

## PENERAPAN MACHINE LEARNING DALAM *PREDICTIVE MAINTENANCE* PADA UNIT *INTERCOOLER TURBO* MESIN INDUK NKK SEMT-PIELSTICK 12PC4-2V-570 PADA KAPAL PASSENGER

Invinandri Joko Ahmad <sup>1</sup>, Nurvita Arumsari, S.Si., M.Si. <sup>2</sup>, George Endri Kusuma, S.T., M.Sc.Eng <sup>3</sup>

Program Studi D-IV Teknik Permesinan Kapal, Jurusan Teknik Permesinan Kapal, Politeknik Perkapalan Negeri Surabaya, Indonesia<sup>1\*</sup>

Program Studi D-IV Teknik Permesinan kapal, Jurusan Teknik Permesinan Kapal, Politeknik Perkapalan Negeri Surabaya, Indonesia<sup>2</sup>

Program Studi D-IV Teknik Permesinan Kapal, Jurusan Teknik Permesinan Kapal, Politeknik Perkapalan Negeri Surabaya, Indonesia<sup>3</sup>

Email: [<sup>1\\*</sup>](mailto:invinandrijoko@student.ppns.ac.id); [<sup>2\\*</sup>](mailto:arum@ppns.ac.id); [<sup>3\\*</sup>](mailto:kusuma.george@ppns.ac.id)

---

**Abstract** Predictive maintenance (PdM) is used to predict when equipment may fail and assist in providing adequate monitoring and maintenance planning before potential future failures. In this research there are several topics of discussion regarding predictive maintenance on the NKK-SEMT PIELSTICK 12PC4-2V-570 main engine turbo intercooler unit on passenger ships including how to model predictive maintenance using machine learning using supervised learning methods, namely decision tree and random forest (rf). Decision tree, then what models and methods have a high accuracy value and finally to get the operating hours of the equipment until it fails. The decision tree approach is used to predict category or class labels and construct knowledge based on training data and labels. Random forest is a collection of decision trees generated from random sample selection with different node splitting rules. This model utilises a subset of the features in each tree, then finds the best threshold to split the data. Through turbo system performance anomalies, the use of machine learning with supervised learning methods can process previously classified data to get output in the form of predictive maintenance. Based on the results of predictive maintenance modelling, the decision tree method is the most accurate method with a ratio of training data and testing data division of 60:40 resulting in the highest accuracy. With MAE value 0, RMSE 0, RAE 0%, TP rate 1, FP rate 0, recall 1, F-measure 1, and Accuracy 100%. By using a decision tree model with a data set comparison ratio of 60:40, predict fututre is performed, the equipment is predicted to fail after operating for the next 3112 hours.

---

**Keyword:** Predictive Maintenance, Machine Learning, Main Engine, Intercooler Turbo

---

### 1. PENDAHULUAN

Pemeliharaan peralatan atau sistem yang efektif dan efisien sangat penting untuk operasi perusahaan atau organisasi. Meminimalkan biaya pemeliharaan memiliki dampak positif yang signifikan pada berbagai aspek operasional dan keberlanjutan perusahaan atau organisasi. Mendeteksi potensi kerusakan sebelum terjadi dapat meningkatkan umur peralatan, karena memungkinkan prediksi kapan peralatan atau sistem mungkin mengalami masalah

*Predictive maintenance* (PdM) digunakan untuk memprediksi kapan peralatan mungkin mengalami kegagalan dan membantu dalam menyediakan pemantauan serta perencanaan pemeliharaan yang memadai sebelum potensi kegagalan dimasa mendatang. Melalui implementasi PdM, perusahaan atau organisasi dapat menganalisa sumber penyebab kegagalan, mengurangi masalah reliabilitas peralatan, frekuensi pemeliharaan, dan pada akhirnya mengurangi potensi biaya yang timbul dari tindakan pemeliharaan preventif yang berlebihan [1].

Perawatan dapat dilakukan secara berkala

atau berdasarkan kondisi peralatan atau mesin tersebut. Perawatan berdasarkan kondisi peralatan atau mesin, atau yang biasa disebut dengan *condition based maintenance* (CBM), merupakan perawatan yang dilakukan dengan cara memantau kondisi peralatan atau mesin secara berkala. Informasi yang diperoleh dari pemantauan tersebut kemudian digunakan untuk menentukan jenis dan waktu perawatan yang diperlukan. Seperti halnya sensor, sensor dapat digunakan untuk mengumpulkan data secara *real-time* dari suatu peralatan atau mesin. Data tersebut kemudian dianalisis untuk mendeteksi adanya perubahan kondisi yang dapat mengindikasikan adanya kerusakan.

Suatu sistem turbo tidak lepas dari peralatan *intercooler*. *Intercooler* merupakan komponen yang tergolong *heat exchanger*, memiliki fungsi melepas panas udara yang akan masuk ke dalam ruang bakar mesin. Apabila terdapat kegagalan pada *intercooler* maka dapat mengganggu kinerja mesin induk. Apabila *intercooler* tidak berfungsi secara optimal, pembakaran akan terganggu dan menyebabkan penurunan efisiensi, mengakibatkan kerugian tenaga yang dapat mengurangi performa

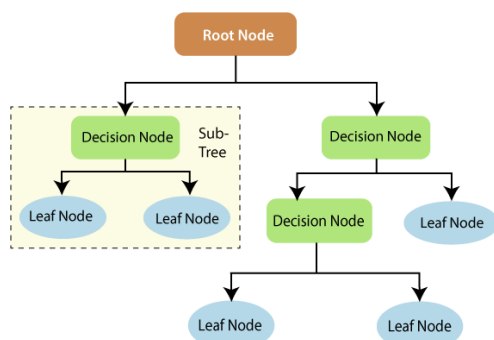
maksimal. Oleh karena itu, perawatan dan perbaikan secara teratur dan terstruktur diperlukan untuk menjaga kinerja intercooler yang optimal.

Pada penelitian ini, guna melakukan *predictive maintenace* digunakan metode *supervised learning*, yaitu *decission tree* dan *random forest*. *Decision tree* merupakan salah satu metode dengan cara menerapkan sistem pengklasifikasian untuk mengelola informasi data. Pendekatan ini digunakan untuk memprediksi kategori atau label kelas dan menyusun pengetahuan berdasarkan data pelatihan dan labelnya. *decission tree* memiliki kelebihan karena mudah dan visualisasi sehingga mempermudah dalam penyajian data [2]. *Random forest* merupakan kumpulan *decision tree* yang dihasilkan dari pemilihan sampel secara acak dengan aturan membelah simpul yang berbeda. Model ini memanfaatkan *subset* dari fitur pada setiap pohon, lalu mencari ambang batas yang terbaik untuk memisahkan data. Salah satu kelebihan metode *random forest* adalah kemampuannya untuk menganalisis data dengan jumlah observasi yang lebih sedikit dibandingkan dengan jumlah variabel penjelas dalam data set [3]

## 2. METODOLOGI

### 2.1 Decision Tree

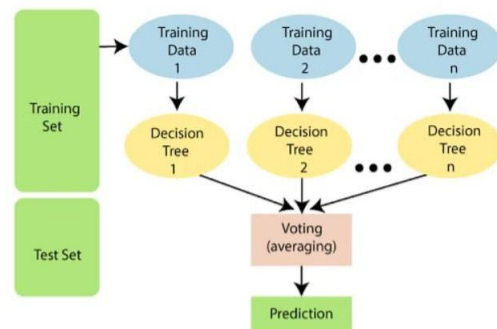
*Decision Tree* merupakan salah satu metode yang sering digunakan dalam proses data mining dengan cara menerapkan sistem pengklasifikasian untuk mengelola informasi yang besar dalam konteks penambangan data. Pendekatan ini digunakan untuk memprediksi kategori atau label kelas, menyusun pengetahuan berdasarkan data pelatihan dan labelnya, dan mengklasifikasikan data yang baru diperoleh. mengelola informasi yang besar dalam konteks penambangan data. Pendekatan ini digunakan untuk memprediksi kategori atau label kelas, menyusun pengetahuan berdasarkan data pelatihan dan labelnya, dan mengklasifikasikan data yang baru diperoleh



Gambar 2.1 Decision Tree Algorithm  
 Sumber : Sinambela, 2023

### 2.2 Random Forest

*Random forest* merupakan kumpulan *decision tree* yang dihasilkan dari pemilihan sampel secara acak dengan aturan membelah simpul yang berbeda. Model ini memanfaatkan subset dari fitur pada setiap pohon, lalu mencari ambang batas yang terbaik untuk memisahkan data. Hasilnya adalah sejumlah pohon yang ditrial dan akan menghasilkan prediksi-prediksi yang berbeda. Hasil prediksi tersebut dapat diinterpretasikan dengan dua cara, yang sering digunakan adalah dipertimbangkan sebagai kelas yang benar berdasarkan suara terbanyak, yang kedua dilakukan menggunakan *scikit-learn* dengan mengimplementasikan algoritmanya berdasarkan rata-rata dari hasil tersebut sehingga menghasilkan prediksi yang akurat. Walaupun secara teoritis berbeda, tetapi rata-rata probabilitas dari *random forest* yang telah ditrial (*scikit-learn*) dan prediksi dengan suara terbanyak, hasilnya tidak akan jauh berbeda dan sering kali mengarah ke hasil yang sebanding [3].



Gambar 2.2 Random Forest Algorithm  
 Sumber : Sinambela, 2023

### 2.3 Perhitungan Error

Evaluasi model performa merupakan salah satu aspek penting dalam membangun model *predictive maintenance*. Beberapa evaluasi performa model yang umum digunakan di antaranya *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dan *Relative Absolute Error* (RAE). Setelah melakukan perhitungan nilai prediksi pada dataset *training* dengan melatih model dalam bentuk metode *supervised learning*, yang bertujuan untuk meminimalkan *cost value for the training set*, langkah selanjutnya adalah memasukkan koefisien (atau bobot) ke dalam model. Proses ini melibatkan penggunaan data *testing* untuk mengevaluasi keakuratan estimasi model terhadap dataset pengujian. Evaluasi dilakukan menggunakan metrik seperti *Mean Squared Error* (MSE), *Root Mean Squared Error* (RMSE). Metrik kinerja ini memiliki rentang nilai antara 0 dan  $\infty$ , dan masing-masing menunjukkan kecocokan yang semakin baik ketika nilai estimasi mendekati 0 [4].

Untuk mencari nilai MAE dapat menggunakan persamaan 1.

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (1)$$

Untuk mencari nilai RMSE dapat menggunakan persamaan 2.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (h(x^{(i)}) - y^{(i)})^2} \quad (2)$$

Untuk mencari nilai RAE dapat menggunakan persamaan 3.

$$RAE = \frac{\sum_{i=1}^m |P_{(ij)} - T_j|}{\sum_{i=1}^m |T_j| - \frac{1}{m} \sum_{i=1}^m |T_j|} \quad (3)$$

## 2.4 Perhitungan Accuracy Model

Selain itu performa *machine learning* dapat dilihat dari beberapa indikator. Dalam penelitian yang dilakukan oleh itu performa *machine learning* dapat dilihat dari beberapa indikator. Dalam penelitian yang dilakukan oleh (Biosci & Dellosa, 2023) kinerja klasifikasi dapat dilihat melalui nilai kinerja klasifikasi dapat dilihat melalui nilai *True positive rate (TP)* ini adalah data positif yang diprediksi benar, *false positive rate (FP)* berupa data negatif yang salah diprediksi sebagai data positif, *precision* menggambarkan tingkat keakuratan antara data yang diminta dan hasil prediksi model. Oleh karena itu, *precision* adalah perbandingan antara prediksi positif dengan semua hasil yang diprediksi sebagai positif. Dari semua prediksi yang benar mengidentifikasi kelas positif, *precision* mengukur berapa banyak di antaranya yang benar benar positif. Nilai *precision* dapat dihitung dengan persamaan 4.

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

*Recall* menggambarkan seberapa baik model dalam mendeteksi kembali sebuah informasi. Oleh karena itu, *recall* adalah perbandingan antara prediksi benar positif dengan semua data yang memang benar positif. *Recall* dapat dihitung dengan persamaan 5.

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

*F-measure* menggambarkan perbandingan rata-rata antara *precision* dan *recall*. Ketika dataset memiliki jumlah data *false negative* dan *false positive* yang samir sama (simetris), kita dapat menggunakan *accuracy*. Namun jika jumlahnya tidak mendekati, maka sebaiknya menggunakan *F-measure* sebagai acuan. *F-Measure* dapat dihitung dengan persamaan 6.

$$F - Measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (6)$$

Matthews Correlation coefficient (MCC), Mengukur kualitas dari pemodelan *machine learning*. Tugasnya mengukur selisih antara nilai

prediksi dengan nilai aktual atau mengukur korelasi antara label yang diprediksi dan label sebenarnya. Dengan memperhitungan data positif dan negatif. MCC dapat dihitung menggunakan persamaan 7.

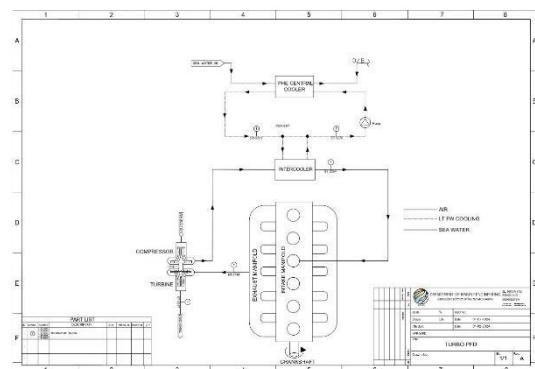
$$MCC = \frac{(TP \times TN - FP \times FN)}{\sqrt{(TP+FP) \times (TP+FN) \times (TN+FP) \times (TN+FN)}} \quad (7)$$

*Receiver Operating Characteristik Area (ROC)*, Menggambarkan klasifikasi biner kinerja model. Kurva ROC membandingkan *TP rate* terhadap *FP Rate* pada berbagai ambang batas klasifikasi, dan arean ROC adalah area dibawah kurva dan terakhir ada *precision-recall curve area (PRC)* merupakan Representasi grafis dari klasifikasi biner kinerja model. Kurva *precision-recall* menggambarkan *precision vs recall* pada berbagai ambang batas klasifikasi, sedangkan area PRC area mewakili area dibawah kurva. merangkum trade-off antara tingkat positif sebenarnya dan nilai prediksi positif untuk model prediktif menggunakan ambang batas probabilitas yang berbeda.

## 3. HASIL DAN PEMBAHASAN

### 3.1 Sistem Turbo

Parameter operasi dipilih sesuai dengan alur proses yang terkait sistem kerja *turbocharger*. *Exhaust gas* yang dihasilkan oleh proses pembakaran akan diarahkan untuk melalui *turbine side*, *turbine side* yang terhubung akan akan memutar *compressore side*. Sehingga udara akan terkompres oleh kompresor. Namun udara yang dihasilkan oleh kompresor memiliki suhu yang tinggi, sehingga perlu diturunkan. Udara yang terkompresi akan masuk kedalam *intercooler* untuk diturunkan suhunya menggunakan media air tawar. Air tawar tersebut akan didinginkan di *central cooler* menggunakan air laut. Udara yang telah melalui *intercooler* akan mengalami penurunan suhu. Udara tersebut yang kemudian akan masuk kedalam *engine (intake)*.



Gambar 1.1 Skema Sistem Turbo

### 3.2 Pelabelan Data

Proses *labeling* data untuk *machine learning* dilakukan dengan membandingkan data yang telah tercatat dengan *standard* yang telah ditetapkan oleh *engine maker*. Pelabelan data dilakukan untuk model *Random Forest* dan *Decision Tree* dengan klasifikasi yang dijelaskan pada Tabel 3.1

Tabel 3.1 Label Klasifikasi Data

| Deskripsi                                      | Klasifikasi | Jumlah Data |
|------------------------------------------------|-------------|-------------|
| Intercooler Air Outlet Temp (side A) $\geq 65$ | Abnormal    | 6           |
| Intercooler Air Outlet Temp (side A) $< 65$    | Normal      | 808         |

Pada Tabel 3.1 dijelaskan mengenai klasifikasi yang digunakan untuk pelabelan data. Klasifikasi yang digunakan berjumlah 2 klasifikasi, yaitu normal dan abnormal. Data normal berupa data sensor *Intercooler Air Outlet Temp* (0254) yang memiliki suhu kurang dari  $65^{\circ}\text{C}$ . Sedangkan data Abnormal berupa data *Intercooler Air Outlet Temp* yang memiliki suhu lebih dari atau sama dengan  $65^{\circ}\text{C}$ . Pada Tabel 3.2 contoh data yang telah diberi label data

Tabel 3.2 Data sensor yang telah diklasifikasi

| Time    | 0202 | 0204 | 0205  | 0235 | 0237 | 0254 | 0273 | 0276 | Keputusan |
|---------|------|------|-------|------|------|------|------|------|-----------|
| 04 : 00 | 379  | 152  | 10366 | 484  | 370  | 54   | 80   | 61   | Normal    |
| 08 : 00 | 375  | 150  | 10097 | 483  | 373  | 54   | 80   | 61   | Normal    |
| 20 : 00 | 379  | 152  | 10420 | 483  | 366  | 54   | 85   | 65   | Normal    |
| 24 : 00 | 379  | 152  | 10353 | 486  | 373  | 54   | 77   | 63   | Normal    |
| 04 : 00 | 382  | 153  | 11043 | 482  | 358  | 54   | 80   | 60   | Normal    |
| 20 : 00 | 384  | 154  | 11049 | 482  | 357  | 54   | 77   | 61   | Normal    |
| 24 : 00 | 383  | 153  | 11049 | 484  | 358  | 54   | 80   | 61   | Normal    |
| 04 : 00 | 377  | 151  | 9645  | 482  | 377  | 54   | 80   | 58   | Normal    |
| 08 : 00 | 370  | 148  | 8027  | 466  | 389  | 52   | 80   | 53   | Normal    |
| 18 : 26 | 370  | 148  | 8027  | 466  | 389  | 65   | 80   | 53   | Abnormal  |
| 18 : 32 | 370  | 148  | 8027  | 466  | 389  | 65   | 80   | 53   | Abnormal  |

Sebelum pembuatan model *predictive maintenance*, dilakukan pembagian *dataset* menjadi 2. Yaitu data *training* dan data *testing*. Data *Training* dipakai untuk membuat prediksi atau menjalankan model yang akan digunakan untuk mengklasifikasikan, sedangkan data *Testing* digunakan untuk mengevaluasi ketepatan model tersebut terhadap *dataset*. Rasio pembagian *dataset training* dan *testing* dilakukan dengan beberapa rasio. Untuk mencari rasio pembagian *dataset* mana yang akan memiliki nilai akurasi tertinggi terhadap pemodelan *predictive maintenance*. Pada Tabel 4.7 ditampilkan jumlah *dataset* berdasarkan rasio pembagian.

Tabel 3.3 pembagian dataset

| No | Jenis Data | Rasio | Jumlah Data | Total |
|----|------------|-------|-------------|-------|
| 1  | Training   | 90%   | 733         | 814   |
|    | Testing    | 10%   | 81          |       |
| 2  | Training   | 80%   | 651         | 814   |
|    | Testing    | 20%   | 163         |       |
| 3  | Training   | 70%   | 570         | 814   |
|    | Testing    | 30%   | 244         |       |
| 4  | Training   | 60%   | 488         | 814   |
|    | Testing    | 40%   | 326         |       |

### 3.3 Pemodelan Supervised Learning

Setelah pembagian *dataset* dilakukan, langkah selanjutnya melakukan pembuatan model *Predictive Maintenance* menggunakan metode *Random Forest* dan *Decision Tree*. Pemodelan dilakukan menggunakan *dataset* yang telah diklasifikasikan (pelabelan) dan dibagi menjadi data *training* dan data *testing* sesuai dengan rasio yang ingin diuji dengan bantuan *software WEKA TOOLS 3.8.6*. Data yang akan dianalisis ke dalam *software* terlebih dahulu diubah dari bentuk file *excel*, yang kemudian akan disimpan ke dalam bentuk ekstensi file *.CSV (comma delimited)*. Selanjutnya file tersebut akan diubah menjadi format *.arff (attribute-relation file format)* sebelum dianalisis menggunakan *weka software*.

```
@relation Dataset

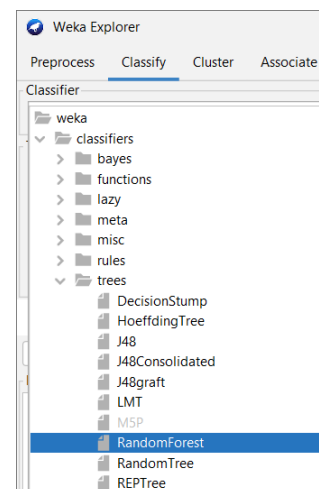
@attribute RPM_ENGINE_2 numeric
@attribute PROPELLING_RPM numeric
@attribute A_TURBOCHARGE_RPM numeric
@attribute A_Exhaust_Gas_Temp_Inlet_Turbo_Charge numeric
@attribute Exhaust_Gas_Temp_Outlet_Turbo_Charge numeric
@attribute Intercooler_Air_Outlet_Temp_(Side_A) numeric
@attribute Turbocharger_Cooling_Water_Temp_Side_A numeric
@attribute Intercooler_Water_Outlet_Temp_Side_A numeric
@attribute species {NORMAL,ABNORMAL}

@data

379 152 10366 484 370 54 80 61 NORMAL
375 150 10097 483 373 54 80 61 NORMAL
379 152 10420 483 366 54 85 65 NORMAL
379 152 10353 486 373 54 77 63 NORMAL
382 153 11043 482 358 54 80 60 NORMAL
384 154 11049 482 357 54 77 61 NORMAL
383 153 11049 484 358 54 80 61 NORMAL
382 153 10988 482 358 54 80 60 NORMAL
382 153 11190 482 357 54 82 60 NORMAL
382 152 10927 483 358 54 80 60 NORMAL
381 152 10945 482 358 54 80 60 NORMAL
382 153 11153 482 357 54 82 60 NORMAL
335 135 3968 388 321 51 77 50 NORMAL
382 153 10805 482 358 54 80 63 NORMAL
383 153 11013 484 358 54 80 61 NORMAL
384 153 11269 484 358 54 82 85 NORMAL
```

Gambar 3.2 konversi data format .arff

Setelah data berhasil diunggah, selanjutnya melakukan pemilihan metode yang ingin digunakan untuk melakukan *classify*. Pemilihan metode dapat dilakukan dengan menuju *command tab Classifier > Choose > Trees (random forest dan decision tree)*.



Gambar 3.3 Tangkapan Layar Fitur Classifier Pada Weka Software

### 3.4 Pemodelan *Random Forest*

#### A. *Random Forest*

Tabel 3.4 Detailed Accuracy By Class (Rf)

| Training | Testing | TP | FP | Precision | Recall | F-Measure | MCC | ROC | PRC   | Class    |
|----------|---------|----|----|-----------|--------|-----------|-----|-----|-------|----------|
| 90       | 10      | 1  | 1  | 0.992     | 1      | 0.996     | -   | 0.5 | 0.992 | Normal   |
|          |         | 0  | 0  | -         | 0      | -         | -   | 0.5 | 0.008 | Abnormal |
| 80       | 20      | 1  | 1  | 0.992     | 1      | 0.996     | -   | 1   | 1     | Normal   |
|          |         | 0  | 0  | -         | 0      | -         | -   | 1   | 1     | Abnormal |
| 70       | 30      | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Normal   |
|          |         | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Abnormal |
| 60       | 40      | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Normal   |
|          |         | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Abnormal |

Berdasarkan hasil pada Tabel 3.4, diketahui bahwa *percentage split* yang menghasilkan akurasi tertinggi berdasarkan nilai TF, FP, *Precision*, *Recall*, *F-measure*, MCC, ROC, dan PRC adalah *percentage split* dengan rasio *training* dan *testing* 70,30 dan 60,40. Dengan memiliki nilai TF, *Precision*, *Recall*, *F-measure*, MCC, ROC, dan PRC adalah 1. Sedangkan *percentage split* yang menghasilkan akurasi terendah adalah *percentage split* dengan rasio *training* dan *testing* 90:10. dengan nilai TP adalah 1, nilai FP adalah 1, nilai *Precision* adalah 0.992, nilai *Recall* adalah 1, nilai *F-measure* adalah 0.996, nilai ROC adalah 0.5, dan nilai PRC adalah 0.008.

Tabel 3.5 Confusion Matrix Model Rf

| Training | Testing | Normal | Abnormal |
|----------|---------|--------|----------|
| 90       | 10      | 727    | 0        |
|          |         | 6      | 0        |
| 80       | 20      | 646    | 0        |
|          |         | 5      | 0        |
| 70       | 30      | 567    | 0        |
|          |         | 0      | 3        |
| 60       | 40      | 485    | 0        |
|          |         | 0      | 3        |

Pada Tabel 3.5 menampilkan hasil *confusion matrix* dari model *random forest*. Dimana pada rasio *percentage split* 90:10 dan 80:20 terdapat data yang mengalami kesalahan klasifikasi. Pada rasio *percentage split* 90:10 terdapat 6 data yang mengalami kesalahan pelabelan pada label normal ke label abnormal. Sedangkan pada rasio *percentage split* 80:20 terdapat 5 data yang mengalami kesalahan pelabelan pada label normal ke label abnormal. Model *random forest* mampu mengklasifikasikan dengan baik dengan nilai *precision* lebih dari 90%. Selain itu untuk nilai *recall* memiliki nilai diatas 90% yang menunjukkan bahwa pemodelan *random forest* mampu mengklasifikasikan sebagian besar data *true positive* diantara data prediksi *true positive*. Sehingga prediksi menggunakan model *random forest* pada data test menghasilkan akurasi diatas 90%.

#### B. Decision Tree

Tabel 3.6 Detailed Accuracy By Class

| Training | Testing | TP | FP | Precision | Recall | F-Measure | MCC | ROC | PRC   | Class    |
|----------|---------|----|----|-----------|--------|-----------|-----|-----|-------|----------|
| 90       | 10      | 1  | 1  | 0.992     | 1      | 0.996     | -   | 0.5 | 0.992 | Normal   |
|          |         | 0  | 0  | -         | 0      | -         | -   | 0.5 | 0.008 | Abnormal |
| 80       | 20      | 1  | 1  | 0.992     | 1      | 0.996     | -   | 0.5 | 0.992 | Normal   |
|          |         | 0  | 0  | -         | 0      | -         | -   | 0.5 | 0.008 | Abnormal |
| 70       | 30      | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Normal   |
|          |         | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Abnormal |
| 60       | 40      | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Normal   |
|          |         | 1  | 0  | 1         | 1      | 1         | 1   | 1   | 1     | Abnormal |

Berdasarkan hasil dari Tabel 3.6 , diketahui bahwa *percentage split* yang menghasilkan akurasi tertinggi berdasarkan nilai TF, FP, *Precision*, *Recall*, *F-measure*, MCC, ROC, dan PRC adalah *percentage split* dengan rasio *training* dan *testing* 70,30 dan 60,40. Dengan memiliki nilai TF, *Precision*, *Recall*, *F-measure*, MCC, ROC, dan PRC adalah 1. Sedangkan *percentage split* yang menghasilkan akurasi terendah adalah *percentage split* dengan rasio *training* dan *testing* 90:10. dengan nilai TP adalah 1, nilai FP adalah 1, nilai *Precision* adalah 0.992, nilai *Recall* adalah 1, nilai *F-measure* adalah 0.996, nilai ROC adalah 0.5, dan nilai PRC adalah 0.008.

Tabel 3.7 Confusion Matrix Model Dt

| Training | Testing | Normal | Abnormal |
|----------|---------|--------|----------|
| 90       | 10      | 727    | 0        |
|          |         | 6      | 0        |
| 80       | 20      | 646    | 0        |
|          |         | 5      | 0        |
| 70       | 30      | 567    | 0        |
|          |         | 0      | 3        |
| 60       | 40      | 485    | 0        |
|          |         | 0      | 3        |

Pada Tabel 3.7 menampilkan hasil *confusion matrix* dari model *decision tree*. Dimana pada rasio *percentage split* 90:10 dan 80:20 terdapat data yang mengalami klasifikasi. Pada rasio *percentage split* 90:10 terdapat 6 data yang mengalami kesalahan pelabelan pada label normal ke label abnormal. Sedangkan pada rasio *percentage split* 80:20 terdapat 5 data yang mengalami kesalahan pelabelan pada label normal ke label abnormal. Model *decision tree* mampu mengklasifikasikan dengan baik dengan nilai *precision* lebih dari 90%. Selain itu untuk nilai *recall* memiliki nilai diatas 90% yang menunjukkan bahwa pemodelan *random forest* mampu mengklasifikasikan sebagian besar data *true positive* diantara data prediksi *true positive*. Sehingga prediksi menggunakan model *random forest* pada data test menghasilkan akurasi diatas 90%.



### 3.5 Evaluasi Pemodelan *Supervised Learning* A. *Random Forest*

Tabel 3.5 Detail Hasil Evaluasi Model (Rf)

| Training | Testing | MAE    | RMSE   | RAE      | Corect   | Incorect |
|----------|---------|--------|--------|----------|----------|----------|
| 90       | 10      | 0.0082 | 0.0905 | 40.8532% | 99.1814% | 0.8186%  |
| 80       | 20      | 0.0091 | 0.0673 | 46.2814% | 99.232%  | 0.768%   |
| 70       | 30      | 0.0035 | 0.0302 | 16.2685% | 100%     | 0%       |
| 60       | 40      | 0.0032 | 0.0261 | 17.3462% | 100%     | 0%       |

Berdasarkan hasil dari Tabel 3.8 uji coba metode *Random Forest* dengan menggunakan 4 kombinasi *percentage split* data *training* dan data *testing*, diketahui bahwa *percentage split* yang menghasilkan akurasi tertinggi berdasarkan nilai MAE, RMSE, RAE, dan Ketepatan adalah *percentage split* dengan rasio *training* dan *testing* 60,40. Dengan memiliki nilai MAE adalah 0.0032, nilai RMSE adalah 0.0261, nilai RAE adalah 17.3462%, dan Ketepatan sebesar 100%.

### B. *Decision Tree*

Tabel 3.6 Detail Hasil Evaluasi Model Dt

| Training | Testing | MAE    | RMSE   | RAE      | Corect   | Incorect |
|----------|---------|--------|--------|----------|----------|----------|
| 90       | 10      | 0.0082 | 0.0905 | 40.8532% | 99.1814% | 0.8186%  |
| 80       | 20      | 0.0137 | 0.0873 | 69.9509% | 99.232%  | 0.768%   |
| 70       | 30      | 0      | 0      | 0%       | 100%     | 0%       |
| 60       | 40      | 0      | 0      | 0%       | 100%     | 0%       |

Berdasarkan hasil dari Tabel 3.9 uji coba metode *Decision Tree* dengan menggunakan 4 kombinasi *percentage split* data *training* dan data *testing*, diketahui bahwa *percentage split* yang menghasilkan akurasi tertinggi berdasarkan nilai MAE, RMSE, RAE, dan Ketepatan adalah *percentage split* dengan rasio *training* dan *testing* 60,40 dan 70,30. Dengan memiliki nilai MAE adalah 0, nilai RMSE adalah 0, nilai RAE adalah 0%, dan Ketepatan sebesar 100%

### 3.6 Predict Future Intercooler Turbo

Pada penelitian ini, penulis ingin melakukan prediksi *failure* berdasarkan data yang dihasilkan sensor pada sistem *turbo charger* pada mesin induk kapal pada saat beroperasi. Pada prediksi kegagalan *intercooler turbo* menggunakan data *testing* dengan parameter terbaik pada setiap masing-masing metode.

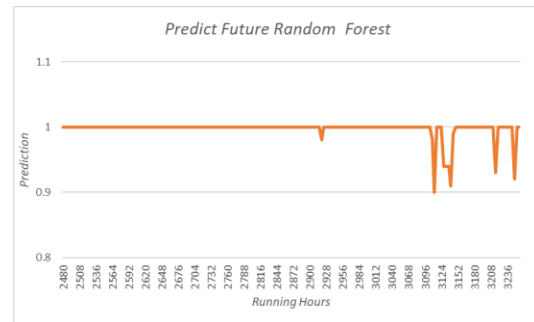
#### A. *Random Forest*

Prediksi kegagalan menggunakan metode *random forest* dilakukan dengan parameter algoritma *random forest* terbaik dengan rasio perbandingan data *training* dan data *testing* sebesar 60:40. Berdasarkan pemodelan metode *random forest* untuk melakukan *predictive maintenance* dengan rasio pembagian data *training* dan data *testing* sebesar 60:40 menghasilkan nilai akurasi tertinggi dengan nilai diatas 85%. Hal tersebut didapatkan dari *confusion matrix*.

Tabel 3.7 Detail Hasil Evaluasi Model Rf Data Testing

| Testing | MAE    | RMSE   | RAE     | Corect | Incorect |
|---------|--------|--------|---------|--------|----------|
| 40      | 0.0038 | 0.0166 | 5.9988% | 100%   | 0%       |

Berdasarkan hasil dari Tabel 3.10 uji coba metode *random forest* dengan menggunakan data *testing* menghasilkan akurasi tertinggi berdasarkan nilai MAE, RMSE, RAE, dan Ketepatan adalah *percentage split*. Dengan memiliki nilai MAE adalah 0.0038, nilai RMSE adalah 0.0166, nilai RAE adalah 5.9988%, dan Ketepatan sebesar 100%. Prediksi kegagalan dilakukan menggunakan bantuan *test option suplied test set* pada software *WEKA Tools*.



Gambar 3.4 Grafik Output Prediksi Kegagalan

Pada Gambar 3.4 menampilkan contoh output dari *predictive maintenance*. Pada tahap ini dilakukan perhitungan operasional peralatan pada periode waktu tertentu hingga komponen mengalami kegagalan. sensor memiliki satuan jam, karena data sensor yang dikumpulkan di *record* setiap 4 jam selama mesin beroperasi. Berdasarkan hasil prediksi untuk kerusakan *intercooler turbo* akan terjadi setelah beroperasi selama 3112 jam kedepan.

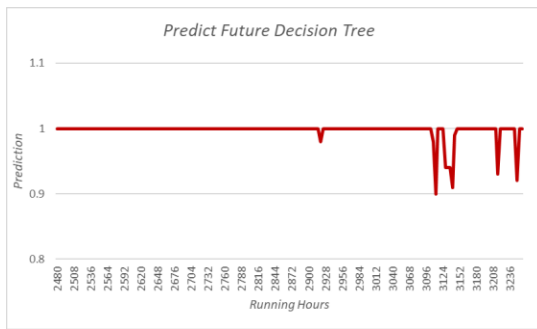
#### B. *Decicion Tree*

Prediksi kegagalan menggunakan metode *decision tree* dilakukan dengan parameter algoritma *decision tree* terbaik dengan rasio perbandingan data *training* dan data *testing* sebesar 60:40. Berdasarkan pemodelan metode *decision tree* untuk melakukan *predictive maintenance* dengan rasio pembagian data *training* dan data *testing* sebesar 60:40 menghasilkan nilai akurasi tertinggi dengan nilai diatas 85%. Hal tersebut didapatkan dari *confusion matrix*.

Tabel 3.11 Detail Hasil Evaluasi Model Dt Data Testing

| Testing | MAE | RMSE | RAE | Corect | Incorect |
|---------|-----|------|-----|--------|----------|
| 40      | 0   | 0    | 0%  | 100%   | 0%       |

Berdasarkan hasil dari Tabel 3.11 uji coba metode *random forest* dengan menggunakan data *testing* menghasilkan akurasi tertinggi berdasarkan nilai MAE, RMSE, RAE, dan Ketepatan adalah *percentage split*. Dengan memiliki nilai MAE adalah 0, nilai RMSE adalah 0, nilai RAE adalah 0%, dan Ketepatan sebesar 100%. Prediksi kegagalan dilakukan menggunakan bantuan *test option suplied test set* pada software *WEKA Tools*.

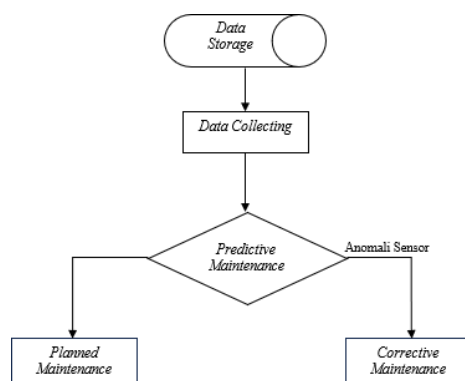


Gambar 3.5 Grafik Output Prediksi Kegagalan

Pada Gambar 3.5 menampilkan contoh output dari *predictive maintenance*. Pada tahap ini dilakukan perhitungan operasi pada periode waktu tertentu hingga komponen mengalami kegagalan. Sensor yang dihitung memiliki satuan jam, karena data sensor yang dikumpulkan di *record* setiap 4 jam selama mesin beroperasi. Berdasarkan hasil prediksi untuk kerusakan *intercooler turbo* akan terjadi setelah beroperasi selama 2588 jam kedepan.

### 3.7 Strategi Perawatan

Berdasarkan pembuatan model *predictive maintenance* yang telah dibuat. Didapatkan hasil bahwa penerapan *machine learning* untuk melakukan *predictive maintenance* memiliki tingkat akurasi yang cukup baik dalam mendeteksi kegagalan *intercooler turbo* pada mesin induk kapal. Penentuan strategi perawatan dapat dilihat pada Gambar 3.6



Gambar 3. 6 Skema Strategi Perawatan

Berdasarkan skema strategi perawatan, langkah pertama dalam *predictive maintenance* adalah pengumpulan data yang telah ter-*record* pada *engine log sheet*. Data tersebut berupa hasil rekaman dari setiap sensor pada sistem yang terdapat di *engine room*. Data tersebut akan diolah menggunakan pemodelan *predictive maintenance* dengan metode yang sudah ditentukan. Pada penelitian ini digunakan metode *random forest* dan *decision tree*. Dimana dalam penentuan parameter algoritma terbaik, dilakukan dengan menggunakan beberapa rasio perbandingan dataset pada setiap metodenya. Adapun pembagian data *training* dan data *testing* yang

ingin dibandingkan untuk mencari hasil akurasi tertinggi ialah, 90:10, 80:20, 70:30, dan 60:40. Setelah didapatkan rasio pembagian dataset yang menghasilkan akurasi tertinggi, yaitu pada penelitian ini menggunakan rasio 60:40. Selanjutnya menentukan pemilihan *maintenance* yang akan dilakukan, antara *planned maintenance* atau *corrective maintenance*.

#### A. Corrective Maintenance

*Corrective maintenance* adalah jenis perawatan atau pemeliharaan yang dilakukan setelah terjadi kegagalan atau kerusakan pada peralatan. Tujuan dari *corrective maintenance* adalah untuk memperbaiki kerusakan tersebut dan mengembalikan sistem atau peralatan tersebut ke dalam kondisi yang berfungsi dengan baik. Kerusakan peralatan/*equipment* dapat mengganggu operasional kapal. Kegagalan pada *intercooler* akan menyebabkan penurunan efisiensi mesin. *Corrective maintenance* dilakukan saat terdapat anomali pada sistem turbo saat kapal sedang beroperasi, maka sensor akan mendeteksi anomali tersebut dan akan menyalakan alarm pada *engine room*. Ketika alarm berbunyi ABK akan langsung melakukan pengecekan pada peralatan yang mengalami kegagalan. Jika terjadi kerusakan, maka akan langsung diperbaiki (*corrective maintenance*). Sehingga peralatan dapat berfungsi dengan normal kembali dan mencegah kerusakan yang lebih parah. Selain itu kapal dapat kembali beroperasi.

#### B. Planned Maintenance

*Planned maintenance* atau perawatan terencana adalah jenis perawatan yang dilakukan secara teratur dan direncanakan sebelumnya untuk menjaga kinerja optimal dari sistem dan peralatan.. Tujuan utama dari *planned maintenance* adalah untuk mencegah terjadinya kerusakan atau kegagalan yang tidak terduga, serta memperpanjang umur pakai peralatan. Mengacu pada peraturan kementerian perhubungan republik indonesia HK.103/I/4/DJPL-14 untuk kapal penumpang wajib melakukan *docking*/pelimbungan setiap setahun sekali. Pelimbungan/*docking* yang dilakukan merupakan pemeriksaan *renewal survey*, mencakup pengecekan kondisi struktural bangunan kapal, termasuk pemeriksaan lapisan luar bagian bawah kapal serta pengecekan mesin dan peralatan tambahannya, untuk memastikan kapal tetap sesuai dengan standar yang ditetapkan. Mengacu pada *rules* BKI class Part.1 Vol. 7 Sec. 2, mewajibkan kapal penumpang melakukan *renewal survey* setiap setahun sekali. Perawatan *intercooler turbo* dapat dilakukan bersamaan saat dilakukannya *docking* tahunan. Hal ini dilakukan untuk mencegah terjadinya kerusakan saat kapal beroperasi. Selain itu, dapat menghemat waktu

perbaikan/*downtime* karena perawatan dilakukan bersamaan dengan *docking* tahunan.

Selanjutnya melakukan perhitungan terkait jam operasional kapal untuk mengetahui interval perawatan. Kapal *passenger* yang dijadikan objek penelitian membutuhkan 32 jam dalam melakukan sekali perjalanan/pelayaran, sedangkan dalam seminggu kapal melakukan perjalanan sebanyak 4 kali dengan rute yang sama, sehingga dalam seminggu kapal beroperasi selama 128 jam, sehingga dalam 1 tahun kapal beroperasi selama 6.144 jam, sehingga dalam 5 tahu kapal beroperasi selama 30.720 jam. Pada operasional kapal juga terdapat waktu *downtime*, dimana *downtime* tersebut terjadi ketika kapal menjalani *docking* untuk melakukan *survey* tahunan (*renewal survey*), untuk kapal *passenger docking* dilakukan 1 kali dalam setahun. Berikut jadwal *docking* yang dilakukan kapal *passenger*

Tabel 3.14 Jadwal Docking Kapal

| Tanggal Docking    | Lama Docking (Hari) | Lama Docking (Jam) |
|--------------------|---------------------|--------------------|
| 7 - 19 November    | 13 Hari             | 312 Jam            |
| 16 - 24 Februari   | 9 Hari              | 216 Jam            |
| 6 - 11 Maret       | 6 Hari              | 144 Jam            |
| 25 Maret - 1 April | 8 Hari              | 192 Jam            |
| 7 - 18 Juni        | 12 Hari             | 288 Jam            |
| Total              | 48 Hari             | 1.152 Jam          |

selama beroperasi.

Setelah mengetahui total operasional kapal dan waktu *downtime* (*docking*) kapal selanjutnya dilakukan perhitungan mengenai operasional kapal aktual untuk melakukan prediksi interval perawatan dengan rumus sebagai berikut.

$$TJO = \sum_{i=1}^n Ti - T_{downtime}$$

$$= 30.720 \text{ jam} - 1.152 \text{ jam}$$

$$= 29.568 \text{ jam}$$

Setelah didapatkan total jam operasional aktual selanjutnya melakukan perhitungan interval perawatan dengan rumus sebagai berikut.

$$IP = \frac{\text{Total Jam operasional (TJO)}}{\text{Jumlah Docking yang Dilakukan}}$$

$$= \frac{29.568 \text{ jam}}{5}$$

$$= 5913.6 \text{ jam}$$

Dari perhitungan interval perawatan yang dilakukan bahwa akan dilakukan perawatan setiap setelah 5914 jam beroperasi.

#### 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, berikut akan diuraikan kesimpulan yang dapat ditarik dari analisis mendalam terhadap data dan temuan yang telah disajikan sebelumnya.

Pemodelan *predictive maintenance* berhasil dilakukan dengan menggunakan *machine learning* dengan data parameter operasional sistem *intercooler turbo* mesin induk kapal. Dengan target parameter jam operasional yang dibutuhkan peralatan hingga mencapai kegagalan. Pemodelan yang dilakukan menggunakan bantuan *software weka tools* dengan *test option* menggunakan *supplied test*.

Pemodelan *decision tree* merupakan metode yang paling akurat dalam pengembangan model *predictive maintenance* pada sistem turbo mesin induk. Dengan rasio pembagian data set *training* dan *testing* 60:40 menghasilkan akurasi tertinggi, ditinjau dari nilai MAE = 0, RMSE = 0, RAE = 0%, Correct = 100%, TP Rate = 1, Fp Rate = 0, Recall = 1, F-Measure = 1, MCC = 1, ROC = 1

Berdasarkan hasil *predict future* menggunakan *option test supplied test*, dengan metode *decision tree* dengan rasio pembagian data set *training* dan *testing* 60:40 atau pemodelan dengan metode yang menghasilkan nilai akurasi tertinggi. Bahwa peralatan membutuhkan 3112 jam operasional hingga mengalami kegagalan. Sehingga untuk jenis *maintenance* yang dilakukan cukup menerapkan *corrective maintenance*. Yaitu ketika sensor mendeteksi anomali baru akan dilakukan perbaikan. Sehingga peralatan dapat berfungsi kembali dengan segera tanpa harus menunggu waktu *docking*.

#### 5. DAFTAR PUSTAKA

- [1] R. Kizito, P. Scruggs, X. Li, R. Kress, M. Devinney, and T. Berg, "The application of random forest to predictive maintenance," *IISE Annu. Conf. Expo 2018*, pp. 354–359, 2018.
- [2] E. Priyanti, "Penerapan Decision Tree Untuk Klasifikasi Tingkat Pendapatan," *IJCIT (Indonesian J. Comput. Inf. Technol.)*, vol. 7, no. 1, pp. 7–12, 2022.
- [3] D. P. Sinambela, H. Naparin, M. Zulfadhilah, and N. Hidayah, "Implementasi Algoritma Decision Tree dan Random Forest dalam Prediksi Perdarahan Pascasalin," *J. Inf. dan Teknol.*, vol. 5, no. 3, pp. 58–64, 2023, doi: 10.60083/jidt.v5i3.393.
- [4] W. K. O. Ho, B. S. Tang, and S. W. Wong, "Predicting property prices with machine learning algorithms," *J. Prop. Res.*, vol. 38, no. 1, pp. 48–70, 2021, doi: 10.1080/09599916.2020.1832558.