LLNL-TR-854185

Fou(FOURCORN1

Lawrence Livermore National Laboratory

Grid Event Signature Library Analytics Report: Signature Matching Tool Development Efforts

Unclassified

Christabella Annalicia, Jhi-Young Joo

September 12, 2023

Disclaimer

Lawrence Livermore National Laboratory is operated by Lawrence Livermore National Security, LLC, for the U.S. Department of Energy, National Nuclear Security Administration under Contract DE-AC52-07NA27344.

Executive Summary

This report describes the purpose and features of the Signature Matching Tool (SMT), employed in the Department of Energy (DOE) Grid Event Signature Library (GESL). The SMT supports a user of GESL to identify snippets of electric signatures, usually from sensor devices measuring electric characteristics such as phase voltages and currents, frequency, etc., suspected to represent certain events in the power grid but are not known to the user. The SMT uses a classification method to identify an event of the unknown signature, using the repository of known and labeled signatures in the GESL. The classifier applies a local binary classifier per node (LCN) approach to the unique event tag taxonomy used in the GESL, where training phases are separated based on the Primary labels in the taxonomy, sensor type, and voltage level. Results show that this method helps with computing time during training, in comparison to a flat, multinomial classifier, and produces acceptable average accuracy of 83% across all Primary labels. The report concludes with planned future work including integration to the web interface and API.

Table of Contents

Section 1. Introduction **5**
Section 2. Methodology **6**
2.1 GESL Database 6
2.2 Classification Method 7
Section 3. Results **12**
Section 4. Conclusion and Next Steps **13**
Appendix A. References **14**

Table of Figures

Figure 1. Breakdown of GESL signatures based on sensor types (left) and breakdown of GESL signatures based on voltage level (right). **6**
Figure 2. Event tag subtree for 'Conditions' Primary label. **7**
Figure 3. LCN approach applied on a subtree under the Events Primary node. The red, dashed lines represent the nodes that have RF binary classifiers trained the labels of that node. **8**
Figure 4. Example format of input signature $x(t)$ of length N. **8**
Figure 5. Determining Secondary label for an input signature $x(t)$. **10**
Figure 6. Determining Tertiary label for an input signature $x(t)$ according to predicted $Spred$. **10**

Table of Tables

Table 1. Accuracy results of multi-level classifier applied on transmission PMU data, rounded up to 1 decimal place. **12**

Section 1

Introduction

The Grid Event Signature Library (GESL) [1] is an open-source database that contains a wide variety of gird event signatures, recorded from anonymized sources coming from the electric power grid. The GESL is the product of the joint collaboration between Oak Ridge National Laboratory (ORNL) and Lawrence Livermore National Laboratory (LLNL) under Department of Energy Office of Electricity (DOE OE) program, in the hopes that more power grid data is available for use in research projects and educational fields centered around artificial intelligence and machine learning (AI/ML) applications.

The Signature Matching Tool was initially developed in the older version of the GESL where users input their signature in a comma separated values (csv) format and are shown GESL signatures that are similar to it. This process is done by matching an input signature to GESL signatures that have similar statistical characteristics, bounded by a predetermined threshold.

However, due to the diversity of the signatures within GESL, which hosts signatures recorded by different sensor types located at various locations in a transmission and distribution (T&D) network, a threshold-based statistical matching method may not be the best for the Signature Matching Tool. Furthermore, some signatures may result in similar statistical characteristics even though they are drastically different; for example, the current of an arcing fault is sometimes not differentiable from a normal load current.

In recent years, Machine Learning (ML) methods observe a rising trend in the power systems field due to its advantages over traditional methods. Therefore, a supervised ML method is adopted in this task, where the Signature Matching Tool becomes a classification problem to classify labels for a set of input features. This report is structured as follows: Section 2 describes a background on the classification problem, including the dataset used, a breakdown on the ML classification method used, and the expected output; Section 3 shows the results of the ML classification method when applied to a testing dataset; finally, Section 4 summarizes the report, including ideas for next steps.

Section 2

Methodology

2.1 GESL Database

The GESL is a database of 2,634 power grid event signatures that are obtained from ten different providers; these signatures came from different voltage levels and sensor types, recording different metrics such as voltage, current, frequency, and acoustics. **Figure 1** shows a breakdown detailing where the signatures came from, where **Figure 1 (left)** shows the breakdown by sensor type, which includes:

- Phasor data, collected from sensors such as:
  - Phasor Measurement Units (PMUs), recording synchrophasor data.
  - Frequency Disturbance Recorders (FDRs), which is a type of low voltage sensor recording from the customer end.
- Waveform data, collected from sensors such as:
  - Point-on-Wave (POW) or generic waveform sensors.
  - Optical sensors, which records waveform data in the same form similar to generic POW sensors.
  - Other types of sensors, including Ultra High Frequency (UHF) sensors, high-frequency current transformers (HFCTs), power/current transformers (PT/CT), and acoustics emissions sensors.

Figure 1 (right) shows the breakdown according to voltage level, i.e. low voltage (LV), medium voltage (MV), and high voltage (HV).

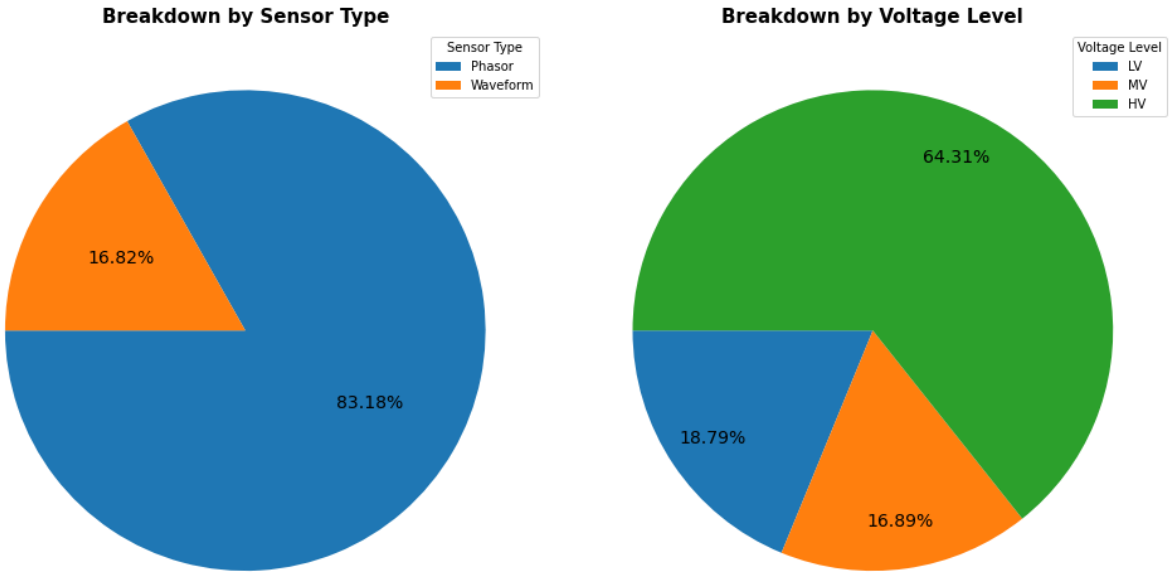

Figure 1. Breakdown of GESL signatures based on sensor types (left) and breakdown of GESL signatures based on voltage level (right).

GESL is a grid data repository[1] that is an open-source database with signatures that were obfuscated, so as to retain the anonymity of providers; this allows GESL to remain open-source and available to use for everyone. GESL signatures also adopt a global format such that there is consistency with how they are all formatted, allowing ease of use for users who want to compare data between two different providers. Finally, the GESL has a unique labeling scheme, herein referred to as the event tag taxonomy, where each signature was labeled by a subject matter expert (SME); this event tag taxonomy is structured in a hierarchical format of three-levels: Primary, Secondary, and Tertiary. Using this taxonomy, signatures can have one or more labels to describe it. This top-down taxonomy allows the event tag taxonomy to be scalable such that it can expanded upon when signatures with new labels are ingested into the database. **Figure 2** shows a breakdown of a subtree of the event tag taxonomy, showing the labels within the 'Conditions' Primary label.

Figure 2. Event tag subtree for the 'Conditions' Primary label.

2.2 Classification Method

The classification problem that the Signature Matching Tool is aiming to solve is hierarchical and multi-label in nature. The event tags taxonomy, which contains the many labels that describe each signature, are structured in a hierarchical manner, represented by a tree with three levels. In this work, the Primary label is set to be the root nodes, i.e. there are five separate trees representing the five Primary labels: **Events**, **State**, **Equipment**, **Conditions**, and **Phase**. Each signature in the GESL, or each feature set, may have more than one label describing it, meaning that the classifier must set multiple labels for each signature when it is classifying [2].

In order to solve a hierarchical, multi-label classification problem, the authors of [3] review multiple hierarchical classifiers across different application domains, where several classifiers can accommodate a

[1] There are other types of data repositories of grid events, such as the Electric Power Research Institute (EPRI) Disturbance Waveform Library (https://pqdl.epri.com) or the real-time streaming repository National Infrastructure for AI on the Electric Grid (NI4AI) developed by PingThings.

multi-label factor. From the classifiers reviewed in the paper, the local binary classifier per node (LCN) approach is adapted for the Signature Matching Tool.

The LCN approach is described at a high-level in **Figure 3** below, where in each node in the event tag taxonomy excluding the root (Primary) nodes, a binary classifier is trained on the node label, represented by the red, dashed lines. (Note that here, **Figure 3** shows the LCN approach applied on a subtree of the 'Events' Primary label.) A few classifiers were used for this work:

1. Random Forest (RF): 1,000 number of decision tree voters of depth 8 **(RF_8)** and depth 10 **(RF_10)** for each RF.
2. Support Vector Machine (SVM): Radial basis function (RBF) **(SVM_RBF)** kernel is used, which is the default setting in the `scikit-learn` [4] Python package used in this classification work.
3. (Categorical) Naïve Bayesian (**NB**).

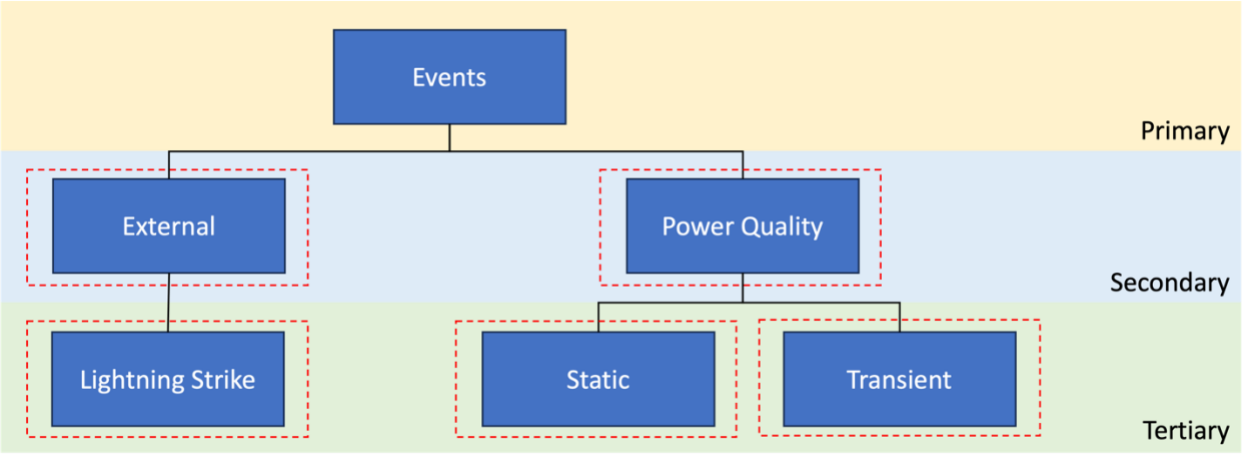

Figure 3. LCN approach applied on a subtree under the Events Primary node. The red, dashed lines represent the nodes that have RF binary classifiers trained the labels of that node.

After that, at a high-level, an input signature is passed into binary classifiers successively at each level until an appropriate label for each Primary label is chosen. An advantage of the LCN approach is that the labels that do not have the highest probability in classification will be disregarded as the classifier goes down each level, i.e. if the classifier does not classify an input signature as 'Power Quality', then the labels associated with 'Power Quality' ('Static' and 'Transient') will be disregarded.

Possible modes of input

Timestamp	Voltage	Current	Frequency
t_0	V_0	I_0	f_0
t_1	V_1	I_1	f_1
...	...	...	...
t_{N-1}	V_{N-1}	I_{N-1}	f_{N-1}

Figure 4. Example format of input signature $\mathbf{x(t)}$ of length N.

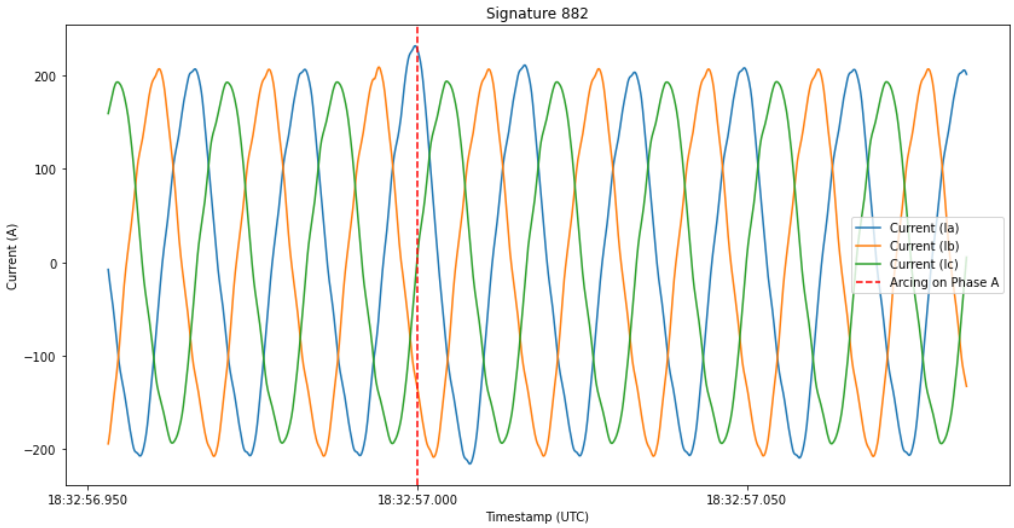

Figure 5. Example format of Signature 882, pulled from the GESL.

The processes are highlighted in more detail below.

1. An unlabeled signature, $x(t)$, of length N is inputted in the form shown in **Figure 4** (**Figure 5** shows an example of how $x(t)$ looks like, pulled from the GESL). The modes can vary for each signature, but the required modes are voltage and frequency; for three-phase voltages and currents, each channel is considered separately. A timestamp column is present in all GESL signatures but is not required for the Signature Matching Tool.
2. The unlabeled signature is preprocessed:
    a. Values are standardized to have a mean of 0 and standard deviation of 1.
    b. If phase angles are included in the input, then assume that the phase angles are not unwrapped. Then, phase angles are unwrapped in the preprocessing stage.
    c. Calculate mean, variance, skewness, and kurtosis (collectively referred to as statistical moments) of each mode in the input signature. These will be the feature set for the input signature.
3. The feature set is passed to a multi-level binary classifier for a single Primary node (Events, State, Equipment, Conditions, and Phase). The process for choosing a Secondary label is highlighted in the sub-steps below and in **Figure 6**.
    a. Input feature set to each binary classifier in the Secondary level representing the labels $S_1, S_2, \dots, S_{N_{Sec}}$, where N_{Sec} is the total number of Secondary labels under the chosen Primary label. Each binary classifier outputs the class probability of itself, i.e. each binary classifier will output the probability of the input feature set to be classified as the positive or 'True' label of itself, denoted by $p(True_{S_1}), p(True_{S_2}), \dots, p(True_{S_{N_{Sec}}})$.
    b. The Secondary label that has the highest probability, S_{pred}, will be chosen by using the equation:

$$S_{pred} = argmax\{p(True_{S_1}), p(True_{S_2}), \dots, p(True_{S_{N_{Sec}}})\}$$

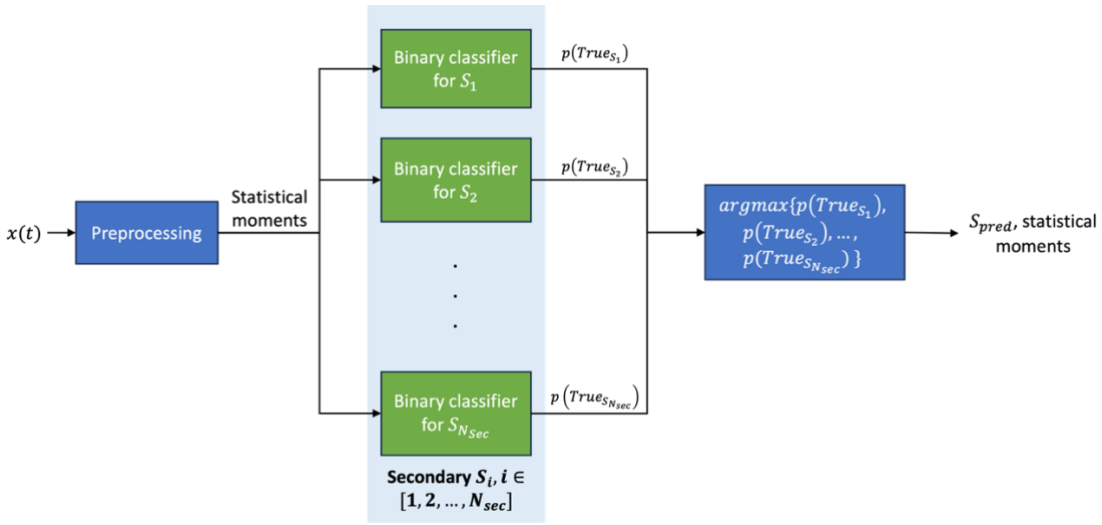

Figure 6. Determining Secondary label for an input signature x(t).

4. Input S_{pred} and the feature set to the next step in the multi-level binary classifier. This process is to choose the Tertiary label of the signature, highlighted in the sub-steps below and in **Figure 7**.
   - **a.** Input feature set to each binary classifier in the Tertiary level representing the labels $T_{S_{pred},1}, T_{S_{pred},2}, \dots, T_{S_{pred},N_{T\subset S_{pred}}}$, where $N_{T\subset S_{pred}}$ is the total number of Tertiary labels under S_{pred}. Similar to before, each binary classifier outputs the class probability of itself, denoted by $p\left(True_{T_{S_{pred},1}}\right), p\left(True_{T_{S_{pred},2}}\right), \dots, p\left(True_{T_{S_{pred},N_{T\subset S_{pred}}}}\right)$.
   - **b.** The Tertiary label has the highest probability, T_{pred} is chosen by the equation:

$$T_{pred} = argmax\left\{p\left(True_{T_{S_{pred},1}}\right), p\left(True_{T_{S_{pred},2}}\right), \dots, p\left(True_{T_{S_{pred},N_{T\subset S_{pred}}}}\right)\right\}$$

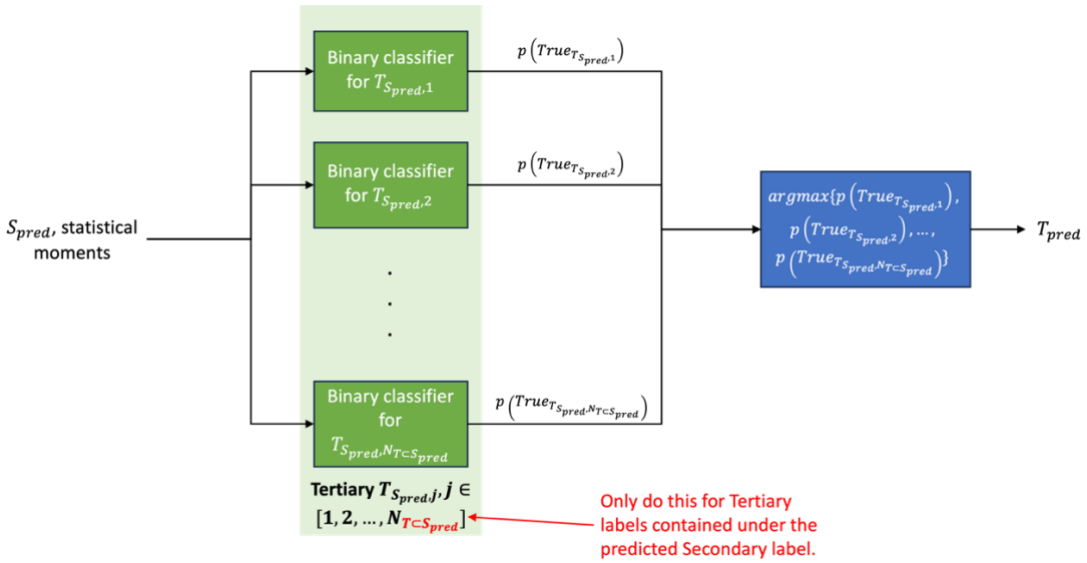

Figure 7. Determining Tertiary label for an input signature x(t) according to predicted S_{pred}.

5. Repeat Steps 3 and 4 for all Primary labels.

To verify the accuracy of the multi-level binary classifier, data from the high-voltage transmission PMU level is used, as it contains the highest number of signatures in the GESL. The training and testing datasets are separated into a 95% - 5% ratio, where that training and testing are repeated for each Primary label.

The accuracy of each Primary label classification is calculated using the equation below:

$$Accuracy = \frac{Number\ of\ correctly\ predicted\ Tertiary\ label}{Total\ number\ of\ signatures\ in\ the\ training\ dataset} \times 100\%$$

Section 3

Results

The accuracy results of the classifier, applied on transmission PMU data, is shown in **Table 1**.

Table 1. Accuracy results of multi-level classifier applied on transmission PMU data, rounded up to 1 decimal place.

Primary Labels	Accuracy (%)			
	RF_8	RF_10	SVM_RBF	NB
Events	80.2	**82.4**	59.6	**41.1**
State	**100.0**	**100.0**	97.0	**78.7**
Equipment	**75.3**	76.3	78.2	**78.6**
Conditions	79.7	**87.3**	**48.1**	**48.1**
Phase	**80.0**	74.0	40.0	**36.0**

In general, the RF classifiers, regardless of depth, works better than other classifiers for most Primary labels. The Naïve Bayesian classifier, on the other hand, did the worst in classifying most Primary labels, but did the best in classifying signatures under the Equipment primary label.

Section 4

Conclusion and Next Steps

A LCN approach is utilized in solving a hierarchical, multi-label problem, thereby defining the Signature Matching Tool. Different classifiers are used to test the efficacy of each one under different Primary labels, where the Random Forest classifier is found to be the best in classifying most Primary labels.

Following up on this work, efforts in transitioning the current code to a web-based version are underway such that the code can be integrated into the GESL website and Application Programming Interface (API). To achieve this, there are several questions/considerations to address to ensure a smooth transition:

1. **Input formatting** – What is the specific format required by input files?
    - For example, voltage and/or frequency modes are required in the input file, but should they be labeled in a specific format?
    - Is the input a phasor measurement or waveform measurement?
    - Is the input the raw measurements or is it pre-standardized?
    - How do we choose the appropriate classifiers if the users wish to anonymize their inputs? Currently, classifiers are divided and trained based on sensor type and voltage level; how do we ensure input files are inputted to the correct class of classifiers if they are anonymous?
2. **Output** – Will the output of the Signature Matching Tool only contain predicted event tags? If not, what are other useful outputs that the Signature Matching Tool should give?
3. **Classifier improvements**
    - Should other complex classifiers be used for the Signature Matching Tool? For example, SVM with other kernels, such as linear and polynomial kernels, are not used due to high computational time, but it may be effective to have them trained and check for accuracy.
    - If different classifiers are used for each Primary label, what is a good validation metric to encapsulate the performance of the Signature Matching Tool as a whole?

Furthermore, for the next fiscal year, the team will be working in collaboration with ORNL to develop visualizations to help distinguish the various AI/ML applications that can be performed using the GESL signatures.

Appendix A

References

[1] Oak Ridge National Laboratory, Lawrence Livermore National Laboratory, Grid Event Signature Library, 2023.
[2] M. -L. Zhang and Z. -H. Zhou, "A Review on Multi-Label Learning Algorithms," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1819-1837, Aug. 2014, doi: 10.1109/TKDE.2013.39.
[3] C. N. Silla Jr. and A. A. Freitas, "A survey of hierarchical classification across different application domains," April 2010.
[4] Pedregosa et al., "Scikit-learn: Machine Learning in Python," in *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.