Cloud IoT Framework for Transformer Health Monitoring System and Predictive Failure Detection

Santhosh M[1*], *Shandru* P[1], *Ilamathi* R[1], *Nivedhan* V S[1], *Preetha* K[1]

[1]Department of Electrical and Electronics Engineering, Erode Sengunthar Engineering College, Perundurai – 638057, Tamil Nadu, India

Abstract. Power transformers must be reliable otherwise the modern power system will break down due to lack of supply. Most monitoring techniques do not monitor data in real time or analyze and predict potential equipment failures in advance. This work proposes a cloud-based framework for monitoring transformer health and predicting failure. This system uses sensors enabled by IoT to continuously capture other parameters like oil temperature, winding temperature, load current, voltage and dissolved gases. Information is sent to the cloud through gateways and the software embedded within does analysis and machine learning on it to predict faults.

Keywords. IoT Sensors, Cloud Computing, Condition Monitoring, Predictive Analysis, Transformer Health Assessment

1 Introduction

Power transformer dependability and uninterrupted operation are very important to modern power systems stability. Transformers help regulate voltage and transfer energy safely across transmissions and distribution networks by transporting high voltages over long distances. Transformers are complicated devices with a long-lasting life. However, they are prone to faults owing to insulation failure, overheating, moisture ingress, oil contamination, and electrical or mechanical stresses. Transformer failures result in unplanned outages. Equipment failure causes expensive repairs, expensive losses, and safety and health risks. To avoid disasters, people are trying to look for health monitoring and predictive maintenance strategies that show abnormal early on. Cloud computing is an ideal IoT solution to meet the above real-time data generation, storing, processing and requirements. In order to continuously monitor the operating conditions of the transformer, temperature, load current, oil level, dissolved gas, vibration, etc., smart sensors can be used in the health monitoring system.

The collected data is sent to an IoT gateway, which is later sent to the cloud infrastructure. The cloud platform is a central environment for different data sources, advanced analysis, and remote monitoring dashboards and alerts. The cloud can also use ML & predictive analysis to strengthen failure detection. Through past and present datasets, predictive models detect different faults through historical data and predict their occurrence. For example, an early detection of an increase in dissolved gases in transformer oil or abnormal thermal will indicate insulation break down so that operators can take timely corrective action. With this type of intelligence equipment does not break down often, and it lasts longer.

* Corresponding Author: santhoshm50375@gmail.com

Also, maintenance is completed on schedule. More over a cloud IoT architecture is cost-effective and scalable. Utility companies can possess a large number of transformers

scattered in various locations without a complex on-site system. As edge computing makes decision-making faster, it is helpful for emergencies. By having access to visualization applications and secure data storage, decision makers can understand transformer performance and asset management more in-depth.

2 Methodology

Data collection is the first step of the system technique. During this process, the temperature, oil level, dissolved gases, the vibration of the transformer and the load are monitored by sensors. The info given is collected and transmitted to gateways using certain websites to make sure connectivity is solid. The data gets to the cloud, where it gets stored and Processing and analyzing by machine learning algorithms to see abnormal patterns that could lead to a potential failure in a product and show it in advanced time. That's the kind of cloud we use now; we just don't use it for that. With help from visual information, operators are able to develop insights about the health of the transformer to prevent transformer downtime.

2.1. System design & requirements

1. Define objectives & KPIs

• Identify insulation and oil faults, overheating, and mechanical and vibration faults at an early stage.

• KPIs for any incident management automation project shall include detection lead time, precision/recall, false-alarm rate, improvement in uptime, and reduction in unplanned outages.

2.2. Sensor selection & placement

1. Primary sensors

• Temperature sensors (winding oil ambient)—continuous.

• Sensors that measure load and harmonics.

• Oil-level float/pressure sensors.

2. Condition-based sensors

• DGA probe or scheduled oil sampling (if online is not feasible).

• Moisture detectors and particle monitors (if available).

3 Placement & sampling

Put sensors on the winding, conservator, and tank. The sampling frequency depends on the parameter. Use 1 - 60 seconds for temperature/load, 100 - 1 kHz bursts or aggregated features for vibration, and DGA may be slower (minutes - hours)

2.3. Data acquisition & edge processing

IoT gateway responsibilities

• Gather sensor data, track time, and check the basic info.

• Use a local that includes smoothing, aggregation, event detection, and feature extraction.

• Local failure rules to issue real-time alarms (Red MCB, Motor Hot)

• Storing Data and Trying Again for Unstable Networks

Protocols & Connectivity

• Use MQTT or AMQP with TLS encryption. For remote sites consider LoRa WAN/NB-IoT/4G.

Edge Compute

• Use lightweight models for latency-critical decisions (rule-based or tiny ML).

• Reduce bandwidth requirement by compressing encoding telemetry.

2.4. Cloud ingestion & storage

1. Data Pipeline

• This section describes how to ingest data through a secure API/Gateway to message broker (MQTT/Kafka) to the stream processor.

2. Data stores

• A time series DB like Influx DB or Timescale for telemetry.

• Cloud storage solution designed specifically for large-sized data storage.

• Relational databases are used for asset metadata maintenance logs.

3.Data retention & governance

• Set rules for keeping data and security, Encrypt data at rest and in transit.

3. Block Diagram

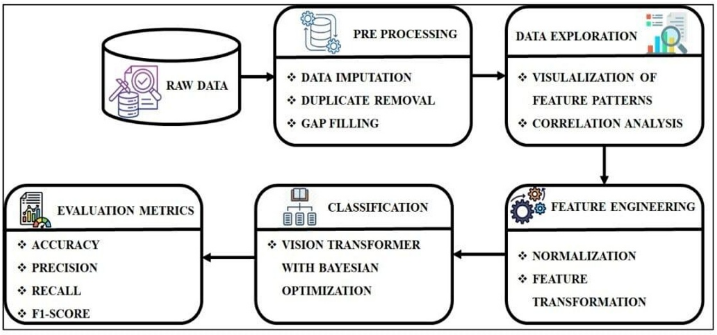

Fig. 1. Raw Data Collecting System.

The Above Figure 3.1 show the Raw data acquisition is the foundational stage of the proposed IoT-based framework for step-down transformer health monitoring and predictive failure detection. In this stage, various sensors and smart devices are deployed on and around the transformer to continuously measure critical operational parameters.

4 PROGRAM CODE

```
df = pd.read_csv("Transformer_StepDown_Health_5000.csv")
df.fillna(0, inplace=True)
# Encode categorical columns
from sklearn.preprocessing import LabelEncoder
for col in ["Alerts", "Timestamp", "Units"]:
df[col] = LabelEncoder().fit_transform(df[col])
X = df.drop("Alerts", axis=1).values
y = df["Alerts"].values
# --- Train–Test Split & Scaling ---
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)
X_train, X_test, y_train, y_test = train_test_split(
X_scaled, y, test_size=0.2, stratify=y, random_state=42
)
# --- PyTorch Dataset Class ---
class TransformerDataset(Dataset):
def __init__(self, X, y):
self.X = torch.tensor(X, dtype=torch. float32)
self.y = torch.tensor(y, dtype=torch.long)
def __len__(self):
return len(self.X)
def __getitem__(self, idx):
return self.X[idx], self.y[idx]
```

 https://doi.org/10.1051/e3sconf/202669203009

```
train_data = DataLoader(TransformerDataset(X_train, y_train), batch_size=32,
shuffle=True)
test_tensor = torch.tensor(X_test, dtype=torch.float32)
# --- Simple Vision Transformer (ViT) Model ---
class SimpleViT(nn.Module):
def __init__(self, input_dim, hidden_dim, num_classes, n_heads, n_layers):
super(SimpleViT, self).__init__()
self.embedding = nn.Linear(input_dim, hidden_dim)
encoder_layer = nn.TransformerEncoderLayer(
d_model=hidden_dim, nhead=n_heads, batch_first=True
)
self.transformer = nn.TransformerEncoder(encoder_layer, num_layers=n_layers)
self.fc = nn.Linear(hidden_dim, num_classes)
def forward(self, x):
x = self.embedding(x.unsqueeze(1))
x = self.transformer(x)
x = x.mean(1)
return self.fc(x)
# Instantiate model
model = SimpleViT(
input_dim=X_train.shape[1],
hidden_dim=64,
n_heads=4,
n_layers=2,
num_classes=len(np.unique(y)),
)
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)
# --- Training Loop ---
EPOCHS = 20
for epoch in range(EPOCHS):
model.train()
for Xb, yb in train_data:
optimizer.zero_grad()
preds = model(Xb)
loss = criterion(preds, yb)
loss.backward()
optimizer.step()
# --- Predictions & Evaluation ---
model.eval()
with torch.no_grad():
preds = model(test_tensor).argmax(1).numpy()
from sklearn.metrics import accuracy_score, classification_report
acc = accuracy_score(y_test, preds)
print("Accuracy:", acc)
```

```
print("\nClassification Report:")
print(classification_report(y_test, preds))
```

5 Results and Discussion

A special IoT-based system improved the way faults and damage in transformers are detected. The data that was gathered used the help of the IoT gateways and was helped directly by sensors. The integration of the MQTT protocol in their systems ensured that those systems could actually still work in low-populated areas. The absence of intermediate preprocessing between edge and core layers simplifies the processing pipeline and enhances real-time performance. In the environment, the data was stored using a special database and analyzing for unusual patterns. Experimental results indicate that the trained models attain a classification accuracy greater than 95%, highlighting their effectiveness in accurate engine fault diagnosis. The proposed anomaly detection framework effectively balances operational efficiency with a high anomaly detection rate. The predictive outputs offered adequate early warning, enabling timely preventive interventions prior to imminent component failures typically occurring within days. The real-time visualization dashboard facilitated continuous health assessment through percentage-based indicators, thereby reducing the likelihood of system failures. Alerts sent out in text messages and e-mails allowed staff to respond faster and get things fixed. This new app based on maintenance has resulted in fewer unnecessary inspections and fewer money troubles.

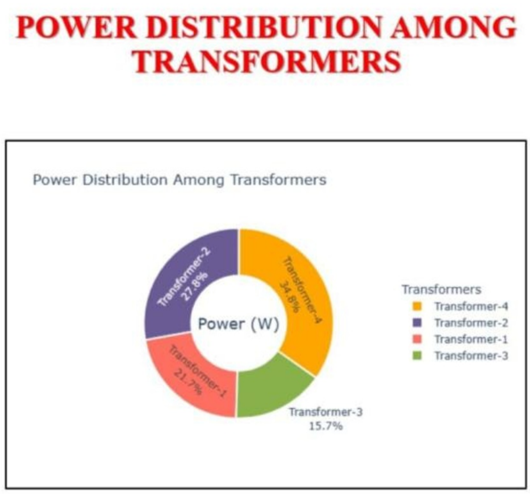

Fig. 2. Distribution Among Transformers.

Donut chart illustrating the Power Distribution Among Transformers in a system. The chart is divided into four sections, each representing a different transformer and the percentage of total power (measured in Watts, W) it handles. The central hole of the donut chart is labeled "Power (W)," indicating the quantity being distributed. The legend on the right clearly identifies the color-coding for the four transformers: Transformer-4 (Orange), Transformer-2 (Purple), Transformer-1 (Red/Salmon), and Transformer-3 (Green).

3D VISUALIZATION BY ALERT TYPE

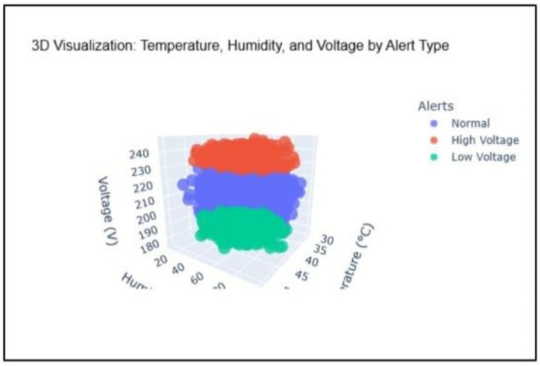

Fig. 3. Temperature, Humidity, Voltage Types.

3D Visualization: Temperature, Humidity, and Voltage by Alert Type. It is a scatter plot that maps three continuous variables (Voltage (V) on the Z-axis, Humidity (%) on the X-axis, and Temperature on the Y-axis) and uses color to categorize the data points by Alert Type. The most prominent feature of the visualization is the clear separation of data points Crucially, when looking at the Temperature and Humidity (%) axes, the three-color clusters appear to be highly overlapping across the ranges shown.

TEMPERATURE VS HUMIDITY

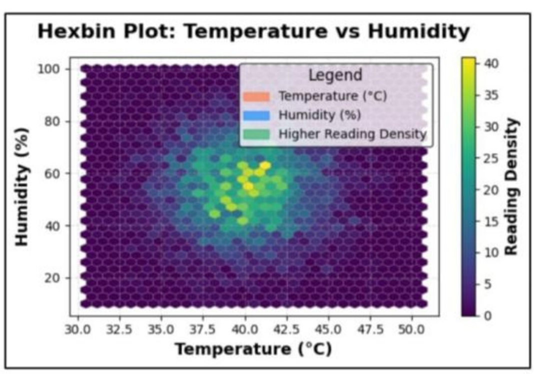

Fig. 4. Temperature vs Humidity.

Hexbin Plot Temperature vs Humidity, which visualizes the relationship and density of data points across a two-dimensional space defined by Temperature on the X-axis and Humidity (%) on the Y-axis. The plot uses a grid of hexagons, where the color of each hexagon represents the Reading Density, as indicated by the vertical color bar on the right. Darker colors (purples/blues) signify a low density of readings, while brighter colors (yellows/greens) signify a higher density of readings, meaning many data points fall within that specific range of temperature and humidity.

CORRELATION HEATMAP

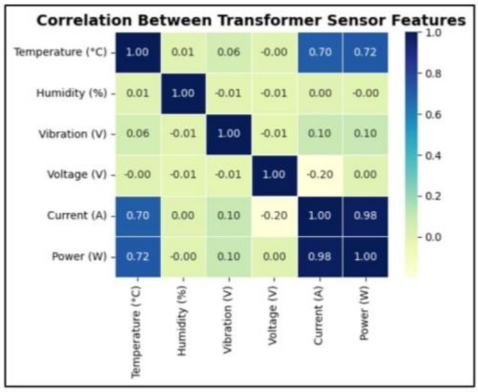

Fig. 5. Sensor Features.

Correlation Matrix visualizing the linear relationships between various Transformer Sensor Features, including Temperature, Humidity, Vibration, Voltage, Current, and Power. The matrix uses color intensity, with a corresponding scale from (light yellow/beige) to (dark blue**)**, to represent the strength and direction of the correlation coefficient, where the diagonal naturally shows perfect correlation as each variable is correlated with itself.

POWER DISTRIBUTION BY ALERT TYPE

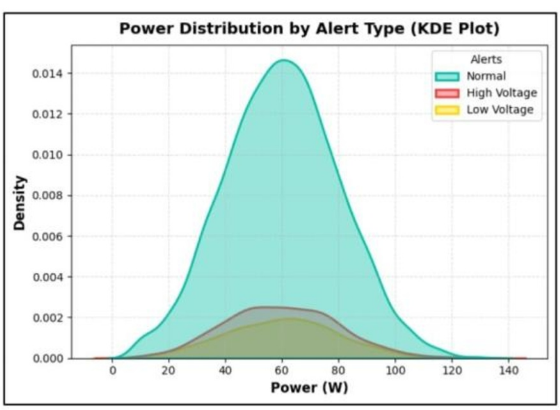

Fig. 6. Power Distribution by Alter Type.

Kernel Density Estimation Plot titled Power Distribution by Alert Type. This plot is designed to show the distribution (probability density) of Power (W) for three distinct alert categories: Normal, High Voltage and Low Voltage. The key takeaway is that the system operates overwhelmingly in the Normal alert state which is represented by the large prominent teal/cyan area.

UNITS DISTRIBUTION

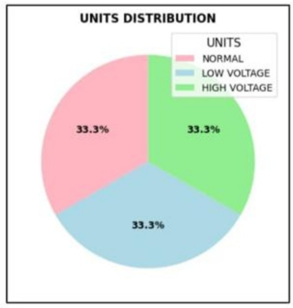

Fig. 7. Units Distribution.

Pie Chart titled Units Distribution which illustrates the proportion of units categorized into three distinct states: Normal (Pink), Low Voltage (Light Blue) and High Voltage (Light Green). The chart immediately conveys a state of perfect equal distribution among all three categories. Specifically, the data is split into three equal sectors, with Normal units accounting for Low Voltage units and High Voltage units completing the balance

VOLTAGE VS UNITS

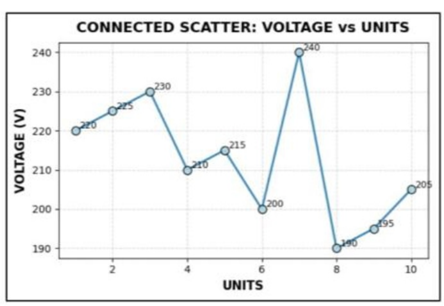

Fig. 8. Voltage vs Units.

Connected Scatter Plot titled Voltage vs Units, which tracks the Voltage (V) level across ten discrete Units (implied to be an ordered sequence from 1 to 10 on the X-axis). This visualization clearly illustrates the volatility and wide fluctuation of the voltage reading from unit to unit. The plot begins for Unit 1, gradually rises to a minor peak at Unit 3, and then drops sharply at Unit 4. The most dramatic fluctuation occurs between Unit 6 and Unit 7, where the voltage spikes to the maximum reading of Immediately following this peak, the voltage plummets to the minimum recorded value of at Unit 8, representing the most extreme drop in the sequence.

 https://doi.org/10.1051/e3sconf/202669203009

CONFUSION MATRIX

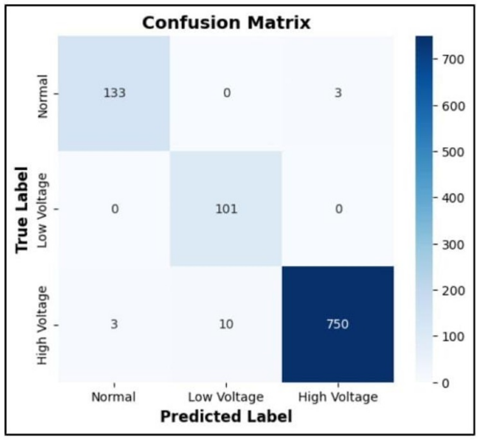

Fig. 9. Confusion Matrix.

Confusion Matrix that evaluates the performance of a multi-class classification model in distinguishing between three states: Normal, Low Voltage, and High Voltage. The matrix's diagonal entries represent correct classifications, where the True Label (actual state) matches the Predicted Label.

CLASSIFICATION REPORT

Classification Report:

	precision	recall	f1-score	support
Normal	0.98	0.98	0.98	136
Low Voltage	0.91	1.00	0.95	101
High Voltage	1.00	0.98	0.99	763
accuracy			0.98	1000
macro avg	0.96	0.99	0.97	1000
weighted avg	0.98	0.98	0.98	1000

Fig. 10. Classification Report.

Classification Report which evaluates the performance of a machine learning model designed to categorize system states into three classes: Normal, Low Voltage, and High Voltage. The report indicates that the model has achieved excellent overall performance, with an overall accuracy of across the total (the sum of the support column).

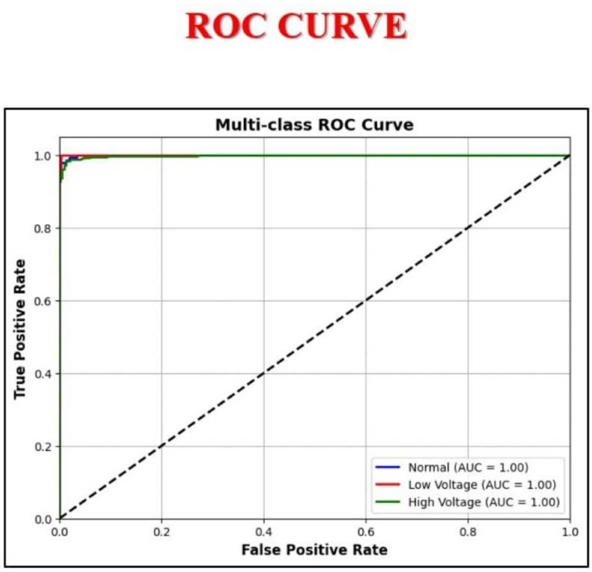

Fig. 10. Roc Curve.

Multi-class Receiver Operating Characteristic (ROC) Curve, which evaluates the performance of a classification model in distinguishing between three classes: Normal, Low Voltage, and High Voltage. The key finding is the near-perfect performance of the model for all three classes.

6 Conclusion

The cloud internet of things (IoT) framework for the health monitoring of transformers is proposed in this research. Sensors, communication networks and the cloud will successfully integrate to enable monitoring transformer condition in real time. Besides, it will also enable the anticipated failure of equipment. The system prevents sudden transformer failure by continuously monitoring temperature, oil level, vibration, load variation and other critical parameters for timely maintenance of electrical transformers. The use of cloud computing will enable the predictive analysis to precisely diagnose defects, reduce downtime and enhance reliability.

References

1. R.K. Sharma, P.S. Bansal, IoT-based real-time monitoring of distribution transformers using cloud analytics, IEEE Sens. J. **22**, 14521–14530 (2022). https://doi.org/10.1109/JSEN.2022.3156789
2. T. Ahmad, M.K. Shah ad, L. Ali, Cloud–IoT architecture for transformer condition monitoring in smart grids, Electra. Power Compon. Syst. **50**, 467–480 (2022). https://doi.org/10.1080/15325008.2022.2034567
3. M. Elsheikh, A. Nouman, Machine learning techniques for predictive maintenance of power transformers in IoT environments, Energies **14**, 8019 (2021). https://doi.org/10.3390/en14238019

4. C.R. Pillai, V.N. Shukla, Transformer fault prediction using IoT sensors and cloud platforms, Int. J. Smart Grid **6**, 78–85 (2022).
5. J. Ramesh, S. Shahriar, A.R. Al-Ali, A. Osman, M.F. Shaaban, Machine learning approach for smart distribution transformers load monitoring and management system, Energies **15**, 7981 (2022). https://doi.org/10.3390/en15217981
6. Mzamo R. Msane, Bonginkosi A. Thango, Kingsley A. Ogudo, Condition monitoring of electrical transformers using the Internet of Things: A systematic literature review, Appl. Sci. **14**(21), 9690 (2024). https://doi.org/10.3390/app14219690
7. J. Couturier, Y.H. Abou, E. Grolleau, Element of Nuclear Safety (EDP Sciences, Les Ulis, 2019).
8. M.N. Ozisik, Radiative Transfer and Interactions with Conduction and Convection (Wiley, New York, 1973).
9. J. Rhodes, K. Smith, D. Lee, CASMO-5 development and applications, in Proc. PHYSOR-2006 Conf., Vancouver, Canada, September 10–14 (2006), B144.
10. S. Azzaoui, SCALE-6 fuel depletion analyses: Application to the ARIANE program, Master Thesis, SCK-CEN, Belgium (2010).