An Ideation Report

On

Electric Motor Temperature detection

Using Machine Learning

Submitted to

KIIT Deemed to be University

By-

- Satyam Kumar – 1707141

Under the guidance of – Proff. Ansuman Patnayak

SCHOOL OF ELECTRONICS ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

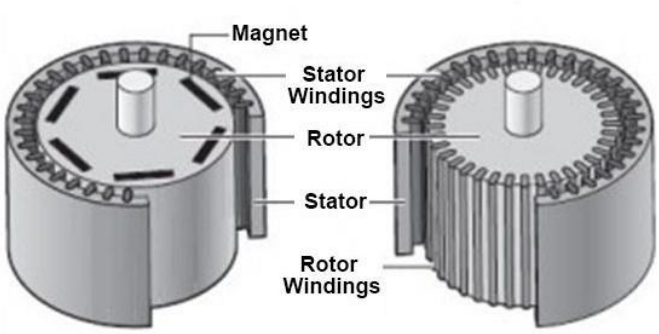
BHUBANESWAR, ODISHA – 751024 ,APRIL 2020

Abstract

The main purpose of the data set's recording is to be able to model the rotor temperatures of a PMSM (Permanent Magnet Synchronous Motor) 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 can't be administered properly without redundant modelling. In addition, precise thermal modelling gets more and more important with the rising relevance of functional safety.

We will be designing a model with appropriate feature engineering, that estimates the rotor temperature in a causal manner, In order to maintain real-time capability. The temperature estimation in production will be deployed on best-cost hardware of traction drives in an automotive environment, where lean computation and lightweight implementation is key.

So, what is a PMSM?



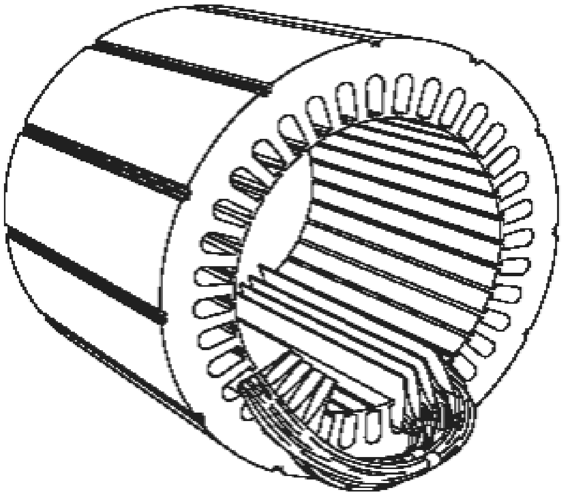
A Synchronous motor is a machine that converts AC electrical Power to Mechanical Power at a synchronous speed. A permanent-magnet synchronous motor (PMSM) uses permanent magnets. embedded in the steel rotor to create a constant magnetic field. The stator carries windings connected to an AC supply to produce a rotating magnetic field. At synchronous speed the rotor poles lock to the rotating magnetic field. Permanent magnet synchronous motors are similar to brushless DC motors. Neodymium Magnets are the most commonly used magnets in these motors.

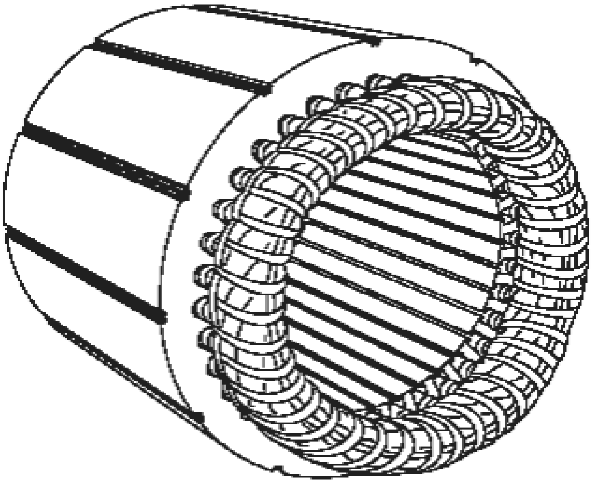
Because of the constant magnetic field in the rotor these cannot use induction windings for starting. These motors require a variable-frequency power source to start.

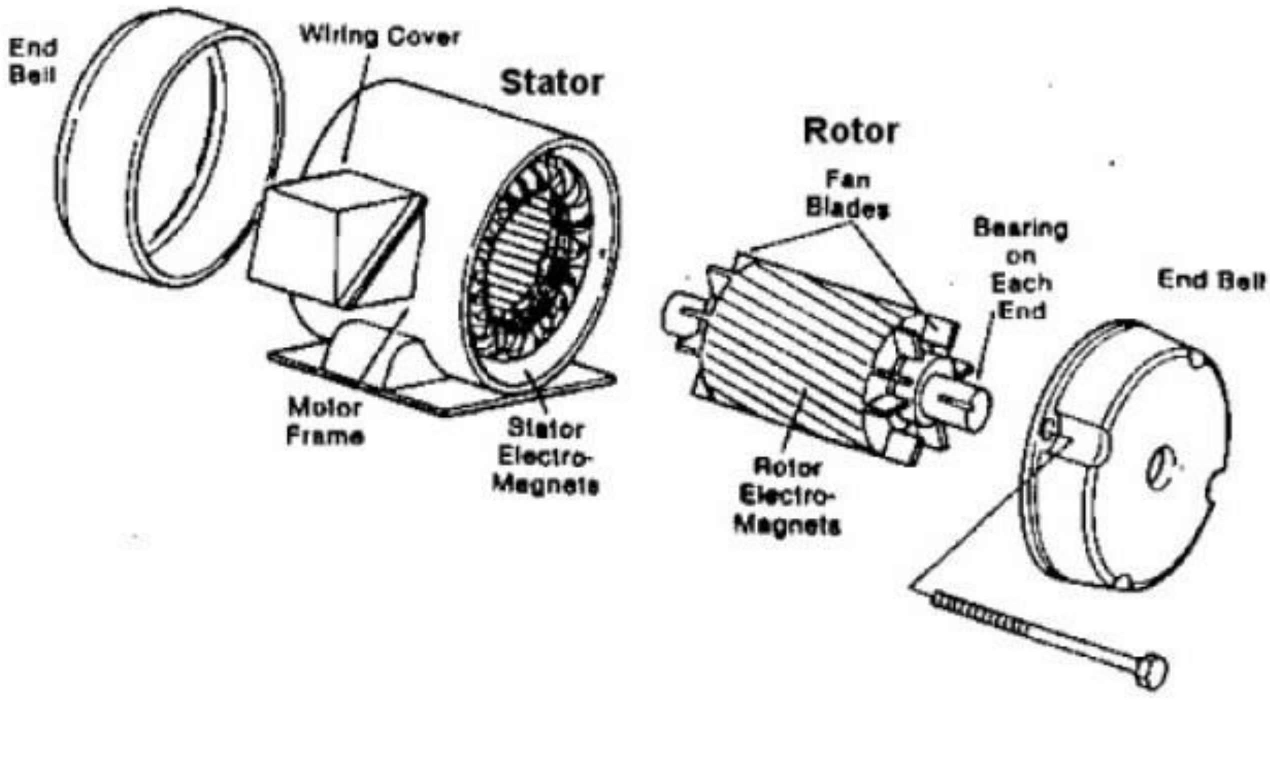
The main difference between a permanent magnet synchronous motor and an Asynchronous motor is the rotor.

Permanent magnet motors have been used as gearless elevator motors since 2000.

A Better View :







PMSM Rotor Temperature Prediction

OBJECTIVE- The rotor temperature of any motor is difficult to measure as it is a rotating part. Placing any sensors to measure this difficult to measure temperature would result in increase in costs and also increase the weight of the motor. Measurement of quantities like temperature, torque of the rotor is important in order to design control systems to effectively control the motor.

1. Problem Statement

Predicting rotor temperature of the rotor of a Permanent Magnet Synchronous Motor (PMSM) given other sensor measurements during operation. by creating various Regressor models and selecting the best one depending on the Accuracy and Mean Square Error of the model.

2. Data Description

A brief description of data attributes:

- Ambient - **Ambient temperature**
- Coolant - **Coolant temperature**
- u_d - **Voltage d-component (Active component)**
- u_q - **Voltage q-component (Reactive component)**
- Motor speed - **Speed of the motor**
- Torque - **Torque induced by current**
- i_d - **Current d-component (Active component)**
- i_q - **Current q-component (Reactive component)**
- pm - **Permanent Magnet Surface temperature**
- Stator_yoke - **Stator yoke temperature**
- Stator_tooth - **Stator tooth temperature**
- Stator_winding - **Stator Stator_winding temperature**
- Profile_id - **Measurement Session id**

Target features are pm as the dependent variable, and other values as the independent variable.

3. Data Preparation

The Dataset in which we'll be working our model on,

```
In [1]: import pandas as pd
   ...: import numpy as np
   ...: import matplotlib.pyplot as plt
   ...: import seaborn as sns
   ...: from sklearn.metrics import accuracy_score,
confusion_matrix, r2_score
   ...: from sklearn.metrics import mean_absolute_error
   ...:
   ...: #import dataset
   ...: dataset=pd.read_csv('pmsm_temperature_data.csv')

In [2]:
```

Name	Type	Size	Value
dataset	DataFrame	(998070, 13)	Column names: ambient, coolant, u_d, u_q, motor_speed, torque, i_d, i ...

dataset - DataFrame

Index	ambient	coolant	u_d	u_q	motor_speed	torque	i_d	i_q	stator_yoke	:ator_toot	tor_windi	pm	profile_id
0	-0.75	-1.12	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.07	-2.02	-2.52	4
1	-0.77	-1.12	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
2	-0.78	-1.12	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
3	-0.78	-1.12	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
4	-0.77	-1.12	0.34	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
5	-0.76	-1.12	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
6	-0.75	-1.12	0.34	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
7	-0.74	-1.11	0.34	-1.31	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
8	-0.73	-1.11	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
9	-0.73	-1.11	0.34	-1.31	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
10	-0.72	-1.11	0.34	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
11	-0.72	-1.11	0.33	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
12	-0.70	-1.11	0.34	-1.31	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
13	-0.68	-1.11	0.34	-1.30	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4
14	-0.65	-1.11	0.34	-1.31	-1.22	-0.25	1.03	-0.25	-1.83	-2.06	-2.02	-2.52	4

Format | Resize | Background color | Column min/max | Save and Close | Close

A comprehensive csv files containing all measurement sessions and features. As one can see there are total of 998070 rows which represents the readings recorded for 140hrs. Sample rate is 2 Hz (One row per 0.5 seconds). Distinctive sessions are identified with "profile_id".

The dataset provided to us has already been feature scaled, ie. Converted into a simpler format in order to make the regression model more accurate. In case one wants to find the original data, the following process may come handful,

scaled_array = (original_array - mean_of_array)/std_of_array

In Sklearn, each array column appears to be scaled in this way. To find the original data, we can simply rearrange the above, or alternatively just calculate the standard deviation and mean of each column in the unscaled data. We can then use this to transform the scaled data back to the original data at any time.

4. Modelling and Evaluation

Algorithm used – **The sequence of sensor values are given as an input that is the independent variables and Temperature of the permanent magnet ie. The rotor is to be evaluated which is the dependent variable. Thus, the regression models will be based on**

- **Multiple linear Regression**
- **Random Forest Regression**
- **Support Vector Regression**

Then the data will be splitted into train and test variables with the help of train_test_split from sklearn. model_selection

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2, random_state = 0)
```

After Creating our Regressor model we will apply k-Fold Cross Validation method which will divide the dataset into n no. of folds, for better accuracy.

```
from sklearn.model_selection import cross_val_score
accuracies = cross_val_score(estimator = regressor, X = X_train, y = Y_train, cv = 10)
accuracies.mean()
accuracies.std()
```

Evaluation – From sklearn. metrics library we'll be using the following to test our regressor model

- **mean_absolute_error**
- **accuracy_score**
- **r2_Score (Coefficient of Determination) have been used.**

```
In [2]: from sklearn.metrics import accuracy_score, r2_score
   ...: from sklearn.metrics import mean_absolute_error

In [3]:
```

Why are we doing this ?

A Simple answer to this question is because of the functional safety of the Synchronous Motor . As the intricate structure of an electric traction drive, direct measurement with thermal sensors is not possible for rotor temperatures, thus instead of sensing the rotor temperature through different sensor which might prove to be costlier and more complex because of the rotating nature of the rotor we can tackle that problem by building a model which will successfully predict the temperature of the rotor by taking into account the different features of the Synchronous motor.