Fatigue Life Prediction of Additively Manufactured Part Using Machine Learning Algorithms for Varying Deposition Attributes

Project report submitted in partial fulfilment of the

requirements for the degree of

BACHELOR OF TECHNOLOGY

in

Mechanical Engineering

by

Antara Parui

(Roll No. 210103019)

Hrishi Inani

(Roll No. 210103120)

Under the supervision of

Prof. Swarup Bag

to

DEPARTMENT OF MECHANICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

GUWAHATI – 781039, INDIA

[January-May 2024]

CERTIFICATE

This is to certify that the work contained in this project report entitled **"Fatigue Life Prediction of Additively Manufactured Part Using Machine Learning Algorithms for Varying Deposition Attributes"** submitted by **Antara Parui (210103019)** and **Hrishi Inani (210103120)** to the Indian Institute of Technology Guwahati towards the partial requirement of **Bachelor of Technology** in **Mechanical Engineering** is a bona fide work carried out by them under my supervision and that it has not been submitted elsewhere for the award of any degree.

(Prof. Swarup Bag)

Mechanical Engineering

IIT GUWAHATI

April 2024

DECLARATION

We declare that this written submission represents our ideas in our own words, and where others' ideas or words have been included, we have adequately cited and referenced the sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated, or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Antara Parui **Hrishi Inani**

Roll No. 210103019 **Roll No. 210103120**

Date: 27/04/2024

APPROVAL SHEET

This project report entitled "**Fatigue Life Prediction of Additively Manufactured Part Using Machine Learning Algorithms for Varying Deposition Attributes**" by **Antara Parui (210103019)** and **Hrishi Inani (210103120)** is approved for the degree of Bachelor of Technology.

Examiners

_____

_____

_____

Supervisor

Prof. Swarup Bag

Chairman

_____

Date: 27/04/2024

Place: Guwahati

ABSTRACT

Additive Manufacturing (AM) is an advanced manufacturing technique that creates three-dimensional objects by adding layer by layer material, based on a digital 3D model. One of the major issues faced by additive manufacturing is the uneven quality of printed goods, which is primarily determined by different processing parameters such as layer thickness and printing speed. It is crucial to assess the fatigue characteristics of metallic structural materials. Many factors, including AM processing parameters, microstructure, residual stress, surface roughness, porosities, post-treatments, etc., affect the fatigue properties of materials that are additively manufactured. Machine learning models are capable of assimilating information and making inferences from reliable training datasets. Thus, they can be used for predicting fatigue life.

This report presents a comprehensive evaluation of machine learning (ML) models for predicting the fatigue cycle of additively manufactured materials. Five ML algorithms, including Random Forest, Decision Tree, XGBoost, Gradient Boosting Regressor, and KNN, were employed to model fatigue behaviour, with their predictive performance assessed using the R^2 and visual inspection of predicted versus experimental values. Additionally, feature importance analysis was conducted to identify influential attributes in predicting fatigue life values. The results highlight variations in predictive accuracy among the ML models and provide insights into the relative importance of different attributes. This study contributes to advancing fatigue prediction methodologies for additively manufactured materials and informs material design and processing optimization efforts to enhance component durability and reliability.

***Keywords**: Additive manufacturing, machine learning, fatigue strength, fatigue life prediction.*

LIST OF ACRONYMS USED

AM	Additive Manufacturing
FL	Fatigue Life
ML	Machine Learning
GB	Gradient Boosting
RF	Random Forest
KNN	K Nearest Neighbours
CRS	Compressive Residual Stress
DT	Decision Tree
GBDT	Gradient-Boosted Decision Trees
SA	Stress Amplitude
YS	Yield Stress
US	Ultimate Stress
RD	Relative Density
SH	Surface Hardness
DHV	Depth of Hardness Variation
SMF	Surface Modification Factor
SR	Surface Roughness
DCRS	Depth of CRS

LIST OF TABLES & FIGURES USED

Figure 1	Datasets utilized for prediction of fatigue life
Figure 2	Dummy Encoded Dataset
Figure 3	Evaluation of different ML Models using R^2 values
Figure 4	Comparison of predicted and experimental fatigue life
Figure 5	Evaluation of Important Features applied for RF
Table 1	Categories of Machine Learning
Table 2	Hyperparameters for RF
Table 3	Hyperparameters for DT
Table 4	Hyperparameters for KNN
Table 5	Hyperparameters for XGBoost
Table 6	Hyperparameters for GB

CONTENT

1. INTRODUCTION **8-11**

1.1 Fatigue Life in Additive Manufacturing 8

1.2 Machine Learning 8-9

1.3 ML Models Used 9-11

1.3.1 RF Regressor Model 9-10

1.3.2 DT Regressor Model 10

1.3.3 GB Regressor Model 10-11

1.3.4 XGBoost Regressor Model 11

1.3.5 KNN 11

2. METHODOLOGY USED FOR FL PREDICTION **12-16**

2.1 Data Description 12

2.2 Data Preprocessing 12-16

2.2.1 Data Encoding 13

2.2.2 Feature Scaling 14

2.2.3 Model Development 14

2.2.4 Hyperparameter Optimization 15

2.2.5 Model Validation 16

2.2.6 Feature Importance Analysis 16

3. RESULT AND DISCUSSIONS **17- 19**

4. CONCLUSIONS AND FUTURE SCOPE **20**

REFERENCES **21**

Chapter 1

Introduction

1.1 Fatigue Life in Additive Manufacturing

Fatigue life in Additive Manufacturing (AM) refers to the number of cycles a component or material can endure under alternating loads before experiencing failure [1]. This failure can manifest as crack formation, sudden fractures, or loss of load-bearing capability in the case of structures. The fatigue strength of additively manufactured metals depends on several factors, including material properties, geometric features, and build orientation. Optimizing these factors is crucial to ensure the durability and reliability of AM-produced components. Key attributes influencing the fatigue strength of additively manufactured metals include yield stress, ultimate stress, elongation, relative density, surface hardness, depth of hardness variation, surface compressive residual stress (CRS), maximum CRS, depth of CRS, surface modification factor, surface roughness, and stress amplitude. Predictive models can be developed using these attributes as input parameters to estimate the fatigue life cycle of additively manufactured parts or metals, aiding in process optimization and component design [1].

1.2 Machine Learning

Machine learning (ML), a subset of artificial intelligence (AI), is defined as computer programming that uses sample data or prior knowledge to optimize a performance criterion.

Table 1: Categories of Machine Learning [2].

Supervised Learning	**Unsupervised Learning**	**Semi-supervised Learning**	**Reinforcement Learning**
Provide input, output and feedback to build model. Example Algorithms Linear regressions i. Sales forecasting. ii. Risk assessment. Support vector machines i. Image classification. ii. Performance comparison. Decision trees i. Predictive analytics. ii. Pricing.	Use deep learning to arrive at conclusions and patterns through unlabelled training data. Example Algorithms K-means clustering i. Performance monitoring. ii. Searcher intent. Artificial neural networks i. Generate new, synthetic data. ii. Data mining and pattern recognition.	Builds a model through a mix of labelled and unlabelled data, a set of categories, suggestions and exampled labels. Example Algorithms Generative adversarial networks i. Audio and video manipulation. ii. Data creation. Self-trained Naïve Bayes classifier i. Natural language processing.	Self-interpreting but based on a system of reward and punishments learned through trial and error, seeking maximum reward. Example Algorithms Q-learning i. Policy creation. ii. Consumption reduction. Model-based value estimation i. Linear tasks. ii. Estimating parameters.

1.3 ML Models Used

In this section, the five typical ML models used for the fatigue life prediction of AM metal parts are presented in detail.

1.3.1 Random Forest Regressor Model

RF is an integrated learning algorithm and is based on DT. RF is a typical supervised machine learning algorithm that integrates integrated learning strategies with nonlinear statistical methods. Randomly extracted sample features are used to build multiple unrelated DTs. Every tree acquires the prediction results in parallel with the extracted sample characteristics. To get the prediction results for the entire forest, the outputs of each tree are averaged and combined. Multiple sub-datasets are created for RF training, using the bootstrap

method. Each sub-dataset is then modelled using a DT without pruning; this type of training is commonly known as bagging.

The two steps of the RF random process are as follows. RF first generates each subset $s_m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ by resampling at random a specific portion of the entire datase $s_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where, $x_1 = [x_{n1}, x_{n2}, \ldots, x_{ni}]$ and $y_1 = [y_{n1}, y_{n2}, \ldots, y_{ni}]$ are the input and output vectors, respectively. Second, RF randomly chooses P (P < M) input features from a pool of M input features to be used in leaf node splitting during each DT's training. The diversity of DT in the forest is guaranteed by these two actions. After training, the kth decision tree's output function can be defined as $f(x, s_m^k)$ and the average prediction result of the RF comprising k DT is: $\hat{y}_{rf} = \frac{1}{k}\sum_{i=1}^{k} f(x, s_m^k)$ where $\hat{y}_{rf}$ is the target variable's RF prediction [3].

1.3.2 Decision Tree Regressor Model

Decision Tree Regressor works by partitioning the feature space into regions and fitting a simple model (typically a constant value) within each region. The decision tree starts with the entire dataset as the root node. It then looks for the feature and the split point that best separates the data into different groups. Once the splitting stops, the final regions of the feature space are called leaf nodes. Each leaf node contains the predicted value for the target variable. To make a prediction for a new data point, it traverses down the tree based on the feature values of the data point until it reaches a leaf node. The predicted value for the target variable is then the value associated with that leaf node [4].

1.3.3 Gradient Boosting Regressor Model

GBR is made up of three elements: a loss function (which needs to be optimized), a weak learner (used for making predictions) and an additive model (which adds weak learners to minimize the loss function). The main idea behind this technique is to build new base-learners that have the highest possible correlation with the negative gradient of the loss function, associated with the whole ensemble. The following is the detailed mathematical formulation of GBR. For an additive model of the form [4]:

$$F(x) = \sum_{m=1}^{M} \gamma_m h_m(x) \qquad (1)$$

where $h_m(x)$ are the basis functions, usually called weak learners in the context of boosting. GBR uses decision trees of fixed size as weak learners. It constructs the additive model in a stage-by-stage forward manner similar to other boosting methods.

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \quad (2)$$

1.3.4 XGBoost

Chen et al. proposed the XGBoost (eXtreme Gradient boosting) algorithm, a tree-model-based boosting algorithm. XGBoost functions by adding a new tree in every iteration to minimize the residuals from the previous prediction. Thus, this gradient-based optimization method produces better accuracy than using individual decision trees or random forests. Additionally, regularization terms are incorporated into XGBoost's loss function to help manage model complexity and avoid overfitting. It improves training efficiency through effective split-finding algorithms and feature-level parallel processing. It is actually an enhanced version of GBDT algorithm, which consists of numerous decision trees. But there are some ways in which the XGBoost algorithm is different from the GBDT algorithm. The primary distinction is the use of second-order derivatives in conjunction with the expansion of the loss function into the Taylor second-order series, which results in a more efficient model solution than the GBDT algorithm. Finding the ideal parameters is a crucial step in the model training process [4].

1.3.5 K-Nearest Neighbour Model

KNN is a non-parametric algorithm which performs well with non-linear data and can handle complex decision boundaries. It is necessary to specify the hyperparameter K, which is the number of neighbours. In general, smoother predictions are produced by bigger values of K, whereas more complicated and potentially overfitting predictions are produced by smaller values of K.

The distance between the feature vectors of each data point in the training set and the feature vector of the new data point x is determined. Any distance metric may be utilized, including Manhattan distance and Euclidean distance. Based on the calculated distances, the K training data points that are closest to the new data point x are chosen. The average or median of the target values of the K nearest neighbours is used as the predicted target value for the new data point x [5].

Chapter 2

Methodology Used for Fatigue Life Prediction

2.1 Data Description

To build the ML models, relevant experimental Fatigue life data of LPBF AlSi10Mg samples was collected from the literature. The data shows the variation of fatigue life in cycles with different attributes such Yield Stress, Ultimate Stress, Relative Density, Surface Hardness and Roughness, Depth of Hardness Variation, Stress Amplitude, etc.

Set Of Sample	Sample No.	Yield stress (MPa)	Ultimate stress (MPa)	Elongation(%)	Relative density (%)	Surface hardness(Hv)	Depth of hardness variation (µm)	Surface CRS (MPa)	Maximum CRS (MPa)	Depth of CRS (µm)	Surface modification factor	Surface roughness, Ra(µm)	Stress amplitude(MPa)	Fatigue life(Cycle)
AB	1	273	393	2.5	99.53	101.4	0	-11.0	-57.6	100	0.0	4.34	60	376456
AB	2	273	393	2.5	99.53	101.4	0	-11.0	-57.6	100	0.0	4.34	30	1244634
AB	3	273	393	2.5	99.53	101.4	0	-11.0	-57.6	100	0.0	4.34	10	2203568
AB	4	273	393	2.5	99.53	101.4	0	-11.0	-57.6	100	0.0	4.34	20	1646240
AB	5	273	393	2.5	99.53	101.4	0	-11.0	-57.6	100	0.0	4.34	40	568620

Set Of Sample	Sample No.	Yield stress (MPa)	Ultimate stress (MPa)	Elongation(%)	Relative density (%)	Surface hardness(Hv)	Depth of hardness variation (µm)	Surface CRS (MPa)	Maximum CRS (MPa)	Depth of CRS (µm)	Surface modification factor	Surface roughness, Ra(µm)	Stress amplitude(MPa)	Fatigue life(Cycle)
AB + HT + SP2	85	201	265	13.0	99.57	72.0	190	-31.7	-103.8	640	1.0	4.87	110	3000000
AB + HT + SP2	86	201	265	13.0	99.57	72.0	190	-31.7	-103.8	640	1.0	4.87	110	3000000
AB + HT + SP2	87	201	265	13.0	99.57	72.0	190	-31.7	-103.8	640	1.0	4.87	140	906500
AB + HT + SP2	88	201	265	13.0	99.57	72.0	190	-31.7	-103.8	640	1.0	4.87	130	1430700
AB + HT + SP2	89	201	265	13.0	99.57	72.0	190	-31.7	-103.8	640	1.0	4.87	130	1750120

Figure 1: Dataset utilized for prediction of fatigue cycles.

2.2 Data Preprocessing

Pre-processing data is intended to transform the raw data to a format that is easier and more effective to use for future processing steps.

2.2.1 Data Encoding

It describes the procedure that transforms textual or category data into numerical format so that algorithms can use it as input. The majority of machine learning algorithms operate with numerical data rather than text or categorical variables, which is why encoding is necessary. It can be done in many ways including One Hot Encoding and Dummy Encoding [6]:

1. One Hot Encoding

One Hot Encoding technique is used when the features are nominal (do not have any order). In one hot encoding, for each level of a categorical feature, we create a new variable. Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category.

2. Dummy Encoding

Dummy encoding method transforms the categorical variable into a set of binary variables (also known as dummy variables). The dummy encoding is a small improvement over one-hot-encoding. In the case of one-hot encoding, for N categories in a variable, it uses N binary variables whereas Dummy encoding uses N-1 features to represent N labels/categories.

For our dataset we have used Dummy Encoding.

Depth of hardness variation (µm)	Surface CRS (MPa)	Maximum CRS (MPa)	Depth of CRS (µm)	Surface modification factor	Surface roughness, Ra(µm)	Stress amplitude(MPa)	Set Of Sample_AB	Set Of Sample_AB + HT	Set Of Sample_AB + HT + SP1	Set Of Sample_AB + HT + SP2	Set Of Sample_AB + SP1	Set Of Sample_AB + SP2
0	-11.0	-57.6	100	0.0	4.34	60	1	0	0	0	0	0
0	-11.0	-57.6	100	0.0	4.34	30	1	0	0	0	0	0
0	-11.0	-57.6	100	0.0	4.34	10	1	0	0	0	0	0
0	-11.0	-57.6	100	0.0	4.34	20	1	0	0	0	0	0
0	-11.0	-57.6	100	0.0	4.34	40	1	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
190	-31.7	-103.8	640	1.0	4.87	110	0	0	0	1	0	0
190	-31.7	-103.8	640	1.0	4.87	110	0	0	0	1	0	0
190	-31.7	-103.8	640	1.0	4.87	140	0	0	0	1	0	0
190	-31.7	-103.8	640	1.0	4.87	130	0	0	0	1	0	0
190	-31.7	-103.8	640	1.0	4.87	130	0	0	0	1	0	0

Figure 2: Dummy Encoded Dataset.

2.2.2 Feature Scaling

Feature scaling is a data preprocessing technique which is used to transform the values of features or variables in a dataset to a similar scale. The goal is to prevent features with

higher values from predominating and to make sure that every feature contributes equally to the model.

Our dataset includes features that have varying ranges, orders of magnitude, and units of measurement, thus feature scaling becomes important. There are two main methods for scaling that are employed:

1. Standard Scaling

It is calculated by subtracting the mean and dividing by the standard deviation. This technique rescales a feature value such that it has a distribution with a mean value of 0 and variance equal to 1 [7].

$$\text{Formula: } x' = \frac{x - \bar{x}}{\sigma} \tag{3}$$

where σ is the standard deviation of the feature vector, and $\bar{x}$ is the average of the feature vector.

2. Min-Max Normalization (or Min-Max scaling)

This technique rescales a feature with a distribution value between 0 and 1. The formula for the same is as follows [7]:

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{4}$$

Here, x is the feature, and min() and max() are the minima and maximum values of the feature, respectively.

We have used StandardScaler from sklearn.preprocessing package.

2.2.3 Model Development

A series of models is built for the Fatigue Life predictions using 5 different algorithms namely RF, DT, KNN, XGBoost, GB. The train and test data are split into 80% and 20%, respectively, where the train data is used for building model and hyper parameter optimization and the test data are used to check performance of the model on the unseen data. The validation of the model is done by comparing the performance of the model on the train and test datasets to verify whether the model is overfitting or underfitting in the present work. The scikit-learn library is used for importing algorithms for our models.

2.2.4 Hyper Parameter Optimization

Hyper parameter tuning is required to improve the performance of the model. Especially in decision tree-based models if the hyperparameters are kept at default values, the model tries to build deeper trees which eventually gives better results on the train data but performs poorly on test data. Hence, to restrict the model from overfitting and to generalize the model, hyper-parameters are tuned.

Table 2: Hyperparameters for RF.

Hyper Parameters	Values
n_estimators	10
max_depth	40
criterion	mse (default)
min_samples_split	2 (default)
max_features	auto (default)

Table 3: Hyperparameters for DT.

Hyper Parameters	Values
min_samples_split	5
max_depth	none (default)
criterion	mse (default)
min_samples_split	2 (default)
max_features	none (default)

Table 4: Hyperparameters for KNN.

Hyper Parameters	Values
n_neighbors	2
weights	uniform (default)
metric	minkowski (default)
leaf_size	30 (default)
algorithm	auto (default)

Table 5: Hyperparameters for XGBoost.

Hyper Parameters	Values
min_samples_split	5
max_depth	none (default)
criterion	mse (default)
min_samples_split	2 (default)
max_features	none (default)

Table 6: Hyperparameters for GB.

Hyper Parameters	Values
n_estimators	80
max_depth	3 (default)
learning_rate	0.1 (default)
lambda	1 (default)

2.2.5 Model Validation

After developing the models, we compared the predicted results with the actual data. To analyse this, the experimental and predicted curves are plotted together to visualize the closeness of the data points. The performance of the developed models can be analysed using few metrics. As this was a regression problem, the metrics can be mean squared error, mean absolute error, and R^2 Scores. The following are the formulae for the same [8]:

$$mse = \frac{1}{N}[\sum(y_a - yp)]^2 \quad (5)$$

$$mae = \frac{1}{N}\Sigma|y_a - y_p| \quad (6)$$

$$R^2 = 1 - \frac{\sum(y_a - y_p)^2}{\sum(y_a - y_m)^2} \quad (7)$$

where, y_a is the actual value, y_p is the predicted value and y_m is mean of the actual values.

2.2.6 Feature Importance Analysis

Feature Importance analysis is carried out to determine the most influencing independent variables on the target column. It is done by calculating the score for all input features in a model to establish the importance of each feature in the decision-making process. The higher the score for a feature, the larger effect it has on the model to predict a certain variable.

Chapter 3

Results and Discussions

The fatigue cycle of additively manufactured material was predicted utilizing five distinct machine learning (ML) models: Random Forest, Decision Tree, XGBoost, Gradient Boosting Regressor, and KNN. Each model's performance was evaluated using R^2 to assess its predictive accuracy. Furthermore, the capabilities of these models were inspected through graphs depicting predicted versus experimental values for both training and testing datasets.

The R^2 values for each model are summarized in Figure 3, providing a quantitative measure of their predictive performance.

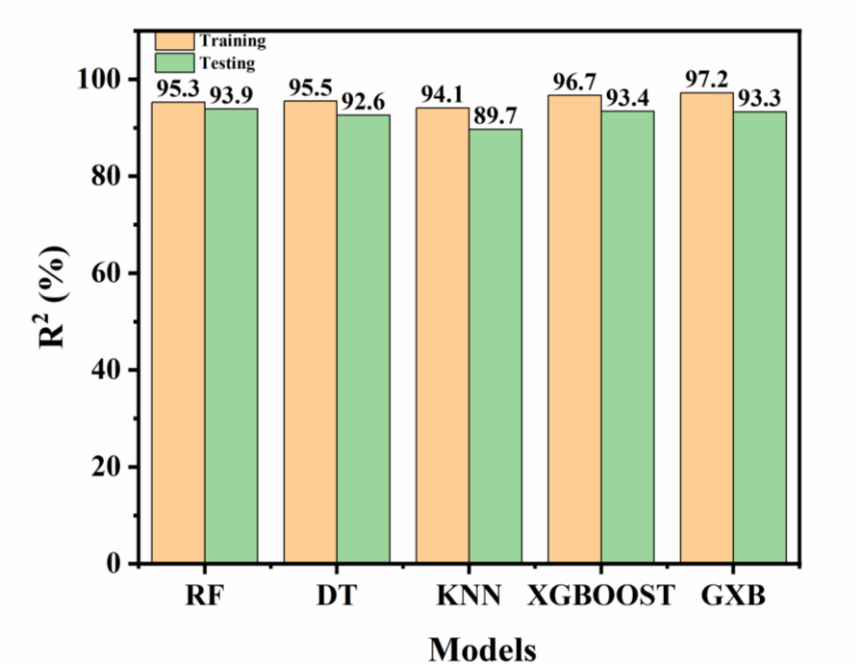

Figure 3: Evaluation of different ML Models using R^2 values.

The variability in R^2 values highlights differences in the models' abilities to explain the variance in fatigue cycle predictions. Models with higher R^2 values demonstrate stronger correlations between predicted and actual values, indicating superior predictive performance.

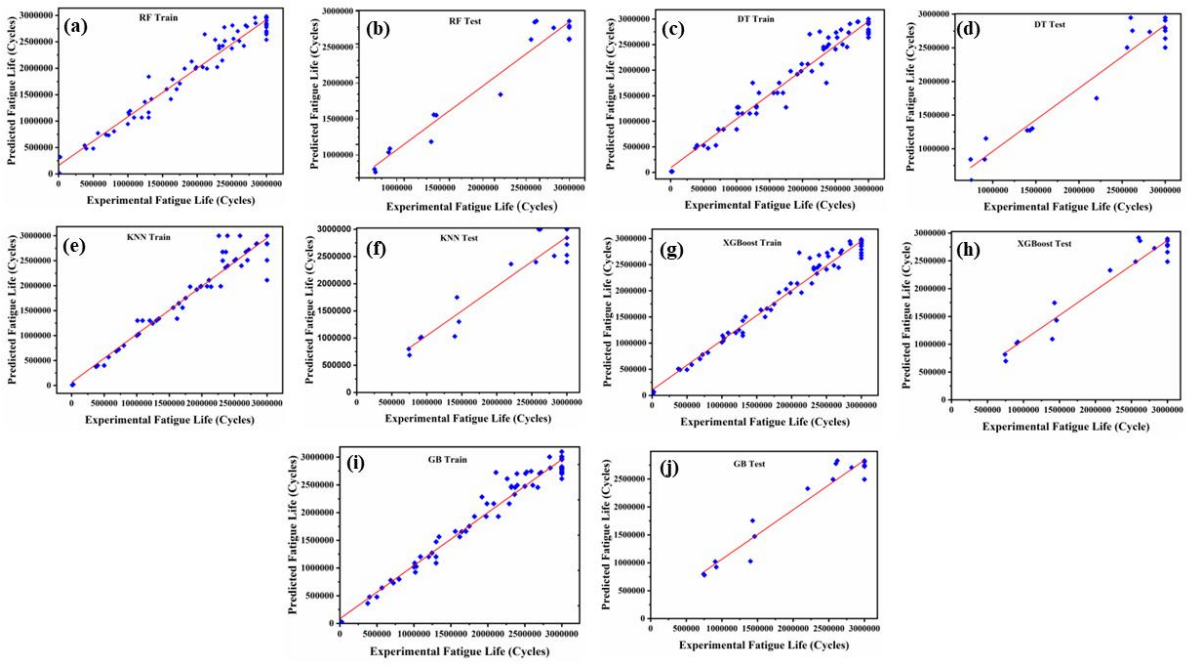

Figure 4: Comparison of predicted and experimental fatigue life on training and testing datasets.

Visual inspection of the predicted versus experimental value graphs for both training and testing datasets provides additional insights into the models' performance. From these graphs, one can observe that the model that is providing us the best results from the used five models for the prediction of unknown data is RF.

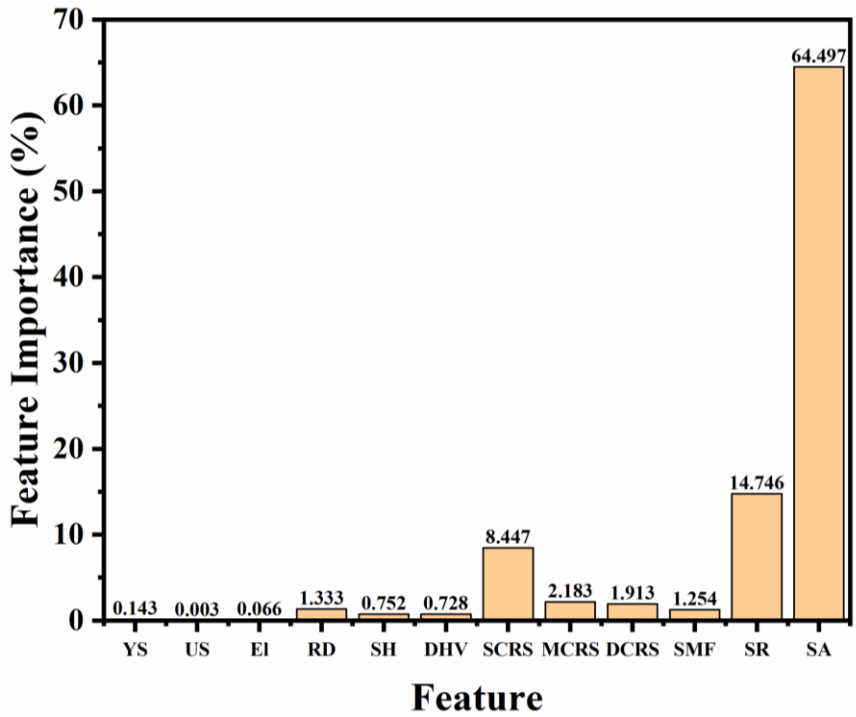

Figure 5: Evaluation of Important Features applied for RF.

Figure 5 shows the feature importance of different features used for the prediction of fatigue life by RF model. It is observed that Stress Amplitude has a significant impact on the fatigue

life of the material. Also, Surface Roughness and Surface CRS play an important role compared to the other features or attributes used for the prediction. Along with these features, which set of samples is used for the prediction also defines the variation in the fatigue life.

Chapter 4

Conclusions And Future Scopes

4.1 Conclusions

In this phase of our BTP, we utilized 5 ML models namely RF, DT, KNN, XGBoost, GB to predict fatigue life of additively manufactured parts. Through rigorous analysis, we optimized model performance through feature importance assessment and hyperparameter tuning in all the models. We also understood how different attributes, such as surface thickness, stress amplitude, surface hardness, relative density, surface CRS, etc. impact the fatigue behaviour of additively manufactured parts.

4.2 Future work

In the following phases we plan to apply some other complex deep learning algorithms/models to predict the fatigue strength of additively manufactured parts and refine them to improve the predictive accuracy.

References

[1] A history of fatigue. Engng Fract Mech 1996;54:263-300, Schutz W.

[2] Learning in Big Data: Introduction to Machine Learning, El Bouchefry, Khadija, de Souza, Rafael S.

[3] A random forest regression with Bayesian optimization-based method for fatigue strength prediction of ferrous alloys. Junyu Guo, Xueping Zan, Lin Wang , Lijun Lei a, Chuangjie Ou , Song Bai.

[4] A. Keprate and R. M. C. Ratnayake, "Using gradient boosting regressor to predict stress intensity factor of a crack propagating in small bore piping," 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, 2017.

[5] Machine Learning Based Predictions of Fatigue Crack Growth Rate of Additively Manufactured Ti6Al4V. Nithin Konda, Raviraj Verma, Rengaswamy Jayaganthan.

[6] M. K. Dahouda and I. Joe, "A Deep-Learned Embedding Technique for Categorical Features Encoding," in IEEE Access, vol. 9, pp. 114381-114391, 2021, doi: 10.1109/ACCESS.2021.3104357.

[7] Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance. Md Manjurul Ahsan, M.A. Parvez Mahmud, Pritom Kumar Saha, Kishor Datta Gupta, Zahed Siddique.

[8] Machine Learning Essentials: Practical Guide in R. Alboukadei Kassambara.