A MACHINE LEARNING APPROACH TO CHARACTERIZING AND DETECTING ELECTRONIC DEVICES

Cassie Seubert and David Daigle
Schweitzer Engineering Laboratories, Inc.
2545 NE Hopkins Ct., Pullman, WA 99163

ABSTRACT

The unauthorized use of computers and electronic devices by foreign intelligence and disgruntled insiders poses a threat to the government. However, the consumption of energy by electronic devices induces measurable signatures back into the local power system. The detection and monitoring of these signatures provides a method of identifying unauthorized equipment use and potential counterintelligence problems.

Previous works in this field focus on distinguishing between devices with a wide diversity in loads. We advance this research by focusing on nearly identical devices in the same class. By using device startup transient features and machine learning algorithms, we show that it is possible to identify unique devices in an electrically noisy office setting. This advances the state of the art and enables a more robust detection algorithm that is well-suited for counterintelligence efforts.

Keywords: Non-intrusive load monitoring, switch mode power supplies, machine learning classification, load signatures, startup transient

1 INTRODUCTION

The operation of electrical devices is detectable by observing a device's energy consumption. In this paper, we will demonstrate that monitoring the signatures induced back onto the power system by electrical devices is a valid counterintelligence method for monitoring spaces.

The process of using electrical signatures to detect and classify particular devices on the power system is called Non-Intrusive Load Monitoring (NLM), or load disaggregation, and has been an area of research for over three decades [1]. However, the research to date has several limitations that make application to counterintelligence problems difficult. Traditionally, the research has focused on identifying typical household loads in a residential environment. The devices that are usually compared in NLM research are those with disparate circuitry and uses, such as ovens, refrigerators, computers, and other household appliances [2] [3] [4] [5] [6] [7] [8] [9] [10]. Generally speaking, these appliances are easy to detect and characterize using low-fidelity measurements and simple algorithms. In addition, the total load present in houses is typically very low in noise amplitude compared to industrial or commercial loads. These qualities of household devices and loads make disaggregation reasonably intuitive.

In contrast, a generic office building will have loads dominated by power electronic devices, including a wide variety of computer equipment. Power electronic devices generate their own electromagnetic interference and filter out noise from other devices. When many of these devices are present on a system, they can make certain detection methods challenging—particularly those that rely heavily on characterizing high frequency signatures. In this environment, device signatures are difficult to detect, and therefore many of the proposed signatures in the NLM literature will be below the noise floor. Finally, the electrical distance between the sensor and the device is much smaller in a residential space than in a commercial building. High-frequency device signatures attenuate over distance, so NLM at a distance complicates the problem.

We address these gaps in the NLM research by developing a method to classify a variety of nearly identical devices that is robust enough to perform well in a noisy electrical environment. Switch mode power supplies (SMPS) are a class of electronic devices that are used to power a variety of appliances in both the commercial and residential spaces. Example applications of SMPS include powering computer equipment, cell phone chargers, and LED lighting. In this research, we focused on classifying a sub-class of power supplies that are used to power industrial equipment. We chose this class of power supplies because products in this sector are similar between manufacturers, allowing us to fulfill the goal of classifying devices that are nearly identical.

After collecting load data from these devices in a controlled environment, we used feature engineering to construct distinct but noise-resistant signatures for each device we tested. Our algorithm identifies which power supply is operating inside a small commercial facility with over 96 percent accuracy. Our systematic approach demonstrates that even devices with nearly identical specifications can be detected and distinguished on a noisy power system. This success is a promising step toward using load signatures to monitor spaces, establish baselines, and detect unusual behavior.

This paper is organized as follows: Section 2 will provide a technical background to NLM and examine in detail prior research in this space. Section 3 will outline the details of our data collection process and our analytical approach. In Section 4 we present our results, and in Section 5 we discuss our conclusions.

2 BACKGROUND

2.1 ELECTRICAL SIGNATURES

Electrical devices connected to the power grid draw current that induces detectable, measurable signatures onto the power system. These measurable effects are dependent on the device circuitry. For example, an ideal resistive load draws current that is linearly proportional to the voltage applied across its terminals. This ideal case induces few distinguishing features. On the other hand, devices that contain active electronics draw currents in a complex and time-varying fashion, yielding waveforms rich with features. The basis for non-intrusive load monitoring is that real-world loads induce distinct signatures based on each device's unique circuitry.

Voltage and current measurements can both be used to sense a device's energy consumption signature; however, this study uses current because it provides better localization. A measurement of voltage at a wall outlet on one branch of a building electrical system can theoretically sense device signatures from all branches in the network. On the other hand, a measurement of the current flowing through a branch will only contain information about the devices connected to downstream points, thus providing better localization.

2.2 PREVIOUS WORKS IN NON-INTRUSIVE LOAD MONITORING

NLM research began in the 1980s [1]. In its original form, devices were fingerprinted by observing the change in a household's power consumption when the devices turned on or off. This method was adequate to identify and classify drastically different devices all with relatively large loads. A plot of power consumption versus time from Hart's original research is shown in Figure 1. However, this method has major drawbacks. As Hart asserted in his original research [1, p. 1887], "The NALM has a restricted set of target appliances, as it is not currently suitable for testing very small devices… nor can it distinguish between electrically identical appliances".

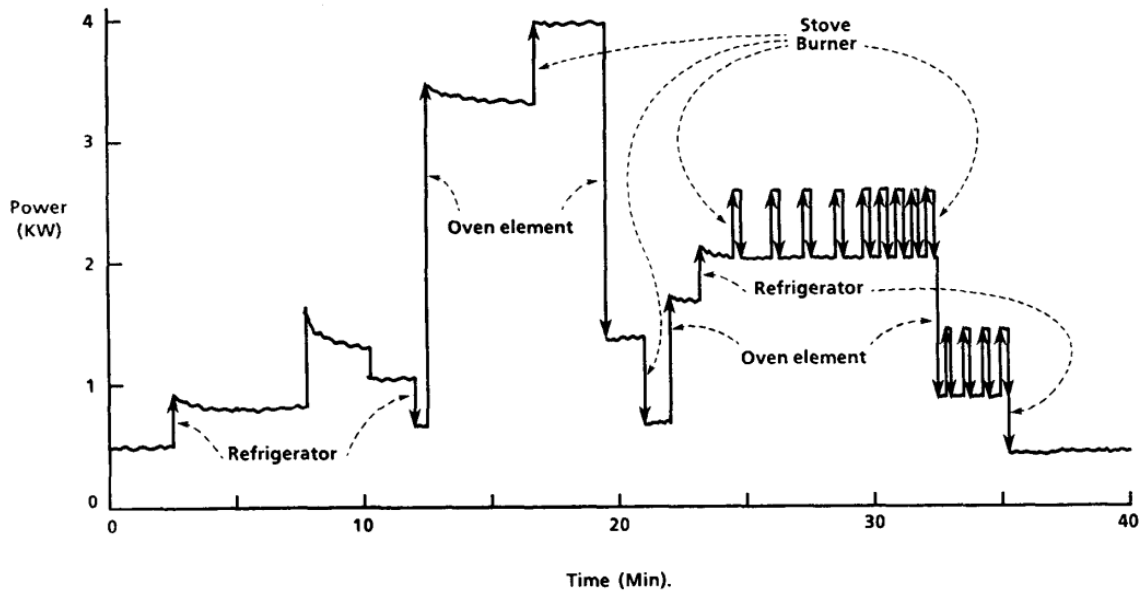

Figure 1 Hart's plot of power consumption versus time that shows load changes due to devices operating on the system. Presented in [1].

Since the 1980s, research in this field has advanced considerably. There has been a substantial effort to improve the robustness of NLM algorithms by using different features present in the electronic signatures and employing more sophisticated pattern matching algorithms. Much of the research focuses on using features present during a device's steady state operation. Steady-state features used have included current draw, phase difference, real power, reactive power, harmonic distortion, and high frequency distortion [2] [4] [5] [6] [11] [12] [13] [14].

Steady state features have a few limitations to their usefulness in detecting devices on real systems. These features are difficult to decouple when more than one device is operating at a time. One infeasible solution to this problem is to collect signatures of all possible device combinations [15]. However, this quickly becomes impractical for even a small collection of devices. Another limitation is that high-frequency based signatures are difficult to detect in many cases due to attenuation in realistic test cases, which we demonstrate in Section 3.

Some research has focused on extracting startup transients for devices as signatures [2] [4] [6] [7] [8] [10] [1] which we assert is a much more robust approach for distinguishing between lower-power devices in a non-residential environment. Startup transient signatures are typically quick, and the likelihood is low that two devices will turn on at the exact same time. This makes identifying devices when more than one is operating much easier than with steady-state features. Additionally, startup transient signatures are detectable even in electrically noisy environments and from a distance.

Other research that investigates the use of startup transients for device identification focuses almost solely on typical residential loads like air conditioners, ovens, or hair dryers [2] [4] [7] [8] [10] [16]. When studies do include devices like televisions or laptops, they usually only include one such device as part of a collection of other household appliances [4] [7] [8]. This research usually takes place inside a residence, which as discussed above is electrically cleaner, and therefore signatures are more easily identifiable.

Many pattern-matching techniques have been proposed to distinguish among electronic devices once the signatures are extracted. Some research uses optimization strategies to reconstruct an aggregate signal from the observed waveforms [2]. Other research applies machine learning classifiers such as naïve Bayes classifiers, support vector machines, hidden Markov models, or artificial neural networks [2] [3] [4] [7] [9] [10] [11] [14]. We will demonstrate that with features extracted using power engineering knowledge, simple and fast pattern matching algorithms can be powerful.

Our study is unique in its focus on differentiating between similar devices. We apply this research and demonstrate its effectiveness in a noisy test environment and at an appreciable distance. We also quantify the performance of a representative selection of machine learning classifiers to identify which algorithms perform best for this type of problem. For these reasons, we believe these techniques contribute to the state of the art of non-intrusive load monitoring.

3 METHOD

3.1 TEST SETUP

3.1.1 Devices

A total of ten industrial switch mode power supplies were chosen for classification. All power supplies convert a single phase 120 V 50/60 Hz input to a 24 V DC output. The selected power supplies are shown in Table I. During testing, each power supply was burdened with a 7.5 Ω resistive load to maintain an equal level of loading for each device.

TABLE I
LIST OF TESTED DEVICES

Manufacturer	**Model**	**Rated Output Current**
Delta	EOE12010005	10 A
Meanwell	DRP-240-24	10 A
Omron	S8VK-G24024	10 A
Phoenix Contact	QUINT-PS-100-240AC/24DC/10	10 A
Phoenix Contact	QUINT-PS-100-240AC/24DC/5	5 A
Schneider	ABL8REM24050	5 A
Schneider	ABL7RP2410	10 A
Siemens	6ES7 307-1EA00-0AA0	5 A
TDK	DLP240-24-1/E	10 A
TDK	DPP120-24-1	5 A

3.1.2 Data Acquisition

We monitored the current using a Flex-Core M1V-50-5 50A/5V current transformer connected directly to a National Instruments (NI) 9220 analog-to-digital converter. The current transformer has a -3 db bandwidth of 30 Hz to 40 kHz. The NI-9220 uses a 16-bit, 100 kHz analog-to-digital converter with a ±10 V range. As configured, the test setup affords 3.05 mA/bit. This data acquisition platform allowed us to record currents at various places upstream from the device under test (DUT). The sensor locations are shown in Figures 2, 3, and 4.

The goal of this research is to be able to use characteristics from current measurements directly at the DUT to detect the presence of the device from an upstream location. Using machine learning terminology, the measurements at the DUT are called the training dataset and the measurements taken at other locations are called test datasets. We used the training dataset to build machine learning classifiers that can predict which devices are present in the test datasets.

3.1.3 Test Scenarios

Measurements of the device current were taken in the following three scenarios:

- **Training Dataset—**In the first measurement scenario, we measured the current directly entering the DUT. This is shown in Figure 2.

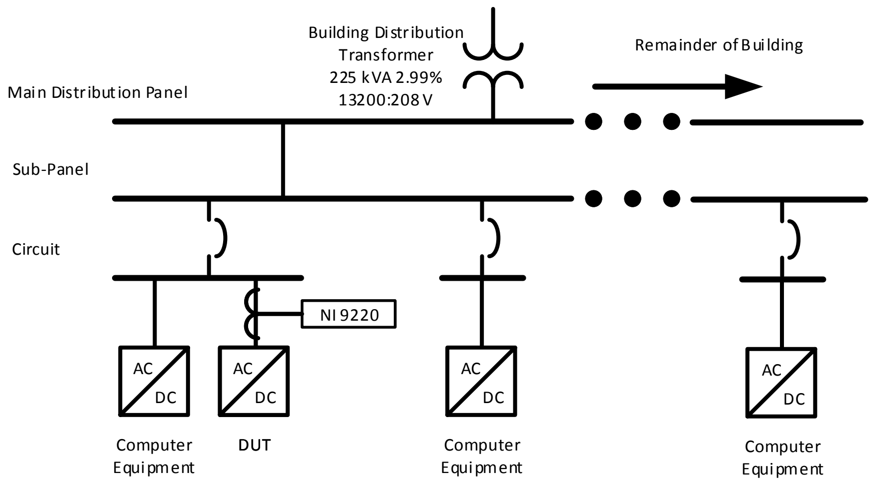

Figure 2 Sensor placed directly at the device under test. These measurements were used to collect signatures to train models.

- **Test Dataset 1—Sensor at Breaker Panel with 3.5 A Load:** In the second scenario, we measured the current at a large, three-phase breaker panel with current draw from other devices on the system measured at nominally 3.5 A RMS. The background loading consisted of four Dell desktop computers, two Dell laptops and their accessories, networking infrastructure, and other peripherals. This scenario was developed primarily to validate that the algorithm was working as expected. The DUT was connected downstream from the panel on the same phase the as the current sensor. This is shown in Figure 3.

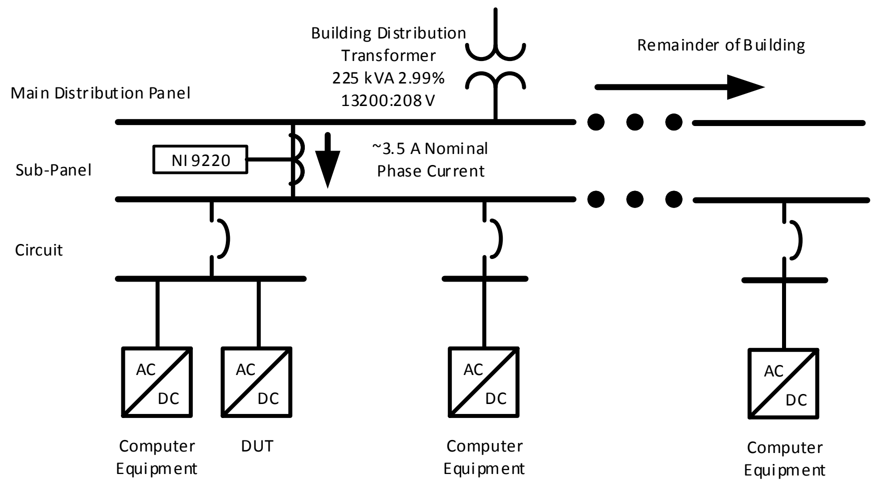

Figure 3 Sensor placed at breaker panel with 3.5 A load. These measurements were used to collect signatures to test the trained models.

- **Test Dataset 2—Sensor at Breaker Panel with 27 A Load:** In the third scenario, we measured the current at the same large, three-phase breaker panel, but with the current draw from other devices on the system measured nominally at 27 A RMS. The upstream load measurements consisted of the same loads as in Test Dataset 2 but with additional server and network equipment along with a 20 A resistive load to contribute to the loading. The loading of this scenario was equivalent to a small commercial facility, with the equivalent three-phase current being nearly eight times the average US household load. As in Test Dataset 2, the DUT was connected downstream from the panel on the same phase as the current sensor. This is shown in Figure 4.

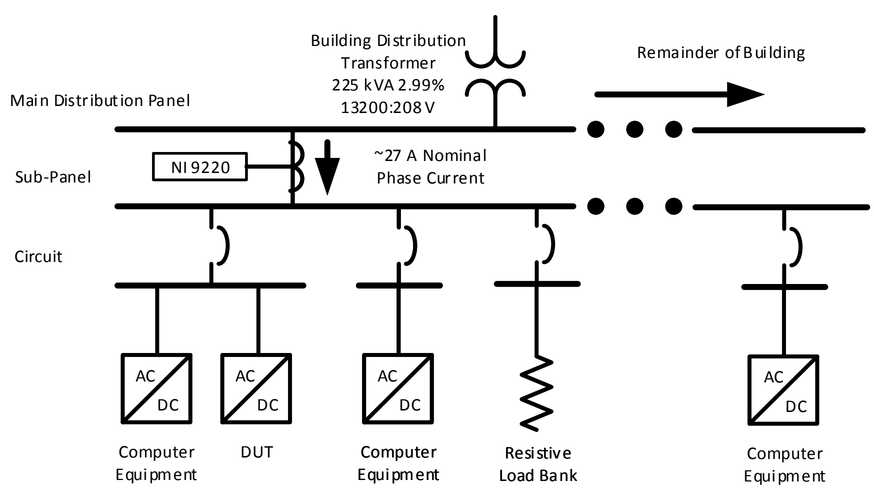

Figure 4 Sensor placed at breaker panel with 27 A load. These measurements were used to collect signatures to test the trained models.

Table II summarizes these scenarios.

TABLE II
LIST OF MEASUREMENT SCENARIOS

Dataset	**Measurement Location**	**Additional Load on System Captured in Measurements**
Training Dataset	Directly at device	0 A
Test Dataset 1	Upstream breaker panel	3.5 A
Test Dataset 2	Upstream breaker panel	27 A

3.2 FEATURE EXTRACTION

Features were extracted from the raw data to form device signatures. We investigated the use of high frequency steady state features and lower frequency startup transient features.

3.2.1 High Frequency Steady State Features

In this section, we demonstrate that high frequency features become less realistic to use as the system size grows. For these tests we used a Tektronix DPO 7104 oscilloscope sampling at 1 MSps in "high-resolution" mode instead of the National Instruments DAQ.

A common method of detecting switch mode power supplies is by analyzing high frequency interference produced by the power supply [13]. This interference is produced by the internal switching circuitry of the power supply, and since manufacturers implement these switching patterns in different ways, they can in some cases be used to identify the presence of particular power supplies. This method will resolve high frequencies when measuring current flowing directly into or slight upstream from the DUT. However, we found that these signatures rapidly attenuate as the test conditions become more realistic. Figure 5a shows the high frequency components of the current flow measured directly into a Phoenix Contact Quint 10 A power supply. In a follow-up test, we measured current at a nearby upstream breaker panel where there was nominally about 3.5 A of constant load from four idling Dell desktop computers and peripherals. The wiring between the DUT and the measurement location was roughly 40 feet. In this test case, the switching frequencies were practically undetectable, as shown in Figure 5b.

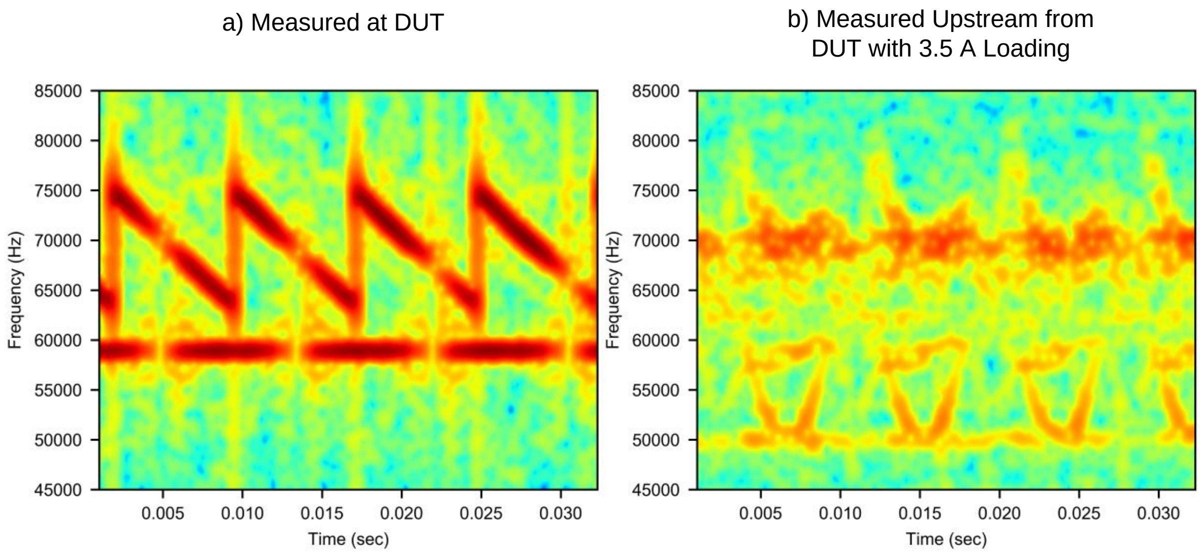

Figure 5 High frequency quantities measured directly into the Phoenix Contact Quint 10 A power supply shows high frequency switching interference (Fig. 5a). These high frequency switching patterns are attenuated and corrupted when measured upstream from the DUT (Fig. 5b).

There are three primary reasons for this. The first is due to impedance of the wiring between the measurement location and the DUT. The second is due to the front end filters of other power electronic devices (computers and peripherals) contributing to the attenuation. The final notable reason is that other power electronics with similar switching frequency ranges corrupt the measurement of the device under test. We concluded on this basis that high switching frequencies are a fragile detection method that is only useful in very specific test cases where there is minimal power electronic load present and minimal electrical distance between the measurement point and the devices being detected.

3.2.2 Startup Transient Features

Current inrush from the device's power-on creates a startup transient that is visible in the current waveform and is less susceptible to the aforementioned attenuation issues experienced with higher frequency measurements. Precise detection of these transients on a general system is an important problem that deserves its own research focus. Therefore, we chose to decouple event detection from this study. As a result, our event detector was tuned to work well to detect device transients on our test systems. A more broadly applicable detector would be needed to further this research on other systems.

Once a startup event was detected, we time-aligned the recorded data so that the 60 Hz fundamental frequency component started at zero degrees and then calculated RMS values for each of the first 200 cycles of the transient. Figure 6 shows an example of this processing on one startup transient for the Phoenix Contact Quint 10 A power supply. The extracted RMS startup transient forms a 200-dimension feature vector.

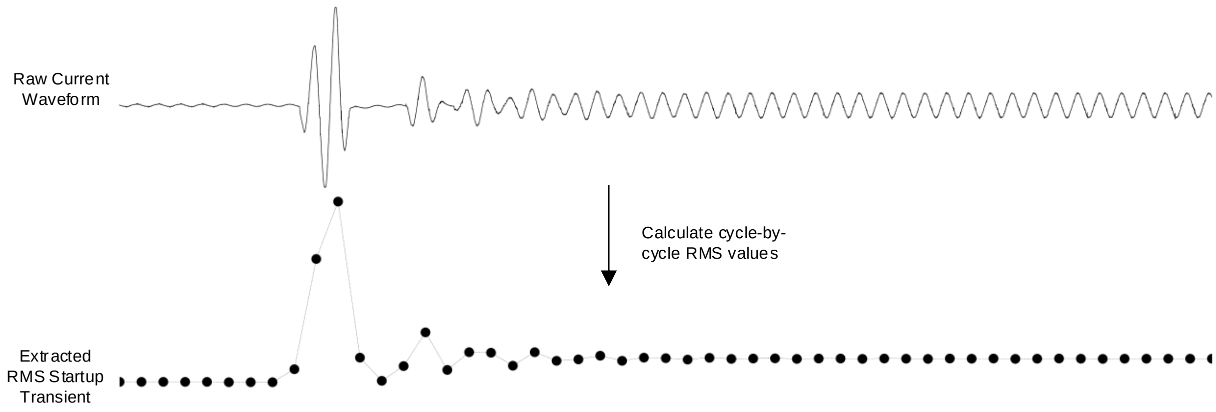

Figure 6 200-dimension feature vectors were obtained by calculating cycle-by-cycle RMS values from device startup transients. Shown are the first 50 calculated values.

We calculated the cycle-aligned RMS feature vectors as in Figure 6 for 100 startup transients for each device-scenario pair shown by Tables I and II. Figure 7 shows the first 50 cycles of these 100 feature vectors extracted from each of the 10 power supplies from the training dataset. Despite these power supplies having similar designs and identical loads, it is apparent that the startup transients differ widely.

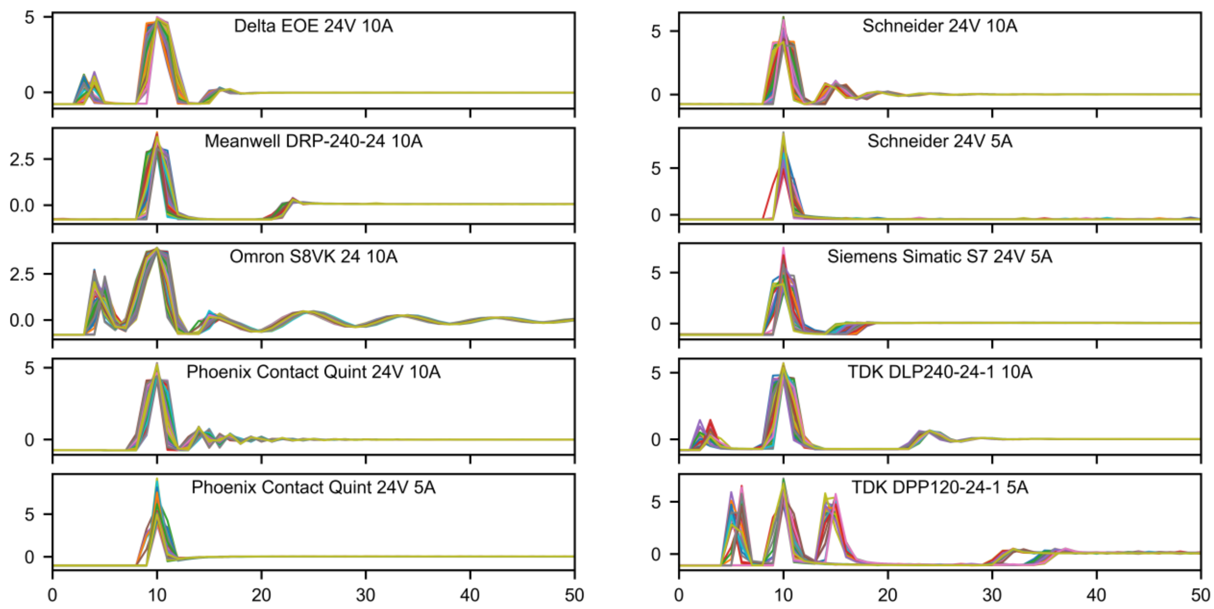

Figure 7 The first 50 cycles of the startup transient RMS values for each power supply measured directly at the device. All 100 repetitions are shown.

3.3 MACHINE LEARNING CLASSIFICATION

In this section, we give a short primer on machine learning, introduce some mathematical details, and outline our analytical approach to classifying device signatures.

3.3.1 Machine Learning Primer

Machine learning is the task of allowing mathematical algorithms to find distinct patterns in data. These algorithms can range from simple models like linear regression to complicated models like deep neural networks. Classification is a type of machine learning that aims to assign data to a class based on observed values of that class.

To robustly quantify which classification algorithms performed best, we completed analysis using six classification models: Linear Discriminant Analysis, Logistic Regression, Support Vector Machines, Decision Trees, Multi-Layer Perceptron Neural Network, and K-Nearest Neighbors. These models were chosen because they represent a selection of mathematically distinct models. In this section, we present a high-level explanation of each of these models.

For the following discussion, each individual 200-dimension startup transient feature vector will be referred to as a data instance. Each instance is paired with exactly one power supply label, which we will call a class. The goal of these classification algorithms is to model the mapping from data instances to class labels so that new, unseen data can be categorized according to that function. The observed data used to estimate that function are called the training data, and in this study are the measurements taken directly at the device. The data used to evaluate the performance of the estimated classifier are called the test data, and in this study are the measurements taken at the breaker panel.

Linear Discriminant Analysis
Linear Discriminant Analysis (LDA) aims to maximize the probability that an instance belongs to a class given the observed data for that instance by using Bayes' theorem. LDA assumes that the distribution of observed data given membership to a class follows a multivariate Gaussian distribution and additionally assumes that the covariance matrix for that class is the same for all classes. These assumptions yield linear decision boundaries between classes which depend on the multivariate Gaussian parameter estimates obtained from the observed data [17].

Logistic Regression
Logistic Regression (LR) is similar to the well-known linear regression which takes an m dimensional feature vector input and maps it to a dependent continuous variable. When using linear regression, the goal is to find the parameters that describe the line that best fits the data. Logistic regression is similar, but instead turns the unbounded linear relationship into a classifier by transforming the modeled line into a bounded logit function. A decision boundary is then selected such that a feature vector is assigned to a class if the modeled logit function is above that threshold. The goal is to find the set of parameters describing the transformed line that best maps the data to the correct class [17].

Support Vector Machines
In its most basic form, the Support Vector Machine (SVM) classifier is a linear classifier that finds the hyperplane that best separates the data. For simplicity, if we assume the data instances belong to one of two classes and consist of only two dimensions so that the data can be visualized on the coordinate plane, then the SVM finds the line that separates the two classes so that the distance between the line and the

data instances is maximized. This line is found by using simple vector calculations and quadratic optimization that easily generalizes to higher dimensions.

SVMs are powerful classifiers because they can easily be modified to solve nonlinear classification problems. It is often the case that a collection of data instances cannot be separated by a hyperplane. In this situation, we can project the data into a higher dimension where the data is linearly separable and solve the classification problem in this higher dimension [17] [18].

Random Forest

A Random Forest (RF) classifier is an ensemble of Decision Tree (DT) classifiers. A DT is a classifier that breaks down the classification problem into answering a series of yes/no questions. These questions can be easily visualized in a tree structure, hence the name.

The crux of using a DT classifier is to construct a tree that is split on features that give the most information. This is done by maximizing the information gain of a feature which is a function of entropy. DTs are useful in that the classification results are easily explainable by examining the tree structure; however, they are prone to overfitting. An RF is used to mitigate the effects of overfitting in DTs. The RF classifier works by computing k decision trees, each built using a different random subset of data instances. The final classification is determined by looking at what a majority of the DTs decided [18].

Multi-Layer Perceptron Neural Network

A single-layer perceptron is a simple binary classification model. As shown in Figure 8, feature values are input into a directed graph and multiplied by weights, summed together, and then given a binary classification based on a threshold value. The goal in training a perceptron network is to find the value for the weights that classifies the training data correctly as many times as possible.

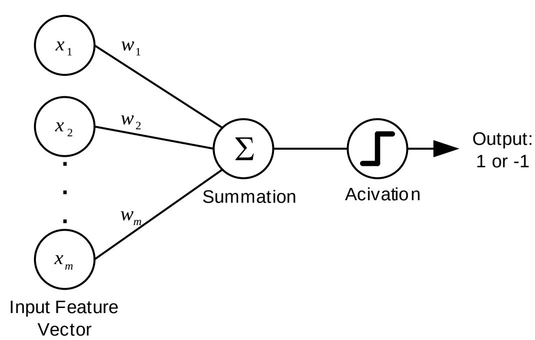

Figure 8   Example of a single layer perceptron binary classification model.

A multi-layer perceptron neural network (NN) is just a generalization of this concept. Figure 9 shows a generic neural network [18].

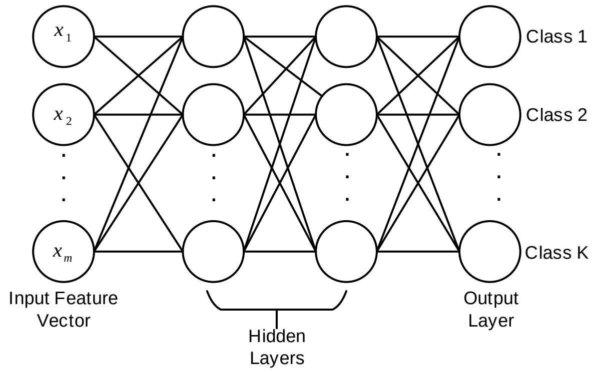

Figure 9 Example of a generic multi-layer perceptron neural network with two hidden layers.

K-Nearest Neighbor

K-Nearest Neighbors (KNN) is one of the more intuitive classifiers; however, it is distinctly different from the above classifiers. Rather than building a model from observed data to classify unseen data, KNN will use all the observed data to assign a label to a new data point each time. For each new data point, KNN finds the closest k observed data instances and assigns a label based on the majority label among those k instances. Despite its simplicity, KNN is often one of the best performing classifiers because it can learn complicated, nonlinear decision boundaries. However, because KNN requires all of the training data each time it classifies a new point, the memory requirements and computation time can be limiting [18].

3.3.2 Machine Learning Implementation

The above classifiers were fit using the training data and then evaluated using the testing data as shown in Figure 10. In this section, we describe the implementation details of building and scoring these machine learning classifiers.

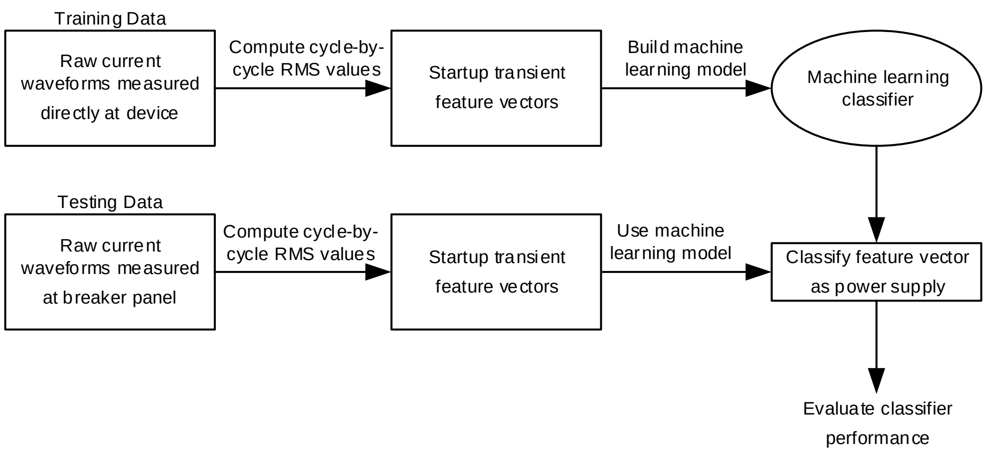

Figure 10 High-level overview of the machine learning process for this study. Cycle-aligned RMS features are extracted from the raw current startup transients. A machine learning model is built from the training data and then evaluated against the testing data.

Parameter Tuning

Each of the models described above requires hyperparameters that are input to the model. For example, the K-Nearest Neighbors classifier requires the number of nearest neighbors to be chosen before the model is fit. Finding the parameters that perform best for a given classification problem can be done by evaluating the model performance for a range of possible parameter values and then selecting the best performing value. This process is called parameter tuning and is shown in Figure 11.

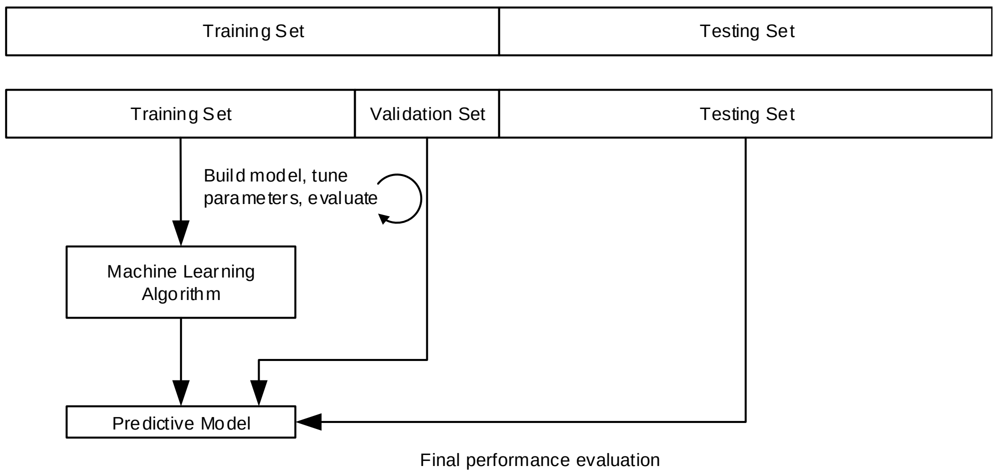

Figure 11   Machine learning classification model fitting process [18].

Scoring

There are several ways to compare the final performance of each model on the test data. Figure 12 shows a simple confusion matrix for a binary classification problem.

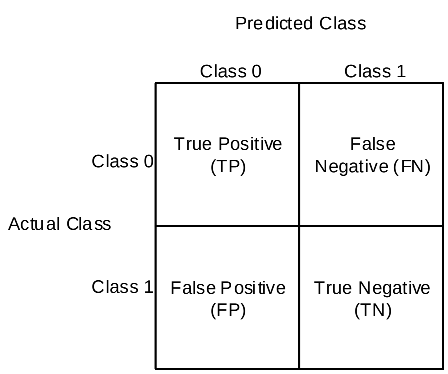

Figure 12   Confusion matrix for a simple binary classification problem.

Accuracy is an intuitive metric that calculates the total number of true positives and true negatives over the total number of instances.

$$ACC = \frac{TP + TN}{FP + FN + TP + TN}$$

This metric is a good choice when the number of instances in each class is relatively equal and when the impact of a false positive is similar to the impact of a false negative. For example, when determining if a patient has cancer, a false negative is much more severe than a false positive. Other metrics such as precision and recall score the impact of false positives and false negatives separately, and there are combined metrics such as the f1-score that compute a weighted average of precision and recall [18].

However, because we have an equal number of instances in each class and are not concerned with weighing false positives and false negatives differently, accuracy is a good measure to evaluate classifier performance for our problem and is used to report results in the next section.

Dimensionality Reduction

The transient features extracted for our analysis consist of 200 values for each turn-on event. In some cases, this number of features for each instance could result in over-fitting of the classifier. Additionally, there could be problems with memory, storage, and computation time when using this much data. Therefore, we evaluated the performance of each of the classifiers on both the raw data and on a reduced dataset using Principal Component Analysis (PCA).

PCA is a technique that transforms feature vectors in m dimensional space to d dimensional space where $d << m$. This is achieved by identifying which orthogonal directions in m dimensional space account for the most variance in the dataset, and then projecting those orthogonal vectors into d dimensions. PCA effectively gets rid of highly correlated features and results in a much more succinct dataset [18].

To perform PCA, we first compute the eigenvalues and eigenvectors of the covariance matrix of the feature vectors. The eigenvectors determine a new feature space for transforming the data, and the eigenvalue associated to each eigenvector describes the magnitude of that component. We can then select the top d eigenvectors for the principal component decomposition.

To select a good value for d, sort the eigenvalues in order of decreasing magnitude and compute the amount of variance explained for each eigenvalue by dividing by the sum of all the eigenvalues [18]. Figure 13 shows the results of this computation on the training data. It is clear that approximately 12 components explain a vast majority of the variance among the 200 dimensions of the original feature vectors. Thus, we chose to decompose our data to 12 features to evaluate the performance of a reduced feature space.

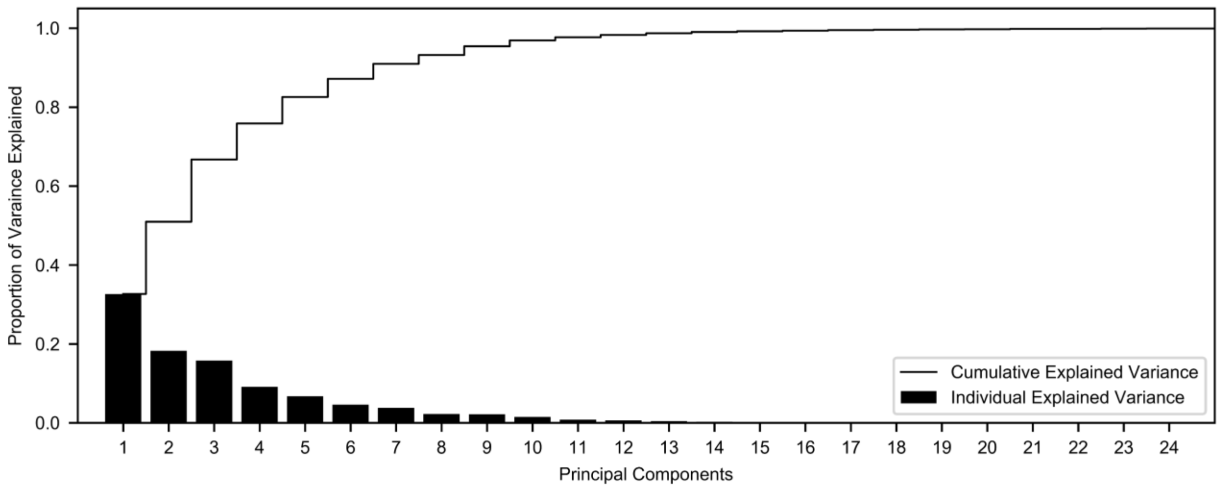

Figure 13 The proportion of variance explained by each principal component found using the PCA technique.

4 Results

We used the measurements taken directly at the device to train and parameter tune each of the six machine learning models described in Section 3. Each model was built from the raw training dataset, which contains 200-dimension feature vectors, and from the dimensionality reduced training dataset, which contains12-dimension feature vectors. This section presents the results of testing those models against the test datasets.

4.1 Test Dataset 1: Sensor at Breaker Panel, 3.5 A Nominal Load

Table III shows the classification results of each of the six classifiers built from both the raw and dimensionally reduced feature vectors from Test Dataset 1 (Figure 3).

TABLE III
CLASSIFIER ACCURACY ON TEST DATASET 1

Classifier	**Accuracy**	
	Raw	**PCA**
LDA	.805	.915
LR	.997	.947
SVM	.997	.943
RF	.669	.887
NN	.994	.923
KNN	.997	.965

Many classifiers reach over 99 percent accuracy when using the raw dataset and over 90 percent accuracy when using the PCA-reduced dataset. The presence of other devices on the system does little to attenuate or corrupt the startup transients of the devices of interest.

Figure 14 shows confusion matrices of the best performing classifiers using the raw data and the PCA-reduced data respectively. A confusion matrix helps visualize the accuracy of a classification algorithm. The number in each box represents the percentage of times the device listed for that row was classified as the device listed for that column. Using the raw feature vectors, we achieved 99.7 percent accuracy with three classifiers. Pictured in Figure 14a is the confusion matrix using the Logistic Regression classifier. Shown in Figure 14b, the PCA-reduced feature vectors achieved the best accuracy at 96.5 percent using the K-Nearest Neighbors classifier. Figure 14b demonstrates that the loss in accuracy when using the PCA-reduced dataset is due to the classifier confusing the Phoenix Contact Quint 10 A with the Schneider 10 A power supplies.

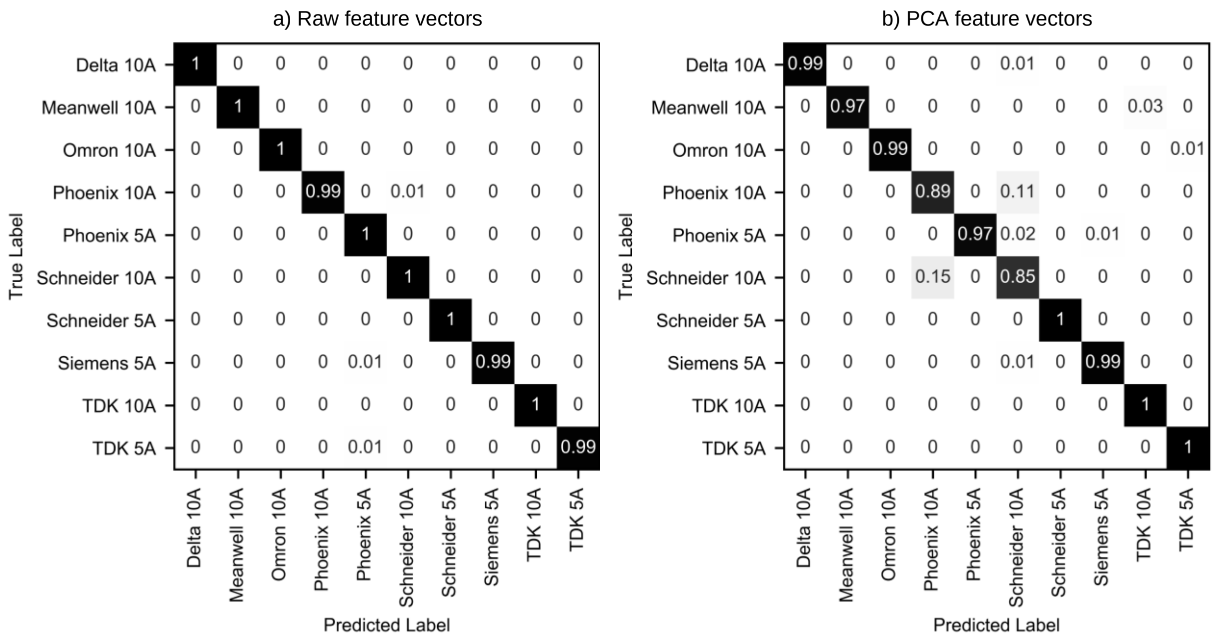

Figure 14 Results of classification on Test Dataset 1 where the sensor was placed at the breaker panel with a 3.5 A nominal load. Using the raw feature vectors (Figure 14a) yields 99.7% accuracy with the Logistic Regression classifier. Using the reduced PCA feature vectors (Figure 14b) yields 96.5% accuracy with the K-Nearest Neighbors classifier.

4.2 Test Dataset 2: Sensor at Breaker Panel, 27 A Nominal Load

Table IV shows the classification results of each of the six classifiers built from both the raw and dimensionality-reduced feature vectors from Test Dataset 2 (Figure 4).

TABLE IV
CLASSIFIER ACCURACY ON TEST DATASET 2

Classifier	**Accuracy**	
	Raw	**PCA**
LDA	.673	.751
LR	.868	.811
SVM	.865	.722
RF	.495	.684
NN	.967	.761
KNN	.964	.878

The increased system load does not affect the results significantly, with two of the classifiers having above 95 percent accuracy on the raw feature vectors. Accuracy is affected when reducing the feature vectors from 200 dimensions to 12; however, two of the six classifiers still achieved over 80 percent accuracy using the PCA-reduced feature vectors. Figure 15 shows confusion matrices of the best performing classifiers using the raw data and the PCA-reduced data respectively. Figure 15a shows the confusion matrix using the Neural Network classifier, which achieved the greatest accuracy at 96.7 percent. Shown in Figure 15b, the PCA-reduced feature vectors achieved the best accuracy at 87.8 percent using the K-Nearest Neighbors classifier. Figure 15b demonstrates that the reduced accuracy in using the PCA-reduced data is due to confusion between the Delta EOE 10 A and Phoenix Contact Quint 10 A power supplies, the Phoenix Contact Quint 10 A and the Schneider 10 A power supplies, and the TDK 10 A and Meanwell 10 A power supplies.

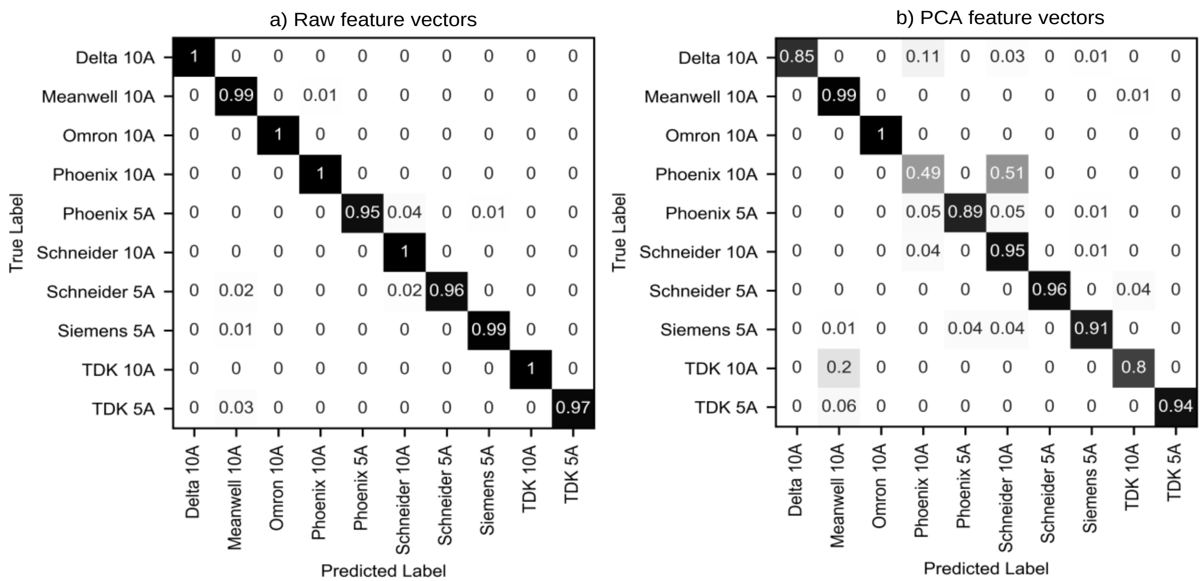

Figure 15 Results of classification on Test Dataset 2 where the sensor was placed at the breaker panel with a 27 A nominal load. Using the raw feature vectors (Figure 15a) yields 96.7% accuracy with the Neural Network classifier. Using the reduced PCA feature vectors (Figure 15b) yields 87.8% accuracy with the K-Nearest Neighbors classifier.

5 Conclusion

We were able to use startup transients and classification algorithms to differentiate with high accuracy among identical-class power supplies operating on the local power system. These results indicate that in a typical office environment it is possible to detect the presence of nearly identical devices at a measurement point that is both physically and electrically distant from the device of interest. Our systematic approach demonstrates in a controlled fashion that nearly identical devices can be accurately discerned. The identical-class power supplies used in this study contain similar circuitry to devices commonly used in office equipment and information systems. Further, the success of reducing the feature vectors by a factor of 16 establishes the viability of implementing these techniques in an embedded device. Thus, our results indicate that the methods presented in this paper will yield high success when monitoring spaces for counterintelligence purposes.

Future work is needed to develop these techniques into a full counterintelligence strategy. Notably, a robust event detector is necessary to detect startup transients so that they can be extracted for machine learning classification. Such an event detector would need to work on generic systems regardless of electrical noise. Next, these techniques need to be evaluated against increasing system loads to identify the limits of event detection and signature extraction. We had substantial success at a system load of 27 A, a similar loading to typical small office buildings. However, when we tested our techniques on a system loaded at over 200 A, we had inconsistent results. Thus, it is necessary to determine a theoretical and experimental limit to the methods presented in this paper. Finally, since this paper establishes that startup transients are robust signatures for NLM of switch mode power supplies, we can build a corpus of load signatures of electronic devices that contain switch mode power supplies (computers, printers, etc.), and evaluate the techniques presented here against a broader collection of devices.

6 REFERENCES

[1] G. W. Hart, "Nonintrusive Appliance Load Monitoring," *Proceedings of the IEEE,* vol. 80, no. 12, pp. 1870-1891, 1992.

[2] A. Zoha, A. Gluhak, M. A. Imran and S. Rajasegarar, "Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey," *Sensors,* no. 12, pp. 16838-16866, 2012.

[3] J. Liang, S. K. K. Ng, G. Kendall and J. W. M. Cheng, "Load Signature Study—Part I: Basic Concept, Structure, and Methodology," *IEEE Transactions on Power Delivery,* vol. 25, no. 2, pp. 551-559, 2010.

[4] M. Zeifman and K. Roth, "Nonintrusive Appliance Load Monitoring: Review and Outlook," *IEEE Transactions on Consumer Electronics,* vol. 57, no. 1, 2011.

[5] M. Kahl, T. Kriechbaumer, A. Ul Haq and H.-A. Jacobsen, "Appliance Classification Across Multiple High Frequency Energy Datasets," *IEEE International Conference on Smart Grid Communications,* pp. 147-152, 2017.

[6] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford and P. Armstrong, "Power Signature Analysis," *IEEE Power and Energy Magazine,* pp. 56-63, 2003.

[7] H.-H. Chang, K.-L. Lian, Y.-C. Su and W.-J. Lee, "Power-Spectrum-Based Wavelet Transform for Nonintrusive Demand Monitoring and Load Identification," *IEEE Transactions on Industry Applications,* vol. 50, no. 3, pp. 2081-2089, 2014.

[8] M. Zeifman, C. Akers and K. Roth, "Nonintrusive Monitoring of Miscellaneous and Electronic Loads," *2015 IEEE International Conference on Consumer Electronics,* pp. 305-308, 2015.

[9] J. A. Mueller, A. Sankara, K. J. W. Kimball and B. McMillin, "Hidden Markov Models for Nonintrusive Appliance Load Monitoring," *North American Power Symposium,* 2014.

[10] J. M. Gillis, J. A. Chung and W. G. Morsi, "Designing New Orthogonal High-Order Wavelets for Nonintrusive Load Monitoring," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 3, pp. 2578-2589, 2018.

[11] M. Kahl, T. Kriechbaumer, A. Ul Haq and H.-A. Jacobsen, "Appliance Classification Across Multiple High Frequency Datasets," *IEEE International Conference on Smart Grid Communications,* pp. 147-152, 2017.

[12] G. Laput, Y. Zhang and C. Harrison, "Synthetic Sensors: Towards General-Purpose Sensing," Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp. 3989-3999, 2017.

[13] S. Gupta, M. S. Reynolds and S. N. Patel, "ElectriSense: Single-Point Sensing Using EMI for Electrical Event Detection and Classification in the Home," UbiComp'10 - Proceedings of the 2010 ACM Conference on Ubiquitous Computing, pp. 139-148, 2017.

[14] Y. Liu and M. Chen, "A Review of Nonintrusive Load Monitoring and Its Application in Commercial Building," IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems, pp. 623-629, 2014.

[15] D. Srinivasan, W. S. Ng and A. C. Liew, "Neural-Network-Based Signature Recognition for Harmonic Source Identification," *IEEE Transactions on Power Delivery,* vol. 21, no. 1, pp. 398-405, 2006.

[16] S. R. Arrachman, M. Adiatmoko, A. Soeprijanto, M. Syai'in, M. Sidik and N. Rohiem, "Smart Meter based on Time Series Modify and Extreme Learning Machine," *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology,* pp. 86-92, 2017.

[17] T. Hastie, R. Tibshirani and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition, Springer Science, 2009.

[18] S. Raschka, Python Machine Learning, Packt Publishing, 2016.