Cyber-Physical Data Analytics to Enable Resilient Electric Grid

A. Srivastava, Washington State University

Associate Professor, Washington State University
Director, Smart Grid Demonstration and Research Investigation Lab
Technical Lead, UI-ASSIST Center
Contact: anurag.k.srivastava@wsu.edu
June 19th, 2019

IEEE PES SC Big Data & Analytics for Power Systems Webinar
What is resiliency?

How data analytics relate to resiliency?

How do we measure and enable resiliency?

How data analytics help and use cases

Learning based on projects: DOE CREDC, NSF FW-HTF, ARPA-E RIAPS, GMLC 1.3.9 Idaho Falls, GMLC: City of Cordova (DOE RADIANCE Project), DOE AGGREGATE, DOE UI-ASSIST
What is resiliency?

How data analytics relate to resiliency?

How do we measure and enable resiliency?

How data analytics help and use cases

Learning based on projects: DOE CREDC, NSF FW-HTF, ARPA-E RIAPS, GMLC 1.3.9 Idaho Falls, GMLC: City of Cordova (DOE RADIANCE Project), DOE AGGREGATE, DOE UI-ASSIST
There are millions of People in the world today who have no access to energy after extreme events.
An EMP weapon or strong solar flare can be even more destructive to the grid. Not a single/double contingency (as in security).
Power Grid: Reliable but Not Resilient

Electric Grid Resiliency

Resilience: The ability to supply its critical load through (and in spite of) extreme contingencies and low resource availability
Withstand any sudden inclement weather or human attack on the infrastructure.

Respond quickly, to restore balance in the community as quickly as possible, after an inevitable attack.

Adapt to abrupt and new operating conditions, while maintaining smooth functionality, both locally and globally.

Predict or Prevent future attacks based on patterns of past experiences, or reliable forecasts.
What is resiliency?

How data analytics relate to resiliency?

How do we measure and enable resiliency?

How data analytics help and use cases

Learning based on projects: DOE CREDC, NSF FW-HTF, ARPA-E RIAPS, GMLC 1.3.9 Idaho Falls, GMLC: City of Cordova (DOE RADIANCE Project), DOE AGGREGATE, DOE UI-ASSIST
Planning
- Design and long term planning
- Short term planning

Operation
- Situational awareness
- Decision support

RESILIENCY

Switch location?
DER/Storage location?
Weather Events?
Cyber Attack?
Power Systems Data: Example of fixed data

Fixed Data (Assets)
- 7,500 generation plants
- 75,000 substations
- 300,000 miles transmission (100,000 lines and transformers)
- 2.2 million miles distribution (1 million distribution feeders)
- 300 million customers

Credit: Prof Anjan Bose, WSU, TAMU NSF SPOKE
Data Collection by PMUs: Example of Operational Data

- PMU sampling rates: 30 per second
- Assume 100 values per second

If we assume all 100 points in a sub are PMUs
- Average data rate per sub is 10K/sec
- Average data rate for the total of 100 subs in a BA is 1M/sec
- Average data rate for the RC is then 10M/sec

Data Analytics Needed for Making Sense of this Steaming Operational Data for Cyber or Physical Events !!!!

Credit: Prof Anjan Bose, WSU
Connecting Data Analytics with Resiliency

- Predicts the future based on past patterns.
- Explores and examines data from multiple disconnected sources.
- Develops new analytical methods and machine learning models.
- Leverages data for relevant applications.
- Delivers actionable insights from the data.
- Stores and processes data for insights.
- Designs and creates data reports using various reporting tools.
- Queries databases and packages data for insights.

Enhancing Resiliency (Ukraine Case Study)
What is resiliency?

How data analytics relate to resiliency?

How do we measure and enable resiliency?

How data analytics help and use cases

Learning based on projects: DOE CREDC, NSF FW-HTF, ARPA-E RIAPS, GMLC 1.3.9 Idaho Falls, GMLC: City of Cordova (DOE RADIANCE Project), DOE AGGREGATE, DOE UI-ASSIST
Can we measure resiliency?

Red – Not Resilient
Purple – Resilient
Green – Super Resilient

System Plane

Attack Plane

Quantify design for better systems

Real-time Vulnerability Quantification

Initial Level Of Resilience

Proximity to collapse

Time taken To collapse

How much Tolerance?

How much Money

How much Money
Multi-criteria Decision for Physical Resiliency

- Analytical Hierarchical Process
- Topology Parameters
- Weather Parameters
- Infrastructure Parameter
Overview of resiliency quantification process

Weights assigned to factors using pairwise comparison, or can be used defined according to requirement

Interaction Index $\lambda$ is determined – models interdependency between factors considered

$$ C_\mu(f) = \int f d\mu = \sum_{i=1}^{n} (f(x_i) - f(x_{i-1})) \mu(A_{(i)}) $$

Choquet Integral to combine the factors into single resiliency value
What is resiliency?

How data analytics relate to resiliency?

How do we measure and enable resiliency?

How data analytics help and use cases

Learning based on projects: DOE CREDC, NSF Microgrid, DOE ARPA-E, GMLC 1.3.9 Idaho Falls, GMLC: City of Cordova (DOE RADIANCE Project), DOE AGGREGATE
Resiliency requires knowing the threat

Situational Awareness is necessary to take decision

Data analytics helps in enhanced awareness

- Predicts the future based on past patterns.
- Explores and examines data from multiple disconnected sources.
- Develop new analytical methods and machine learning models.
- Leverage data for relevant applications.
- Deliver actionable insights from the data.
- Store and process the data for insights.
- Design and create data reports using various reporting tools.
- Query database and package data for insights.

Data Science and Analytics

Data Processing, Database, Interfacing, Management

Statistical (Regression, clustering)

System and relational analysis (Graph Theory)

Machine Learning (Deep learning, reinforcement)

Visualization (Cognitive science)
Data Science Everywhere

- XBox Face Recognition
- Facebook Face recognition
- Google Home/Maps Prediction
- Amazon Alexa
- Apple Siri
- Self-driving cars
What can go wrong with Data Analytics
Low Quality AI

Trained with Bad Data
Use Case I: Anomaly Detection, Classification, Event Detection and Root Cause Analysis using PMU Data

Data

- Physical
  - PMU measurements
  - CT/PT measurements
  - Breaker status
  - Relay operations

- Cyber
  - Network data
    - Pcaps, netflows, Ilds alerts
  - Hosts
    - Event logs, Ilds alerts

Diagram:

- Anomaly: YES → Physical Event → YES → Normal Operation Status
- Anomaly: YES → Cyber Event → YES → Normal Operation Status
- Anomaly: NO → Physical Event → NO → Normal Operation Status
- Anomaly: NO → Cyber Event → NO → Normal Operation Status
Kronos: Real-Time Power Event on Heterogenous Data Stream

Goal: Build a lightweight Knowledge Base from power events and their semantic & temporal relationships for explainable event prediction and cause analysis, directly from cyber and power data streams.

Long-term Goal: An interactive NLP-based Question & Answering system for resilient Cyber-power system (imagine a “Siri” or IBM Watson system for cyber-power event and resiliency analysis)

With Dr. Wu and Dr. Hahn, WSU and Siemens
Problem Statement

- Input: Streams of events and physical entities (PMU, etc), ontology
- Output: A dynamically maintained knowledgebase (Kronos)
Global Data Streaming framework

1. accumulate changes of data/ground truth
   - Cache Data
     - $M_1, M_2, M_3, M_4, M_k$
     - Ensemble Model

   “Run-time stack”

2. Retrain
   - 2.1 update base detectors
   - 2.2 accumulate changes of models
   - 2.3 update ensembler

3. Detect
   - Model updates triggered by:
     1. distribution change $\epsilon$
     2. loss of accuracy $\epsilon'$ (when having training data)
     3. Base detection change $\epsilon''$

Windowing technique which updates component weight after each example.
Incremental classifier for the ensemble learning which is trained between component reweighting.
Online drift detector that allows the shorten drift reaction time.

Compare Both Windows
"Replace If there's change"

event entities (stars)
Options?

Linear regression
find straight line $y = \alpha + \beta x$ to provide a "best" fit for the data points w.r.t least-squares

Chebyshev method
Determine a lower bound of the percentage of data that exists within $k$ standard deviations from $\mu$

$$P(|X - \mu| \leq k\sigma) \geq (1 - \frac{1}{k^2})$$

$\mu$: mean, $\sigma$: standard deviation, $k$: number of standard deviations from the mean.

DBSCAN

- DBSCAN uses two thresholds radius $\varepsilon$ and $\text{min}$. 
- A data point is a center node if it has more than $\text{min}$ $\varepsilon$-neighbors (points within distance $\varepsilon$); 
- Two centers are reachable if they are in $\varepsilon$-neighbor of each other; a cluster is a sequence of reachable centers and their $\varepsilon$-neighbors. 
- New clusters is formed after the event ends. Points far away from any cluster are outliers.

Does standalone method suffice?
LSTM Auto-encoder Model

• The model consists of two RNNs – the encoder LSTM and the decoder LSTM as shown in Figure

• The input to the model is a sequence of vectors (PMU data)

• The encoder LSTM reads in this sequence

• Once input vector is read, the decoder LSTM takes over and outputs a prediction for the target sequence

• The encoder can be seen as ‘creating a list’ of new inputs and previously constructed list (learned weights).

• The decoder essentially unrolls this list, with the hidden to output weights extracting the element at the top of the list and the hidden to hidden weights extracting the rest of the list.

• Thus the LSTM weights are learned using the auto encoder method.
No Single Winner!
Needs tuning effort
Lack of training data
Anomaly Detection with Ensemble

1. Base Detectors
   - Regression
   - Chebyshev
   - DBSCAN
   - LSTM

2. Normalization of Base Detector Scores

3. MLE-Ensemble

4. Inference Algorithm

5. Unflagging Anomalies detected in Transient Window

6. Bad Data Detected

Model $Y_{MLE} (\alpha, \beta)$

Detection of Transient Window Using Prony Analysis

Data Window from PMU/PDC

(online) Learning

Data X

Outlier Scores

$D_1, D_2, D_3, D_4$
Maximum Likelihood Estimator (MLE)

- No Single Winner! -> ensemble-based
- Needs tuning effort -> learning best integration
- Lack of training data -> Unsupervised detection

Data Set $X$ → Compute Sensitivity $\Psi$ and Specificity $\eta$ → Learn Weights $\alpha$ and $\beta$ → Using EM algorithm fit $Y_{MLE}$ → Final learned weights $\alpha$, $\beta$

\[
y_{MLE} = \sum_{i=1}^{k} (f_i(x) \log \alpha_i + \log \beta_i)
\]
Given a PMU detector $D$ and PMU data $X$, denote the actual anomaly data set as $B_T$, and the anomaly reported by $D$ as $B_D$, the performance of $D$ is evaluated using three metrics as follows.

**Precision:** Precision measures the fraction of true anomaly data in the reported ones from $D$, defined as

$$Precision = \frac{|B_D \cap B_T|}{|B_D|}$$

**Recall:** Recall measures the ability of $D$ in finding all outliers, defined as

$$Recall = \frac{|B_D \cap B_T|}{|B_T|}$$

**False Positive:** False positive (FP) evaluates the possibility of false anomaly data detection; the smaller, the better.

$$FP = 1 - \frac{|B_D \cap B_T|}{|B_D|}$$
### Simulation results for SyncAD

#### RTDS simulated PMU data (1.5 hours)

<table>
<thead>
<tr>
<th>Method</th>
<th>Recall</th>
<th>Precision</th>
<th>False positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear Regression</td>
<td>0.9021</td>
<td>0.8565</td>
<td>0.1435</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>0.8821</td>
<td>0.8821</td>
<td>0.1179</td>
</tr>
<tr>
<td>Chebyshev</td>
<td>0.9154</td>
<td>0.8754</td>
<td>0.1246</td>
</tr>
<tr>
<td>LSTM</td>
<td>0.9298</td>
<td>0.8554</td>
<td>0.1446</td>
</tr>
<tr>
<td><strong>MLE ensemble</strong></td>
<td><strong>0.9351</strong></td>
<td><strong>0.8913</strong></td>
<td><strong>0.1087</strong></td>
</tr>
</tbody>
</table>

Tests on the RTDS simulated PMU data (1.5 hours, 5% bad data points, 5%-10% range)

<table>
<thead>
<tr>
<th>Method</th>
<th>Recall</th>
<th>Precision</th>
<th>False positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear Regression</td>
<td>0.7854</td>
<td>0.7655</td>
<td>0.2345</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>0.7216</td>
<td>0.7015</td>
<td>0.2985</td>
</tr>
<tr>
<td>Chebyshev</td>
<td>0.8125</td>
<td>0.7542</td>
<td>0.2458</td>
</tr>
<tr>
<td>LSTM</td>
<td>0.8298</td>
<td>0.7754</td>
<td>0.2246</td>
</tr>
<tr>
<td><strong>MLE ensemble</strong></td>
<td><strong>0.8912</strong></td>
<td><strong>0.9021</strong></td>
<td><strong>0.0979</strong></td>
</tr>
</tbody>
</table>

Tests on the RTDS simulated PMU data (1.5 hours, 10% bad data points, 10%-20% range)
Results with SyncAD using Real PMU Data
Event Detection Algorithm Architecture

Data Window from SyncAD

Computation of Active and Reactive Flows

DBSACN Algorithm

Cluster Change Detected?

Get New Data Window

Events Detected?

Decision Tree

Collection Of Cluster Change instances in V, I, P, Q and Fz

Undetected Events

Events Detected?

X

V

I

Fz

P

Q
Event Classification Process

- Decision Tree: Active Power Event, Reactive Power Event and Fault Events.

- Cluster changes in P and I: Active Power Event

- Cluster change in V and Q: Reactive Power Event.

- Cluster changes V, I, P, Q and Fq: Fault event.
## Simulation Results

<table>
<thead>
<tr>
<th>S No.</th>
<th>Time (s)</th>
<th>Reactive Event (Bus)</th>
<th>Active Event (Bus)</th>
<th>Fault Event (Bus)</th>
<th>Actual Events</th>
<th>Detected Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>109</td>
<td>9,7</td>
<td>-</td>
<td>-</td>
<td>Cap Bank Closed</td>
<td>Reactive</td>
</tr>
<tr>
<td>2</td>
<td>119</td>
<td>-</td>
<td>-</td>
<td>6,9,7,13,2</td>
<td>Three Phase fault</td>
<td>Fault</td>
</tr>
<tr>
<td>3</td>
<td>132</td>
<td>9,7</td>
<td>-</td>
<td>-</td>
<td>Cap Bank Opened</td>
<td>Reactive</td>
</tr>
<tr>
<td>4</td>
<td>148</td>
<td>-</td>
<td>7</td>
<td>-</td>
<td>P load decreased</td>
<td>Active</td>
</tr>
<tr>
<td>5</td>
<td>158</td>
<td>9,7,13</td>
<td>-</td>
<td>-</td>
<td>Cap Bank Closed</td>
<td>Reactive</td>
</tr>
<tr>
<td>6</td>
<td>168</td>
<td>-</td>
<td>-</td>
<td>6,9,7,13,2</td>
<td>Three Phase fault</td>
<td>Fault</td>
</tr>
<tr>
<td>7</td>
<td>179</td>
<td>9,7</td>
<td>-</td>
<td>-</td>
<td>Cap bank Opened</td>
<td>Reactive</td>
</tr>
<tr>
<td>8</td>
<td>188</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>P load increased</td>
<td>No Detection</td>
</tr>
<tr>
<td>9</td>
<td>198</td>
<td>9,7</td>
<td>-</td>
<td>-</td>
<td>Q load increased</td>
<td>Reactive</td>
</tr>
<tr>
<td>10</td>
<td>209</td>
<td>-</td>
<td>9</td>
<td>-</td>
<td>P load decreased</td>
<td>Active</td>
</tr>
<tr>
<td>11</td>
<td>219</td>
<td>9,7</td>
<td>-</td>
<td>-</td>
<td>Q load decreased</td>
<td>Reactive</td>
</tr>
<tr>
<td>12</td>
<td>229</td>
<td>-</td>
<td>2</td>
<td>-</td>
<td>Gen Drop</td>
<td>Active</td>
</tr>
</tbody>
</table>
Case 1: keyword search (e.g., fault, reactive)
Case 2: Ontology and Correlation Monitoring (blue edges)
Case 3: Map interaction (google map)
Extending for microPMU

- **Objective**: Sensor data analytics (specifically using μPMU) for anomaly detection, classification, event detection, root cause analysis and explanation generation

- **Tasks and Deliverables**:
  - Provenance-aware Anomaly Detection and Prediction
  - Online Event Detection using μPMU datasets
  - Root cause detection based on statistical causality models
  - Explanation generation and interpretation

---

**Event Detection**

- Time-series modeling
  - Olympic model, Regression model, …

- Fault detection
  - (Outliers, Changed points, Missing data)
  - Simple threshold, DBScan, KSigma, …

**Correlation Detection between Events**

- (Extracting edges via Granger Causality)

**Provenance Generating explanations**

**Root Cause Analysis**

via probabilistic models
Use case II: Cyber-physical Data Analytics in Protection Failure

- Protection Mal-operation is #1 concern according to NERC
- Protection and associated control is becoming more digital
A fault occurs on line 2-3. Relays 7 and 8 are expected to open their corresponding breakers but relay 7 doesn’t respond.

To compensate relay’s 7 malfunction, relays 1, 3, 10 and 12 should open their corresponding breakers but relay 1 malfunctions.
# Hypothesis Generation

<table>
<thead>
<tr>
<th>Hypothesis #</th>
<th>Location of fault</th>
<th>Initial Incident</th>
<th>Consequential Incident</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual Scenario</td>
<td>Line 2-3</td>
<td>Breaker 8 tripped Relay 7 malfunctioned</td>
<td>Breakers 3,10,12 tripped Relay 1 malfunctioned</td>
</tr>
<tr>
<td>Hypothesis 1</td>
<td>Line 2-4</td>
<td>Breaker 10 tripped Relay 9 malfunctioned</td>
<td>Breakers 3,8,12 tripped Relay 1 malfunctioned Relay 6 Tripped</td>
</tr>
<tr>
<td>Hypothesis 2</td>
<td>Line 2-1-2</td>
<td>Breaker 3 tripped Relay 4 malfunctioned</td>
<td>Breakers 8,10,12 tripped Relay 1 malfunctioned Relay 6 Tripped</td>
</tr>
<tr>
<td>Hypothesis 3</td>
<td>Line 1-5</td>
<td>Breaker 6 tripped Relay 5 malfunctioned</td>
<td>Relay 2, 3, 4 malfunctioned Breakers 8,10,12 tripped</td>
</tr>
<tr>
<td>Hypothesis 4</td>
<td>Line 2-5</td>
<td>Breaker 12 tripped Relay 11 malfunctioned</td>
<td>Breakers 3, 8, 10 tripped Relay 1 malfunctioned Relay 6 Tripped</td>
</tr>
</tbody>
</table>

Cyber Physical Security Analytics for Anomalies in Transmission Protection Systems
Data Analytics For Event Classification

SCADA
- Breaker Status and Topology of the System
- Breaker Status Change

Streaming PMU Data
- PMU Data
- Autoencoder
- Fault Detection (Physical Data)
- IF-Else Conditions based Final Decision
- Cyber Attack
- Physical Fault

Streaming Cyber Data
- Cyber Data
- Signature Based Algorithm
- Intrusion Detection (Cyber Data)
- Cyber-Physical

Cyber Physical Security Analytics for Anomalies in Transmission Protection Systems
Simulating Cyber Attack on a Relay

1. Attacker sends an e-mail with malware
2. E-mail recipient opens the e-mail and the malware gets installed quietly
3. Using the information that malware gets, hacker is able to take control of the e-mail recipient's PC and get access of two-level password
4. Analyse IEC 61850 protocol (GOOSE, SMV packet) information and relay setting file
5. Manipulate MMS packet and relay configuration session information
6. Takes control of circuit breaker or change the setting of relay
Detect Intrusion Using Cyber Data From Relay.

Attack Scenario For Relay Communication between Relay and Unauthorized IP Address -(Attacker)

Detecting an Intrusion:
Detect Intrusion Using Physical Data From PMU

Algorithm Description:

- Basic Idea: Reconstruction of input feature vector with minimum loss (Mean Square Error)

- Train the algorithm on input data consisting of no anomalies. Output Result: Reconstructed input feature vector with low MSE.

- Test the algorithm on input data consisting of anomalies. Output Result: Reconstructed input feature vector with high MSE.

- We want our algorithm to have high MSE on input data consisting of anomalies and low MSE on input data consisting of no anomalies.
Detect Intrusion Using Physical Data From PMU

Architecture Of Stacked Autoencoder

Loss Function: Mean Squared Error
Optimizer: ADAM

: Input Feature Vector
: Reconstructed Output Feature Vector
Detect Intrusion Using Physical Data From PMU

Dataset Description:

<table>
<thead>
<tr>
<th>Dataset</th>
<th># PMU Readings (Total: 37500)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Dataset (No Fault)</td>
<td>22250</td>
</tr>
<tr>
<td>Testing Dataset (No Fault)</td>
<td>11250</td>
</tr>
<tr>
<td>Validation Dataset (Fault)</td>
<td>4000</td>
</tr>
</tbody>
</table>

Types Of Validation Dataset:

<table>
<thead>
<tr>
<th>Validation Dataset</th>
<th>PMU Readings (# Normal Instances)</th>
<th>PMU Readings (# Anomalous Instances)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1</td>
<td>3979</td>
<td>21</td>
</tr>
<tr>
<td>Type 2 (Synthetic Minority Oversampling -SMOTE)</td>
<td>3979</td>
<td>3979</td>
</tr>
</tbody>
</table>
Evaluate Metrics

The intersection between actual values and predicted values yield four possible situations:

- **True Positive (TP)**: Positive instances correctly classified.
- **False Positive (FP)**: Negative instances classified as positive.
- **True Negative (TN)**: Negative instances correctly classified as negative.
- **False Negative (FN)**: Positive instances classified as negative.

**Classification Measures:**

Accuracy is calculated as the number of correctly classified instances over total number of instances evaluated.

\[
\text{Accuracy} = \frac{TP + TN}{\text{Total instances}}
\]

Precision is the percentage of correctly predicted instances over the total instances predicted for positive class.

\[
\text{Precision} = \frac{TP}{TP + FP}
\]

Recall is the percentage of correctly classified instances over the total actual instances for the positive class.

\[
\text{Recall} = \frac{TP}{TP + FN}
\]

F-Measure is a measure of test accuracy.

\[
\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]
Detect Intrusion Using Physical Data From PMU

Autoencoder Evaluation On Type 1 (Validation Dataset)

<table>
<thead>
<tr>
<th>Threshold (Test Data)</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.003617 (Minimum)</td>
<td>5.50%</td>
<td>0.99</td>
<td>0.06</td>
<td>0.09</td>
</tr>
<tr>
<td>0.003621 (Mean)</td>
<td>50.25%</td>
<td>0.99</td>
<td>0.50</td>
<td>0.66</td>
</tr>
<tr>
<td>0.003625 (Maximum)</td>
<td>99.48%</td>
<td>1.0</td>
<td>0.99</td>
<td>1.00</td>
</tr>
</tbody>
</table>
Decision Based On Data Analytics and Validation Using Additional Non-Streaming Data

- PMU 2 and 3 show highest MSE among all PMUs
- It can be determined that most probably the fault could have occurred in the line from bus 2 and 3
Use case III: Load/ DER Disaggregation

- Increasing PV penetration
  - Behind-the-meter (Invisible)
  - PV with meters (Visible)
- Invisible solar photovoltaic not monitored
- Invisible to utilities and system operators
- Visibility into behind-the-meter solar generation is limited

- System net load is a key input when scheduling for the short-term operation (minutes/ hours)
- DER estimation can help with voltage control and CVR, especially given high percentage of rooftop PV

- Exact load estimation can help with cold load pickup after the outage
- DER estimation can also help for resiliency driven outage management with DERs in microgrids/ active distribution system
How do we gain more visibility into the PV generation and load behind the meter?

- Data-driven methodology can be used to estimate (ML prediction) the power generation of invisible solar power sites in short time scale and load/DER can be disaggregated.

With Dr. Gebremedhin, WSU
Supervised Machine Learning

- A computer system learns from data, which represent some “past experiences” of an application domain

- **Our focus:** learn a **target function** that can be used to predict the values of a continuous value, i.e. *load, power generation*

- The task is commonly called **Regression**: a specific type of **Supervised learning**, complementary to **Classification**

- **Supervision:** the data (observations, measurements, etc.) are labeled with pre-defined values
Regression Problem Formulation

• **Data**: a set of data records (also called examples, instances) described by
  - *k attributes*: $X_1, X_2, \ldots, X_k$ (e.g. weather, voltage, power)
  - a target value (*y*): each example has a pre-defined value (e.g. power estimation)

• **Goal**: learn a regression model from the data that can be used for estimating values of new (future or test) instances
  - We may use a simple model like linear regression or a more complex model
    - **General** $y = F ( X_0, X_1, X_2, \ldots, X_k ) + \varepsilon$
    - **Linear** $y = X_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k + \varepsilon$

• **Evaluation**: accuracy of estimation
PV/Load Disaggregation

• Data Properties
  ▪ High sampling rate
  ▪ Heterogeneous data (e.g. different sampling rate, different nature)
  ▪ Many missing points

• Weather data: 4 to 9 variables
  ▪ Diffused irradiance, global irradiance, humidity, temperature,
    wind direction, wind speed, dew point, pressure, rain

• Challenges
  ▪ Large scale data
    • More than 31,540,000 rows of data per year (based on HNEI data)
  ▪ How can we represent data to expedite model training?
    • Feature extraction
  ▪ How can we use future data to upgrade the model?
    • Online Learning, Deep Learning
PV/Load Disaggregation ML Pipeline

Data Preparation

Model Training

Validation

Modification (Feature Selection, Segmentation Setting, ML Model Selection)
Data Preparation

- **Low-pass filter**
  - to remove high frequency noise data

- **Average-based sliding window method**
  - to fill missing values

- **Process weather station text data**
  - to generate clean and structured data

- **Spline interpolation**
  - to match different sampling rates

- **Synchronization**
  - timestamp adaptation for different sources of data
Data Segmentation and Feature Extraction Implementation

\[
\begin{align*}
S_{11}, S_{12}, \ldots, S_{1k} & \quad T_{11}, T_{12}, \ldots, T_{1m} & \quad C_{11}, C_{12}, \ldots, C_{1p} \\
S_{21}, S_{22}, \ldots, S_{2k} & \quad T_{21}, T_{22}, \ldots, T_{2m} & \quad C_{21}, C_{22}, \ldots, C_{2p} \\
& \quad \vdots & \quad \vdots \\
S_{n1}, S_{n2}, \ldots, S_{nk} & \quad T_{n1}, T_{n2}, \ldots, T_{nm} & \quad C_{n1}, C_{n2}, \ldots, C_{np}
\end{align*}
\]

\[\text{meter Data} \quad \text{Time} \quad \text{Weather}\]

Number of features for each segment:

\[
(\text{PMU}_{\text{size}} + W_{\text{size}}) \times \text{number of SF} + \text{number of TR features}
\]
Datasets (working with HNEI)

• Real Data (Maui Hawaii) --- One Year
  - Smart-meter transformer data
  - Smart-meter load data (target value)
  - Smart-meter solar panel data (target value)
  - Weather data in text format (9 parameters)

• GridLab-D (IEEE 123 Node Test Feeder) --- One Year
  - PMU data: current and voltage for the entire network
  - Weather data (low sampling rate data)
  - Seattle weather data (4 parameters)
  - 5 Solar panels in the network
  - Load data (in progress)
Case Studies

• Real data, Load estimation
  - Target: load value
  - Input: transformer smart-meter data and weather data

• Real data, Solar panel power estimation
  - Target: solar panel power generation
  - Input: weather data

• GridLab-D data, Solar panel power estimation
  - Target: solar panel power generation
  - Input: weather data

• (Ongoing) GridLab-D data, Load estimation
Accuracy Metrics

- **R-squared**
  - Also called *coefficient of determination* (the square of the correlation coefficient)
  - Represents the fraction of the variance in y that can be explained by the regression model

\[
R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i(y_i - \hat{y}_i)^2}{\sum_i(y_i - \bar{y})^2}
\]

- **RMSE**
  - The Root Mean Square Error for prediction

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}
\]
Real Data, Load Estimation

• Investigated importance of features
  ▪ Most valuable feature: irradiance

• Models we used
  ▪ Linear regression, decision tree based regression, deep learning (LSTM)

• Future direction
  ▪ Train model based on Load and PV capacity from multiple locations
  ▪ Weather variation: a generalized model that works for weather condition in any location
  ▪ Active Learning: collect data for load based on PV estimation and update load estimation model

<table>
<thead>
<tr>
<th>Scenario/ RMSE (%)</th>
<th>Transformer Only</th>
<th>Transformer &amp; PV</th>
<th>Context-aware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario 1</td>
<td>20 to 40%</td>
<td>12 to 41%</td>
<td>4 to 18%</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>18 to 36%</td>
<td>10 to 3%</td>
<td>4 to 13%</td>
</tr>
<tr>
<td>Scenario 3</td>
<td>8 to 28%</td>
<td>7 to 12%</td>
<td>2 to 11%</td>
</tr>
</tbody>
</table>

Each scenario corresponds to a specific train/test split:
S1: 3 months train, one month test
S2: 6 months train, one month test
S3: 11 months train, one month test

Models: LR, DT, LSTM
Real Data, PV Estimation

- We found that power generation is highly dependent on context
- Using transformer smart meter data results in higher accuracy
- Models we used
  - Decision tree based
  - Linear regression
  - MLP
- Future direction
  - Use LSTM
  - Train models for different scenarios (PV capacity, different weather conditions)

Each scenario corresponds to a specific train/test split:
S1: 3 months train, one month test
S2: 6 months train, one month test
S3: 11 months train, one month test

| Scenario/\n| \n| R² | Transformer Only | Transformer and Load | Context-aware |
|---|---|---|---|
| Scenario 1 | 0.56 | 0.63 | 0.83 |
| Scenario 2 | 0.66 | 0.65 | 0.81 |
| Scenario 3 | 0.72 | 0.77 | 0.89 |

Model: DT

Estimation for a day
Use Case IV: Cyber-Physical Analytics for the Transactive Energy Systems

Data Analytics Based Anomaly Detection in TES

- Detecting malicious activity within a TES environment is challenging due to the diverse group of participants
  - Prosumers
  - Market Participants
  - Communication networks
  - Transmission and Distribution networks

- Systems and networks are vulnerable to diverse attacks

- Physical and cyber data are available for monitoring.

- Huge information flow: Manual detection will be difficult
Possible Cyber Events in TES

### Denial of Service

- An attacker sends several packets to the host in order to cause the unavailability of the resources.
- This can be detected by analyzing cyber data.

### Manipulation of Bid Values

- The Bid Price and Bid Quantities are altered to random values through a malware injection.
- The impact can be seen in the cyber, physical, and market information.
Architecture of Proposed Ensemble Decision Tree Algorithm

- **Input Data**
  - Cyber Data
  - Physical Data
  - Market Data

- **Ensemble Decision Tree Algorithm**

- **Detect System State**
  - (Cyber Anomaly, Physical Anomaly, Normal Operation)

- **Check Cyber System Data**
  - (log Files, Server Accesses, Network Traffic)

- **Validate Physical System**
  - (V, I, P, Q)

# Combine all three types of data in to a single data frame
- Given the data, the ensemble decision tree algorithm detects/classifies cyber and physical events in the Transactive Energy System.

- The proposed algorithm is based on the concept of bagging in which each single decision tree model is built from a random subset composing the ensemble of decision trees.

- “Confidence” and “Probability” measures are computed while predicting a class at a node.

- After all single decision tree model are created their prediction is combined using a combiner measure called “Plurality” and then the final class is predicted.
Ensemble Model: Measures

• CONFIDENCE:
  • Imposes a confidence at every node in an individual decision tree model.
  • The confidence is calculated using lower bound of "Wilson Score Interval" formula.
  • Core Idea of imposing 'confidence' is to balance the proportion of the predicted class with uncertainty of a small number of instances.

\[
\hat{p} + \frac{1}{2n} z^2 - z \sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}}
\]

• \( \hat{p} \) = proportion of instances for the predicted class at a particular node.
• \( n \) : the number of instances at that node.
• \( z = 1.96 \)

• PROBABILITY:
  • The probability of a class at a certain node is the percentage of instances for that class at the node.
  • Additive smoothing tends to pull the class probabilities towards their overall proportion in the dataset.
  • If there are only a few instances in the leaf leading to the prediction, this smoothing can have a significant impact. If there are a high number of instances, the smoothing will have little effect.

\[
\text{Probability}_{\text{class}} = \frac{\text{node_class_count} + \text{class_prior}}{\text{node_total_count} + 1}
\]

• node_total_count : Total number of instances in the node.
• node_class_count : Total number of instances for the class of interest at a given node.
• class_prior : Percentage of instances for that class over the total instances in the dataset.

• PLURALITY :
  • Plurality is based on the premise that each model in the ensemble has one vote for a prediction.
  • The class with most votes is returned as the final prediction.
  • The resulting prediction is computed by averaging the confidence of the models that are predicting the right value.
The simulated power system includes a 9-bus transmission system and one feeder with transactive components at node 7. The HVAC devices in each house will participate in the power market.

TESP Test bed

TESP is a framework designed by PNNL that simulates transactive systems. It includes various software modules and a number of agents in the form of smart houses.

- Mininet has been used to create a network that helps the house controllers get and send messages from/to the FNCS broker.
- The network has a tree topology with the central controller as the root node. There are 30 TE Agents. Each agent has a house controller.
- The house controller sends the temperature values, bid values to the FNCS broker, which publishes the information for the other components to read.
- The attack performed in the study aims at the house controller and manipulating the bid values which are then sent to the FNCS.

Simulation Scenarios

**Scenario 1** (High Impact): Manipulating bid price and quantity to an arbitrary large value. (Aim to impact the operation of power grid)

**Scenario 2** (Low Impact): Manipulating bid price and quantity to some reasonable values. (Aim to gain benefits from the market)

### Input Features
- Bid price
- Bid quantity
- LMP
- Voltages
- Generators real and reactive power outputs
- Total demand
- Flows
- Class

### Dataset Description

**Output Labels**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>ΔBid Price</th>
<th>ΔBid Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Cyber Attack</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>Physical Outage</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Scenario 1 (High Impact):**
- Manipulating bid price and quantity to an arbitrary large value.
- Aim to lead series of events in the operation of power grid.

**Scenario 2 (Low Impact):**
- Manipulating bid price and quantity to some reasonable values.
- Aim to gain benefits from the market.

**Trained-Tested using 80% and 20% split, respectively.**
Results: Impact of Cyber Attack on TES

Scenario 1: Manipulating bid price and quantity to an arbitrary large value.

Scenario 2: Manipulating bid price and quantity to some reasonable values.
Evaluation Metric Results

Confusion matrix is a table containing the predictions and actual values of objective field classes. The intersection between actual values and predicted values yield four possible situations:

- True Positive (TP): Positive instances correctly classified.
- False Positive (FP): Negative instances classified as positive.
- True Negative (TN): Negative instances correctly classified as negative.
- False Negative (FN): Positive instances classified as negative.

Classification Measures:

Accuracy is calculated as the number of correctly classified instances over total number of instances evaluated.

\[
\text{Accuracy} = \frac{TP + TN}{\text{Total instances}}
\]

Precision is the percentage of correctly predicted instances over the total instances predicted for positive class.

\[
\text{Precision} = \frac{TP}{TP + FP}
\]

Recall is the percentage of correctly classified instances over the total actual instances for the positive class.

\[
\text{Recall} = \frac{TP}{TP + FN}
\]

F-Measure is a measure of test accuracy.

\[
\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

Phi-Coefficient is the correlation coefficient between the predicted and actual values.

\[
\text{Phi Coefficient} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}}
\]

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>Phi-Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario 1</td>
<td>92.2%</td>
<td>92.0%</td>
<td>91.1%</td>
<td>0.92</td>
<td>0.83</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>99.97%</td>
<td>83.32%</td>
<td>99.98%</td>
<td>0.87</td>
<td>0.89</td>
</tr>
</tbody>
</table>
Use Case V: Enabling Resiliency

Cordova, AK, serves as a perfect microcosm for prototyping innovations for the entire power grid across the nation.

- Tsunamis
- Avalanches
- Power outages disrupting fishing business
CANVASS?

- Canvass stands for Cyber-Attacks and Network Vulnerability Analytics Software for Smart Grids
- It enables unfavorable physical and cyber event simulation for power systems
- Free, open-source, platform-independent resiliency-computation toolkit
- It has default restoration and resiliency computation algorithms – with ability for user to define own metrics and scenarios.
- It enables easy power system modeling and interdisciplinary resiliency engineering research by abstracting lower level (hard-to-learn) open-source:
  - power simulation software [GridLAB-D],
  - network analysis library [NetworkX],
  - OS-based socket libraries [TCP/IP]
  - Packet Manipulation library [ScaPy]
- It can interface with Real-Time Simulation software through socket programming.
- Multiple interdependent infrastructure modeling, such as cyber-physical power grid, along with crew transport network.
- It can interface with Real-Time Simulation software through socket programming.

https://sgdril.eecs.wsu.edu/research-interests-and-grants/industrial-grade-products/pycanvass/
Measuring Resiliency

**Weather Center**

- Real-Time Weather Inputs
  - Real Time Weather Data (Temp, Humidity)
  - Temperature Forecast (next hour, next day, next week)
  - Event Forecast (next hour, next day, next week)
  - Precipitation Forecast
- Cyber-Physical Power Grid Data
  - Power Flow State Variables (f, P, Q, V, theta, p.f.) - Estimated/Observed
  - Network Communication State Variables - Bandwidth, Latency, Round Trip Time, Drop Rate
  - Diesel (DER) Generation Capacity
  - Solar Generation Rate, Battery SOC

**NOC**

- Historically Available Data and Semantic Information
  - Load Profile
  - Demand Schedule
  - Generation Schedule
  - Time of Day, Type of Day
  - Load Priority List
- List of Probability of an Event
  - WEATHER-RELATED
    - Tropical Storm
    - Wind
    - Hurricane
    - Solar Flares
    - Earthquake
    - Volcano
    - Mild Rain
    - Heavy Rain
    - Flooding
    - Snow
    - Hail
    - Blizzard
    - Freezing Temperature
    - Strong Winds
    - Cyclones
    - Solar Eclipse
  - CYBER
    - Data Packet Modification
    - Denial of Service
    - Bad Data Injection
    - Eavesdropping

**EMS/DMS**

- Data Pre-processing and Conversion to Logistic Variables
- Impact Metric
CyPhyR: Cyber-Physical Resiliency Tool
ANALYSIS & MANAGEMENT LAYER

Data driven and AI based learning decision support

Anomaly flag, Classification and root cause analysis

DETECTION MAKING LAYER

Operator Visualization/ Human Factor

DATABASE

Data for Training and Validation

PHYSICAL LAYER

GRID modeling

Protection Relays & Sensors

SENSOR LAYER

Control Center

CYBER

Network monitoring software

Protection software

SCADA

DATABASE

Cyber DATABASE

Cyber data Network packets

HYPERSIM

IncSys EMS

PINGThings

PMU DATABASE

Voltage and Current phase Frequency

3-phase voltage and current values Breaker status
Takeaway#1: Resiliency is a Complex Problem

- **Power Grid Resiliency**
  - Multiple switch
  - Macrogrid
  - Minigrid
  - Microgrid
  - Nanogrid
  - Graceful disintegration and interconnection
  - Flexible management and control of resources
  - Economic and market incentive

- **Flexible Infrastructure**
  - Communication
    - Authentication
    - Encryption
  - Computation
    - Access control
    - Attestation
    - Forensics
    - Patch management
    - Software Audits
  - System Management
    - Intrusion Detection
    - Event Monitoring/Analytics
    - Security Assessment

- **Secure Cyber Infrastructure**

- **Resilient Power Control Applications**
  - Generation
    - Automatic Generation Control
    - Governor Control
    - Automatic Voltage Regulation
    - Protection
  - Transmission
    - State Estimation
    - VAR Compensation
    - Protection
  - Distribution
    - Load Shedding
    - Protection
    - Advanced Metering Infrastructures

- Resiliency metric is a MCDM problem
- Resiliency is characteristics of the system
Data Analytics and machine learning approaches needs to be applied after analyzing the power system problem carefully. Finding match between machine learning strength and power system problem to be solved is important.

Machine learning is only applicable in data-rich problems if no system model is available (e.g. forecasting).

If model is available with rich data set, typically it will be two step approach: apply machine learning to narrow down your possible options and refine it with model based approach (e.g. event detection).

Machine learning will not give a good results based on state of the art for highly complex and dynamic problems (e.g. transient stability, contingency analysis).

Validation and metric is important for these evolving solution technologies.
Takeaway#3: Get Involved in PMU Data Analytics and Applications

- NASPI White Paper on Data Quality Requirements for PMU based Control Applications
- IEEE Synchrophasor based Power Grid Operation as part of Bulk Power System Operation. White paper on a) Challenges and Solutions in Implementing PMU based Applications in Control Center) and b) Quality-Aware Applications
- https://sgdrl.eecs.wsu.edu/workshop_conferences/real-time-data-analytics-for-the-resilient-electric-grid/
Thank You

Acknowledgement: DOE, NSF, INL, Siemens, EPRI, PNNL, IEEE PES Big Data

Anurag K. Srivastava
anurag.k.srivastava@wsu.edu