Enhanced Algorithm Blend and Turing for Improving Quality-of-Service of Multiple Datasets

Meer J. Khan[1], Mohsin A. Farhad[1], Nurgul Nalgozhina[1] and Abdul Razaque[1]

[1]*International Information Technology University, Manas St. 34/1, Almaty, 050040, Kazakhstan*

Abstract

This article proposes a new strategy built on a strengthened Machine Learning (ML) engine to enable better metrics and results. It tackles issues with bias, overfitting, low accuracy, and poor generalization in the current techniques. The proposed model's mechanics can be summarized to the following: a) applications of a chosen set of supervised learning (SL) algorithms on the experimental dataset and storage of the determined readings in a logical table (LT) construct, b) creation of a blend of algorithms based on in-parallel tuning of the model to enhance classifier learning, c) expansion of a logical 3-D cube, and LT, that define the algorithms for ensuring optimum fitting for the e) Limiting the over/under learning by employing Local Error (LE) and Global Error (GE) boundaries, and optimizing the internals of the model by creating Local Gain (LG) and Global Gain (GG) functions. Additionally, it supports the validity of the suggested model, which includes sub-algorithms for a variety of real-world data sets for both prescriptive and predictive analytics. We create a special method called "parallel tuning processes" for better performance/fitting. Any type of dataset and predicting objective can be used with this model. We offer entire mathematical models, algorithms, and all essential frameworks, visualizations, and diagrams to facilitate the execution of the given plan. Several simulation results are shown with an in-depth examination of the data to demonstrate the viability of the suggested system.

Keywords

Data mining, predictive modeling, machine learning, algorithm blending, tuning integration, optimum fitting, overfitting, QoS, logical table

1. Introduction

The performance of machine learning depends significantly on the properties of the algorithms [1]. Modern algorithms used in machine learning (ML) let systems make more accurate decisions and successfully forecast events [2–3]. Under-fitting, over-fitting, bias, mistakes, and poorer accuracy are a few of the ML difficulties still to be solved. Numerous algorithms have been developed, although they may not be fully capable of solving these problems [5-7]. However, given the availability of digital and e-data, notably in the fields of Big data and social networking data, ML has advanced significantly in data science [8–9]. Predictive analytics in ML uses a variety of models, approaches, and algorithms [10–11]. The focus of the study has been combining models and characteristics.

The suggested solution is used to train the model for the determination of the supporting algorithm and then blend it to produce higher prediction accuracy and classifier fitness. The following subsections introduce the concept, questions, contributions, and anatomy of the study. Our motivation stems from the need to better predictive analytics by combining social networking data with professional and academic data. Several cutting-edge algorithms have been investigated to better comprehend the suggested solution [15-18]. This work seeks to answer/address the following general research questions:

- Can matching, fitness, and accuracy scores be used to mix and fine-tune algorithms?

DTESI 2023: Proceedings of the 8th International Conference on Digital Technologies in Education, Science and Industry, December 06–07, 2023, Almaty, Kazakhstan
j.meer@iitu.edu.kz (M.J. Khan); m.farhad@iitu.edu.kz (M.A. Farhad); n.nalgozhina@iitu.edu.kz (N. Nalgozhina), a.razaque@iitu.edu.kz (A. Razaque)

0009-0004-3147-1400 (M.J. Khan); 0000-0001-9737-3273 (M.A. Farhad); 0000-0003-0254-8670 (N. Nalgozhina), 0000-0003-0409-3526 (A. Razaque)

- Does the combined version perform better than a single algorithmic model when it comes to addressing bias, over- or under-fitting, poor generalizability, and low accuracy?
- Can algorithms be added and removed algorithmically in real-time as the classifier processes the provided raw data for ML predictions?
- Is there a logical space where overfitting and underfitting can be pushed to achieve the best fitting? Is it possible to use or program an error to control the model's lower and upper bounds in order to prevent bias and overlearning?
- Can a model (such as a mixed one) learn from its errors (such as incorrect predictions)?
- Lastly, can a blended model improve from bad predictions?

1.1 Research contributions

We observe the potential of data science in the creation of new algorithms as well as enhancements to current approaches taking into account the most recent requirements and applications in the big data environment. The following is a summary of the paper's contributions:

- To develop a tuning and blended process, the proposed solution can quantify the matching, fitness, and accuracy scores.
- Our suggested method demonstrated that blending may be done based on how well each algorithm, as chosen from the pool of options, fits the sort of data model being trained. This method guarantees that the model is neither biased nor overfit in comparison to any particular algorithm being tested.
- We trained the suggested classifier so it can learn to either add a good-fit algorithm (one with improved metrics) or remove a bad-fit algorithm (one with inferior metrics).
- We have created enhanced metrics that control the model's performance in one particular area. In this manner, the suggested model is simultaneously cross-checked and trained on the data. The specifics of eWPM are not the subject of this essay, though.
- For maximum fitness, we have changed our model to z-space coordinates in 3-D logical space to visualize the architecture of our proposed model.
- We have created a unique method for applying error, a crucial statistic in machine learning. The proposed model is taught to use minimal and maximum error bounds when administering. As a result, it is possible to maintain acceptable bias and fitness, including overlearning. For both GG and LG, the lowest and maximum error bounds are 20% and 80%, respectively.
- We have developed improved metrics, which regulate the dimension of the performance of the model. This way, the proposed model is cross-checked in parallel while it learns from the data (during training). However, the detailed discussions of eWPM are outside of the scope of this article.

1.2 Research problem

Knowing which algorithm or model is ideal for the problem(s) or data to be used is the main obstacle data scientists, researchers, and analysts face. There are a lot of ambiguities and unknowns. Depending on the variables and/or features of the dataset, each method may respond differently. For different classifiers, evaluation of a model or method is essential. The selection of a classifier type based on fresh or unexplored data becomes extremely difficult. Scientists and researchers have adopted the normal practice of evaluating some of the classifiers that the algorithm generates. We view this as a research challenge to determine if an improved model and algorithms can be provided for such predictive modeling using improved blending and tuning. These algorithms are specially constructed with built-in parallel processing.

1.3 Mechanics of parent research

This research manifests itself in the following sub-components:

- Enhanced Algorithm Blend and Tuning (eABT): Based on the fitness scores and increased accuracy of several SL algorithms, it builds the enhanced machine learning engine (iMLE) model. Based on what is commonly known as Ensembling, Bagging, and Boosting, it applies the existing ML and predictive modeling techniques (e.g., Logistic Regression, Linear Regression, Multiple Regression, Bayesian, Decision trees, SVM, and Classification) to determine the ideal blend until the improved metrics are measured. This plan makes use of the approaches' capability for parallel processing. Errors and incorrect predictions made during a test are fed back to the algorithm in this phase so that it can continue to learn from them (artificial intelligence).
- Enhanced Feature Engineering and Selection (eFES): It improves feature engineering (such as feature creation and trans-formation) and feature selection in order to extract more data (predictor variables that were previously unknown) and the best set of features/attributes for the relationship between "predictor-target" variables. It ensures that features with the highest fitness scores are included, while those with the lowest are removed. Each feature is automatically validated by the eFES model.
- Third, Enhanced Weighted Performance Metric: This component creates a novel metric based on accepted metrics, while the prior two components make sure the model is neither overfit nor underfit. With the help of this metric, ML-based blended models can be trained while maintaining the integrity of their overall performance metrics. Improved Cross Confirmation and Split (iCCS): To improve on current validation techniques like cross-validation, this block of the model looks at and applies a special approach for train-test data splitting. In comparison to the other three (iMLE) components, this one is currently in its infancy.

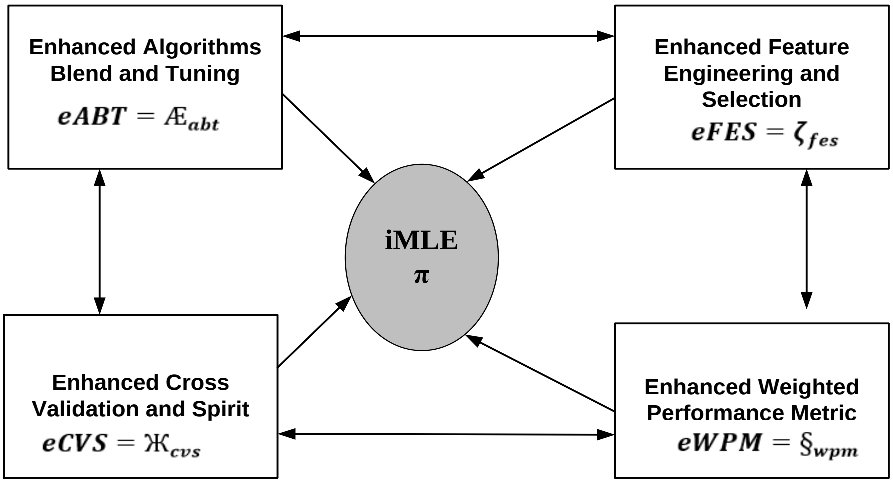

Figure 1: The higher notion of an improved machine-learning engine

1.4 Research methodology

The foundation of our proposed paradigm will be built on extensive study, familiarity, and evaluation. As a result, we put these methods into practice using already-existing libraries and reported the outcomes. For a variety of research issues and datasets, we reviewed the literature and recorded their conclusions. As may be seen in the results in the following sections, we compared some of the outcomes with our suggested model. Future research is still needed for several of the methods, including Random Forest (DT type), Dimensionality Reduction (PCA), and Boosting techniques. Furthermore, this article does not cover unsupervised or semi-supervised learning. The goal of this work is to understand the current status of these algorithms in machine learning so that we can advance it with the work we conduct, as described in this paper. In conclusion, because of the nature of the problem and the datasets that the algorithms are designed for, a genuine comparison can be difficult. In other words, in some circumstances, one may perform better than the other and the opposite outcomes may be seen. The scope of this article does not include a comprehensive examination or comparison of each through experimentation. the general comparison of several of the algorithms/techniques under consideration. The Support vector machine (SVMs), K-Nearest Neighbor (KNN), Logistic Regression, Multiple Linear Regression (MLR), Decision tree (DR), and Graph-based boosting (GB) algorithms were tested in 10 experiments (using 1000+ data samples) for different datasets. The final results are shown in Figure 2 [20]. Implemented and contrasted with current methods is the iMLE. We employed CART and ID3 for DT, and it was shown that iMLE was effective with bigger amounts of data. It is crucial to note that we discovered through our own research and analysis of the literature that the results of different algorithms are not always reliable. This observation served as one of the driving forces behind our decision to conduct this research in order to stabilize the results and develop a modified version of these algorithms that might be applicable to a wide range of problems and datasets.

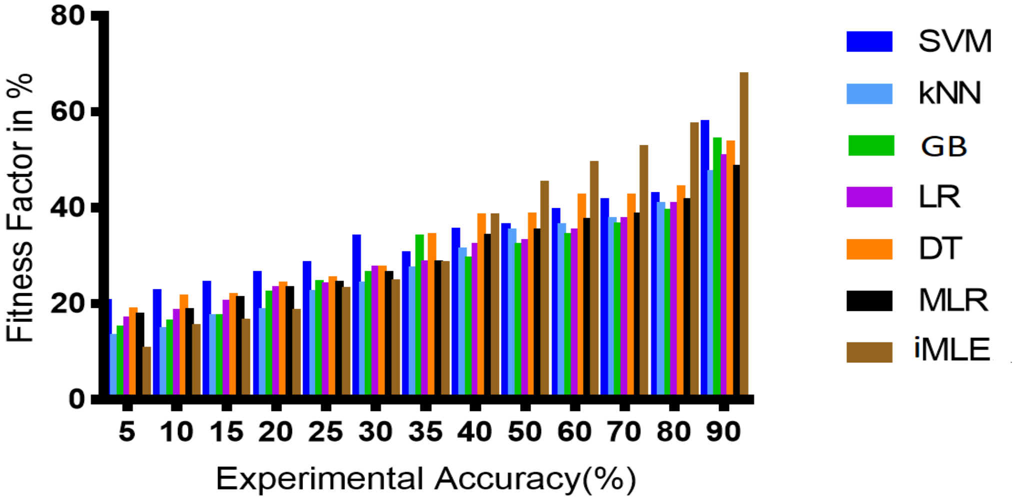

Figure 2: Comparison of known algorithms using the accuracy-fitness relationship

1.5 Paper structure

The remainder of the paper is organized as follows:

Section 2 presents the proposed enhanced algorithm blend and Turing. Section 3 provides experimental results and discussion. Section 4 concludes the entire paper and provides future work.

2. Enhanced algorithm blend and Turing

The most recent developments in ML research have demonstrated a tremendous potential for parallel algorithm evaluation [21]. We make use of a research opportunity to advance the state of the art in machine learning, as we mentioned in earlier parts. The theorems, internal equations, and algorithms that collectively form the suggested model are presented in the following sections.

2.1 Theory and mathematical model

The constructs, theorems, and algorithms that form the foundation of the eABT model and framework are developed in this section. The eABT model is made up of three main parts: the logical table, the internals (three theorems), and the final framework.

2.1.1 eABT-logical table

This article does not cover all of the eABT Logical Table (LT)'s modular details; see the appendix for further information. We provide a succinct introduction to this eMLEE module's discussion on eABT. LT runs in memory and receives dynamic updates. It monitors the algorithms. As the ML process develops to get the ultimate optimum fitting after it has taken into account all of the algorithms from the pool, A(x, y, z) = {A1, A2, ..., An}. This facilitates the best blending and tweaking. This logical table contains data based on three dimensions, where 'x' denotes over-fitness, 'y' denotes under-fitness, and 'z' denotes optimum fitness.

2.1.2 eABT-internals

It covers the matching factors that are given below:

Theorem 1: Algorithm Pre-processing - (Algo.Eval ({A_({1,,,.n)})

Proof: *There must exist a matching factor (M.F) for optimum fitness between two algorithms being evaluated, so let ∇d(A_1, A_2) be the Euclidean distance between two algorithms, such that Op.F{0:1} ≥ 0.5, as optimum Fitness scores and 0.2<err|Err{0:1} < 0.8, are bounds of LE (err) in GE (Err). Let S(x. y, z) determine the suitability scores for the fitness of the given algorithm in 3-D space, with the pointers being x = over-fitness, y = under-fitness, and z = optimum fitness. Let ψ(n) be the classifier function, that the model learns to be able to classify the optimum blend of algorithms for a given dataset and problem.*

Algorithm evaluation deals with applying various algorithms one by one to observe the outcomes (i.e. measures) and then prepare the model for risk estimation and algorithm blending. This construct also compares the two algorithms at the same time and then groups them based on Euclidean distance for similarity scores in terms of fitness. This sub-model finally produces the set of algorithms for the best fit for a given dataset. The end goal is to engineer the master function ψ(n).

Let us define that standard distance function,

$$\nabla d(A_1, A_2) = \sqrt{\left(\sum_{i=1}^{M} (H_i(p_i) - \left(H_i(p_i)\right)^2)\right)} \tag{1}$$

Where S' is the strategy function for all dimensions, and p is the training problem set (dataset vector) in the distribution of time τ. (D, T) being the arguments for the distance function in ∅.

$$\mathbb{S}(x.y, z) = \sum_{j=1}^{N} \emptyset(D_j,\ \tau(p_j, S'_{x,y,z}) \tag{2}$$

Where

$$\emptyset(D, t) = \begin{cases} \dfrac{\bar{T}}{(D+1)}\ ,\ \ t = \infty \\ \dfrac{t}{(D+1)}\ \ \ e \in N,\ \ \bar{T} \geq T \end{cases} \tag{3}$$

Let us assume a raw dataset to be DS(sig, noi), 'sig' shows the signal and 'noi' shows the noisy component of the dataset, and a class classifier function Ć, with loss function as L(x, y, z) | (0:1), for which we iterate in n-sample blocks such that Loss function remain in the defined boundary as estimated, for which the feature sets exist in Function $F = \{f_1, f_2, f_3, \dots \dots .., f_n\}$ with optimum score > 0.5 as determined in the theorem. The classifier in the distribution D, for n blocks of data sample, for the upper bounds of the generalization error, is given by:

$$\acute{C}(x, y, z) = L(x) \le L(y) \le L(z) + \frac{1}{n}\sum_{k=1}^{n}(U.Des - L.Des)^k + \sqrt[3]{\frac{-\log\left(\frac{p}{2}\right)}{2\varphi}} \tag{4}$$

Where φ the pattern identified as signals (removing noises) with the probability of 1 – p. Here we construct a simple rule to estimate the loss function in signal data(S) and noisy data(N) that impacts the classifier design such that:

$$L(DS(S(x, y, z), N(x, y, z) = \begin{cases} 0, & (N(x, y, z) \ge 0.5 \ \ge S(x, y, z)) \\ 1, & (S(x, y, z) \ge 0.5 \ \ge N(x, y, z)) \end{cases} \tag{5}$$

Thus, for each algorithm in the pool, the Optimum Fitness simply becomes:

$$Op.F\{0{:}1\} = \ \left\|(\acute{C}(x, y, z) - \ L(DS(S(x, y, z), N(x, y, z) \right\|^2 \times log\frac{(err)^2}{(err + Err)} \tag{6}$$

Table 1
Quantized comparison of typical differences

Op. F(x)	Op. F(x, y)	Op. F(x, y, z)
0.00028723	0.00238712	0.03489120
0.234356	0.93	0.87
0.659023	+0.73	+0.78

As stated earlier, our model is based on 3-D space for x, y, z dimensions that algorithm uses to optimize the fitness of the blend to the given data. It must be noted that such approach is taken so the model gets very generalizable by design for any given data with any type of features. Thus, we manipulate matrix (real-valued) space to engineer the blend, thus, using Frobenius norm form:

$$\|M\|(x, y, z) = \sqrt{\sum_{x=1}^{X}\sum_{y=1}^{Y}\sum_{z=1}^{Z} M_{x,y,z}^2} \tag{7}$$

The matrix manipulation for each dimension to evaluate the blend can be built as:

$$\psi(n) = \underset{(x,y,z\,\in Z)}{Extract}(\,\|M\|(x, y, z)\,) + \log\sum\nolimits_{k=1}^{Z} \gamma^{z+1} \ . Op.F\{0{:}1\} \tag{8}$$

The illustration in Fig. 3 supports the mechanics of the theorem stated above. As we observe the visualization of three functions, err|Err, Op.F and Cost (C) being rotated based on distance function, as shown . Thus, Blender Filter Switch (logical) connect the value to the Suitability and Evaluation Score 3-D logical construct. The value of ψ(n) in each dimension swing between 0 and 1, based on Op.F response from Tuning Synthesizer block.

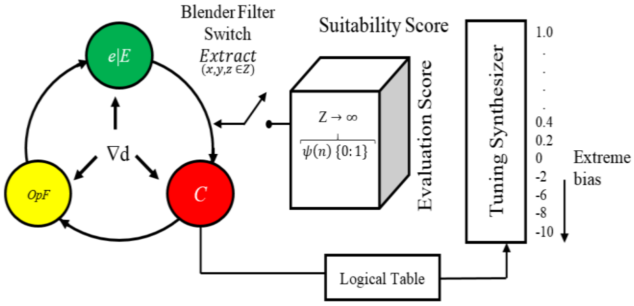

Figure 3: Illustration of Internals of the eABT (Theorem 1 - Algorithm Pre-processing)

Theorem 2: Risk Estimation, Local Errors, and Metrics Evaluation – $Algo.Risk_{emp}[f]$
AGE(Err) is in bound of all LE err(n), for which each occurrence of the error at any point in x and y space, exists inside all theoretical values of Err, such that $err(A) \in Err(A+1)$, *where* $0.8 < e < 0.2$. *Let there be a maximum risk function (Φ) with mean square error as MSE on the set of features as* $F = \{f_1, f_2, f_3, \ldots\ldots, f_n\}$, *and unknown means as* $\{m_1, m_2, m_3, \ldots\ldots, m_n\}$.

Construction – We implement the maximum and minimum error bounds logical limit to ensure the optimum fitness (i.e. avoiding overfitting and underfitting) in-terms of the errors to be controlled by upper and lower bounds. We first force error to be at low threshold and then to be high threshold. Once the algorithm has learned the max(e:0.8) and min(e:0.2) bounds, it then learns to stay in between, and accuracy is maintained.

$$max(e{:}\,0.8) \;=\; \frac{1}{E}\sum_{i=1}^{Ne}\{(RMSE_i) \;-\; (100 \;+\; 0.2)/E\} \tag{9}$$

$$min(e{:}\,0.2) \;=\; \frac{1}{E}\sum_{i=1}^{Ne}\{(RMSE_i) \;-\; (100 \;+\; 0.8)/E\} \tag{10}$$

Let us define our optimum error (dynamically governed by algorithm tuning and blending process), as

$$E_{opt}\,(max, min\,|e_{(x,y,z)} \;=\; \left(\frac{\int \partial(x,y,z)}{(1-E)}\right) \times \sum_{j\,=\,1}^{Nt}\left(\max(e)_j \;-\; \min(e)_j\right) \,*\, g.f \tag{11}$$

$g.f$ is the error gain factor and is produced by eABT algorithm. The errors produced by each algorithm, tend to increase when they are blended with each other, and errors of local and global functions must stay in the limit defined. For general Machine Learning modeling, there are two categories of errors, i) Estimation and ii) Approximation. Collectively, we can call it generalization error, in which our goal becomes a search for a special function $f'(x,y)$, that tends to minimize the risk of learning in the target space (i.e. X, Y, Z), shown as:

$$Risk[f']_{x,y,z} \;=\; \int_{X\,\times\,Y\,\times\,Z} L\,(y,\;\; f'(x,y))\,P(x,y,z)dxdydz \tag{12}$$

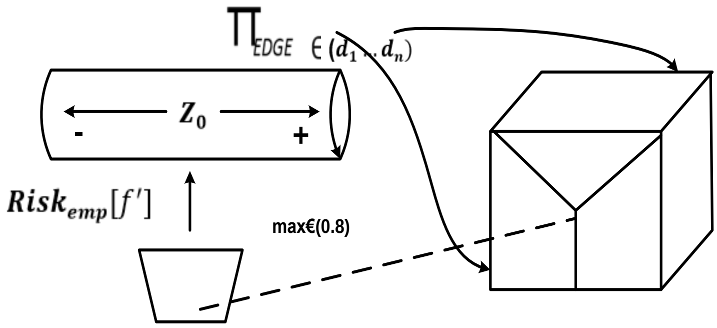

Figure 4: Illustration of minimum and maximum error bounds for risk estimation global function

Fig. 4 shows the concept of correlation of error bounds and risk function. As we propose the novel idea of limiting the error between 20% to 80 % for optimum realistic fitness for the real-world prediction, it shows that the Risk Estimation (emp) stays in bounds of logical cube shown on the right. The center point shows the ideal co-variance of function ʐ. Thus, $P(x, y, z)$ will be unknown at this stage. We will have to approximate based on well-known mathematical and statistical learning theory, known as 'empirical risk minimization principle':

$$Risk_{emp}[f'] = \frac{1}{m} \sum_{i=1}^{m} L\,(y^i\ , f'\,(x^i)) \tag{13}$$

Here, we need to satisfy two conditions, as
i) $\lim_{m \to \infty} Risk_{emp}[f'] = Risk[f']_{x,y,z}$ and
ii) $\lim_{m \to \infty} \min_{f \in H} Risk_{emp}[f'] = \min_{f \in H} Risk[f]_{x,y,z}$.
These two conditions will be valid when H is relatively small. The second condition requires minimal should converge and thus we can construct the following bound, that is being held valid with probability of 1 – δ is $R[f] \leq Risk_{emp}[f'] + ʐ$, where ʐ is given by:

$$ʐ = \sqrt{\left(\frac{h\ ln\left(\frac{2m}{h} + 1\right) - ln\left(\frac{\delta}{4}\right)}{m}\right)} \tag{14}$$

The sub-estimator function is $\widehat{m_k} = c(F_k, \vartheta)$, where ϑ is positive regularization parameters and it is observed that c(f, 0) = F such that, $\vartheta = 0 \mid \widehat{m_\iota} = F_k$. We deduce that $\vartheta = \infty$, corresponds to maximal shrinking, that is $\{\widehat{m_k} = 0, for\ k = 1,,,,,, n\}$. Here, we can apply Cross validation techniques (CV) and Stein's unbiased risk estimate (SURE), where popular estimators are (ridge), (lasso) and (pretest). For Loss and risk estimation, we utilize the foundation of squared error loss function, also known as compound loss,

$$Loss_n(F, c(k, \vartheta), D) = \frac{1}{n} \sum_{k=1}^{n} (c(F_k, \vartheta\,) - m_k)^2 \tag{15}$$

Where $D = \{d_1, d_2, \dots \dots \dots, d_n\}$ shows the distribution of Features F{x, y, z}. It should be observed that Loss is highly dependent on 'D' through value of 'm'. Finally, we can construct the regularization parameter for which the algorithm blend will fit the model to maximum relevance, such that: $\vartheta(D) = \max_{\vartheta \in [0,\infty)} B\,(c(.,\ \vartheta), \pi)$, thus:

$$Algo.Risk_{emp}[f] = Risk[f']_{x,y,z} \times \prod_{EDGE(Loss_n)} \zeta_n \tag{16}$$

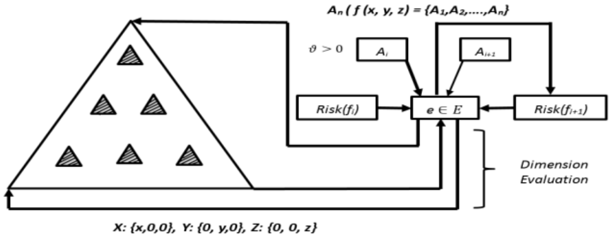

Figure 5: Illustration of risk estimation function in 3-d space for in-bound LE and GE

3. Experimental results

This section presents various experimental results with necessary discussion and information. Each figure is accompanied with detailed information and comments to elaborate on the experimental analysis of the proposed model.

3.1 Datasets and tools

The data from the many domains listed below have all been used. Some datasets come in raw, CSV, and SQL light formats and include field descriptions and parameters. To put all of our input data into the SQL Server data warehouse, we converted it. Some datasets have been discovered to be perfect for making predictions about the stock market, epidemics, and criminal control.

- Iris species, Credit Card Fraud Detection, http://www.Kaggle.com Flight Delays and Cancellations in 2022, Human Resource Analytics, daily stock market predictions news 1.88 million US wildfires, a dataset of SMS spam, a classification of Twitter users' genders, Retail Data Analytics, Wisconsin Brest Cancer Data Set College Scoreboard, US Department of Education US Mass Shootings, Adult Census Income, Fatal Police Shootings, Exercise Pattern Prediction, Death in the United States, Data from Netflix Prize Diabetes Database for Pima Indians, WUZZUF Job Listings, Student Survey, FiveThirtyEight, S&P 600 Stock Data Zika virus outbreak, alcohol consumption among students statistics on education, Center for Storm Prediction;
- http://snap.standford.edu – Facebook, Twitter, Wiki and bitcoin data set;
- Social networking APIs;
- https://www.reddit.com/r/bigquery/wiki/das;
- https://docs.microsoft.com/en-us/azure/sql-database/sql-database-public-data-sets;
- https://docs.google.com/forms/d/1l57Un32YH6SkltntirUeLVpgfn33BfJuFLcYupg43oE/viewform?edit_requested=true - online questionnaire from students across 12 campuses in the world;
- http://archive.ics.uci.edu/ml/index.php - Iris, Car Evaluation, Heart disease data set, Bank Marketing Data;
- https://aws.amazon.com/datasets/ - Enron Email Data, Japan Census data, 1000 Genomics Project;
- https://cloud.google.com/bigquery/public-data/ - We are experimenting it using BigQuery in our Sandbox environment and will publish results in the future.

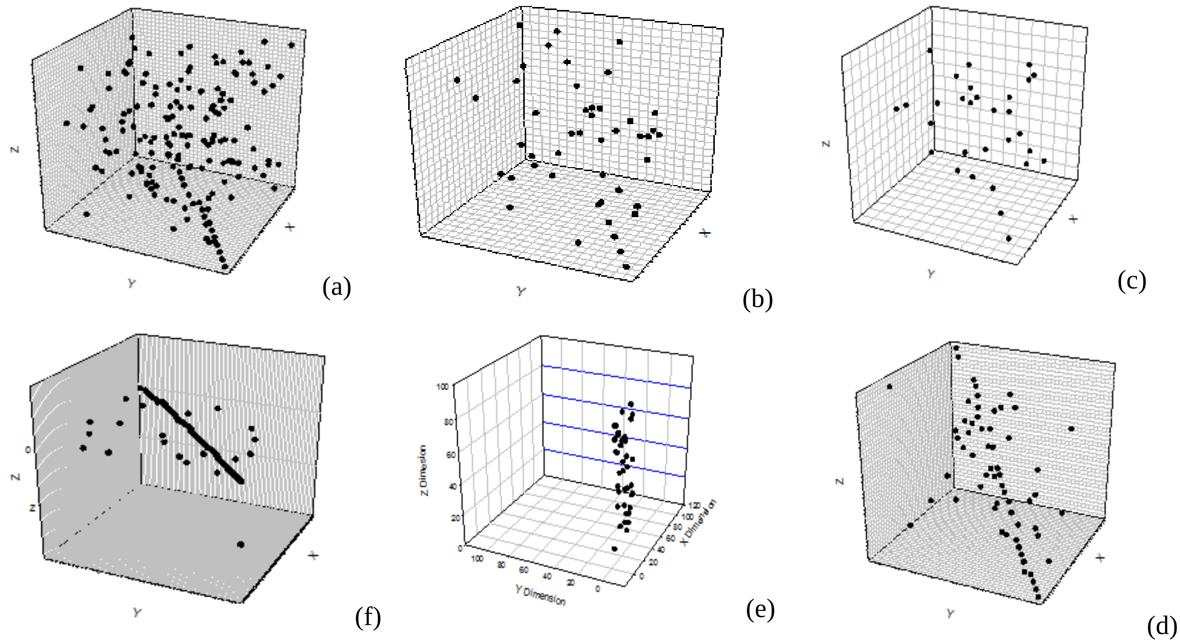

Figure 6: The significant dispersion of all fitness functions is demonstrated in (a). (b) indicates that the identification process has started, and (c) indicates that the variables that contribute to over fitting are being eliminated. (d) Demonstrates that the data points are streaming as anticipated, and (e) depicts an enhanced version of (d). (f) In the end, it is seen that the x, y dimensions have shrunk, and the data has been filtered to fall inside the ideal fitness range using the improved and optimized eMLEE algorithm/model.

Our data warehouse uses Microsoft SQL Server (Business Intelligence, SQL Server Analysis Services, and Data Mining). Work is done initially using Microsoft Azure ML tools. The suggested model's fundamental construction and supporting algorithms are carried out in the programming languages C#, Python, and R. Useful Python libraries like Pandas, Numpy, SciPy, Matplotlib, scikit-learn, Statsmodels, ScientificPython, Fuel, SKdata, Fuel, MILK, etc. have been used. Also used were gbm, KlaR, tree, RWeka, ipred, CORELearn, MICE Package, rpart, PARTY, CARET, and randomForest packages for the R programming language. The simulation produces results using the GraphPad Prism.

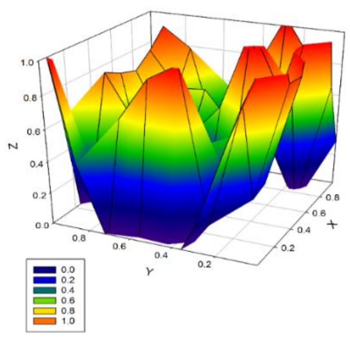

Figure 7 (a): shows the spread of Gain Function for about 200-10 experimental run. This result of the output shows that algorithm blend function was randomly distributed in all axis. This further shows that GG was extremely low, and fitness of the model was very poor

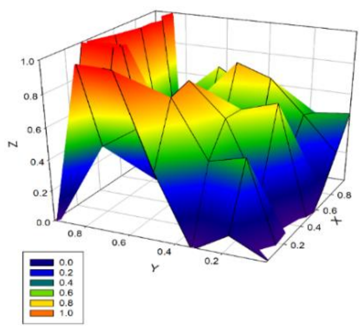

Figure 7 (b): shows the evolution of tuning process. As it is observed that x, and y dimension are fading out and z is out-running the fitness constraints. However, the yellow color in z-axis shows the error above 80 % that is still not acceptable for model application

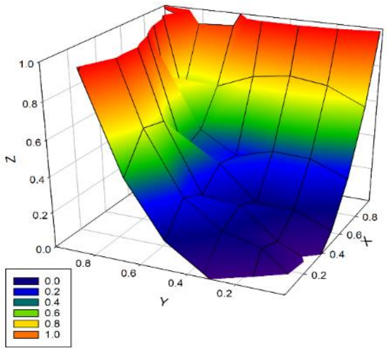

Figure 7 (c): shows the progress of the blending and tuning function with error being reduced

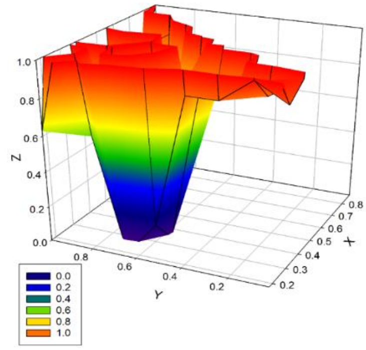

Figure 7 (d): finally shows the error in bounds of (20-80 %) rule and z-dimension has finally been optimized, as defined in the model and algorithm. The value of 1.0 as shown are optimistic values, the realistic values are observed be-tween 0.6 and 0.85

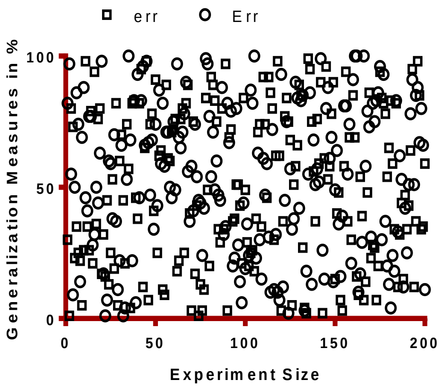

Figure 8 (a): It shows the random distribution generation of local and global complex error function

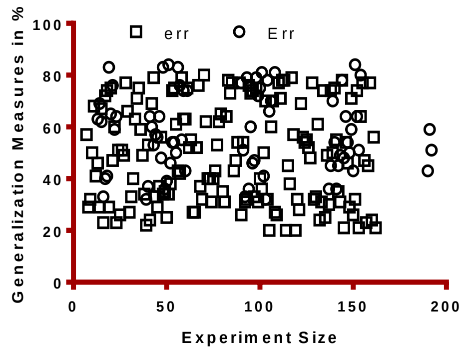

Figure 8 (b): This shows the improved separation of both function at stage 2 of model classifier definition as discussed in the algorithms earlier in section 3

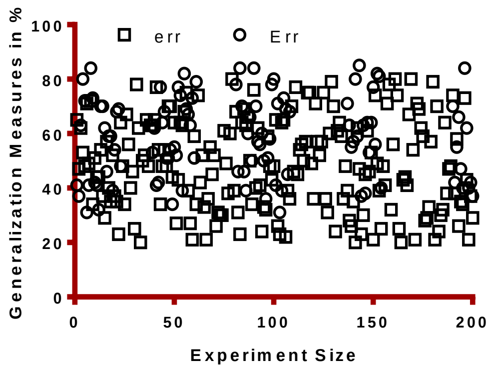

Figure 8 (c): This shows that model is learning (self-teaching) the separation of LE and GE over time

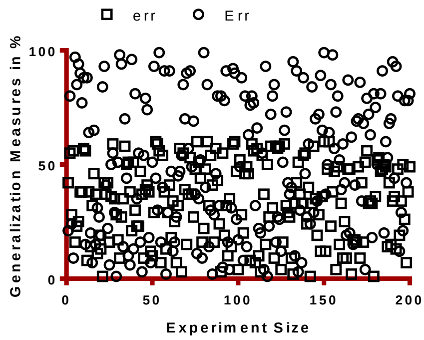

Figure 8 (d): This finally shows the optimum separation of both functions. It must be noted that both function over-laps since one is local and one is global

As described in section 2 earlier in this article, the y-axis displays the percentage from 0 to 100 and the x-axis displays the 200 experimental runs for various random data sets and algorithm blends. This helps the model be more universal for a very large set of data and features. The GE function is denoted by a circle, and the LE function is denoted by a square. It should be emphasized that these results come after accuracy measurements for the model after thousands of iterations. The y-axis shows the % from 0 to 100 and x-axis shows the 200-experimental run for various random data set and algorithm blend to improve the generalization of the model for a very vast set of data and features, as discussed in section 2 earlier in this article. Circle indicate the GE function and square indicates the LE function. It must be noted that these findings follow accuracy measurement for the model as optimum fit for several thousand iterations.

4. Conclusion

The suggested model's most recent developments, including its framework, algorithms, and mathematical components, are reported in this study. The following foundational elements have been used to produce an improved machine learning methodology: the development of a combination of algorithms based on parallel tuning of the model and the classifiers to enhance metrics, the implementation of a selected set of supervised learning (SL) algorithms on the experimental dataset, and the recording of the measured metrics in a logical table construct. Creating a logical 3-D cube that controls the algorithms to ensure the best fitness for the blend being engineered, c) developing the final model so it can learn from its errors (wrong predictions) and teach itself to choose the correct algorithm and eliminate the incorrect one during the training process, d) engineering the blend being engineered to have the best fitness possible, e) and finally, validating the suggested model, which includes sub-algorithms for a variety of real-world data sets for predictive and prescriptive analytics. The blend was seen during several hundred studies, according to this article. To fine-tune the model and generate simulated outcomes for the study, we divided the results based on a 10, 20, and 30-experimental method. In 3D space, the LG and GG functions were developed and optimized. The parallel tuning and blending approach described in this paper has better outcomes and has the ability to generalize to a different collection of data and challenges, it was discovered. We will test new algorithms, particularly in the areas of unsupervised learning. Predicting Educational Relevance For an Efficient Classification of Talent (PERFECT) algorithm Engine (PAE) is a model that we are creating and refining. Three algorithms – Noise Removal and Structured Data Detection (NR-SDD), Good Fit Student (GFS), and Good Fit Job Candidate (GFC) – are included in PAE, which is based on eMLEE. We will try to apply the most recent iteration of the eABT (i.e., eMLEE) model to research, analyze, and validate additional improvements.

5. References

[1] Razaque, Abdul, Mohamed Ben Haj Frej, Gulnara Bektemyssova, Muder Almi'ani, Fathi Amsaad, Aziz Alotaibi, Noor Z. Jhanjhi, Mohsin Ali, Saule Amanzholova, and Majid Alshammari. "Quality of Service Generalization using Parallel Turing Integration Paradigm to Support Machine Learning." Electronics 12, no. 5 (2023): 1129.

[2] Jain, Nishant, and Prasanta K. Jana. "LRF: A logically randomized forest algorithm for classification and regression problems." Expert Systems with Applications 213 (2023): 119225.

[3] Hassaballah, Mahmoud, Yaser M. Wazery, Ibrahim E. Ibrahim, and Aly Farag. "Ecg heartbeat classification using machine learning and metaheuristic optimization for smart healthcare systems." Bioengineering 10, no. 4 (2023): 429.

[4] Shahzadi, Shumaila, Fahad Ahmad, Asma Basharat, Madallah Alruwaili, Saad Alanazi, Mamoona Humayun, Muhammad Rizwan, and Shahid Naseem. "Machine learning empowered security management and quality of service provision in SDN-NFV environment." Comput. Mater. Contin 66, no. 3 (2021): 2723-2749.

[5] Ansari, Homa, Marc Ruβwurm, Mohsin Ali, Sina Montazeri, Alessandro Parizzi, and Xiao Xiang Zhu. "InSAR displacement time series mining: A machine learning approach." In 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, pp. 3301-3304. IEEE, 2021.
[6] Razaque, Abdul, Fathi Amsaad, Dipal Halder, Mohamed Baza, Abobakr Aboshgifa, and Sajal Bhatia. "Analysis of Sentimental Behaviour over Social Data Using Machine Learning Algorithms." In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pp. 396-412. Springer, Cham, 2021.
[7] Al-Jawad, Ahmed, Ioan-Sorin Comşa, Purav Shah, Orhan Gemikonakli, and Ramona Trestian. "An innovative reinforcement learning-based framework for quality of service provisioning over multimedia-based sdn environments." IEEE Transactions on Broadcasting 67, no. 4 (2021): 851-867.
[8] Laiche, Fatima, Asma Ben Letaifa, Imene Elloumi, and Taoufik Aguili. "When machine learning algorithms meet user engagement parameters to predict video QoE." Wireless Personal Communications 116 (2021): 2723-2741.
[9] Guo, N., Li, L., Mukherjee, B. and Shen, G., 2022. Protection against failure of machine-learning-based QoT prediction. Journal of Optical Communications and Networking, 14(7), pp.572-585.
[10] Kumar, Naveen, and Anwar Ahmad. "Machine learning-based QoS and traffic-aware prediction-assisted dynamic network slicing." International Journal of Communication Networks and Distributed Systems 28, no. 1 (2022): 27-42.
[11] Guo, Ningning, Longfei Li, Biswanath Mukherjee, and Gangxiang Shen. "Protection against failure of machine-learning-based QoT prediction." Journal of Optical Communications and Networking 14, no. 7 (2022): 572-585.
[12] Guo, Ningning, Longfei Li, Biswanath Mukherjee, and Gangxiang Shen. "Protection against failure of machine-learning-based QoT prediction." Journal of Optical Communications and Networking 14, no. 7 (2022): 572-585.
[13] Shah, Syed Danial Ali, Mark A. Gregory, Shuo Li, and Ramon Dos Reis Fontes. "SDN enhanced multi-access edge computing (MEC) for E2E mobility and QoS management." IEEE Access 8 (2020): 77459-77469.
[14] Hu, Hexuan, Qingqiu Li, Yunfeng Zhao, and Ye Zhang. "Parallel deep learning algorithms with hybrid attention mechanism for image segmentation of lung tumors." IEEE Transactions on Industrial Informatics 17, no. 4 (2020): 2880-2889.
[15] Maruyama, Yoshihiro. "Symbolic and statistical theories of cognition: towards integrated artificial intelligence." In Software Engineering and Formal Methods. SEFM 2020 Collocated Workshops: ASYDE, CIFMA, and CoSim-CPS, Amsterdam, The Netherlands, September 14–15, 2020, Revised Selected Papers 18, pp. 129-146. Springer International Publishing, 2021.
[16] Li, Li, Yilun Lin, Nanning Zheng, and Fei-Yue Wang. "Parallel learning: A perspective and a framework." IEEE/CAA Journal of Automatica Sinica 4, no. 3 (2017): 389-395.
[17] Vaccaro, Lorenzo, Giuseppe Sansonetti, and Alessandro Micarelli. "An empirical review of automated machine learning." Computers 10, no. 1 (2021): 11.
[18] Golmohammadi, Davood, Mahour Mellat Parast, and Nada Sanders. "The impact of service failures on firm profitability: integrating machine learning and statistical modeling." IEEE Transactions on Engineering Management 69, no. 6 (2020): 3038-3052.
[19] Elshawi, Radwa, Sherif Sakr, Domenico Talia, and Paolo Trunfio. "Big data systems meet machine learning challenges: towards big data science as a service." Big data research 14 (2018): 1-11.
[20] Alzubi, Jafar, Anand Nayyar, and Akshi Kumar. "Machine learning from theory to algorithms: an overview." In Journal of physics: conference series, vol. 1142, p. 012012. IOP Publishing, 2018.
[21] Guo, Ningning, Longfei Li, Biswanath Mukherjee, and Gangxiang Shen. "Protection against failure of machine-learning-based QoT prediction." Journal of Optical Communications and Networking 14, no. 7 (2022): 572-585.