ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

Terrain Specific OBD Data Anomalies Detection and Transmission using Carbon-Footprint Reducing Techniques and Route Prediction for Efficient Regeneration of Energy

Hanu Priya Indiran[1]
[1] Student Member - IEEE, Bachelors in Electronics and Communication Engineering
Kumaraguru College of Technology (Affiliated to Anna University)
Coimbatore, Tamil Nadu, India.

ABSTRACT: This paper proposes a terrain specific OBD Data anomalies detection method with respect to ideal OBD values for the particular vehicle, previous drives' OBD dataset and driving behavior analysis in that specific terrain. The anomalies are transmitted using Carbon- footprint reducing techniques. A Route prediction technique for efficient regeneration of energy based on the wheel's powertrain at various speeds and routes traveled is also affixed with the proposed method. The proposed method monitors the combustion engine and hydraulic tubes in the case of Hybrid vehicles and Motor/Generator & Converter and transmission, Regenerative Braking Systems, the high-voltage battery in case of Electrical vehicles and gets the OBD using OBD II UART and low-power system on a chip microcontroller with integrated Wi-Fi.

Then the data analytics employs AdaBoost algorithm and pattern recognition and comparison techniques to detect anomalies. The anomalies are encrypted by DES Algorithm, Storage virtualization and Storage convergence techniques are used during the transmission of the encrypted anomalies to the automakers. Based on the analytics of the anomalies, version upgrades are made more reliable and OTA (On the Air) software updates can be done in Electric vehicles.

The route-based regenerative energy from wheels powertrain during the drives are estimated and displayed for the efficient choice of routes by the user. This data also provides efficient for performance analysis of regenerative systems at that terrain. The proposed methodologies provide efficiently for automakers to reduce recall expenses, improve cybersecurity response time, increase product quality and operational efficiency, and deliver post-sale vehicle performance and feature enhancements.

KEYWORDS: Terrain Specific OBD Data Anomalies Detection, AdaBoost Algorithm, pattern recognition, and comparison, Onboard Diagnostics(OBD), Regenerative energy-based route prediction, DES Data Encryption, Carbon footprint reduced data transmission.

I. INTRODUCTION

In the era of alarming carbon footprint accounted by vast climate change emissions, Clean vehicle technology keeps up the federal air quality standards. Self-sustained Electric and Hybrid vehicles are the trendsetters for eco-friendly automobiles. These automotive being the need of the era, their reliability and upgradations based on their performance is vital. Figure *1* gives the statistics of the failure of upgraded versions of cars in the year 2017 based on customer reviews.

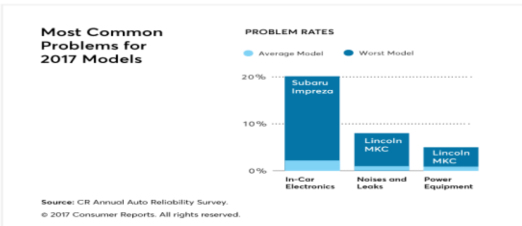

Figure 1

This occurs due to insufficient delivery post-sale vehicle data to determine vehicle performance which is mandatory for reliable upgradations.

A monitoring task allows characterizing the operation mode, recording information, recognizing and indicating abnormalities. Supervision functions indicate the undesirable or not permitted states of the process.

The data analytics can perform a major role in enhancing the reliability of the Research and Development areas of the automakers for manufacturing efficiency enhanced versions of their automotive.

Problem Definition:

Complacency in vehicle performance and not desired by user upgradations is a major issue for automakers. No proper delivery post-sale vehicle performance data record and analysis causes the lag. Performance expected and performance exhibited is to be recorded to formulate better performance which results in efficient Continuous improvement of the vehicle and adjusts it to fit the need of the users more accurately.

Proposed method :

The OBD data from Electric and Hybrid vehicles push by the OBD II UART and ESP32- low-power system on a chip microcontroller with integrated Wi-Fi

are analyzed for anomalies using AdaBoost, pattern recognition and comparison algorithms. The anomalies are DES-encrypted and virtually stored and transmitted to the automakers. A route prediction based on efficient regenerative energy from the wheel's power trains is also employed for route choice of user and performance analysis of regenerative systems. *Figure 2 r*epresents the overall methodology of the electronic and software systems Figure 2

Electronic Systems

The Car diagnostic port in the Electronic control unit of the car is interfaced with the 16 pins of the DL(Data Link) connector. From the connector, the Engine RPM, Engine coolant temperature, Engine load, Fuel pressure, timing advance, flow rate, runtime, absolute load, Gauge pressure, fuel tank level input, Throttle position, fuel-air equivalence ratio are received through OBD II UART and transmitted to the virtual storage by ESP32, a low-power system on a chip microcontroller with integrated Wi-Fi.

The *figure 3.a* represents the Circuit Diagram of the Electronic System:

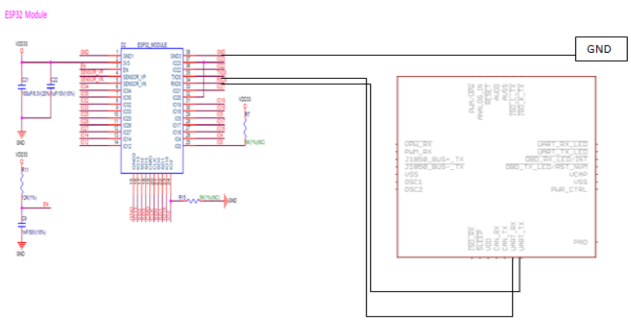

Figure 3.a

The PID stands for Parameter ID. The PIDs of vehicle speed, engine speed (RPM), throttle position, and engine load are defined in SAE J1979 and given in *Table 1*.

ISSN- 2394-5125VOL 7, ISSUE 15, 2020

PID(DEC)	Data bytes returned	Description	FORMULA
12	2	Engine RPM	(256*A+B)/4
5	1	Engine coolant temperature	A-40
4	1	Calculated engine load	A/2.55
35	2	Fuel Rail Gauge Pressure(diesel direct injection)	10*(256*A+B)
14	1	Timing advance	(A/2)-64
16	2	(MAF) air flow rate	(256*A+B)/4
31	2	Run time since engine start	256*A+B
67	2	Absolute load value	(256*A+B)/2.55
10	1	Fuel pressure (gauge pressure)	3*A
47	1	Fuel Tank Level Input	A/2.55
52	4	Oxygen Sensor 1- Fuel–Air Equivalence Ratio	((256*A+B)*2)/65536
69	1	Relative throttle position	A/2.55

Table 1.

Figure 3.*b* shows the system setup in FORD Figo to receive the OBD data through CAN protocol using OBD II UART and ESP32.

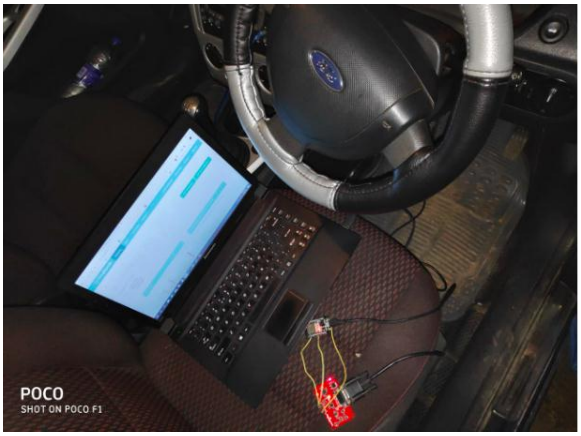

Figure 3.b

Software Systems:

The Algorithms employed by the software systems are as below

Algorithm:

Step 1: Get the accelerometer data

Step 2: Get the gyroscope data

Step 3: Get the altitude from latitude and longitude

Step 4: Determine terrain from the difference in altitude at every instant during the drive with respect to accelero-gyro readings from current location to destination

categorize terrain as :

1. Uphill
2. Downhill
3. Normal

If Uphill :

Step 5: Get mean of all OBD parameters in the dataset of past drives in Uphill terrain (say dataset A)

Step 6: Get ideal values of parameters pre-set by the manufacturer for the particular terrain

Step 7: Get all sensor value (Say dataset C) for a specified interval (dataset of the present)

Step 8: Use the Adaboost Algorithm for driving pattern classification for weak classifiers

- Decision trees

- Logistic Regression
- Support Vector Machines

If safe driving :

Step 9: Compare the values of the Dataset C with the parameters means of Dataset A and ideal values

If anomalies detected :

Step 10: Encrypt anomalies with AES (Advanced Encryption Standard)

Step 11: Storage virtualization and Storage convergence of the encrypted anomalies in the cloud. (which can be pulled by the automakers for their perusal)

If Downhill :

Step 12: Get mean of all OBD parameters in the dataset of past drives in downhill terrain (say dataset B)

Step 13: Get ideal values of parameters pre-set by the manufacturer for the particular terrain

Step 14: Get all sensor value (Say dataset D) for a specified interval (dataset of the present)

Step 15: Use the Adaboost Algorithm for driving pattern classification for weak classifiers

- Decision trees
- Logistic Regression
- Support Vector Machines

If safe driving :

Step 16: Compare the values of the Dataset D with the parameters means of Dataset B and ideal values

If anomalies detected :

Step 17: Encrypt anomalies with AES (Advanced Encryption Standard)

Step 18: Storage virtualization and Storage convergence of the encrypted anomalies in the cloud. (which can be pulled by the automakers for their perusal)

If Normal :

Step 19: Get mean of all OBD parameters in the dataset of past drives in Normal terrain (say dataset E)

Step 20: Get ideal values of parameters pre-set by the manufacturer for the particular terrain

Step 21: Get all sensor value (Say dataset F) for the specified interval (dataset of the present)

Step 22: Use the Adaboost Algorithm for driving pattern classification for weak classifiers

- Decision trees
- Logistic Regression
- Support Vector Machines

If safe driving :

Step 23: Compare the values of the Dataset F with the parameters means of Dataset E and ideal values

If anomalies detected :

Step 24: Encrypt anomalies with AES (Advanced Encryption Standard)

Step 25:Storage virtualization and Storage convergence of the encrypted anomalies in the cloud. (which can be pulled by the automakers for their perusal)

AdaBoost Algorithm :

The week classifiers used are

1. Decision trees

2. Logistic Regression
3. Support Vector Machines

The algorithm combines the above weak classifiers and stands as a strong classifier

$P(x) = sign(\sum_{t=1}^{T} w_t h_t(x))$

The final classifier consists of 'T' weak classifiers. $h_t(x)$ is the output of weak classifier 't' (in this paper, the outputs are limited to 3). w_t is the weight applied to classifier 't' as determined by AdaBoost. So the final output is just a linear combination of all of the weak classifiers, and then we make our final decision simply by looking at the sign of this sum.

Pattern Analysis and Comparison Algorithm:

For the specified terrain:

If predicted on the safe drive by AdaBoost Algorithm **then** Get the Dataset (Say A) of previous drives' OBD and Get average of all the parameters of the dataset.

PseudoCode

```
def get_averages(dataset A):
column_sums = None with open(dataset A) as file:
lines = file.readlines()
rows_of_numbers = [map(float, line.split(','))for line in lines]
sums = map(sum, zip(*rows_of_numbers))
averages = [sum_item / len(lines) for sum_item in sums]
return averages
```

Get Ideal values for the parameters in that terrain and Get Dataset(Say C) for the Present drive .Compare the average parameters of Dataset A and the ideal parameter values with the present drive's OBD data to find anomalies

Pseudocode

```
AverageidealParams = [line.strip() for line in file('averages')]
anomaly = [ ]
for line in [line for line in file('Dataset c')][1:]:
stop = 0
anomaly.append('-')
for item in line.split(','):
if stop : break
for AverageidealParam in AverageidealParams:
if idealParam in item:
anomaly[-1] != AverageidealParam
stop = 1
print anomaly
```

AES Encryption :

The Block size and round size is determined and the anomalies are Encrypted using the methodology in *figure 4.a*

ISSN- 2394-5125VOL 7, ISSUE 15, 2020

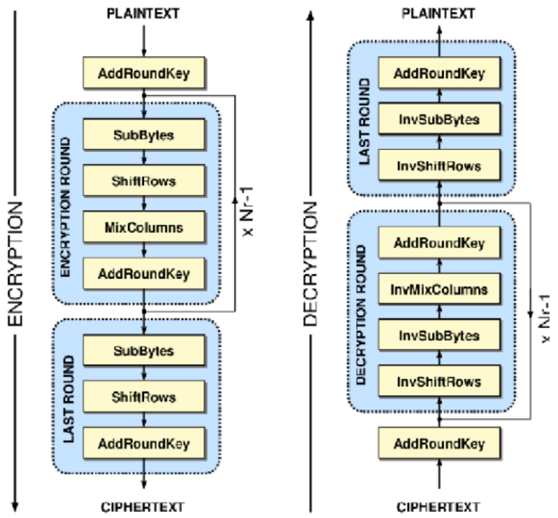

Figure 4.a

Route based Regenerative Energy Mapping:

Algorithm and Calculations:

Step 1: Get the value of RPM produced by the vehicle (N)

Step 2: Get the Number of poles of the machine (P)

Step 3: Get the Flux per pole in Weber (ϕ)

Step 4: Get the Total number of armature conductors **(Z)**

Step 5: Get the Number of parallel paths in the armature winding (A)

Step 6: Get the reverse voltage produce from the motor by using formula

$$E = PZ\frac{\phi N}{60A}$$

Step 7: Get the reverse voltage by using another formula

$$E = k_1 \phi N$$

Step 8: The k_1 (back EMF constant of the motor) revalue will be given in the datasheet of the motor**.**

Step 9: Get the reverse voltage for the period of time.

Total voltage = $E * time\ (minutes)$

The encrypted data are virtually stored by the methodology in *figure 4.b* for improved energy efficiency

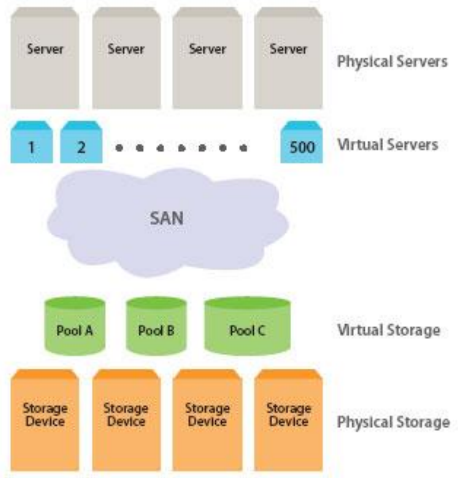

Figure 4.b

Results:

By employing the above methodologies the following results were obtained:

A. OBD values from OBD UART II and ESP32

Table 2 represents the OBD values from OBD UART II and ESP32

Altitude	Bearing	G(x)	G(y)	G(z)	G(calibrated)	Engine Coolant Temperature(°C)	Engine RPM(rpm)	Intake Air Temperature(°C)	Engine Load(%)	Mass Air Flow Rate(	Throttle Position(Manifold)(%
250	211.6	0.3504345	9.163701	-2.9478176	9.1E-04	56	2623.25	30	92.54901886	33.75999832	32.5490188
255	0	-6.745979	4.3944516	-4.0395737	-0.06278998	56	3169.5	30	90.98039246	37.20000076	32.5490188
247	0	-0.9232808	7.7367573	3.8708684	-0.0941295	56	2587.5	30	98.4313736	35.18000031	37.6470603
225	0	-4.7923107	2.1726327	16.387754	0.7736128	57	2753	30	98.03921509	38.04000092	39.215686
225	0	-0.061368175	7.1908793	6.6864495	0.038574874	57	2957	30	96.86274719	39.34999847	38.8235282
225	0	0.69519955	7.344568	6.993827	0.050246455	57	3158.5	30	97.2549057	41.93000031	4
225	0	0.3312809	7.181762	7.118325	0.050316274	57	3430.25	30	95.29412079	48.09000015	40.392158
225	0	1.164463	7.574411	5.586036	-0.014310896	58	2927.25	30	65.49019623	4.010000229	16.470588
225	0	0.8963125	6.7795362	7.9610844	0.08882338	58	2267	30	77.64705658	27.78000069	29.411766
225	0	1.1836166	6.5496926	7.616319	0.050063074	58	2353.75	30	93.33333588	31.42000008	31.764705
225	0	1.3847296	7.4403358	8.037699	0.14437789	59	2325.75	30	94.90196228	31.54000092	32.156864
225	0	-0.2529043	6.645461	7.635473	0.051165044	59	2303	30	93.72549438	30.25	32.156864
225	0	0.3027803	5.475941	7.408311	0.058579795	60	1685.75	30	34.11764908	4.289999962	16.470588
225	0	0.11124419	5.7153616	7.3700037	-0.01134787	60	1258.5	30	34.90196228	4.260000229	16.470588
225	0	0.5805077	5.6770544	7.3412733	-0.021725193	60	1507.75	30	28.23529434	4.139999866	16.470588
225	0	-0.6261699	5.9643583	6.948624	-0.045349956	61	1686.5	30	63.52941132	15.42000008	24.705883
225	0	0.7816206	6.2133555	6.3835926	-0.0694378	61	1667	30	94.11764526	21.10000038	28.627450
225	0	-1.066703	5.533402	6.776242	-0.08260286	61	2378.75	30	100	31.87000084	37.647060
225	0	-1.1433175	5.2652516	8.471336	0.04240471	61	2489.75	30	95.29412079	33.84000015	37.254901
225	0	0.27404988	5.926051	8.442606	0.0708316	62	2089	30	40	6.179999828	16.078432
225	0	-0.79855245	5.3227124	8.298954	0.027306855	62	2012.5	30	48.23529434	11.89999962	20.784313
225	0	-1.0571262	4.1351886	7.4370413	-0.10691422	63	1779.5	30	24.70588303	3.279999971	17.254901
225	0	-0.4442106	5.025831	7.207198	-0.084187865	63	1706.25	30	22.74509811	3.25999999	16.078432
225	0	-0.23352085	4.93964	7.2838125	-0.08355957	63	1428	30	24.70588303	3.25999999	16.078432
225	0	-0.109022364	5.495095	8.567104	0.05657047	63	1376.5	30	25.88235283	3.210000038	16.078432
225	0	-0.4537874	5.533402	7.848844	-9.81E-04	64	1879.5	30	29.41176605	7.119999886	18.823530
225	0	-0.8943205	5.6195936	8.394722	0.0527969	65	1793.75	30	24.70588303	3.450000048	18.823530
225	0	0.101667374	5.198214	8.9406	0.07327539	65	1407.75	30	24.31372643	3.180000067	16.078432
225	0	-0.769822	4.843872	7.944612	-0.029254317	65	1286.5	30	26.66666794	3.980000019	16.470588
225	0	0.63796854	5.15033	8.404299	0.02588135	65	1457.75	30	29.80392265	3.890000105	16.470588
225	0	-0.33886573	4.307571	7.168891	-0.12775195	65	1482	30	27.45098114	3.950000048	16.470588
225	0	-0.19521362	4.2788405	7.207198	-0.126369	66	1380.25	30	28.23529434	3.920000076	16.470588

Table 2

B. AdaBoost Algorithm Accuracy:

Table 3 represents the AdaBoost Algorithm Accuracy and figure 5.a gives the graphical representation of the data

Iteration Time	Gentle AdaBoost (Accuracy %)
5	99.39
10	100
15	100

Table 3

ISSN- 2394-5125VOL 7, ISSUE 15, 2020

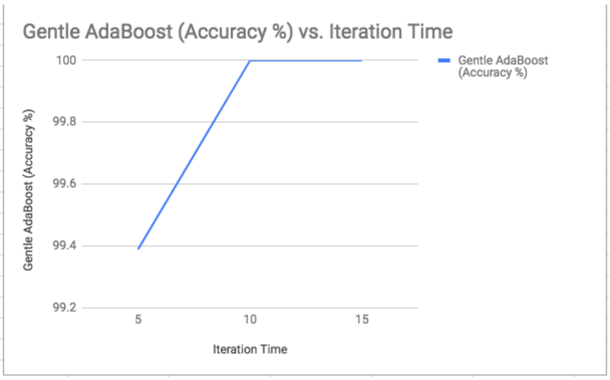

Figure 5.a

C. Average plot of Past Drives in Normal Terrain

Figure 5.b represents the averages of OBD data of Past Drives in Normal Terrain

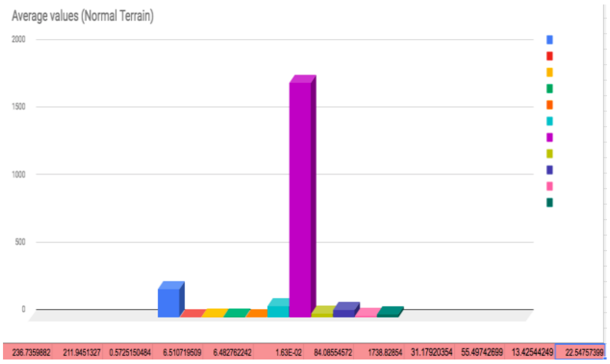

Figure 5.b

D. Anomalies Plot by Pattern Analysis and Comparison

The *figure 5.c* shows the anomalies detected by Pattern Analysis and Comparison

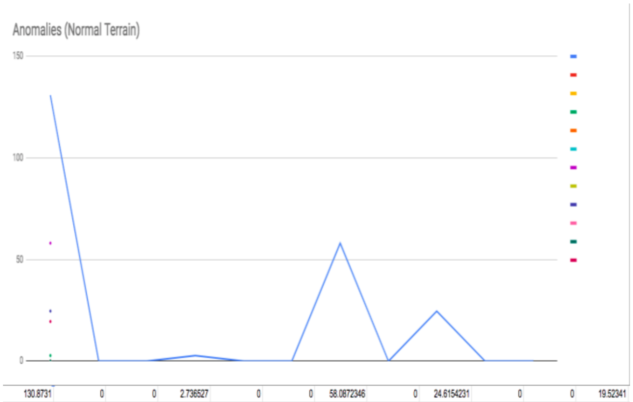

Figure 5.c

E. The Terrain setup in AIRSim Simulator for experimental Analysis

AIRsim is used to simulate terrains and test drive to obtain vehicle characteristics in the specific terrain. In the *figure 5.d* below simulated a drive-in Uphill terrain and its respective characteristics.

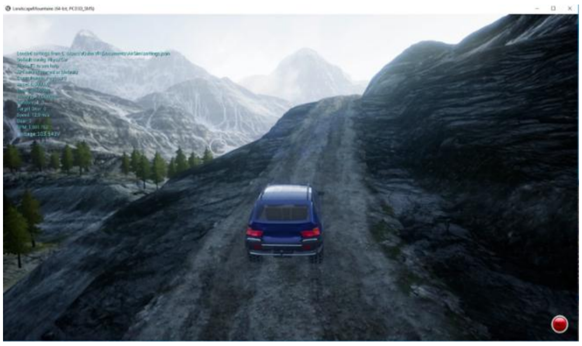

Figure 5.d

In the *figure 5.e* below simulated a drive-in Downhill terrain and its respective characteristics.

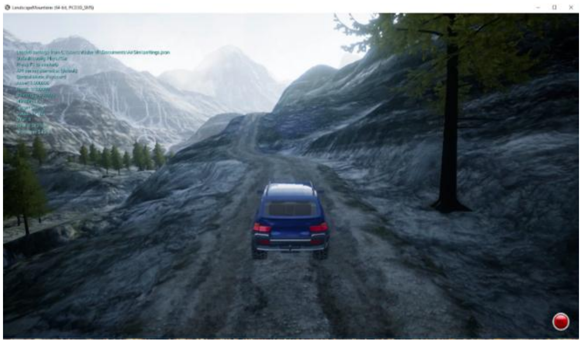

Figure 5.e

F. Route based Regenerative Energy Plot

Regenerative energy from the wheel's power trains is calculated for different paths to the same destination and depicts the more regenerative energy-generating trip.

The *table 4* and *figure 5.f* depicts the more regenerative energy-generating trip.

Total voltage = $E * time\ (minutes)$

Time	**Distance**	r.p.m	**reverse voltage**
21 min	**8.3 km**	**195.810406302821 83**	**8.2240v**
24 min	**10.6km**	**195.810406302821 83**	**9.3989v**
30min	**11.5km**	**195.810406302821 83**	**11.7486v**

Table 4

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

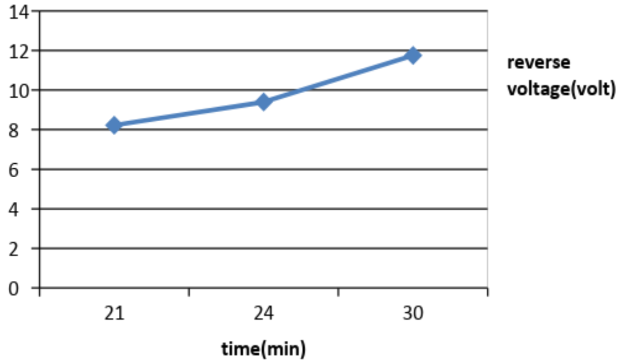

Figure 5.f

Conclusion:

Thus, the proposed methodologies provide efficient by **detecting anomalies**, **encrypting them and posting to the automakers** through **carbon- footprint reducing storage methods.** This data provides well for automakers to **analyze** and **make improvisation**, reduce recall expenses, improve cybersecurity response time, **increase product quality** and operational efficiency and deliver post-sale vehicle performance and feature enhancements. This could help **enhance the performance of Clean Automotive technologies** and **widen their market**, thus contributing to the global cause of **reducing Climate Changing emissions** and **reduced Carbon footprint** which is the need of the hour.

References :

1. Model-Based Fault Detection and Diagnostics
2. Trends in the application of model-based fault detection and diagnosis of technical processes
3. Fault-diagnosis systems: an introduction from fault detection to fault tolerance
4. Prognostics and health management design for rotary machinery systems—Reviews, methodology, and applications
5. Hybrid electric vehicle regenerative braking energy recovery system
6. Customer Reports of New version of cars
7. BD-II_PID
8. motor-calculations
9. emf-equation-of-dc-generator