Forecasting Temperatures of a Synchronous Motor with Permanent Magnets Using Machine Learning

Oleh Soprun[1], Myroslava Bublyk[2][0000-0003-2403-0784], Yurii Matseliukh[3][0000-0002-1721-7703], Vasyl Andrunyk[4][0000-0003-0697-7384], Lyubomyr Chyrun[5][0000-0002-9448-1751], Ivan Dyyak[6][0000-0001-5841-2604], Anatoly Yakovlev[7][0000-0003-0751-7072], Michael Emmerich[8][0000-0002-7342-2090], Oleksandr Osolinsky[9][0000-0002-0136-395X], Anatoliy Sachenko[10][0000-0002-0907-3682]

[1-4]Lviv Polytechnic National University, Lviv, Ukraine
[5-6]Ivan Franko National University of Lviv, Lviv, Ukraine
[7]National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine
[8]Leiden University, Leiden, The Netherlands
[9-10]Ternopil National Economic University, Ternopil, Ukraine

oleh.soprun.sa.2017@lpnu.ua[1], my.bublyk@gmail.com[2], indeed.post@gmail.com[3], vasyl.a.andrunyk@lpnu.ua[4], Lyubomyr.Chyrun@lnu.edu.ua[5], ivan.dyyak@lnu.edu.ua[6], yakovlevkpi@gmail.com[7]

Abstract. This project is based on the data set that comprises several sensor data collected from a PMSM deployed on a test bench. The PMSM represents a German OEM's prototype model. The LEA department at Padeborn University collected test bench measurement. The main purpose of the data set's recording is to be able to model the stator and rotor temperatures of a PMSM in real-time. Due to the intricate structure of an electric traction drive, direct measurement with thermal sensors is not possible for rotor temperatures, and even in case of the stator temperatures, sensor outage or even just deterioration cannot administer properly without redundant modelling. In addition, precise thermal modelling gets more and more important with the rising relevance of functional safety. The main task in this project is to design a model with appropriate feature engineering that estimates four target temperatures casually.

Keywords: Synchronous Motor, Forecasting, Machine Learning,

1 Introduction

Article purpose is predicted the temperature of a synchronous motor with a permanent magnet using machine-learning methods [1-6]. Object of research is permanent-magnet synchronous motor (PMSM). Subject of research is forecasting the temperature of a synchronous motor with a permanent magnet using machine-learning methods based on a data set containing sensory data collected from PMSM placed on a test bench. PMSM is a German prototype model of OBO. The LEA department of the University of Paderborn assembled the measuring stand. The main task in this work is to develop a model that predicts four target temperatures (permanent magnet surface, stator yoke, stator teeth and stator winding) at random. The main difference between a synchronous motor with permanent magnets and an induction motor is the rotor. Investigations presented in [1-6] show that PMSM is approximately 2% more efficient than a premium-efficiency induction motor (IE3), where provided that the stator has the same design and the same variable frequency drive is used for control. In this case, PMSM has the best performance compared to other electric motors, which makes the PMSM study relevant today.

Having accurate engine temperature estimates helps the automotive industry produce engines with lower material content and enables management strategies to use the engine to its maximum potential.

An electric motor with implicit poles has equal inductance along the longitudinal and transverse axes, whereas an electric motor with explicit poles has a transverse inductance not equal to the longitudinal one [1, 7-12]. Also, on a design of a rotor of PMSM divide into such ones:

- Electric motor with surface installation of permanent magnets.
- Electric motor with built-in (incorporated) magnets.

PMSM cannot started when directly connected to the network. To do this, the following options are used:

- Start with an additional engine. To do this, the shaft is connected to the shaft of another electric machine. This method is expensive and practically not used;
- Start in asynchronous mode. The rotors of such electric motors have a short-circuited winding. The start-up takes place in asynchronous mode. After entering the synchronization, the rotor winding is disconnected;
- Start with a frequency converter. The frequency converter is included in the stator winding circuit and supplies voltage to them by gradually increasing the frequency.

2 Literature Review

The study of a permanent magnet synchronous motor is conducted in the laboratory of the University of Paderborn, where the following results are presented in [2-5]:

- Determination of rotor temperature for the inner part of the PMSM using an accurate flow monitor [2]. This investigation presents an extended method for determining the temperature of the PMSM rotor during dynamic operations. The approach is based on the observed flow of the main wave and therefore largely does not depend on the conditions of the coolant. The measurement results prove satisfactory results of the observer's device [2].
- Real-time methods for determining the temperature of PMSM [3]. The authors of [3] believe the PMSM is widely used in automotive traction drives with high operation and other industrial areas. The temperature of the magnet is of great interest in the service life of the device, safe operation and monitoring. Since direct measurement of the magnet temperature is not possible in most cases, this contribution

provides an overview of modern methods for determining the magnet temperature based on the model in the PMSM. This publication provides brief descriptions of these methods, followed by a direct comparison of the disadvantages and advantages, culminating in the prospect of further research in this area [3].

- Synthesis of methods of direct and indirect temperature estimation for PMSM [4]. Maximizing the degree of thermal use of PMSM reduces the weight, volume and cost of the engine. For this reason, real-time temperature information is required. However, this cannot obtain solely by measuring sensors due to costs and design flaws. Thus, in recent years, several methods of direct temperature modelling have investigated, such as heat networks with concentrated parameters (LPTN), as well as methods of indirect temperature estimation based on accurate models of electric motors that detect changes in temperature-sensitive parameters. Because these methods are usually independent of each other, fusion methods can used to increase the accuracy and reliability of the joint assessment. In this study, a Kalman filter (KF) is successfully combined with a low-order LPTN, an electric model permanent magnet temperature monitor, and a built-in winding temperature sensor to achieve this goal. Measurement results for PMSM with a capacity of 50 kW confirm the increased productivity of KF in comparison with individual assessment methods [4].
- Deep residual convolutional and recurrent neural networks for temperature estimation in PMSM [5]. Most traction drive applications that use PMSM do not have precise temperature control capabilities, so safe operation is ensured by expensive, oversized materials and their efficient use. In this paper, deep recurrent and convolutional neural networks with residual connections are empirically evaluated for their applicability to solve the problem of sequential learning of the forecast of latent high-dynamic temperatures inside the SPDM. The search for the model hyper parameter is performed sequentially using Bayesian optimization on different cores of the random number generator in order to assess the consistency of model learning and probabilistic search of promising topologies, as well as optimization strategies. It is found that the root mean square error and normal characteristics of trained neural networks correspond to the established methods of real-time modelling, such as thermal networks with concentrated parameters, without requiring for their design special knowledge in the field [5].

To implement the main task of this study, the method of machine learning Random Forest is chosen [13-21].

Random forest is an ensemble machine learning method for classification, regression, and other tasks that operate by constructing numerous decision-making trees during model training and producing fashion for classes (classifications) or averaged prediction (regression) of constructed trees [22-29]. The algorithm combines two main ideas: the Braiman method of begging and the method of random subspaces proposed by Xo. Advantages of the Random Forest method [30-38]:

- High quality of the received models, in comparison with SVM and boosting, and much better, than at neural networks;
- Ability to efficiently process data with numerous characteristics and classes;

- Insensitivity to scaling (and in general to any monotonous transformations) of values of signs.
- Both continuous and discrete features are treated equally well. There are methods of constructing trees according to data with omitted values of features.
- High scalability of the method.

Disadvantages of the Random Forest method [39-48]:

- The algorithm tends to relearn on some tasks, especially with a lot of noise in the data set;
- Learning large numbers of deep trees can be costly (but can be parallel) and use a lot of memory.

3 Data Set Description

This study uses a data set that contains sensory data collected from the PMSM placed on a test bench. PMSM is a German prototype model from the original equipment manufacturer. The LEA department of the University of Paderborn assembled the measuring stand. The main purpose of recording a data set is to be able to simulate the temperature of the stator and rotor of the SPD in real time. Due to the complex design of the traction drive, direct measurement of the rotor temperature by thermal sensors is not possible, increase in stator temperature, sensor shutdown or even simply deterioration of the quality of work cannot properly monitor without excessive simulation. In addition, accurate thermal modelling is becoming increasingly important with the importance of functional safety. The main features of the selected data set are the next ones:

- ambient - The ambient temperature is measured by a temperature sensor located close to the stator;
- coolant - The temperature of the coolant. The motor is cooled by water. The measurement is performed at the outflow of water;
- u_d - Voltage of the d-component;
- u_q - Q-component voltage;
- motor_speed - Engine speed;
- torque - Torque induced by current;
- i_d - Current d-component;
- i_q - Current of the q-component;
- profile_ID - Each measurement session has a unique identifier.

Target features of the selected data set are the next ones:

- pm – The surface temperature of the permanent magnet, which is the temperature of the Rotor;
- stator_yoke – The stator yoke temperature is measured using a temperature sensor;
- stator_tooth – The temperature of the stator teeth is measured using a temperature sensor;

- stator_winding – The temperature of the stator winding is measured using a temperature sensor.

Additional information about the data set are the next ones:

- All recordings are selected at a frequency of 2 Hz (One row in 0.5 seconds). The data set consists of several measurement sessions, which can distinguished from each other by the column "profile_id". The measurement session can last from one to six hours. The number of sessions: 52.
- The engine is accelerated by manually designed driving cycles, indicating the reference engine speed and reference torque.
- Currents in the d / q coordinates (columns "i_d" and "i_q") and voltages in the coordinates d / q (columns "u_d" and "u_q") are the result of a standard control strategy that tries to follow the reference speed and torque.
- The columns "motor_speed" and "torque" are the resulting values achieved by this strategy, obtained from the specified currents and voltages.
- Most motion cycles denote random wanderings in the velocity-torque plane to simulate real motion cycles more accurately than constant excitements and difficulties.

Given that, the main task of this project is to develop a model that predicts the temperature values of such SPDM elements as stator and rotor according to already collected sensory data from the SPD prototype model, this project should not consider as a development of a global SPD control system in cars or other mechanisms. It is better to consider it as a part of this system (hereinafter: component), which is responsible for predicting stator and rotor temperatures using sensor data from other elements of PMSM, because to obtain data from stator or rotor using temperature sensors is unreliable and commercially unprofitable.

The following are two UML diagrams, namely an activity diagram that describes the process of the machine-learning model and a sequence diagram that describes the interaction between the control system and the model [10-17] (Fig. 1-2).

This diagram on Fig. 1-2 shows the following activities [18-23]:

- Receiving a request from the system to learn the model - the system sends a request to the component of learning the model.
- Getting data for model training - the component receives data from PMSM sensors for training.
- Conducting model training - machine learning model training is performed using the obtained data.
- Saving the trained model - after completion of training the model is saved.
- Receiving a request from the system for forecasting - the system sends a request for forecasting the model.
- Predicting - using the model, the stator temperature values are predicted.
- Sending the forecast result to the system - the forecast results are sent back to the system.

- Expecting new requests from the system - the component expects new requests to learn the model or predict temperature values.

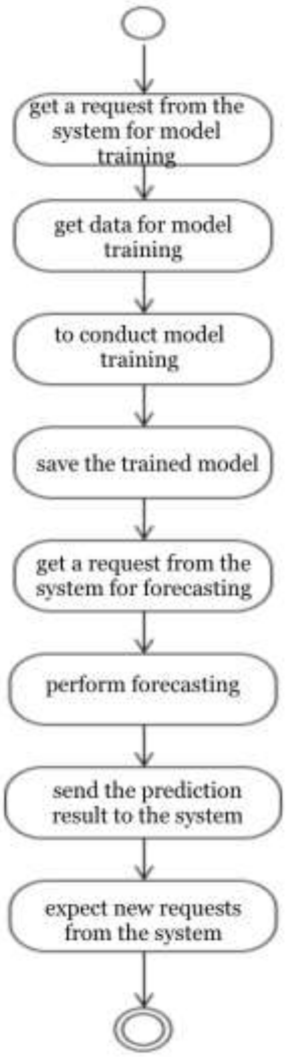

Fig. 1. Activity diagram

The control system sends a request to the component responsible for sensors on the PMSM for data; this component reads data from the sensors and sends data to the control system. The control system requests model training and sends sensor data from the component responsible for the model (hereinafter: CM). The CM performs Random Forest Regression model training and stores the trained model, then sends a message to the model control system ready for forecasting.

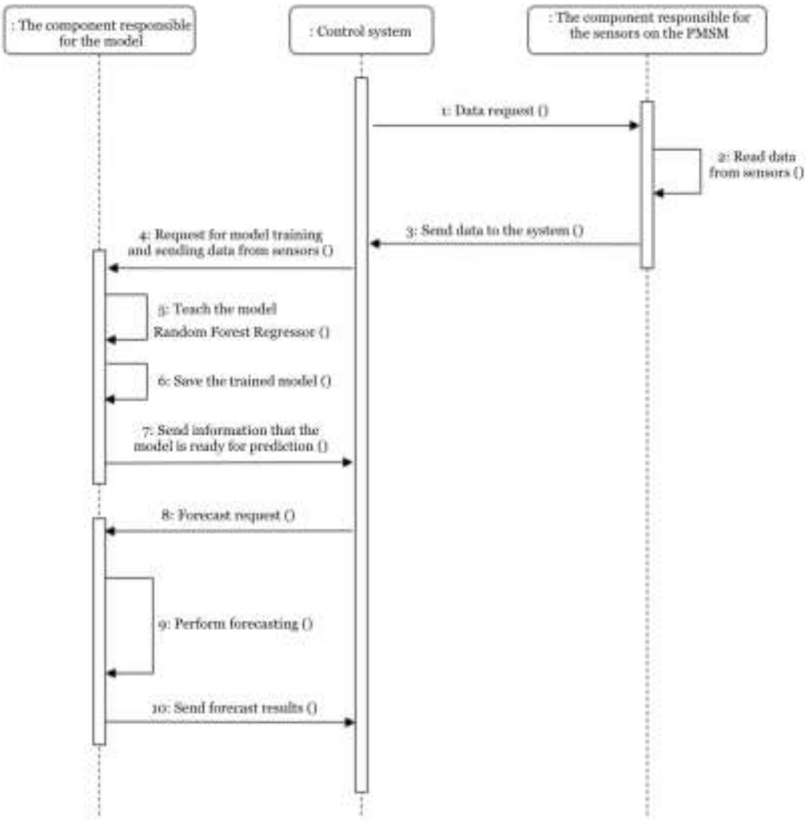

Fig. 2. Sequence diagram.

4 Case Study

4.1 Obtained Results Implementation and Analysis Description

Fig. 3 shows upload data to the format *pandas dataframe*:

	ambient	coolant	u_d	u_q	motor_speed	torque	i_d	i_q	pm	stator_yoke	stator_tooth	stator_winding	profile_id
0	-0.752143	-1.118446	0.327935	-1.297858	-1.222428	-0.250182	1.029572	-0.245860	-2.522071	-1.831422	-2.066143	-2.018033	4
1	-0.771263	-1.117021	0.329665	-1.297686	-1.222429	-0.249133	1.029509	-0.245832	-2.522418	-1.830969	-2.064859	-2.017631	4
2	-0.782892	-1.116681	0.332771	-1.301822	-1.222428	-0.249431	1.029448	-0.245818	-2.522673	-1.830400	-2.064073	-2.017343	4
3	-0.780935	-1.116764	0.333700	-1.301852	-1.222430	-0.248636	1.032845	-0.246955	-2.521639	-1.830333	-2.063137	-2.017632	4
4	-0.774043	-1.116775	0.335206	-1.303118	-1.222429	-0.248701	1.031807	-0.246610	-2.521900	-1.830498	-2.062795	-2.018145	4
5	-0.762936	-1.116955	0.334901	-1.303017	-1.222429	-0.248197	1.031031	-0.246341	-2.522203	-1.831691	-2.062549	-2.017884	4
6	-0.749228	-1.116170	0.335014	-1.302082	-1.222430	-0.247914	1.030493	-0.246162	-2.522538	-1.833012	-2.062115	-2.017243	4
7	-0.738450	-1.113986	0.336256	-1.305155	-1.222432	-0.248321	1.030107	-0.246035	-2.522844	-1.832182	-2.061953	-2.017213	4
8	-0.730910	-1.111828	0.334905	-1.303790	-1.222432	-0.247785	1.029851	-0.245981	-2.522808	-1.831576	-2.062443	-2.017739	4
9	-0.727130	-1.109486	0.335988	-1.305633	-1.222431	-0.248294	1.029636	-0.245886	-2.522677	-1.831438	-2.062317	-2.018180	4

Fig. 3. The first ten lines of the data set.

Fig. 4 shows checking the dataset for NaN (empty) values:

```
ambient           0
coolant           0
u_d               0
u_q               0
motor_speed       0
torque            0
i_d               0
i_q               0
pm                0
stator_yoke       0
stator_tooth      0
stator_winding    0
profile_id        0
dtype: int64
```

Fig. 4. The number of NaN values for each feature of the data set.

Fig. 5-6 show descriptions of the features of the data set.

	ambient	coolant	u_d	u_q	motor_speed	torque	i_d	i_q
count	998070.000000	998070.000000	998070.000000	998070.000000	998070.000000	998070.000000	998070.000000	998070.000000
mean	-0.003905	0.004723	0.004780	-0.005690	-0.006336	-0.003333	0.006043	-0.003194
std	0.993127	1.002423	0.997878	1.002330	1.001229	0.997907	0.998994	0.997912
min	-8.573954	-1.429349	-1.655373	-1.861463	-1.371529	-3.345953	-3.245874	-3.341639
25%	-0.599385	-1.037925	-0.826359	-0.927390	-0.951892	-0.266917	-0.756296	-0.257269
50%	0.266157	-0.177187	0.267542	-0.099618	-0.140246	-0.187246	0.213935	-0.190076
75%	0.686675	0.650709	0.358491	0.852625	0.853584	0.547171	1.013975	0.499260
max	2.967117	2.649032	2.274734	1.793498	2.024164	3.016971	1.060937	2.914185

Fig. 5. Description of the features of the data set (part 1).

pm	stator_yoke	stator_tooth	stator_winding	profile_id
998070.000000	998070.000000	998070.000000	998070.000000	998070.000000
-0.004396	0.000609	-0.002208	-0.003935	50.732001
0.995686	1.001049	0.999597	0.998343	22.073125
-2.631991	-1.834688	-2.066143	-2.019973	4.000000
-0.672308	-0.747265	-0.761951	-0.725622	32.000000
0.094367	-0.057226	0.005085	0.006536	56.000000
0.680691	0.697344	0.772239	0.725660	68.000000
2.917456	2.449158	2.326668	2.653781	81.000000

Fig. 6. Description of the features of the data set (part 2).

Features described above:

- Count is reported the number of non-empty rows in the column.
- Mean is reported as the average value of the data in the column.
- Std specifies the default value of the data deviation in the column.
- Min is reported as the minimum value of data in the column.
- 25%, 50%, and 75% are represent the percentile/quartile of each sign.

- Max is reported as the maximum value in the column.

Fig. 7 shows swing charts for all signs. Fig. 8 shows the correlation matrix.

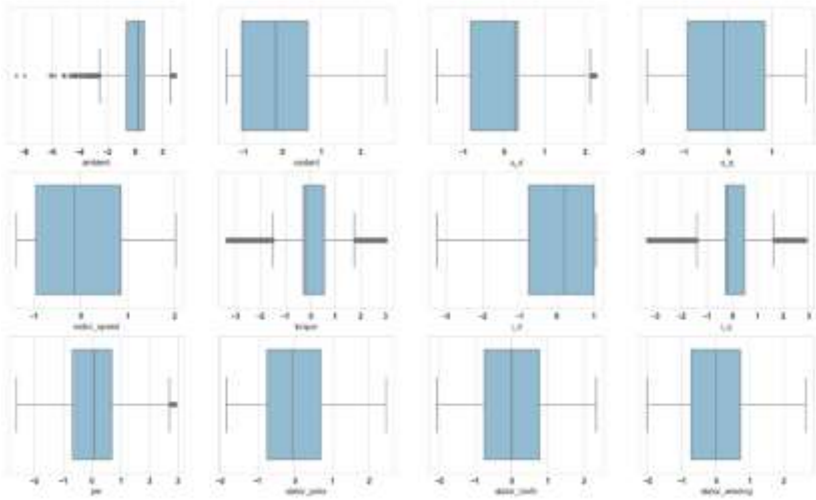

Fig. 7. Swing charts for signs.

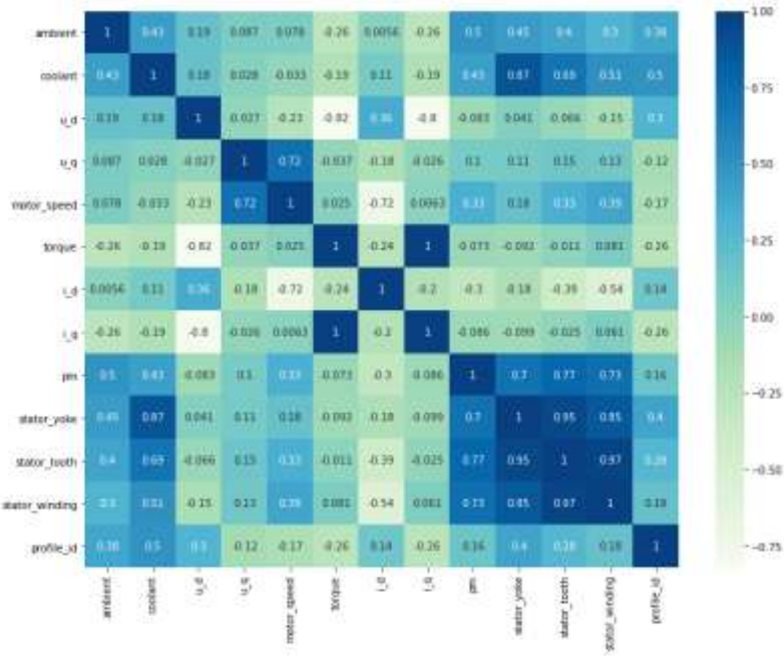

Fig. 8. Correlation matrix of features.

A list of all test runs(profile_id) is presented in Fig.9.

```
[ 4  6 10 11 20 27 29 30 31 32 36 41 42 43 44 45 46 47 48 49 50 51 52 53
 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 73 74 75 76 77 78
 79 80 81 72]
```

Fig. 9. List of test runs or profile_id values.

After studying the data set, the following conclusions can be drawn:

- The data set does not contain NaN values;
- Indicators for test cycles are not incremental.
- The description of the data set does not provide references to the units of measurement used for each of the samples, which complicates the interpretation of the measured values.
- A statistical review of the data set and histograms shows that the data set already has some normalization.
- As is already known, the ambient temperature is measured by a thermal sensor located close to the stator. Therefore, it can be assumed that this will affect the ability of the motor to cool itself. Higher ambient temperatures are likely to increase the temperature for both the motor stator and the rotor.
- The correlation matrix shows that there is a significant correlation between three different stator temperatures.

4.2 The Predicted Sign Determination

As mentioned above, there is a significant correlation between the temperature of the stator winding, the stator yoke, and the stator teeth. This is, of course, because the stator winding is wound around the stator tooth, which in turn is connected to the stator yoke. For better understand the relationship between the three traits, we plot the values of the traits for different randomly selected test cycles.

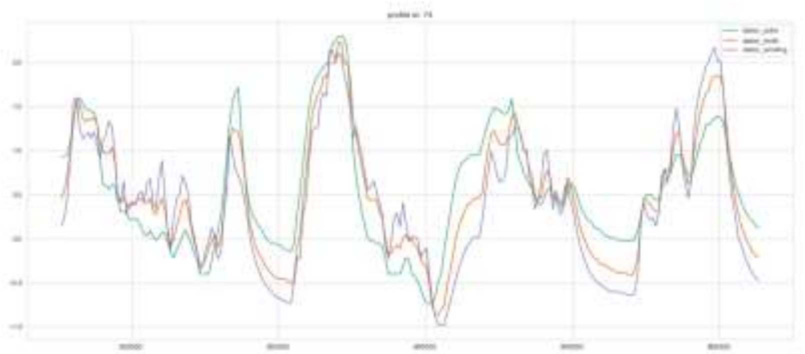

Fig. 10. Graphs of values of stator temperature signs for profile id: 74.

Fig. 11. Graphs of values of stator temperature signs for profile id: 70.

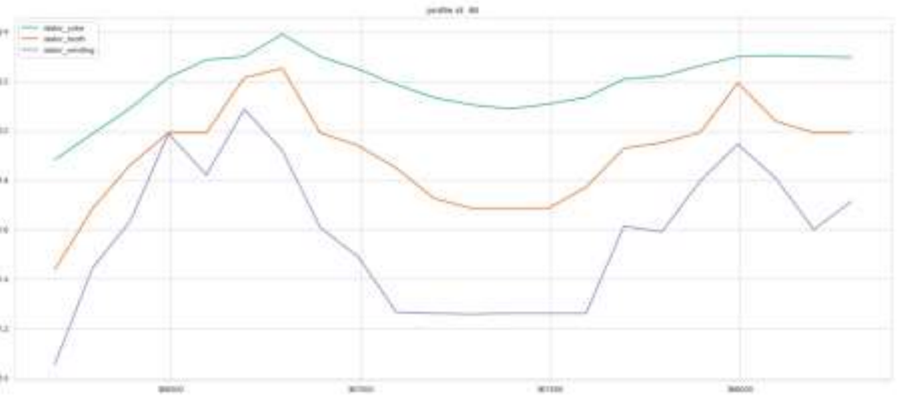

Fig. 12. Graphs of values of stator temperature signs for profile id: 46.

Fig. 13. Graphs of values of stator temperature signs for profile id: 72.

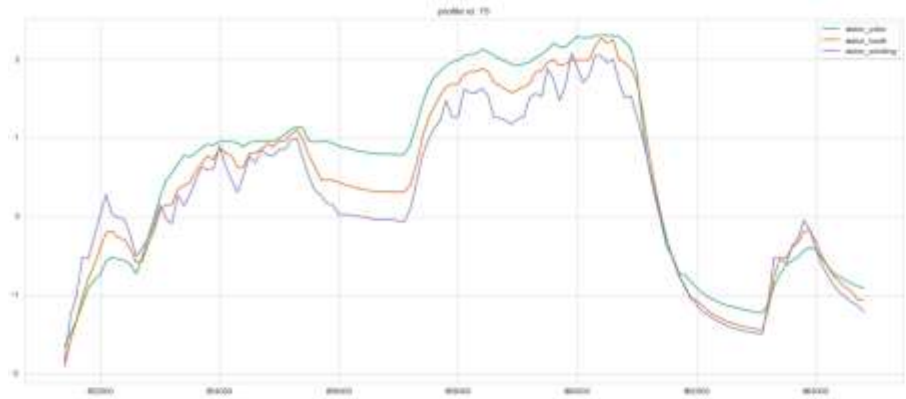

Fig. 14. Graphs of values of stator temperature signs for profile id: 75.

Fig. 15. Graphs of values of stator temperature signs for profile id: 56.

Fig. 16. Graphs of values of stator temperature signs for profile id: 11.

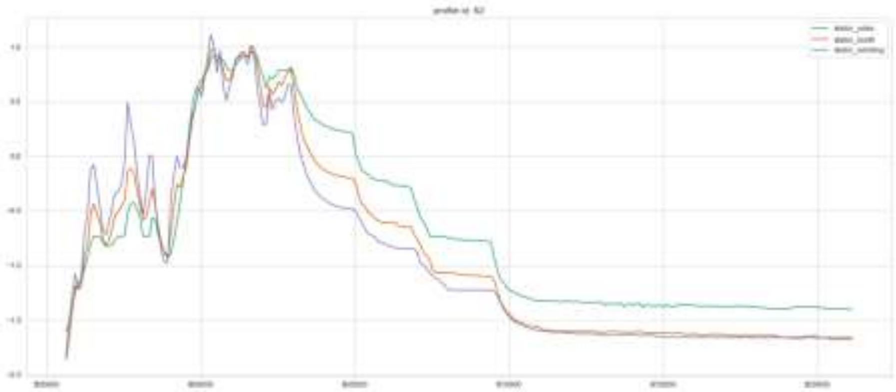

Fig. 17. Graphs of values of stator temperature signs for profile id: 62.

These line charts show that all three temperatures correspond to the same trend. The temperature of the stator winding shows the largest changes, followed by the temperature of the stator teeth and the stator yoke. This is especially noticeable in those diagrams where the stator winding temperature varies greatly. In this case, the temperature of the tooth and the stator yoke increases and decreases more smoothly compared to the temperature recorded on the stator winding. In other words, the heat dissipated by the stator windings requires some time to heat the teeth and the stator yoke due to the thermal inertia of both parts of the stator.

The second observation that can made from these line diagrams is that sometimes the stator yoke temperature is higher than the stator winding. Nor can we determine whether this is related to the normalization method used previously mentioned, or whether these values represent higher temperatures measured per stator yoke.

Line diagrams for comparing stator temperatures with torque and motor speed are presented in Fig.18-21.

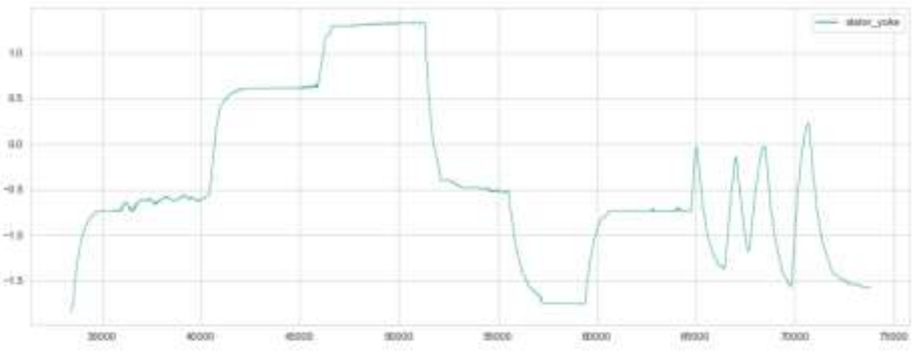

Fig. 18. Graph of stator yoke temperature change for profile_id: 6.

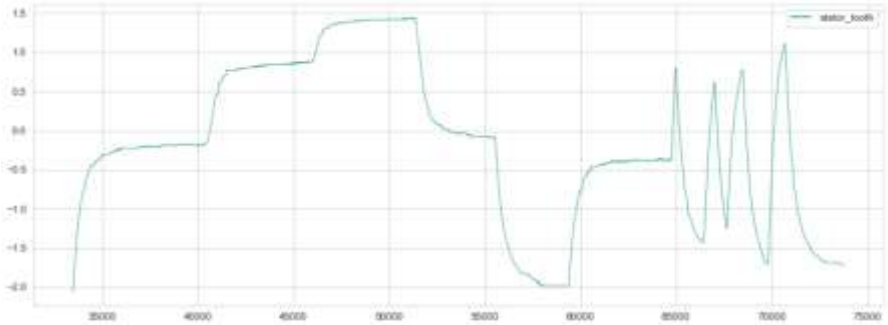

Fig. 19. Graph of stator teeth temperature change for profile_id: 6.

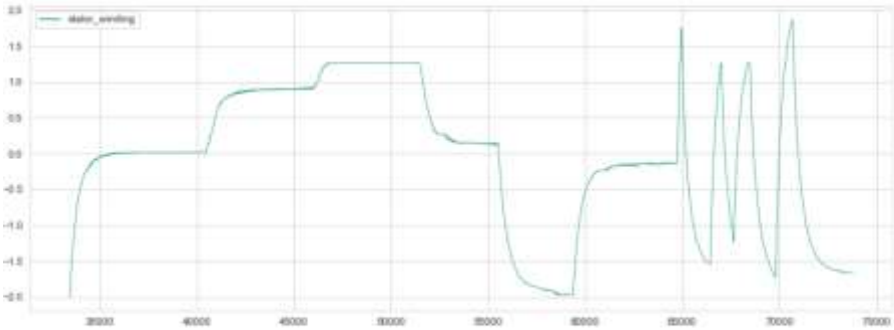

Fig. 20. Graph of stator winding temperature change for profile_id: 6.

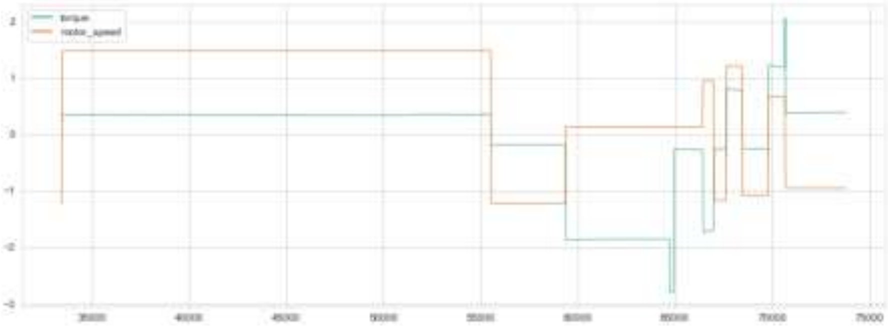

Fig. 21. Graph of changes in torque and engine speed for profile_id: 6.

In the second part of the above test cycle, you can see that there is a relationship between stator temperature, torque, and motor speed. With increasing torque and/or engine speed, the stator temperature increases, and vice versa decreases with decreasing. However, look at the first part of the test run; it is clear that this dependence is not always fulfilled. Even at constant torque and motor speed, the stator winding temperature shows several sudden temperature changes. One or more other variables affect the stator temperature more significantly than torque and motor speed.

4.3 Stator Winding Temperature Prediction

Since the measurement of torque, rotor, and stator temperature of an electric motor is not reliable and economically feasible in commercial programs, we will predict the stator temperature using other available functions in the data set. To do this, remove the torque, rotor and stator temperature from this data set, and use the stator winding temperature as the target value. After that, we train the Random Forest Regressor model to predict the correct stator winding temperature as the output value for the specified input variables. Input variables are:

- Ambient temperature;
- Coolant temperature;
- Voltage of the d-component (u_d);
- Q-component voltage (u_q);
- Engine speed (motor_speed);
- D-component current (i_d);
- The current of the q-component (i_q).

Target variable: stator winding temperature (stator_winding). Rejectable variables are:

- Engine torque;
- Rotor temperature (pm);
- Stator yoke temperature (stator_yoke);
- Stator teeth temperature (stator_tooth).

At the end of the model learning process, the values of the Mean Absolute Error (MAE) and the Mean Square Error (MSE) are calculated.

The MAE is a loss function used for regression. Loss is the average value for the absolute differences between true and predicted values, deviations in any direction from the true values are treated the same. The MAE for this model is: $MAE = 0.05787082149138227$.

The MSE, like the MAE too, treats the deviation in any direction the same. The difference between MSE and MAE is that MSE is more likely to notice large errors because it squares all errors. The closer the values of MAE and MSE to zero, the closer the predicted values of the algorithm to the true ones. Comparing diagrams of the predicted and true values of several test runs are shown in Fig.22-25.

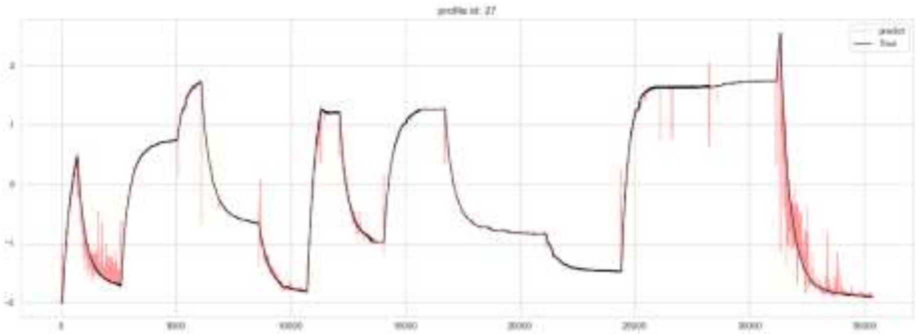

Fig. 22. Comparing diagram of predicted and true values for profile_id: 27.

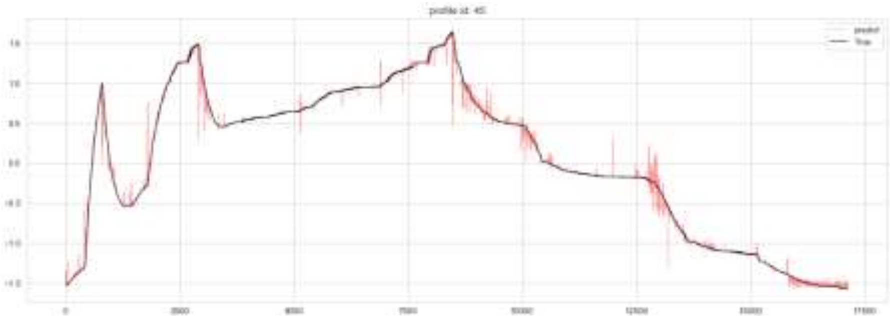

Fig. 23. Comparing diagram of predicted and true values for profile_id: 45.

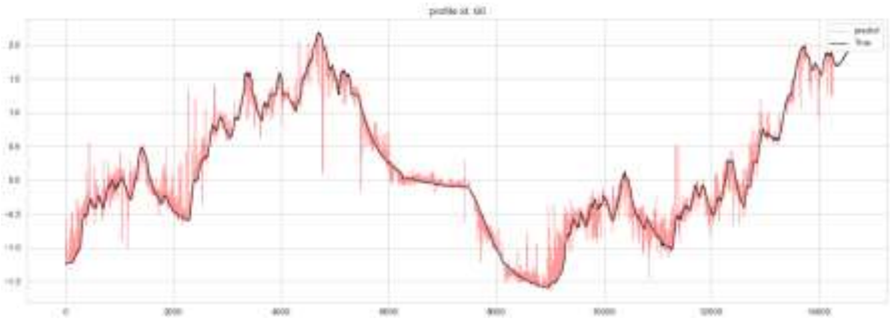

Fig. 24. Comparing diagram of predicted and true values for profile_id: 60.

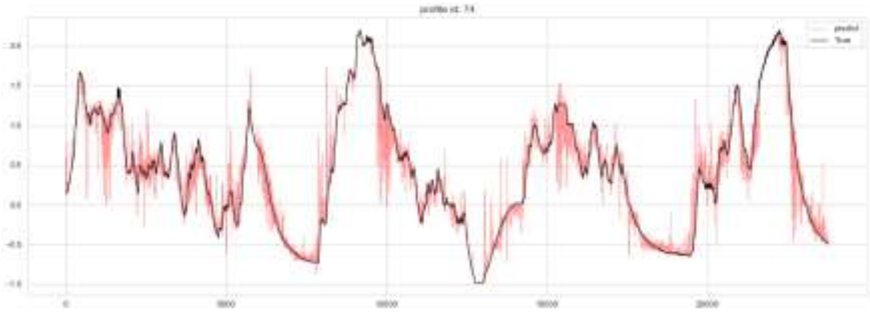

Fig. 25. Comparing diagram of predicted and true values for profile_id: 74.

In these comparing diagrams, it can seen that the model is able to capture the global trend in the stator winding values, but there is still a lot of noise in the predicted values, especially when the true signal shows a large number of changes. To filter the noise generated by the model, apply the smoothing method to the output signal. MSE and MAE values and linear diagram of the output signal for test run id_72 without smoothing: $MAE = 0.04831882742340883; MSE = 0.015408519961455704$.

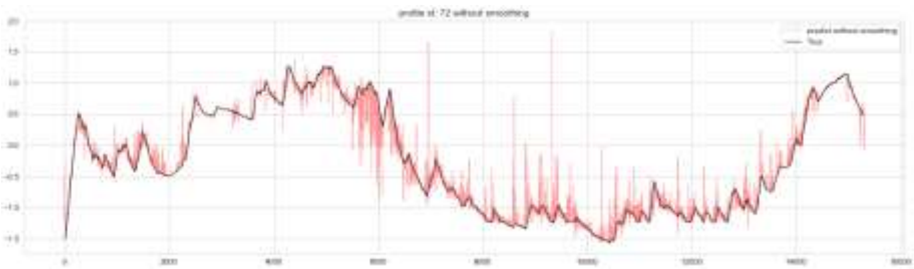

Fig. 26. Comparing diagram of predicted and true values for profile_id: 72, without smoothing.

MSE and MAE values and linear diagram of the output signal for test run id_72 with smoothing: $MAE = 0.07833458099037062; MSE = 0.011039865767098802$.

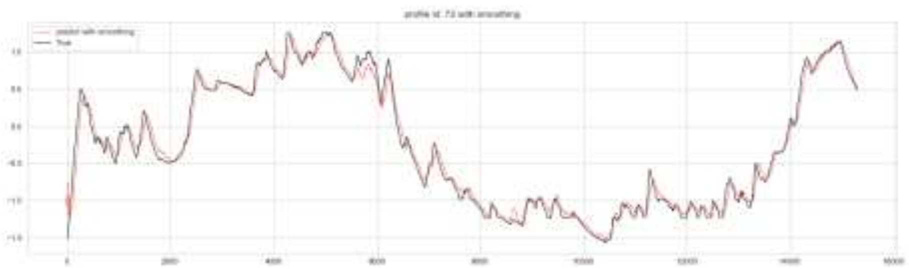

Fig. 27. Comparing diagram predicted and true values for profile_id: 72, with smoothing.

In combination with smoothing, the Random Forest Regressor performs a fairly accurate prediction of the stator winding temperature. Smoothing reduced the noise level in the predicted values and had a positive effect on MSE for this particular technology.

4.4 The accuracy of Random Forest Compared to Other Algorithms and the Learning Time of the Algorithm on Different PCs.

Let's compare Random Forest with such algorithms as k-NN, SVM, and linear regression.

The figures below show the following characteristics: model training time in seconds, model accuracy for test set, MSE and MAE for test set.

Fig. 28. Characteristics of the Random Forest model.

Fig. 29. Characteristics of the k-NN model.

Fig. 30. Characteristics of the Linear Regression model.

Fig. 31. SVM model characteristics.

The following are line graphs of several randomly selected test cycles depicting the models described above colours: black - original data, red - data predicted by Random Forest method, blue - data predicted by k-NN method, green - data predicted by SVM method, purple - data predicted by Linear Regression method.

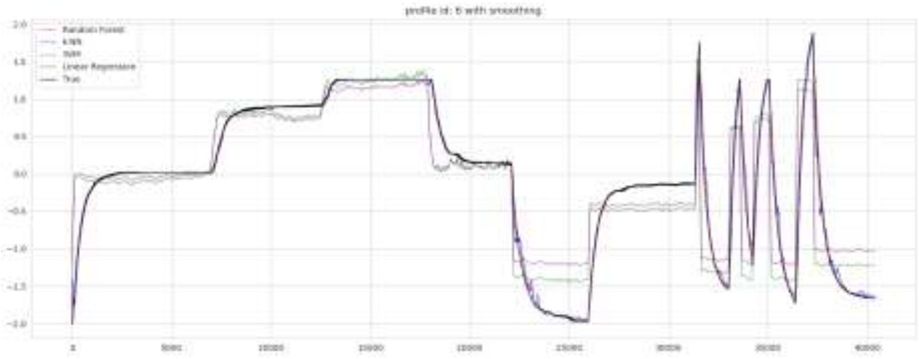

Fig. 32. Comparing diagram of predicted and true values: profile_id: 6.

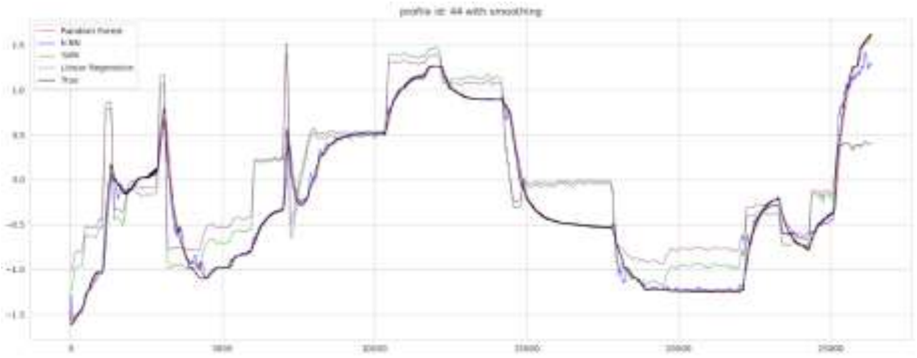

Fig. 33. Comparing diagram of predicted and true values: profile_id: 44.

Fig. 34. Comparing diagram of predicted and true values. profile_id: 69.

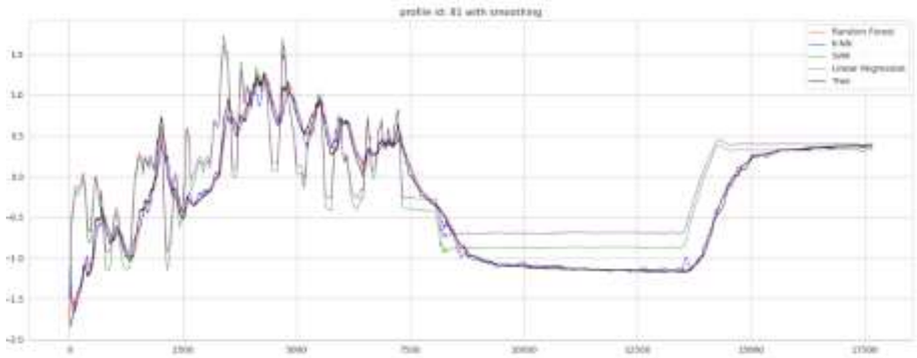

Fig. 35. Comparing diagram of predicted and true values: profile_id: 81.

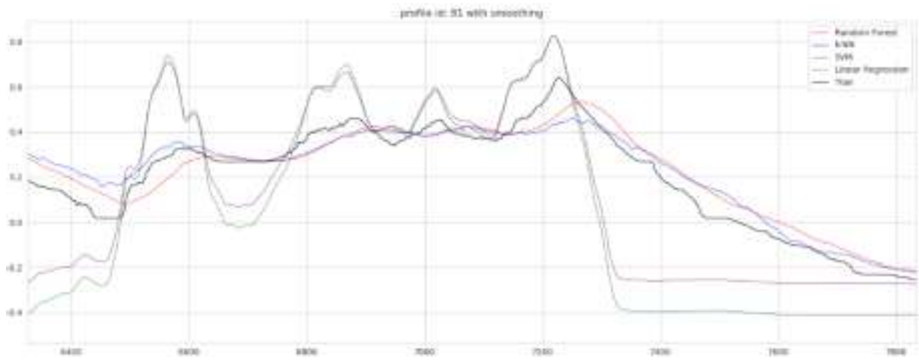

Fig. 36. Part of the comparing diagram of predicted and true values: profile_id: 81.

Also, compare the work Random Forest method on different PCs. Below is a table of components of these PCs (Table 1) and a picture of the characteristics of the robots of the Random Forest method on these PCs.

Table 1. Table of PC components.

	PC 1	PC 2	PC3
CPU name	AMD FX-8350	Intel Core i3-6006U	AMD A8-7050
CPU cores	8	2	4
CPU threads	8	4	4
CPU rate	4.3 Ghz	2 Ghz	2.2 Ghz
Memory	8 GB	8 GB	8 GB
Drive	Samsung SSD 860 EVO	Hynix SC308	Toshiba MQ01ABD100
Drive speed	555 MB/sec	500 MB/sec	225 MB/sec

```
Wait until model fit...
Model fitted. Wall time: 124.76 seconds

Prediction accuracy for the entire test set: 97.99 %

metrics for entire test set :
MSE: 0.01998096705249478
MAE: 0.05742878779411757
```

Fig. 37. Characteristics of the Random Forest model for PC1.

```
Wait until model fit...
Model fitted. Wall time: 123.79 seconds

Prediction accuracy for the entire test set: 98.03 %

metrics for entire test set :
MSE: 0.019631325771931423
MAE: 0.05701921334484496
Wait until model save...
Model saved in Data/RFR_model.pkl.
```

Fig. 38. Characteristics of the Random Forest model for PC2.

```
Wait until model fit...
Wall time: 220.77 seconds

Prediction accuracy for the entire test set: 98.02 %

metrics for entire test set :
MSE: 0.019586497076649903
MAE: 0.05682736683181596
```

Fig. 39. Characteristics of the Random Forest model for PC3.

Thus, comparing the characteristics of different models and line charts for different test runs, we can divide the above models into 2 groups: Random Forest and k-NN that show forecast accuracy above 90%, and Linear Regression and SVM with forecast accuracy less than 65%. K-NN is 113 seconds faster than Random Forest, but 3.1% less accurate. In addition, having tested the Random Forest model on different PCs, we can conclude that for faster learning and forecasting of the model it is necessary to use faster data carriers.

5 Received Results and Discussion

Since the Random Forest Regressor with smoothing performs more than sufficient prediction at the stator temperature (as shown by the various line diagrams shown below), to use a relatively modest model like this is much better than more bulky and requiring more memory alternatives such as k-NN, Linear Regression, and various neural networks.

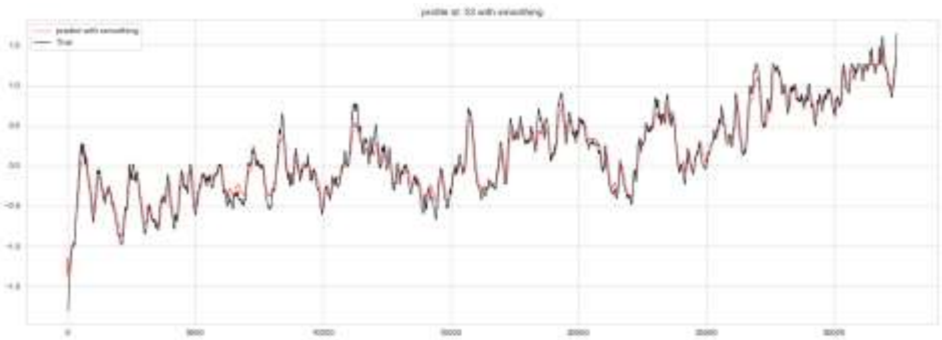

Fig. 40. Comparing diagram of predicted and true values for profile_id: 53.

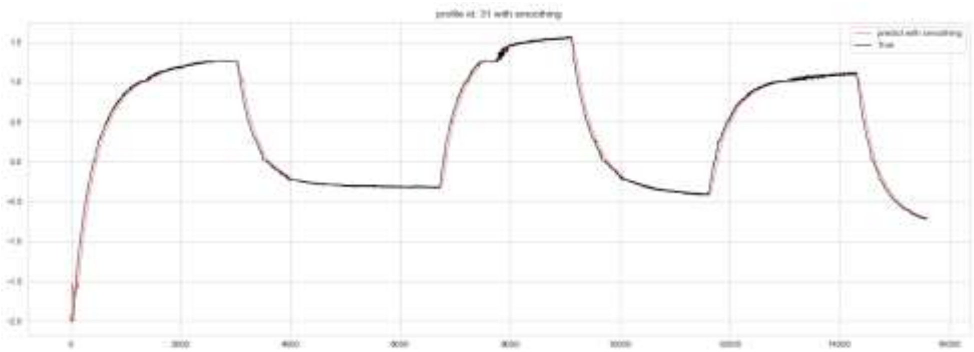

Fig. 41 Comparing diagram of predicted and true values for profile_id: 31.

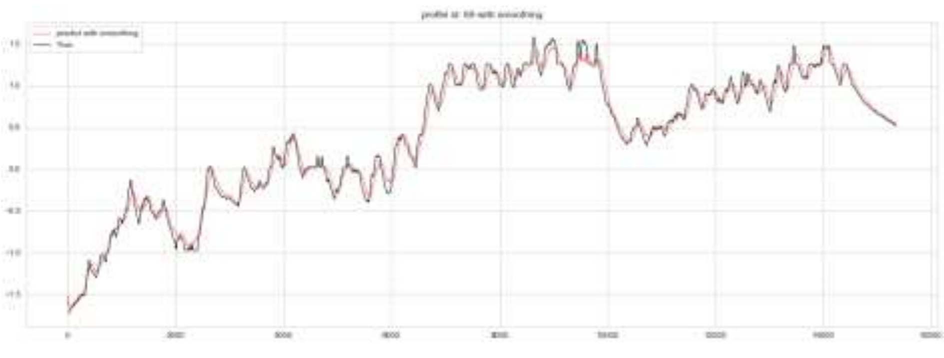

Fig. 42. Comparing diagram of predicted and true values for profile_id: 69.

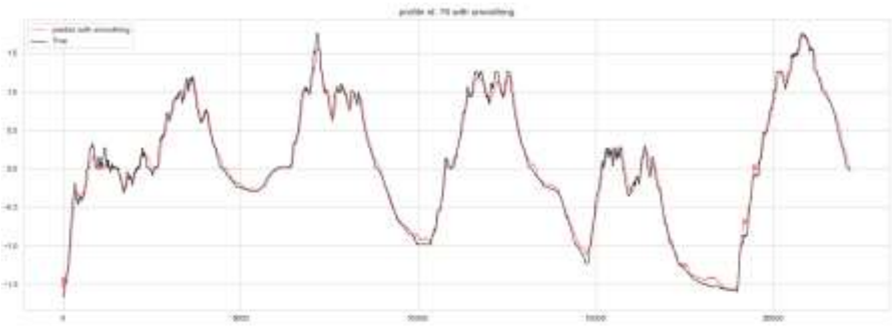

Fig. 43. Comparing diagram of predicted and true values for profile_id: 76.

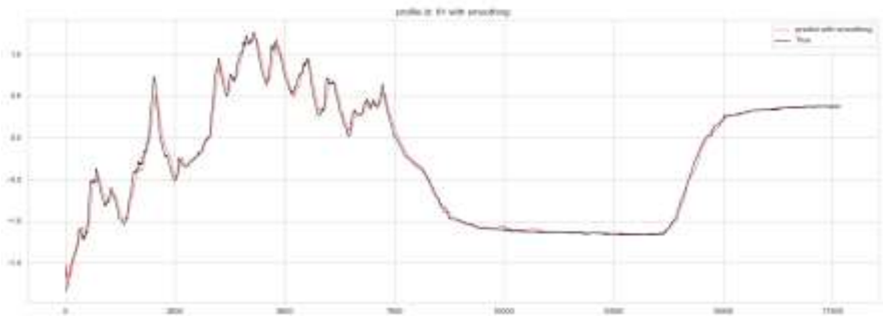

Fig. 44. Comparing diagram of predicted and true values for profile_id: 81.

6 Conclusions

The purpose of this work was to predict the temperature of a synchronous motor with a permanent magnet using machine-learning methods. First of all, in this work, a study of the subject area and a review of publications with research established in the laboratory of the University of Paderborn SDPM. With the help of the unified UML language, diagrams are designed that show how the implemented model can act as a part of the control system of the mechanism that works on SDPM. To implement this project, the method of machine learning Random Forest Regression is chosen. The implementation tool is the Python programming language with add-ins such as NumPy, Pandas, Scikit-learn, Matplotlib, and Seaborn. A comparison of the machine learning method and the means of implementation with analogues is also performed. Because of this project, a study of data from the SDPM sensory data set is conducted, the predicted feature, which was the temperature of the stator winding, is determined, and this feature was predicted. After analysing the obtained forecasting results, we can conclude that the purpose and main objectives of this work are achieved and the trained model can be used to accurately estimate and predict the stator and rotor temperatures of a synchronous motor on permanent magnets.

References

1. Lindegger, Markus: Economic viability, applications and limits of efficient permanent magnet motors.- Switzerland: Swiss Federal Office of Energy (2009).
2. Specht, A., Wallscheid, O., Böcker, J.: Determination of rotor temperature for an interior permanent magnet synchronous machine using a precise flux observer. In: Int. Power Electronics Conference (IPEC-Hiroshima - ECCE ASIA), Hiroshima, 1501-1507. (2014).
3. Wallscheid, O., Huber, T., Peters, W., Böcker, J.: Real-time capable methods to determine the magnet temperature of permanent magnet synchronous motors. In: IECON 2014 - 40th Annual Conference of the Industrial Electronics Society, Dallas, TX, 811-818. (2014).
4. Wallscheid, O., Böcker, J.: Fusion of direct and indirect temperature estimation techniques for permanent magnet synchronous motors. In: IEEE International Electric Machines and Drives Conference (IEMDC), Miami, FL, 2017, 1-8. (2017).
5. Böcker, J., Kirchgässner, W., Wallscheid, O.: Deep Residual Convolutional and Recurrent Neural Networks for Temperature Estimation in Permanent Magnet Synchronous Motors. (2019).
6. Electric Motor Temperature, https://www.kaggle.com/wkirgsn/electric-motor-temperature
7. Larose, D.: Discovering Knowledge in Data: An Introduction to Data Mining. (2004).
8. Beklemysheva, A.: Why Use Python for AI and Machine Learning? https://steelkiwi.com/blog/python-for-ai-and-machine-learning/.
9. Regression loss metrics, https://peltarion.com/knowledge-center/documentation/evaluation-view/regression-loss-metrics.
10. Peleshchak, R.M., Bachynsky, I.Ya.: Electric properties of the interface quantum dot - Matrix. In: Condensed Matter Physics 12(2), 215-223. (2009)
11. Peleshchak, R.M., Kuzyk, O.V.: Self-Consistent deformation-diffusion spatial redistribution of defects in a semiconductor induced by the electric field. In: Ukrainian Journal of Physics 54(7), 702-706. (2009)
12. Peleshchak, R.M., Dan'Kiv, O.O., Kuzyk, O.V.: Modulation of the direction of radiation emitted by an InAs/GaAs heterolaser with InAs quantum dots under the influence of acoustic wave. In: Ukrainian Journal of Physics 57(1), 68-72. (2012)
13. Uhryn, Y.O., Peleshchak, R.M., Brytan, V.B., Velchenko, A.A.: Magnetoresistance based determination of basic parameters of minority charge carriers in solid matter . In: Condensed Matter Physics 20(4), 43702. (2017)
14. Bublyk, M., Rybytska, O., Karpiak, A., Matseliukh, Y.: Structuring the fuzzy knowledge base of the IT industry impact factors. In: Computer sciences and information technologies : Proceed. of the 2018 IEEE 13th International Scientific and Technical Conference (CSIT 2018), (Lviv, Ukraine, September, 11–14, 2018). DOI: https://doi.org/10.1109/STC-CSIT.2018.8526760.
15. Matseliukh, Y., Vysotska, V., Bublyk, M.: Intelligent system of visual simulation of passenger flows. In: CEUR Workshop Proceedings, 2604, 906. (2020) http://ceur-ws.org/Vol-2604/paper60.pdf
16. Bublyk, M., Matseliukh, Y., Motorniuk, U., Terebukh, M.: Intelligent system of passenger transportation by autopiloted electric buses in Smart City. In: CEUR Workshop Proceedings, 2604, 1280. (2020) http://ceur-ws.org/Vol-2604/paper81.pdf
17. Maslak O., Ishchenko S., Grishko N., Maslak, M.: Export-oriented enterprise development of the electrotechnical industry: strategic guidelines. In: 2019 IEEE International Conference on Modern Electrotechnical and Energy Systems (MEES). IEEE, 486-489. (2019)
18. Krislata, I., Katrenko, A., Lytvyn, V., Vysotska, V., Burov, Y.: Traffic flows system development for smart city. In: CEUR Workshop Proceedings, Vol-2565, 280–294. (2020)

19. Peleshko D., Rak T., Lytvyn V., Vysotska V., Noennig J.: Drone monitoring system DROMOS of urban environmental dynamics. In: CEUR Workshop Proceedings, Vol-2565, 178–19. (2020)
20. Emmerich, M., Lytvyn, V., Yevseyeva, I., Fernandes, V. B., Dosyn, D., Vysotska, V.: Preface: Modern Machine Learning Technologies and Data Science (MoMLeT&DS-2019). In: CEUR Workshop Proceedings, Vol-2386. (2019)
21. Lytvyn, V., Vysotska, V., Osypov, M., Slyusarchuk, O., Slyusarchuk, Y.: Development of intellectual system for data de-duplication and distribution in cloud storage. In: Webology, 16(2), pp. 1-42. (2019)
22. Lytvyn, V., Vysotska, V., Shatskykh, V., Kohut, I., Petruchenko, O., Dzyubyk, L., Bobrivetc, V., Panasyuk, V., Sachenko, S., Komar, M.: Design of a recommendation system based on Collaborative Filtering and machine learning considering personal needs of the user. In: Eastern-European Journal of Enterprise Technologies, 4(2-100), 6-28. (2019)
23. Lytvyn, V., Gozhyj, A., Kalinina, I., Vysotska V., Shatskykh, V., Chyrun, L., Borzov, Y.: An intelligent system of the content relevance at the example of films according to user needs. In: CEUR Workshop Proceedings, Vol-2516, 1-23. (2019)
24. Lytvyn, V., Peleshchak, I., Peleshchak, R., Vysotska, V.: Information Encryption Based on the Synthesis of a Neural Network and AES Algorithm. In: 3rd International Conference on Advanced Information and Communications Technologies, AICT, 447-450. (2019)
25. Lytvyn, V., Kowalska-Styczen, A., Peleshko, D., Rak, T., Voloshyn, V., Noennig, J. R., Vysotska, V., Nykolyshyn, L., Pryshchepa, H.: Aviation Aircraft Planning System Project Development. In: Advances in Intelligent Systems and Computing IV, Springer, Cham, 1080, 315-348. (2020)
26. Kravets, P., Burov, Y., Lytvyn, V., Vysotska, V.: Gaming method of ontology clusterization. In: Webology, 16(1), 55-76. (2019)
27. Shu, C., Dosyn, D., Lytvyn, V., Vysotska V., Sachenko, A., Jun, S.: Building of the Predicate Recognition System for the NLP Ontology Learning Module. In: International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS, 2, 802-808. (2019)
28. Kravets, P., Lytvyn, V., Vysotska, V., Burov, Y.: Promoting training of multi-agent systems. In: CEUR Workshop Proceedings, Vol-2608, 364-378. (2020)
29. Batiuk, T., Vysotska, V., Lytvyn, V.: Intelligent System for Socialization by Personal Interests on the Basis of SEO-Technologies and Methods of Machine Learning. In: CEUR workshop proceedings, Vol-2604, 1237-1250. (2020).
30. Puleko, I., Kravchenko, S., Chumakevych, V., Ptashnyk, V.: Method of Machine Learning on the Basis of Discrete Orthogonal Polynomials of Chebyshev. In: Computational Linguistics and Intelligent Systems, COLINS, CEUR workshop proceedings, Vol-2604, 67-76. (2020).
31. Zdebskyi, P., Lytvyn, V., Burov, Y., Rybchak, Z., Kravets, P., Lozynska, O., Holoshchuk, R., Kubinska, S., Dmytriv, A.: Intelligent System for Semantically Similar Sentences Identification and Generation Based on Machine Learning Methods. In: Computational Linguistics and Intelligent Systems, COLINS, CEUR workshop proceedings, Vol-2604, 317-346. (2020).
32. Makara, S., Chyrun, L., Burov, Y., Rybchak, Z., Peleshchak, I., Peleshchak, R., Holoshchuk, R., Kubinska, S., Dmytriv, A.: An Intelligent System for Generating End-User Symptom Recommendations Based on Machine Learning Technology. In: Computational Linguistics and Intelligent Systems, COLINS, CEUR workshop proceedings, Vol-2604, 844-883. (2020).

33. Andrunyk, V., Berko, A., Rusyn, B., Pohreliuk, L., Chyrun, S., Dokhniak, B., Karpov, I., Krylyshyn, M.: Information System of Photostock Web Galleries Based on Machine Learning Technology. In: Computational Linguistics and Intelligent Systems, COLINS, CEUR workshop proceedings, Vol-2604, 1032-1059. (2020).
34. Veres, O., Rishnyak, I., Rishniak, H.: Application of Methods of Machine Learning for the Recognition of Mathematical Expressions. In: CEUR Workshop Proceedings, Vol-2362, 378-389. (2019)
35. Kosarevych, R.J.,Rusyn, B.P.,Korniy, V.V.,Kerod, T.I.: Image Segmentation Based on the Evaluation of the Tendency of Image Elements to form Clusters with the Help of Point Field Characteristics. In: Cybernetics and Systems Analysis,51(5), 704-713. (2015)
36. Varetskyy, Y.,Rusyn, B.,Molga, A.,Ignatovych, A.: A New Method of Fingerprint Key Protection of Grid Credential. In: Advances in Intelligent and Soft Computing, 84, 99-103. (2010)
37. Zhengbing, H., Yatskiv, V., Sachenko, A.: Increasing the Data Transmission Robustness in WSN Using the Modified Error Correction Codes on Residue Number System. In: Elektronika ir Elektrotechnika, 21(1), 76-81. (2015)
38. Sachenko, A., Kochan, V., Kochan, R., Turchenko, V., Tsahouridis, K., Laopoulos,T.: Error compensation in an intelligent sensing instrumentation system. In: Conference Record - IEEE Instrumentation and Measurement Technology Conference, 869-874. (2001)
39. Sachenko, S., Pushkar, M., Rippa, S.: Intellectualization of Accounting System. In: International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applicationsm, 536 – 538. (2007)
40. Sachenko, S., Rippa, S., Krupka, Y.: Pre-Conditions of Ontological Approaches Application for Knowledge Management in Accounting. In: International Workshop on Antelligent Data Acquisition and Advanced Computing Systems: Technology and Applications,. 605-608. (2009)
41. Oksanych, I., Shevchenko, I., Shcherbak, I., Shcherbak, S.: Development of specialized services for predicting the business activity indicators based on micro-service architecture. In: Eastern-European Journal of Enterprise Technologies, 2(2-86), 50-55. (2017)
42. Galushka, I., Shcherbak, S.: Devising a mathematical model for pattern-based enterprise data integration. In: Eastern-European Journal of Enterprise Technologies, 59-64. (2015)
43. Shcherbak, S.S.: Interoperability web application models based on microformats. In: CriMiCo, 21st International Crimean Conference: Microwave and Telecommunication Technology, 57-58. (2011)
44. Babichev, S., Lytvynenko, V., Osypenko, V.: Implementation of the objective clustering inductive technology based on DBSCAN clustering algorithm. In: Proceedings of the 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 1, 479-484. (2017)
45. Babichev, S.A., Gozhyj, A., Kornelyuk, A.I., Lytvynenko, V.I. Objective clustering inductive technology of gene expression profiles based on SOTA clustering algorithm. In: Biopolymers and Cell, 33 (5), 379-392. (2017)
46. Babichev, S., Taif, M.A., Lytvynenko, V., Osypenko, V.: Criterial analysis of gene expression sequences to create the objective clustering inductive technology. In: International Conference on Electronics and Nanotechnology, ELNANO, 244-248. (2017)
47. Kravets, P.: Adaptive method of pursuit game problem solution. In: Modern Problems of Radio Engineering, Telecommunications and Computer Science Proceedings of International Conference, TCSET, 62-65. (2006)
48. Kravets, P.: Game methods of construction of adaptive grid areas. In: The Experience of Designing and Application of CAD Systems in Microelectronics, 513-516. (2003)