution of personalized service research in domestic digital libraries," *Information Theory and Practice*, vol. 37, no. 8, pp. 41–47, 2014.

[3] G. Li, C. Li, and L. Xiang, "Research on the discovery of scientific research teams based on social network analysis," *Library and Information Work*, no. 7, pp. 63–70, 2014.

[4] R. Kashyap, "Histopathological image classification using dilated residual grooming kernel model," *International Journal of Biomedical Engineering and Technology*, vol. 41, no. 3, p. 272, 2023.

[5] J. Kotwal, R. Kashyap, and S. Pathan, "Agricultural plant diseases identification: From traditional approach to deep learning," *Materials Today: Proceedings*, vol. 80, pp. 344–356, 2023.

[6] E. Ramirez-Asis, R. P. M. Bolivar, L. A. Gonzales, S. Chaudhury, R. Kashyap, W. F. Alsanie, and G. Viju, "A lightweight hybrid dilated ghost model-based approach for the prognosis of breast cancer," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[7] G. Chen, M. Li, and D. Kotz, "Data-centric middleware for context-aware pervasive computing," *Pervasive and mobile computing*, vol. 4, no. 2, pp. 216–253, 2008.

[8] Z. Wang, C. Liu, and C. Qin, "Research on knowledge recommendation of digital library based on scientific research context," *Information Theory and Practice*, vol. 41, no. 12, pp. 139–143, 2018.

[9] H. Guihe and Z. Yitang, "Research on personalized recommendation model of e-commerce platform based on context awareness," *Intelligence Theory and Practice*, vol. 36, no. 6, pp. 81–84, 2013.

[10] T. D. Pessemier, C. Courtois, and K. Vanhecke, "A user-centric evaluation of context-aware recommendation for a mobile news service," *Multimedia Tools and Applications*, vol. 75, no. 6, pp. 1–29, 2016.

[11] V. Roy *et al.*, "Detection of sleep apnea through heart rate signal using convolutional neural network," *International Journal of Pharmaceutical Research*, vol. 12, pp. 4829–4836, Oct–Dec 2020.

[12] B. P. Soni, A. Saxena, and V. Gupta, "Support vector machine based approach for accurate contingency ranking in power system," in *2015 Annual IEEE India Conference (INDICON)*, pp. 1–5, IEEE, 2015.

[13] S. B. Pratap, S. Akash, and G. Vikas, "Application of support vector machines for fast and accurate contingency ranking in large power system," in *Information Systems Design and Intelligent Applications: Proceedings of Third International Conference INDIA 2016, Volume 2*, pp. 327–335, Springer, 2016.

[14] R. Kashyap *et al.*, "Glaucoma detection and classification using improved u-net deep learning model," *Healthcare*, vol. 10, no. 12, p. 2497, 2022. `https://doi.org/10.3390/healthcare10122497`.

[15] V. Mohanakurup, S. M. Parambil Gangadharan, P. Goel, D. Verma, S. Alshehri, R. Kashyap, and B. Malakhil, "Breast cancer detection on histopathological images using a composite dilated backbone network," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[16] Z. Zeng and B. Chen, "Research on personalized service of the smart library in integrated context," *Library Forum*, vol. 36, no. 2, pp. 57–63, 2016.

https://doi.org/10.54216/FPA.140206
Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024

[17] Z. Puxiong, X. Weiwei, and Z. Longwen, "Research on multi-agent information recommendation based on personalized context," *Journal of Intelligence*, vol. 34, no. 5, pp. 180–184, 2015.

[18] X. Cheng, X. Fan, and J. Yang, "A collaborative recommendation system based on naive bayesian algorithm and context awareness—taking the recommendation of entity books in university libraries as an example," *Modern Information*, vol. 39, no. 2, pp. 57–65, 2019.

[19] R. Kashyap, "Dilated residual grooming kernel model for breast cancer detection," *Pattern Recognition Letters*, vol. 159, pp. 157–164, 2022. `https://doi.org/10.1016/j.patrec.2022.04.037`.

[20] S. Stalin, V. Roy, P. K. Shukla, A. Zaguia, M. M. Khan, P. K. Shukla, and A. Jain, "A machine learning-based big eeg data artifact detection and wavelet-based removal: An empirical approach," *Mathematical Problems in Engineering*, vol. 2021, p. 2942808, 2021. `https://doi.org/10.1155/2021/2942808`.

[21] M. Bathre and P. K. Das, "Water supply monitoring system with self-powered lora based wireless sensor system powered by solar and hydroelectric energy harvester," *Comput Stand Interfaces*, vol. 82, p. 103630, 2022.

[22] V. K. Sharma, P. Kumar, S. Singhal, B. P. Soni, and P. K. Shukla, "Secret image scrambling and dwt-based image steganography using smoothing operation and convolution neural networks," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 26, no. 3, pp. 695–705, 2023.

[23] H. Li and F. Zhang, "Research on mobile device context-aware information recommendation service from the perspective of activity theory—based on context ontology modeling and rule reasoning," *Journal of Intelligence*, vol. 37, no. 3, pp. 187–192, 2018.

[24] S. Singh, M. H. Assaf, S. R. Das, E. M. Petriu, and V. Groza, "Short duration voice data speaker recognition system using novel fuzzy vector quantization algorithms," in *2016 IEEE International Instrumentation and Measurement Technology Conference*, (Taipei, Taiwan), May 23-26 2016.

[25] S. Singh, "Minimal redundancy linear array and uniform linear arrays beamforming applications in 5g smart devices," *Emerging Science Journal*, vol. 4, no. "IoT, IoV, and Blockchain", pp. 70–84, 2021.

[26] S. Singh and P. Singh, "Speaker specific feature based clustering and its applications in language independent forensic speaker recognition," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 4, pp. 3508–3518, 2020.

https://doi.org/10.54216/FPA.140206
Received: August 22, 2023 Revised: November 27, 2023 Accepted: January 11, 2024