Spectral Clustering for Electrical Phase Identification Using Advanced Metering Infrastructure Voltage Time Series

Logan Blakely
Portland State University

Follow this and additional works at: https://pdxscholar.library.pdx.edu/open_access_etds

Part of the Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Let us know how access to this document benefits you.

Recommended Citation
https://doi.org/10.15760/etd.6567

This Thesis is brought to you for free and open access. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of PDXScholar. Please contact us if we can make this document more accessible: pdxscholar@pdx.edu.
Spectral Clustering for Electrical Phase Identification Using Advanced Metering Infrastructure Voltage Time Series

by
Logan Blakely

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Science

Thesis Committee:
Wu-chi Feng, Chair
Wu-chang Feng
Charles Wright

Portland State University
2018
Abstract
The increasing demand for and prevalence of distributed energy resources (DER) such as solar power, electric vehicles, and energy storage, present a unique set of challenges for integration into a legacy power grid, and accurate models of the low-voltage distribution systems are critical for accurate simulations of DER. Accurate labeling of the phase connections for each customer in a utility model is one area of grid topology that is known to have errors and has implications for the safety, efficiency, and hosting capacity of a distribution system. This research presents a methodology for the phase identification of customers solely using the advanced metering infrastructure (AMI) voltage timeseries. This thesis proposes to use Spectral Clustering, combined with a sliding window ensemble method for utilizing a long-term, time-series dataset that includes missing data, to group customers within a lateral by phase. These clustering phase predictions validate over 90% of the existing phase labels in the model and identify customers where the current phase labels are incorrect in this model. Within this dataset, this methodology produces consistent, high-quality results, verified by validating the clustering phase predictions with the underlying topology of the system, as well as selected examples verified using satellite and street view images publicly available in Google Earth. Further analysis of the results of the Spectral Clustering predictions are also shown to not only validate and improve the phase labels in the utility model, but also show potential in the detection of other types of errors in the topology of the model such as errors in the labeling of connections between customers.
and transformers, unlabeled residential solar power, unlabeled transformers, and locating customers with incomplete information in the model. These results indicate excellent potential for further development of this methodology as a tool for validating and improving existing utility models of the low-voltage side of the distribution system.
List of Figures

Figure 1 - Example power grid, [6].......................................................................................... 3
Figure 2 - Example of phase connections [7]........................................................................... 5
Figure 3 - Transformer connection example, satellite................................................................. 6
Figure 4 - Transformer example, street view............................................................................... 7
Figure 5 - Feeder 1 utility phase labels .................................................................................... 16
Figure 6 - Percentage of time meters are reporting for 3576 customers (Feeder 1, Feeder 2, and Feeder 3)......................................................................................................... 16
Figure 7 - Length of outages of 486 days for 3576 customers (Feeder 1, Feeder 2, and Feeder 3)......................................................................................................................... 17
Figure 8 - Missing data for Feeder 1 .......................................................................................... 18
Figure 9 - 'Out of bounds' voltage examples ............................................................................ 23
Figure 10 - More 'out of bounds' voltage examples .................................................................. 24
Figure 11 - Two nearly identical voltage profiles ...................................................................... 26
Figure 12 - Voltage correlations visually ................................................................................... 29
Figure 13 - Voltage correlations, synthetic example .................................................................. 30
Figure 14 - Clustering Methodology ........................................................................................ 31
Figure 15 - Clustering prediction assignment, [7]..................................................................... 34
Figure 16 - Parameter Sweep, RBF versus polynomial kernels.................................................. 43
Figure 17 - Parameter Determination, PCA versus No PCA using 30 components .................... 44
Figure 18 - Parameter determination, sweep over the number of PCA components.................. 45
Figure 19 - PCA versus without PCA plotting Consistency and Confidence Scores .................. 46
Figure 20 - Parameter determination, Consistency versus Confidence score for different sliding window sizes .............................................................................................................. 47
Figure 21 - Clustering predicted phases .................................................................................... 51
Figure 22 - Feeder 1 Topology Validation .................................................................................. 53
Figure 23 - Incorrect lateral prediction, the area from the blue oval in Figure 22 ....................... 53
Figure 24 - Phase labeling error, example #1, satellite view ...................................................... 54
Figure 25 - Phase labeling error, example #1, street view ............................................................ 55
Figure 26 - Plot of Confidence scores ........................................................................................ 56
Figure 27 - Consistency scores versus Confidence scores in a Monte Carlo simulation .......... 57
Figure 28 - Consistency versus Confidence comparison of Feeder 1, 2, & 3 ............................. 60
Figure 29 - Comparison of Topology Inconsistency Rate for Feeder 1, 2, & 3 ........................... 62
Figure 30 - Comparison of clustering failure between Spectral Clustering (SP) and K-Means (KM) ................................................................................................................................. 65
Figure 31 - Feeder 1 Spectral Clustering Compared to K-Means, Consistency score versus Confidence score .......................................................................................................................... 66
Figure 32 - Feeder 1 Customers where the predicted phase is inconsistent with the topology validation ........................................................................................................................................ 68
Figure 33 - Incorrect transformer prediction ............................................................................. 71
Figure 34 - Street view example #1, original utility labels .................................................................73
Figure 35 - Street view Example #1, clustering and street view labels ........................................ 73
Figure 36 - Street view example #1, four-meter house ....................................................................74
Figure 37 - Street view example #1, four-meter house transformer connection ..............................74
Figure 38 - Street view example #1, transformer 51 connection .......................................................75
Figure 39 - Street view example #1, transformers 50 & 52 ............................................................75
Figure 40 - Street view example #2, original utility labels ..............................................................77
Figure 41 - Street view example #2, clustering and street view labels ................................................77
Figure 42 - Street view example #2, transformers 61 & 62 connections .........................................78
Figure 43 - Street view example #2, phase B continues down the street and serves another customer ........................................................................................................................................78
Figure 44 - Version mismatch verifying clustering and correlation coefficient results ..................80
Figure 45 - Google Earth view of an unmarked transformer ..............................................................81
Figure 46 - Possible three-phase customer ......................................................................................82
Figure 47 - Possible 3-phase customer ..............................................................................................83
Figure 48 - Example of clear issues in the utility model .................................................................84
Figure 49 - Unknown customer example #1 .....................................................................................85
Figure 50 - Unknown customer example #2 ....................................................................................86
List of Tables
Table 1 - Comparison of feeder differences .................................................................15
Table 2 - Spectral Clustering algorithm .................................................................34
Table 3 - Spectral Clustering results breakdown ....................................................50
Table 4 - Feeder 1 utility labels versus predicted labels ........................................50
Table 5 - Feeder 2 results, utility labels versus predicted labels ..............................59
Table 6 - Feeder 3 results, utility labels versus predicted labels ..............................59
Table 7 - K-Means algorithm .................................................................................63
Table 8 - Breakdown of the customers inconsistent with the labeled model topology ....70
Acknowledgements

I would like to take the opportunity to acknowledge Portland State University, Sandia National Laboratories, and the U.S. Department of Energy for having a role in supporting this research. I would also like to personally thank my two thesis advisors Matthew Reno, with Sandia Labs, and Wu-chi Feng, with Portland State University. I am grateful to them both for their support and advice throughout this process.

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC a wholly owned subsidiary of Honeywell International, Inc. for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA0003525. SAND2018-12556 PE
1. Introduction

This research proposes to use a spectral clustering-based methodology to identify electrical utility customers’ electrical phases based on time-series voltage profiles recorded at customer meters. The purpose of this research is to validate and improve existing utility company phase labels in models of the distribution system portion of the power grid.

1.1 Motivation

The drive towards integrating ever greater quantities of renewable energy into the existing power grid poses a unique set of challenges. [1] demonstrates both the push towards more distributed energy resources (DER) such as solar power, wind power, and energy storage devices, as well as the diverse set of challenges in reaching those goals.

Of critical importance as these technologies move forward is accurate simulations of the power grid, and a critical aspect of obtaining accurate simulations is accurate models. At larger scales (high-voltage and medium-voltage), the models used by utility companies tend to be accurate, however, on the low-voltage side which connects individual customers to transformers, the models are more prone to error [2]. The grid is a legacy system that can accumulate model errors over time as new components or sections are added or outages cause unrecorded changes to be made during the restoration process. Over time this can lead to grid models containing erroneous descriptions of the grid. A Geographical Information System (GIS) model may have transformer connection issues, missing transformers, phase connection issues, missing

1
meters, etc. The model will show all these details and more about the system, and those details are often used as is in simulations or for planning. More and more time and effort is going into making sure that the GIS models (and therefore the simulations) are accurate; significant research is going into topology estimation for the low-voltage segment of the distribution system; [2], [3], [4], [5] are just a few examples. Historically, correcting such errors meant sending personnel out to conduct field verification of these systems. That process is both extremely time consuming and expensive, particularly in dense urban areas. The advent of advanced metering infrastructure meters (AMI), or smart meters, has dramatically increased the amount of sensing equipment on the grid. The ‘big data’ that results from that sensing presents opportunities to process that data and gain further understanding of the operations of the grid. This is a perfect application for using machine learning to better understand this data.

1.2 Overview of the Power Grid

This section provides an introduction to the power grid as a whole, as well as the concept of electrical phases which is the main focus of this research. The power grid is broadly divided into two sections, the transmission system and the distribution system. See Figure 1 for an example; although grid characteristics vary, these definitions are generally accurate. The transmission system generally runs from large-scale power generation to substations, and it tends to be high-voltage and long distance. The
distribution systems (within the black oval) runs from substations to customers and is lower voltage and shorter distances. The distribution system is further subdivided into two sections. The medium-voltage section (red oval) runs from substations along streets, and the low-voltage portion (green ovals) generally runs from transformers on street poles to customers. The high-voltage transmission system and medium-voltage distribution system models are fairly accurate, but historically the low-voltage portion of the distribution system was less critical for simulations and therefore the models are less accurate. Consider though, the issue of adding solar power to individual residences, now the models of the low-voltage side are more critical.

Figure 1 - Example power grid, [6]

The distribution system is powered by alternating current (AC) which is generally divided into three distinct phases, Phase A, B, and C. In the sine wave representation of
alternating current, the three phases are offset by 120 degrees each. Standard residential customers will be connected to one of the three phases for standard power usage. Larger commercial customers may use all three phases for higher power usage. In practice, the large AC transmission system will always have all three phases, but at the distribution level, there may be fewer phases since each residential customer only requires a single phase. Figure 2 shows a sample diagram of the phase connections. The medium voltage system is labeled as ‘primary’ and the low voltage system is labeled as ‘secondary’ in that figure. An example of three-phase distribution system power lines running down a street, one for each of the phases is shown at the top of Figure 4. Looking at Figure 3 from Google Earth, there is a transformer plotted in purple that steps the voltage down from the distribution system voltage (12 kV) to the household electrical voltage (120/240 V), serving five customers in green. The green lines represent power lines running to the customers, and the green color represents a utility labeling of Phase B for those customers. Figure 4 shows the same transformer in Google Street View; the cylindrical transformer is connected to the middle of the three wires along the top by a wire running from the transformer and clamped to the middle power line. By convention, the middle wire is generally Phase B. Then, power lines run from the transformer in three directions. Those power lines will be directly connected to customers. All the customers connected to the same transformer must be on the same phase. So, that clamped connection between the transformer and one of the three
power lines is the specific aspect of low-voltage topology that this research is focused on.

*Figure 2 - Example of phase connections [7]*
Figure 3 - Transformer connection example, satellite
1.3 Overview of Research

Electrical phase connection is one facet of the grid topology that is known to contain errors and is crucial for both efficient grid performance and the continuing addition of DER onto the grid. This research focuses on identifying the phases of customers using the voltage profiles of each customer. The research conducted in [8] demonstrates the importance of balanced phases in the presence of solar power as well as looking towards adding more solar power to the grid. Even without the presence of solar power, there are safety and efficiency reasons to desire the load on the three phases to be balanced, [9]. The increasing prevalence of advanced metering infrastructure (AMI) data allows the use of data science in the estimation of grid topology using methods
that were previously not feasible. AMI data profiles are time-series measurements taken at consistent intervals, at anywhere from 1-hour to 1-min intervals [10]. AMI data provides a significant increase in the amount of data available about the system.

Common uses of AMI data include phase identification, [5], [7], [11], [12], load disaggregation [13], [14], and topology estimation [2], [3], [4], [5].

This research proposes to use a methodology based on spectral clustering to cluster electrical utility customers by phase using only the voltage profiles recorded at the customer meters. The profiles will be clustered according the correlation between voltage profiles on the same phase. Spectral Clustering works by performing non-linear dimensionality reduction on the raw data prior to clustering, and that reduction step distinguishes this type of clustering from other clustering methodologies. The voltage measurements portion of the advanced metering infrastructure (AMI) data, taken from residential meters in a city in the northeastern United States, will be used to cluster the customers of three feeder systems into Phase A, B, & C groups. This research is novel in that Spectral Clustering has not been applied to this problem, and in previous approaches to solving this phase identification problem, the substation voltage has also been used. At no time is the substation voltage used in this research. In addition, the results produced by this methodology provide insight into correcting other types of errors, besides phase identification, in the model topology.
1.4 Related Work

The introduction of AMI meters and the big data that they produce is enabling new opportunities to apply data science and machine learning to the challenges in power systems research. There are many current examples of this; here are just a few [15]–[18]. There has also been a variety of previous work in the area of phase identification. Some common approaches as well as several state-of-the-art methods are briefly summarized below.

One method presented in [19] from 2011 is a ‘load summing’ approach. This method relies on the fact that the sum of all the loads connected to a transformer should equal the load on the transformer itself within some margin of error. This approach turns this information into a system of linear equations to solve, and the paper presents several approaches to solving the set of equations generated in this way. The datasets used were two datasets that were artificially generated with a maximum of 250 homes and one actual dataset of 100 homes, with measurements taken at 30-minute intervals.

Another approach is the signal injection approach from this research in 2012 [20]. This requires two pieces of equipment, one to inject a signal and one to read the signal on the customer side. In this way different signals are injected into the different phases to differentiate when they are read at a later time. This appears to work well; although it is time-consuming and requires extra equipment.
Reference [5] from 2013 uses correlation coefficients as a similarity metric in ‘bottom-up grouping’ (hierarchical clustering) to do phase identification and topology estimation. This research uses linear regression to account for some load issues that affect the customers, both individually and in groups, and they note that this helps differentiate the correlation coefficients between customers. Their datasets used 15-minute measurement intervals as well as 1-hour measurement intervals. For the phase identification portion they still rely on having the known substation voltage measurements to compare to. Reference [7] from 2015 also uses a hierarchal clustering technique, taking advantage of the inherent hierarchy in the distribution system. Their data consists of over 10,000 customers over 2 months and five feeders. They claim to have the ground truth through field verification and the utility database, and they mention that they are dealing with missing data in the dataset. The initial choice of centroid was the substation voltage, making the final phase assignment easier. They do note that customer correlation could be used to make the final phase assignment. Correlation was used as the distance metric and the data was broken in 4-day segments in a sliding window approach.

Reference [11] from 2016 uses the constrained k-means algorithm with Principal Component Analysis (PCA) [21] to do phase identification for customers in Southern California. The constrained c-means algorithm takes the original k-means algorithm and adds ‘must-link’ and ‘must-not-link’ constraints to the data [22], [23]. In this case,
customers who were known to be connected to the same physical lateral were grouped using ‘must-link’ constraints so that all those customers were forced to be in the same cluster. The dataset consisted of 3 months of voltage measurement data, with 1-hour measurement intervals. They took batches of 1-month, used PCA to reduce the full voltage profiles down to 2 principal components and then used the constrained k-means algorithm with Euclidean distance to cluster the customers. The phases were identified by comparing the results of the clustering to the substation voltages. One interesting observation they make is that some months of their data (September and October) clustered better than other months (August). Reference [5] also shows similar results of some time periods being easier to cluster than others. They speculate that perhaps shorter measurement intervals may be required to deal with this issue. This also suggests that longer durations of data may be required for phase identification using these methods.

Reference [24] from 2018 in Belgium introduces the Constrained Multi-Tree algorithm which grows a tree using Pearson Correlation Coefficients as a similarity metric, starting with a transformer as the root of the tree. The tree-growing algorithm is inspired by Prim’s Minimum Spanning Tree Algorithm. They show that their method produces better results than the constrained k-means algorithm [23] on their datasets. Their dataset is a group of 89 homes in Belgium with 1-minute, averaged, measurement intervals; it’s important to note that because the distribution system is located in
Belgium, the distribution system is not directly comparable to distribution systems in the United States.

In another approach, [18] in 2018 investigates the use of supervised machine learning techniques to solve the phase identification problem. The training set for the machine learning is a carefully chosen subset of the customers for which manual phase verification must be done prior to employing this technique. That data is then used to train a neural network which predicts the phase for the remaining customers. The dataset used is from Southern California Edison with about a month of data.

Finally, [25] in 2017 proposes to use computer vision techniques on specialized video images to detect the flicker in lights due to the alternating current. This method takes advantage of the distance between the alternations in the three phases. For example, this method might record a video of a light source and then use the information contained in the video, the flicker that is too fast for the human eye to detect, to assign a phase to that building.

Some of the current research, including [11], [24], use some of the topology information, such as the customer-transformer connections labels in the clustering itself as a form of constraint. This implicitly assumes that those labels are completely accurate, and if they are not then this approach simply propagates those errors into the phase identification problem. I have chosen not to use that type of topology
information in the clustering itself, but instead that information is used to validate the clustering results, and actually in some cases leads to discovering other types of topology errors. See the Overview of Spectral Clustering Results section as well as the Additional Advanced Grid Applications section for further details.

One aspect to keep in mind in comparing these methods and research is that it is difficult to directly compare them due to high variability in location, dataset sizes, seasons, availability of substation voltages, and a variety of other factors. Both [5], [11] suggest that their algorithms tend to perform better during certain portions of the year and speculate that seasonal variance may account for that difference. Therefore, comparing methods and research conducted using data from southern California with research conducted using data from locations with much higher seasonal variation may be difficult to compare directly.
2. Dataset

The dataset used in this research was provided by a utility in the northeastern United States. The Advanced Metering Infrastructure (AMI) data covers data over 486 days for a set of feeders of which this research uses 3 feeders with ~1,000 customers per feeder; see Table 1 for a comparison of feeder characteristics. The results using the methodologies covered in this research are shown in each of those three feeders. In addition to the AMI data, the dataset also includes the electrical model, equipment information (wire types, transformer sizes, etc.), customer-transformer connection labels, and GIS coordinates for customers and other equipment. AMI time-series data can have diverse characteristics depending on the implementation by a specific utility. In this case, each customer is labeled with a customer identification number, a phase label, and a generation type, if any. The measurements are collected in 15-minute intervals using the interval average method (instantaneous measurement is the other common approach). Each interval contains a measurement for real power, reactive power, real power output (if any), and voltage. The power and voltage measurements are taken with an accuracy of four decimal places. Figure 5 shows the original utility model phase label distribution for Feeder 1. The 4% of customers that are ‘unlabeled’ have voltage profiles in the dataset but do not have phase labels or geographical coordinates. The following figures below show the customers from the 3 feeders, labeled Feeder 1, Feeder 2, and Feeder 3. Figure 6 shows the percentage of time that
each customers’ meter was reporting over the 486-day period. Figure 7 shows the
length of outages for the same time period, considering all of the customers in those
three feeders. Any periods of ‘outage’ over a week are excluded from these figures and
are considered a time where the meter was not in use. Figure 8 plots the missing data
points in red for Feeder 1, including the longer duration periods where the meters were
not reporting. It is clear from these three figures that missing data, both from outages
and longer periods where meters were not reporting, is an issue for the using the
dataset as whole. These issues will occur in any real-world dataset and approaches for
dealing with these issues are discussed in the Data Preparation section.

Table 1 - Comparison of feeder differences

<table>
<thead>
<tr>
<th>Feeder</th>
<th>Voltage (kV)</th>
<th>Peak Load (MW)</th>
<th>Number of Customers</th>
<th>Feeder Length (KM)</th>
<th>Line Regulators</th>
<th>Capacitors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feeder 1</td>
<td>12.47</td>
<td>2.0</td>
<td>1153</td>
<td>5.4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Feeder 2</td>
<td>12.47</td>
<td>1.8</td>
<td>1309</td>
<td>2.5</td>
<td>1 set, single-phase</td>
<td>450 kVar</td>
</tr>
<tr>
<td>Feeder 3</td>
<td>12.47</td>
<td>1.4</td>
<td>1188</td>
<td>3.2</td>
<td>0</td>
<td>300 kVar 300 kVar</td>
</tr>
</tbody>
</table>

15
Figure 5 - Feeder 1 utility phase labels

Figure 6 - Percentage of time meters are reporting for 3576 customers (Feeder 1, Feeder 2, and Feeder 3)
Figure 7 - Length of outages of 486 days for 3576 customers (Feeder 1, Feeder 2, and Feeder 3)
One unique aspect of this dataset compared to the datasets referenced in the Related Work section is the length of the dataset. With approximately 15 months of data, this is by far the longest of the datasets used of the research mentioned. The dataset is also geographically from the northeastern United States, a location with distinct seasonal variations. Finally, some work was done by the utility in attempting to update and clean the model for Feeder 1. As a result, we would expect that feeder to have fewer errors than the other two, however the phases and network topology are not fully verified in any of the feeders and this presents some interesting challenges.
3. Data Preparation

The considerations in preparing the data fall into four general categories, missing data, erroneous data, noise, and the representation of voltage profiles. As a preprocessing step, the voltage profiles were normalized to a mean of 1.

3.1 Missing Data

Based on the figures in the previous section, Figure 6, Figure 7, and Figure 8, it is clear that missing data is a significant issue. All customers on Feeder 1 have some level of missing data, so there are in fact no voltage profiles that are 100% complete for the observed time period. Much of the previous work on phase identification via clustering has used synthetic datasets, time periods with no missing data, and fully field verified portions of the distribution system. Deciding what approach to use in dealing with the missing data is a critical question and significant time was spent exploring the best approach. The first step was to eliminate customers with too few measurements. There were a small number of customers in the original dataset with fewer than 1,000 data points (~10.5 days). Those customers were excluded from the analysis altogether. The remaining customers were used in the analysis regardless of the percentage of data that was missing. One common approach to deal with missing data is an interpolation scheme. It’s clear that the average of the normalized voltage profiles will be 1 and so that is a plausible value that could be used to replace the missing data. However, phase identification via clustering relies on the minute shifts in voltage of customers on the
same phase together, in contrast to the minute shifts of customers on other phases. By replacing with ones or some other interpolation scheme, the risk is that those minute shifts will be lost or obscured by the interpolated data. Therefore, the decision was made to leave the data ‘missing’ (NaN values) instead of using a replacement/interpolation scheme.

The two approaches that were used in the final analysis were the ‘sliding window’ approach from [7], and a ‘pairwise’ correlation coefficient method for post-clustering analysis. The sliding window approach breaks the full voltage profile into ‘windows’ of arbitrary size and then each window is clustered separately. To deal with the missing data, each window is ‘cleaned’ by removing customers that have missing data during that window’s time period. In this way, customers are not clustered during windows where they are missing data but are included during the remainder of the windows. The ‘pairwise’ correlation coefficient methodology calculates the Pearson correlation coefficients using the entire voltage profiles for one pair of customers individually, simply omitting values where one of the pair of customers has missing data. Both of these approaches, for different applications, worked well in these experiments.

3.2 Erroneous Data

The second main problem with the data is the issue of erroneous data. The first question is, ‘How do we know if a data point is erroneous?’ Several types of erroneous data points were identified during this stage of the research. There were a small
number of customers (~12 in Feeder 1) with abnormally large voltage readings, even when the voltage readings were normalized, the customers were reporting readings ~40 or even larger. There was no explanation in the data or the customer labeling or that was provided by the utility for this group of customers. Therefore, this group was also excluded from the analysis.

There were also a variety of individual voltage readings that are clearly erroneous. Thresholds of < 0.9 and > 1.1 were imposed as clear delineations of ‘out of bounds’ voltage values. Any measurements outside of those thresholds clearly violates the standards set forth in ANSI C84.1. Figure 9 shows a representative example of the type of ‘out of bounds’ voltages that are present in the dataset. This is not intended to remove all erroneous values, but these are certainly erroneous values that are being removed. From the figure we can see two types of erroneous voltages, there is an instance where the meter recorded a 0, and then an instance where the meter recorded both high and low oscillations before stopping to record entirely. It’s unknown the precise cause of these types of erroneous measurements. To deal with these issues, all ‘out of bounds’ voltage measurements were removed (replaced with NaN values the same as ‘missing’ data) as well as one preceding measurement and one subsequent measurement. That reduces the possibility of including erroneous measurements, even if they are ‘within bounds’ in the final analysis. See Figure 10 for further examples of customers with ‘out of bounds’ voltage measurements.
Figure 9 - 'Out of bounds' voltage examples
Figure 10 - More 'out of bounds' voltage examples

3.3 Noisy Data

The third issue to consider in the data preparation process is the issue of ‘noise’ in the data. In this case, ‘noise’ can be defined as any perturbations of the data that are unrelated to the fluctuations due to the electrical phase. This ‘noise’ could be either erroneous measurements that were not filtered out via the ‘out of bounds’ criteria, they could be the individual particularities of each customer, or even daily trends followed by all customers could be considered noise in this case. One possibility is a ‘smoothing’ algorithm; Dynamic Time Warping approach is common for time series data [26]. However, it’s impossible to ascertain precisely the level of noise, and it’s important to
keep in mind that each of the measurements in this dataset was recorded as the 15-minute average value (as opposed to recording the instantaneous value). Thus, there is already a ‘smoothing’ effect that occurs by the utility choosing to use that collection method. Investigating the data itself, it appears that any additional smoothing is likely to remove information relating to the phase identification problem. In Figure 11, there are two highly correlated customers’ (according to the ‘pairwise’ correlation coefficient method) voltage profiles through a short time period plotted together. The two profiles are nearly identical. For this reason, it was decided to not to use a smoothing algorithm during the data preparation phase of the project. For some applications using AMI time-series data, clock synch or clock skew can be an issue with the measurements aligning. In this case, since the voltage measurements are taken using the averaging method, this should take care of any relatively small clock skew issues.
3.4 Voltage Profile Representation

There is the question of how to represent the voltage profiles themselves during the clustering process. All representations were first normalized around a mean of 1 and cleaned using the process described previously to remove clearly erroneous data values. ‘Raw voltage’ profiles are defined as simply the normalized, cleaned profiles without any further processing. [7] proposes two other voltage representations in addition to the raw voltage profiles; what is there called ‘voltage fluctuations’, and ‘discretized binary fluctuations’. I have chosen to call the voltage fluctuations ‘delta voltage’ profiles. Delta voltage is simply the change in voltage between two measurements, so the total
number of data points is reduced by one, and the resulting representation consists of the difference between adjacent data points in the time series. I also implemented the ‘discretized binary fluctuations’ but that representation did not work with the SciPy implementation of spectral clustering, so that method was not used in this research. During the experiments conducted for this research the raw voltage profiles and the delta voltage profiles were compared. The experimental results favored the ‘delta voltage’ representation but were inconclusive overall. See the Results section for more details.
4. Methodology

The conceptual basis of this methodology is that it is possible to cluster the customers based on the correlations between their voltage profiles. This section is organized as follows, the Voltage Profile Correlation section provides an overview of how the voltage profiles are separable by phase, the Clustering Methodology section provides an overview of the clustering process, the Spectral Clustering section gives an introduction to the specific type of clustering used in this research, the 4.3 Evaluation Metrics & Validation section gives an overview of metrics used in this research, and the Parameter Determination section details the chosen parameters for the clustering. Code for this project was primarily implemented in Python using the sklearn libraries. The utility grid model software used was OpenDSS, and Matlab with the GridPV toolbox [27] was used to interface with OpenDSS. Google Earth and Google Earth Street View were used during the validation process of this research.

4.1 Voltage Profile Correlation

Each customer has a voltage profile consisting of measurements at 15-minute intervals. The core concept of this research is that customers on the same phase will tend to have similar variations in voltage compared to customers on different phases. The clustering attempts to cluster based on that similarity. Figure 12 and Figure 13 show examples of this. Figure 12 shows an example of actual voltage profiles and Figure 13 shows synthetic, simplified profiles for illustration. There are six customer voltage profiles
plotted over time (~15 hours), two of each phase. Note there are some large trends followed by all six profiles, these are likely to be daily trends or perhaps are weather related. At the medium level of trends, one can see that the pairs of customers on the same phase have variation that the other four customers do not (note that the differences in question are variations and NOT magnitude). Finally, there are smaller variations that are particular to each customer. The daily trends and individual customer variation are considered noise in the context of this research.

Figure 12 - Voltage correlations visually
4.2 Clustering Methodology

Clustering techniques are a collection of methods within the unsupervised machine learning category. Unsupervised learning implies that the data labels (or categories) are unknown in advance. This is in contrast to supervised machine learning where the training data labels are essential to train the machine learning architecture. Within the context of this dataset, the utility labels for each customer exist but are known to have some level of error, and the level of error itself is unknown. The approach used in this research is not classic unsupervised learning because the utility labels are used to
classify the resulting clusters. See Figure 14 for a flowchart representation of the clustering methodology.

First, a portion of the data is selected for each customer. For example, if the sliding window size is 384 samples (~4 days) then for the first sliding window the first 384 samples for each customer are selected. Then the window is ‘cleaned’ by removing all customers with missing data for that time period. Second, the clustering is performed on that time period for all remaining customers. This step does not involve the utility phase labels. After the clustering, each cluster is assigned a phase based on a majority
vote of the utility labels for the customers that landed in that clustering. Then the predicted phase for each customer in that cluster is the assigned label for that cluster. For example, if a cluster has 10 customers with 7 customers labeled Phase A by the utility, 2 customers labeled Phase B, and 1 customer labeled Phase C, then the assigned cluster label would be Phase A, and all 10 customers would receive a predicted phase of Phase A for that clustering. For the three customers labeled Phase B or Phase C in the utility model, this represents a predicted error in the utility model labels for these customers. Figure 15 shows a simple example of this process. Each sliding window is treated in this same manner, in this example the next portion of data selected would be the subsequent 384 samples. Each sliding window is a ‘vote’ for the final predicted phase. Once all sliding windows have been clustered separately, a final prediction is determined for each customer. The final ensemble prediction for each customer is the majority vote over all of the sliding window predictions.

The sliding window approach from [7] provides several advantages over attempting to cluster the entire time series at one time. First, it provides a way to deal with the missing data issue. Clustering the entire time series at once is problematic because all customers in Feeder 1 have missing data at some point during the whole collected time period. Using the sliding windows, customers with missing data can be excluded from any window where their profile contains missing values, and, ideally, with a small enough window size all customers will receive votes from some of the sliding windows.
Second, using the sliding windows provides an ensemble approach to the clustering. Ensemble machine learning approaches have been demonstrated to achieve superior performance relative to single instances of a technique [28]. In this case, the ensemble gives improved accuracy, better stability (consistent results), and better scalability of the algorithm because only a portion of the data is used at a given step. The sliding window approach also address the possible issue of seasonal variance causing difficulty in cluster; by taking a small window of data at a time, the influence of seasonal variation will be reduced. Continuing with the example of choosing a window size of 384, this provides 121 windows with the ~15-months of data in this dataset. Even if customers are missing data during some of those windows, there are enough windows to provide a number of separate clustering instances and votes for each customers’ predicted phase. This, of course, depends on the distribution of missing data for a particular customer, but this window size works well with the customers present in this dataset.
4.3 Spectral Clustering

This research uses spectral clustering as the clustering algorithm in this methodology. Spectral clustering is a clustering algorithm that first computes a non-linear dimensionality reduction on the input data and then uses the k-means algorithm to cluster that representation of the data. The dimensionality reduction step is the key difference that separates this clustering algorithm from other clustering algorithms. See Table 2 for a summary of the spectral clustering algorithm.

*Table 2 - Spectral Clustering algorithm*

<table>
<thead>
<tr>
<th>Spectral Clustering Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Create an affinity (similarity) matrix using a pairwise kernel</td>
</tr>
<tr>
<td>2. Non-linear dimensionality reduction</td>
</tr>
<tr>
<td>2.1 Compute Laplacian matrix</td>
</tr>
<tr>
<td>2.2 Compute the eigen vectors to use as feature vectors</td>
</tr>
</tbody>
</table>
3. Cluster using the eigen vectors

In Step 1, an affinity (or similarity matrix) is created from the voltage profiles. In the SciPy implementation, this can be a precomputed matrix or one of the built-in pairwise kernels, such as the radial basis function kernel, polynomial kernel, or the cosine similarity kernel. This provides the relative similarity between each pair of customers. Then in Step 2, the Spectral Clustering algorithm does a non-linear dimensionality reduction using the affinity matrix. This is the critical step for the efficacy of the algorithm. The Laplacian matrix is computed from the affinity matrix and then the corresponding eigen vectors are computed from the Laplacian matrix. The eigen vectors are then used as representative feature vectors. In Step 3, the clustering is done using the eigen vectors. The usual (and default for SciPy) clustering scheme is K-means, using the K-Means++ initialization scheme. Effectively this means that the only differences are the choice of similarity (or distance) metric and the dimensionality reduction in Step 2. Creating that representation of the data in terms of the eigen vectors is what gives this methodology its power. For a more detailed explanation of spectral clustering please see [29].

Parameters that need to be determined in advance for the Spectral Clustering algorithm include, the number of clusters and the pairwise kernel to compute the affinity matrix. Then, outside of the spectral clustering algorithm itself, a determination also needs to be made about what representation of the voltage profiles to use as input for the
algorithm and what window size to use for the sliding window ensemble approach. The final parameters were the ‘Radial Basis Function’ kernel to compute the affinity matrix, to use 30 clusters for each sliding window iteration of the algorithm, and a window size of 384 samples (~ 4 days). For a more detailed treatment of how those decisions were made and what parameter sweep experiments were conducted please see the Parameter Determination section.

4.3 Evaluation Metrics & Validation

Two critical components of this research are the issues of evaluating the quality of the clustering techniques and validating the results of which customers are potentially mislabeled by the utility company. The utility knows that some percentage of its customers are labeled as being on the wrong phase, although they are uncertain as to the specific percentage and determining that, as well as re-labeling the incorrectly labeled customers, is the overall goal of this research. It is also reasonable to suspect that there are other types of mislabeling errors in the utility model. If there are certainly phase labeling errors, what other types of errors are likely to be present in the utility model? Possibilities include single-phase customers labeled as three-phase customers, three-phase customers labeled as single-phase customers, customers who are correct geographically but are labeled as being connected to the wrong transformer, customers who are incorrect geographically, customers with errors in their identification numbers, etc. The last one is interesting; in this dataset for Feeder 1, there are 41
customers with IDs and data that are not found in the OpenDSS utility model; ‘not found’ means that those IDs do not have geographic coordinates or phase labels.

Without complete field validation of the model, which is prohibitively expensive for most real-world systems, there will be some percentage of error, and it’s worth considering what types of errors are possible, in what quantities, and how that might affect the results of this type of clustering. In light of all of those considerations, this leads us to the question of how should the different clustering parameters and inputs be evaluated?

Each of the following metrics has advantages and drawbacks and only by considering each of them does the full picture emerge of the quality of this clustering methodology.

The first metric used for comparing clustering instances for this research is Purity. In general, metrics for determining the quality of a clustering require the ground truth labels, which in this case, we know there are some errors in the labels. Purity can be defined as taking the sum of the majority predicted label customers in a cluster divided by the total number of customers over all of the clusters. More formally,

$$\frac{1}{|N|} \sum_{k} \max_{i} [d_k \cap c_i]$$

Where N is the set of customers in the feeder, k ∈ K where K is the set of clusters, i ∈ I where I is the set of classes (phases labels in this case), d_k is the set of customers in
cluster \( k \), and \( c_i \) is the set of customers labeled with class \( i \). This Purity metric allows the comparison of clustering simulations with different parameters. One drawback of this metric is that it is necessary to know that the clustering was successful in general (meaning that there are indeed distinct clusters for the three phases) before considering the Purity score; for example, if all customers were predicted to have phase A, the Purity score would be 1. This check can be accomplished by visual inspection of a plot of the clusters or by considering a ‘cluster balance’ metric. This metric is useful in comparing some of the possible parameter configurations of the clustering algorithms, however a perfect Purity score is not to be expected. It is known that some of the utility labels are inaccurate, so in fact, the interesting portion of the results are the ‘impure’ portion of the clustering. For this reason, other metrics are also necessary to fully quantify the quality of the clustering results.

A Confidence score is the next metric used in this research. Recall that this research uses the ‘sliding window’ approach to utilizing the entire time series of voltage measurements. The Confidence score quantifies how consistently each customer is classified as being on the same phase across each of the windows. More formally,

\[
max_i(v_{ni}) / \sum_i v_{ni}
\]

where \( n \in N \) where \( N \) is the set of the customers in the feeder, \( i \in I \) where \( I \) is the set of classes (phases labels in this case), and \( v \in V_n \) where \( V_n \) is the votes for each phase.
received by customer n. For example, if the window size is 384 samples then there are 121 windows total for this dataset. So, for each window each customer will receive a predicted phase ‘vote’. Consider a situation where a particular customer is predicted to be Phase A on 115 windows, Phase B on 2 windows and is removed (because of missing data) from 4 windows. Then the confidence score for that customer will be \( \frac{115}{117} \approx 0.983 \). A confidence score of 1 is desirable as that implies that all windows predicted that customer to be on the same phase for each window, a high confidence prediction. Bear in mind that this metric implies nothing about the correctness of the predictions, simply a measure of how ‘confident’ the algorithm is about the prediction.

The Consistency score quantifies the consistency of the clustering algorithm over multiple individual instances of applying the method. Recall that the final step in spectral clustering uses k-means, and there is randomness introduced in the initialization of the initial centroids for k-means. The ensemble nature of the sliding window approach will likely reduce that random influence. A Consistency score for a customer is defined as the percentage of individual simulations that the customers was predicted in the final ensemble prediction to be on the same phase. To obtain this score Monte Carlo simulations were run, where the algorithm was repeated some number of times to measure the effects of the randomness introduced. More formally the Consistency score is,

\[
\max_i(s_{ni}) / M
\]
where $i \in I$ where I is the set of classes (phases labels in this case), $n \in N$ where N is the set of the customers in the feeder, $s \in S$ where S is the set of final ensemble predictions for all of the Monte Carlo simulations, and $s_{ni}$ are the final votes for each phase label over all the Monte Carlo simulations, M is the total number of Monte Carlo simulations run. For example, if 200 Monte Carlo simulations are run, and for a particular customer the resulting votes are: 195 runs of the algorithm produced a final prediction of Phase A, and 5 runs produced a prediction of Phase B then the Consistency score would be $(195/200) = 0.975$. A Consistency score of 1 is ideal because that implies the algorithm is producing consistent predictions for that customer between individual runs of the algorithm. Bear in mind that this metric implies nothing about the correctness of those predictions, just the consistency.

The final metric defined to examine the results of this research is called topology validation, and it uses the topology of the utility model itself. Each customer is labeled as being connected to a specific transformer (with the exception of the ‘unknown’ customers) and then that transformer is also, potentially, part of a larger lateral, where a lateral is several nearby transformers on the same phase that are grouped together. Using that topology information, it is possible to validate the algorithm predictions. All customers that are connected to the same transformer must, by definition, be on the same phase. All transformers that are connected to the same lateral must, by definition, be connected to the same phase. Thus, the clustering must also follow these
patterns. If the clustering algorithm has predicted that all customers labeled on the same transformer are the same phase (regardless of whether that is the utility labeled phase or not) that is regarded as a ‘correct’ or topologically validated prediction. If, for example, 4 out 5 customers on a transformer are labeled as one phase and the 5\textsuperscript{th} customer is labeled as a different phase, the 5\textsuperscript{th} customer would be considered ‘inconsistent’ according to topology validation and contribute to the set of ‘inconsistent’ customers. The Inconsistency Rate is defined as the percentage of customers that are considered ‘incorrect’ according to the topology validation divided by the number of customers predicted to be on a different phase from the utility model. More formally,

\[
\frac{|T|}{|\{x_n | (p_n \neq l_n)\}|} \times 100
\]

Where \(T\) is the set of customers considered ‘incorrect’ according to the topology validation method, \(p \in P\) where \(P\) is the set of phase predictions for all customers, \(l \in L\) where \(L\) is the set of utility phase labels for all customers. For example, if 50 customers are predicted to be on a different phase from the utility model and 45 of them are considered ‘correct’ according to the topology validation and 5 of them are considered ‘incorrect’ according to topology validation then the Inconsistency Rate for that clustering would be \((5/50) \times 100 = 10\%\)
4.4 Parameter Determination

The parameters for the Spectral Clustering algorithm were primarily determined using parameter sweeps of plausible combinations of parameters. The choice of the kernel to use is one of the first decisions that needs to be made. Kernel options that are implemented in the sklearn Python libraries include the ‘radial basis function’, ‘polynomial’, ‘sigmoid’, and ‘cosine’. The radial basis function (RBF) kernel was chosen in the end. Radial basis function kernels and polynomial kernels were thoroughly investigated, and the other kernels appear to have similar performance. Figure 16 shows a comparison of the RBF kernels with the Polynomial Degree 3 kernel using the delta voltage input; red x’s indicate clustering failure. The x-axis is the sliding window size given in number of measurements; 384 measurements corresponds to ~4 days. Keep in mind, these results are based on single simulations, and that is the cause of the spikes in the graph. These results show nearly identical performance between the two kernels. The RBF kernel was chosen for the final architecture because in using the raw voltage inputs, the Polynomial kernel failed on the window size of 384.
Principal Component Analysis (PCA) was tested as a possible step prior to using the Spectral Clustering algorithm. For this analysis, the input profiles, either raw voltage or delta voltage, were reduced using PCA and then the reduced form was used as the input to the Spectral Clustering algorithm. Figure 17 shows using PCA with 30 components versus without using PCA in a sliding window size sweep. Feeder 2 shows better Purity without PCA and the other two feeders appear to have nearly identical Purity scores. Figure 18 shows that as long the number of components is large enough, performance in terms of Purity scores is nearly identical. Note WS stands for ‘Window Size’. Finally, Figure 19 shows the Consistency versus Confidence scores over 200 Monte Carlo
simulations using the raw voltage profiles as input. This figure clearly shows that using PCA results in deteriorated performance in both the Consistency and Confidence metrics. Note that there are some low consistency customers plotted in red for the version without PCA, they are simply covered by the blue markers. So, PCA is not useful as a pre-processing step for the Spectral Clustering for this dataset.

Figure 17 - Parameter Determination, PCA versus No PCA using 30 components
Figure 18 - Parameter determination, sweep over the number of PCA components
Determine the size of the sliding windows is another parameter that must be chosen. The larger the window size, the fewer windows need to be clustered but also the fewer ‘votes’ in the ensemble. Looking at Figure 20, the colors represent different window sizes, with the Consistency scores and Confidence scores. Window sizes of either 384 or 960 are the best according to these metrics. The Consistency scores are much lower for a window size of 96, and although larger window sizes do not appear to be much worse than 384 or 960, it’s important to notice the ‘lost’ numbers in the legend. This represents customers that had no ‘clean’ windows at all and were therefore never clustered. The probability of that occurring gets larger as the window sizes increase.

*Figure 19 - PCA versus without PCA plotting Consistency and Confidence Scores*
Although no customers were lost in Feeder 1 for a window size of 960, there were customers lost in Feeder 2 and Feeder 3. For this reason, the smaller window size of 384 was chosen. This matches the results from [7].

![Figure 20 - Parameter determination, Consistency versus Confidence score for different sliding window sizes](image)

Finally, spectral clustering requires that the number of clusters be determined in advance. 30 clusters was used in the final implementation of this algorithm. A range of possible values was used in the experiments. Small numbers of clusters did not work well, sometimes not producing a plausible clustering (all customers predicted on one phase for example). At around 30 clusters, the algorithm consistently produces plausible clustering results. By increasing the number of clusters the risk of obtaining
singleton, or very small, clusters increases. If this were to occur it would invalidate the strategy of using the majority vote of the utility labels in a cluster to assign the predicted phase. Therefore, 30 clusters was chosen as a compromise between those competing factors.
5. Results

The results section is broken into three sections, Overview of Spectral Clustering Results, Comparison of Results for Other Feeders, and Comparison to K-Means.

5.1 Overview of Spectral Clustering Results

Figure 21 shows the clustering results from one run of the Spectral Clustering algorithm. The plots from left to right show the utility labels and the colors show the phase predicted by the clustering, where Green is Phase C, Blue is Phase B, Red is Phase A, and Cyan is Unknown. The first bar graph represents the customers where the utility model did not have phase labels, see Figure 5. There a few customers remaining that have a predicted phase of ‘unknown’. This means that those customers were grouped only with other ‘unknown’ customers for the majority of the sliding windows. For one possible approach to obtaining further information about those customers see the Unknown Customers Locations sections. Working from the assumption that the majority of the utility labels are correct, we would expect to see the clustering predicted phases to match the utility labeling which is what we see here. The clustering matches the utility labeling for ~91% of the customers. The remaining 9% of customers are predicted by the clustering algorithm to be errors in the utility labeling for those customers. In the end ~7.2% are predicted to be errors in the utility phase labeling and ~1.8% are predicted to be other types of errors in the utility model. See Table 3 for a customer breakdown of the Spectral Clustering results, and Table 4 shows the
distribution of the original utility phase labels versus the predicted labels. Since the ‘unknown’ customers do not have geographical coordinates in the model, they are excluded from the results in Table 3 and Table 4.

**Table 3 - Spectral Clustering results breakdown**

<table>
<thead>
<tr>
<th>Feeder 1 Customers</th>
<th>Total Customers</th>
<th>Validated Utility Labels</th>
<th>Corrected Utility Labels</th>
<th>Other Errors Detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customers</td>
<td>1055</td>
<td>957</td>
<td>79</td>
<td>19</td>
</tr>
<tr>
<td>Percentages</td>
<td>100%</td>
<td>~90.7%</td>
<td>~7.5%</td>
<td>~1.8%</td>
</tr>
</tbody>
</table>

**Table 4 - Feeder 1 utility labels versus predicted labels**

<table>
<thead>
<tr>
<th>Clustering Predicted</th>
<th>Utility Label</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>A</td>
<td>506</td>
<td>24</td>
</tr>
<tr>
<td>B</td>
<td>10</td>
<td>229</td>
</tr>
<tr>
<td>C</td>
<td>48</td>
<td>5</td>
</tr>
<tr>
<td>Total</td>
<td>564</td>
<td>258</td>
</tr>
</tbody>
</table>
Two types of validation were done for this methodology. First is the topology validation, shown in Figure 22. Circles represent customers predicted to be on a different phase than the utility label, with green representing prediction that are reasonable given the topology, and red circles are predictions that contrast with the utility that do not seem reasonable given the topology. In total, 98 customers are predicted to be on a different phase than in the utility model, 79 customers’ predicted phases seem reasonable (plotted in green) under the topology validation method and 19 seem unreasonable (plotted in red). Note that the green and red circles mark
transformers so most circles mark multiple customers. In particular, notice the four transformers marked within the blue oval in Figure 22. These four transformers that compose the lateral labeled as Phase B shown under the blue oval are all predicted to be an error in the utility model. Figure 23 shows the Google Earth satellite view of those four transformers and the fifteen customers that they serve. All fifteen are labeled as being on Phase B, however the clustering algorithm has predicted that all fifteen are actually on Phase A. This is a striking example of the topology validation method at work. The clustering predicted an error, and the underlying customer-transformer connection labels strongly indicate that this whole lateral is in fact on Phase A as predicted.

Please see the Additional Advanced Grid Applications section for a detailed analysis of the customers that the Spectral Clustering algorithm predicts are incorrect in the utility model but seem unreasonable given the topology validation method; that section will show that those customers being ‘unreasonable’ based on topology is actually indicative of other types of model errors.
The second type of validation is a subset of customers verified by street view. The topology validation provides the first layer of validation, and unfortunately street view is
inconclusive for the majority of customers, as the street view images are not designed for this type of work. However, a subset of customers can be verified with reasonable certainty using street view. Figure 24 shows the satellite view of the original utility labeling of Phase C for this customer, and Figure 25 shows the street view of the transformer, clearly showing a Phase B connection to the middle wire which verifies the clustering prediction of Phase B for this customer. Up to this point, the two-layer validation accounts in a reasonable way for 98% of the customers, 91% match the original utility labeling, and ~7% are validated using the topology validation and then street view.

![Figure 24 - Phase labeling error, example #1, satellite view](image-url)
Turning to the Consistency and Confidence of the methodology, Figure 26 shows a bar graph of the distribution of the Confidence Score metric for this run of Spectral Clustering. You can see that the majority of customers have high confidence scores, and only a relatively small number of customers have low confidence scores. For a discussion of the calculation of the Confidence Score, please see the Evaluation Metrics & Validation section.
From these results we can see that the Spectral Clustering algorithm results for Feeder 1 are quite promising.

The final step in Spectral Clustering is a $k$-means clustering, and so there is still an element of randomness present in the algorithm. The following results quantify what that randomness looks like in practice.

Figure 27 plots the Consistency scores versus the Confidence scores for each customer over a 200 run Monte Carlo simulation. Each blue marker represents one of the ~1000 customers. As would be expected, all customers with consistency scores < 1 also have lower Confidence scores. The other conclusion to take away from this figure is that the Spectral Clustering algorithm is quite consistent overall. There are only a handful of
customers that are ever given multiple final classifications. Note that the Monte Carlo simulation is only used to explore the consistency of the methodology, given the influence of the random factors introduced by the k-means step. Actual usage of the algorithm does not use a Monte Carlo approach.

![Graph showing Consistency scores versus Confidence scores in a Monte Carlo simulation](image)

*Figure 27 - Consistency scores versus Confidence scores in a Monte Carlo simulation*

The results shown above use the delta voltage profiles as inputs. For Feeder 1 and Feeder 3, delta voltage performed significantly better than the raw voltage profiles as inputs, achieving better consistency scores. However, for Feeder 2 the Consistency score results are better using the raw voltage than the delta voltage. The reason for this
difference in response by the feeders is unknown and will be left to future work to be investigated further.

These results overall show that this methodology works well and is extremely promising as a phase identification tool to validate and improve utility models. Leaving aside the 2% of customers inconsistent with the topology labeling, this method accounts for 98% of the customers in a plausible way. 7% of the customers’ labels have been corrected from the original labeling and the remaining 91% of labels have been verified as correct. The phase predictions are validated with the topology validation as well as a subset verified using street view images.

5.2 Comparison of Results for Other Feeders
The following are results for the other two feeders in the dataset, Feeder 2 and Feeder 3. One factor communicated by the utility company whose data is used in this research is that the feeder model for Feeder 1 is one that they have spent time updating and cleaning for their own purposes. They expect there to be, in general, more errors and discrepancies in the Feeder 2 and Feeder 3 models compared to the Feeder 1 model.

The data characteristics are similar between the three feeders, however there are some differences in the feeder topology. See Table 1 for details. Note that Feeder 1 is the least complex of the three, Feeder 2 has a set of single-phase regulators and a 450 kVar capacitor, and Feeder 3 has two 300 kVar capacitors. That additional complexity could be a factor in the results, and one possible avenue of future work could be to investigate
the specific effects of the line regulators and capacitors on the clustering results. Table 5 and

Table 6 show the distribution of the utility phase labels versus the predicted labels. To compare with Feeder 1, see Table 4.

Table 5 - Feeder 2 results, utility labels versus predicted labels

<table>
<thead>
<tr>
<th>Clustering Predicted</th>
<th>Utility Label</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>320</td>
<td>0</td>
<td>8</td>
<td></td>
<td>328</td>
</tr>
<tr>
<td>B</td>
<td>33</td>
<td>338</td>
<td>1</td>
<td></td>
<td>372</td>
</tr>
<tr>
<td>C</td>
<td>175</td>
<td>16</td>
<td>398</td>
<td></td>
<td>589</td>
</tr>
<tr>
<td>Total</td>
<td>528</td>
<td>354</td>
<td>407</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6 - Feeder 3 results, utility labels versus predicted labels

<table>
<thead>
<tr>
<th>Clustering Predicted</th>
<th>Utility Label</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>354</td>
<td>13</td>
<td>59</td>
<td></td>
<td>426</td>
</tr>
<tr>
<td>B</td>
<td>26</td>
<td>318</td>
<td>1</td>
<td></td>
<td>345</td>
</tr>
<tr>
<td>C</td>
<td>73</td>
<td>29</td>
<td>239</td>
<td></td>
<td>341</td>
</tr>
<tr>
<td>Total</td>
<td>453</td>
<td>360</td>
<td>299</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 28 shows the plot of Consistency scores and Confidence scores over a 200 Monte Carlo simulation of Feeder 1, Feeder 2, and Feeder 3 together. It is clear from the plot that there are more customers that have a Consistency of less than one in the other two feeders.
Figure 29 shows a plot of the Topology Inconsistency Rate (see section 4.3 Evaluation Metrics & Validation for a description of that metric) over a 200 Monte Carlo simulation. The number of customers that are seemingly inconsistent with the topology validation is larger for Feeder 2 and Feeder 3 and also more variable between simulations. The variation between simulations is potentially explainable in that the other two feeders had more customers with Consistency scores less than one and so it expected that the other metrics would reflect that variability. There are several possible explanations for
the larger number of customers that are inconsistent according to the topology validation. One explanation is that it is known that the utility has put more time and effort into improving the Feeder 1 secondary system model than the other two feeders. One evidence of this is that Feeder 1 has 41 customers that are labeled as ‘unknown’ and Feeder 2 and Feeder 3 both have ~20 customers labeled as unknown. Since those customers are ‘unknown’ (lacking geographical coordinates) they do not show up in the topology validation metric. Assuming there are similar rates of ‘unknown’ customers between the feeders, that means there are ~20 customers negatively affecting the topology validation metric in the Feeder 2 and Feeder 3 that are not present in the topology validation for Feeder 1. Another possibility is that the increased complexity of Feeders 2 and 3 (see Table 1) is causing issues with the clustering. Given that the majority of those customers in Feeder 1 labeled inconsistent under the topology validation metric were actually other types of topology errors, it is plausible to hypothesize the same is true for Feeder 2 and Feeder 3.
5.3 Comparison to K-Means

A key question is how Spectral Clustering compares to other possible algorithms for phase identification. Variations of k-means were used in [11], among others, so that was chosen as a baseline comparison. K-means is a member of the Expectation-Maximization (EM) family of techniques. See Table 7 for a summary of the K-Means algorithm. Following that is brief description of the K-Means algorithm, followed by the results comparison.
**Table 7 - K-Means algorithm**

<table>
<thead>
<tr>
<th>K-Means Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the number of clusters</td>
</tr>
<tr>
<td>2. Select initial centroids</td>
</tr>
<tr>
<td>3. Calculate distances from each datapoint to each centroids and assign each datapoint to a cluster</td>
</tr>
<tr>
<td>4. Calculate a new centroid for each cluster based on the average of all the datapoints in the cluster</td>
</tr>
<tr>
<td>5. Go to Step 3 and repeat until convergence or until a maximum number of iterations is reached</td>
</tr>
</tbody>
</table>

In step 1, the user must choose the number of clusters in advance. This is often difficult to determine, and it is a difficulty shared by both k-means and Spectral Clustering. In Step 2, initial centroids (or cluster centers) are chosen. The classic version of k-means initializes the centroids randomly, and this often leads to a sensitivity to poorly chosen initial centroids. The Python SciPy implementation supports initialization using the K-Means++ algorithm [30]. This has been shown to be superior to random initialization, and although k-means remains unstable in terms of producing the same results with each simulation, the K-Means++ initialization provides a better initial starting point than random guessing. In Step 3, the distances from each data point are calculated to each centroid. Classically, Euclidean Distance is used as the distance metrics (and that is the only metric supported in the SciPy implementation). Each data point is assigned to the closest centroid and that becomes its current cluster. In Step 4, a centroid is calculated by taking the average of all the data points in the new clusters. Then Steps 3 and 4 are repeated, either until the algorithm converges (no changes in cluster assignments) or
until a maximum number of iterations is reached. k-means makes the assumption that
the clusters will have similar density and are ‘spherical’ in shape. These assumptions are
not ideal in this situation. There is no guarantee that the actual phase clusters have
similar density. From Figure 5, we can see that the utility phase labeling has one larger
Phase A cluster and two smaller Phase B and Phase C clusters. We know there is some
error in those labels, but it does imply that we cannot expect the clusters to be of equal
density.

It is not necessarily a given that each sliding window clustering gives a plausible
clustering. As discussed, there are some random factors involved in the clustering, so
the initial conditions are critical. Also, due to removing customers with missing data in
the window, it’s likely that some windows will have either insufficient customers to do a
quality clustering, or a mix of customers that is not ideal to produce a quality clustering.
Figure 30 shows the results of looking at each individual sliding window clustering over
20 Monte Carlo simulations, a total of 2420 windows, and examining if that clustering
was a plausible clustering. ‘SP’ stands for ‘spectral clustering and ‘KM’ for k-means.
‘Failure’ to cluster was defined as the number of customers predicted by on Phase A,
Phase B, or Phase C was less than 10% of the customers the clustering was considered
to be a failure. This does not guarantee a quality clustering by any means, however
certainly if the clustering fails to meet this criterion, then it was a failed clustering
attempt. Based on this figure, the k-means windows failed nearly 6x more often than
the spectral clustering windows. As an element of possible future work, this failure information could be applied during the clustering process, discarding windows with clear indicators of failure, to further refine the final results.

Figure 30 - Comparison of clustering failure between Spectral Clustering (SP) and K-Means (KM)

Figure 31 shows the difference in the Consistency score versus Confidence score plot for spectral clustering and k-means over a 200 run Monte Carlo simulation using the sliding window methodology with a window size of 384 described in previous sections. The k-means results in blue show more customers with Consistency scores < 1, and this implies that the Spectral Clustering algorithm produces more consistent results than the K-Means algorithm.
These two figures together show that Spectral Clustering is both more consistent (with itself) in the results that it produces and that more of the sliding windows provide plausible clustering and are therefore more likely to be adding information to the final results. While not speaking conclusively to the accuracy of the final results, both of these indicators suggest that the Spectral Clustering algorithm is more stable overall than the K-means approach by itself.
6. Additional Advanced Grid Applications

Finally, one of the clear issues that arises from the results shown in the previous section is the customers represented in the Topology Inconsistency Rate. In the Feeder 1 results shown in Table 3, there are 19 customers where the predicted phase according to the clustering is an ‘inconsistent’ result based on the model topology. There are also other possible metrics for examining customers that are ‘problematic’ in some way. For example, customers that have a Consistency score < 1 or customers with low Confidence scores should also be considered. This section attempts to provide an explanation for what is occurring with those customers.

To investigate these customers, the results from a 200 run Monte Carlo simulation were used. A set of 22 customers was found that in any or all of the 200 simulations were classified as part of the set of customers that was inconsistent with the topology validation metric for that simulation. The reason that there are 22 customers instead of the original 19 is due to the random effect introduced by the initial conditions in the clustering. The differences between identical runs of the algorithm are small, but they do exist. It’s worth noting that all but 3 of the 22 customers has a Consistency score of 1, so in each of the 200 Monte Carlo simulations 19 of the customers were predicted to be on a consistent phase each time. Figure 32 shows a plot of the 22 customers on the grid model; keep in mind that the circles indicate transformers so some of them contain multiple customers out of the 22 total customers plotted. Each of those 22 customers
and the set as a whole were examined in an effort to explain their apparent inconsistency with the model topology.

First, they were examined to determine if there was some commonality between them to explain their presence as part of the failure rate, working on the hypothesis that the predicted phases for those customers were errors in the Spectral Clustering. However, at this time, no link has been found to connect those customers in this manner, and instead additional errors with the utility model were identified.

*Figure 32 - Feeder 1 Customers where the predicted phase is inconsistent with the topology validation*

Next, the customers were investigated as potentially being a different type error in the utility model, working under the hypothesis that the predicted are actually correct and
there was some error in the underlying model. This avenue of research produced some interesting results which suggest that the majority of those 22 customers with phase predictions inconsistent with the model topology are actually indicative of some other type of error in the utility model, primarily customers who are connected to a different transformer than they labeled as being on. The following results strongly indicate that many of those 22 customers represent additional errors in the utility model and that the methodology from this research can successfully detect not just phase errors, but other types of errors as well.

To investigate the possibility that these customers are actually connected to a different transformer than they are labeled as being on, the 10 most correlated customers were plotted using the pairwise correlation coefficient method. Those correlated customers were then examined for utility phase labels, predicted phase labels, geographical proximity, etc. With a couple of exceptions, all of the 10 most correlated customers predicted phases matched the predicted phase of the customer in question. This lends further weight to the clustering predictions. Then the customer in question as well as the most correlated customers were examined in a version of the grid model overlaid into Google Earth. It’s important to note that there are two different versions of the same feeder model, there is the one that the dataset came from and which is in OpenDSS, and the version model that interfaces with Google Earth; they are slightly different versions which do not agree in all cases. There is an example of this mismatch
in Figure 44. See Table 8 for a breakdown of what was discovered about the 22 customers that did not appear consistent with the model topology. The majority of them are plausibly explainable as being connected to a different transformer than they are labeled in the model. This method also identified a transformer not marked in the utility model, solar panels that were not marked in the model, and finally a group of customers where there are clear issues with the model surrounding those customers (unlabeled homes for example) but not a clear explanation of the error.

<table>
<thead>
<tr>
<th>Total Customers</th>
<th>Transformer Connection Error</th>
<th>Unlabeled Transformer</th>
<th>Unlabeled Solar Panels</th>
<th>Possible 3-phase customer</th>
<th>Inconclusive, but clear issues in the model nearby</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>13</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>6</td>
</tr>
</tbody>
</table>

To illustrate these inconsistencies, below are several examples that are representative of the types of findings for those 22 customers.

6.1 Transformer Connection Errors

Looking at Figure 33, we see an example of a predicted transformer connection error. The customer in question is plotted in yellow; the utility label is Phase B (green lines), the predicted phase is Phase A. All 10 of the most correlated customers are predicted to be on Phase A and they are, all on the transformer plotted in blue which is a Phase A transformer (red lines). This strongly suggests that the customer in yellow is in fact connected to the transformer plotted in blue instead of the transformer plotted in red.
This is a representative example of using the correlation coefficient methodology to locate additional errors in the utility model based on the customers that are ‘inconsistent’ with the labeled model topology.

Figure 33 - Incorrect transformer prediction

In general, the ‘street view’ photos are difficult to follow in enough detail to ascertain the interconnections clearly (including the previous example). However, following are two examples where the street view is reasonably clear and the findings support the hypotheses generated by the Spectral Clustering and the correlation coefficient methodology.

Figure 34 shows the original utility labeling for this section of the feeder. There are four transformers, all labeled as being on Phase A, and there is one house that is unlabeled in
the upper right of the figure. Customers 5 and 6 were on the list of customers that were inconsistent with the model topology. The dataset from OpenDSS also listed four other customers as being on that transformer as well and those customers are not in the Google Earth version of the model. So, all the customers on this street are labeled as Phase A, and there are four customers missing from Google Earth version of the model.

Now, compare this to Figure 35. Figure 35 has the actual phases and interconnections as predicted by the clustering algorithm and correlation coefficient method and further verified by street view. Transformer 51 is the only one of the transformers that is actually on Phase A. Transformers 50 and 52 are on Phase C, and Transformer 53 is on Phase B serving four customers. The house that was unlabeled in the original utility model clearly has four meters on it in Figure 36 and the transformer that serves that building is clearly connected to Phase B in Figure 37 (middle wire). Continuing down the street Customers 5 and 6 were predicted to be on Phase C, Customer 4 was predicted to be on Phase A (as labeled) and Customers 1, 2, and 3 were predicted to be on Phase C. Figure 38 shows transformer 51 connected to the wire nearest the street, Phase A in this case. Figure 39 shows that transformers 50 and 52 are connected to the wire farthest from the street, Phase C in this case. So, the topology shown in Figure 35 is fully validated by street view, confirming the phase predictions as well as identifying four customers that were labeled on the wrong transformer.
Figure 34 - Street view example #1, original utility labels

Figure 35 - Street view Example #1, clustering and street view labels
Figure 36 - Street view example #1, four-meter house

Figure 37 - Street view example #1, four-meter house transformer connection
Next, consider the original utility labeling in Figure 40. Customers 1-5 are labeled on Phase A (red lines) from Transformer 60, and Transformer 61 is labeled as Phase BC (light blue lines) serving Customers 5 and 6. Compare with the clustering and
correlation coefficient predictions confirmed by street view in Figure 41. Customers 1 and 2 were predicted to be Phase A, as labeled by the utility and they are indeed served by transformer 60. Customers 3, 4, and 5 were predicted to be on Phase B. Customer 8 (who was originally excluded from analysis because that customer was labeled as not being single phase) was the most correlated customer to each of Customers 3, 4, and 5, and the prediction is that customers 3, 4, 5, and 8 are connected to the same transformer. This was confirmed by the street view. In Figure 42, we can see transformer 61 connected to Phase A and serving Customer 6, and transformer 62 connected to Phase B serving customer 5 and then continuing down the street. Figure 43 shows that same Phase B line at the next pole down the street and serving customers 3, 4, and 5 on Phase B as predicted.
Figure 40 - Street view example #2, original utility labels

Figure 41 - Street view example #2, clustering and street view labels
Figure 42 - Street view example #2, transformers 61 & 62 connections

Figure 43 - Street view example #2, phase B continues down the street and serves another customer
Figure 44 shows a different situation. Customer 1 in this figure is contained in the set of customers ‘inconsistent’ with the topology validation and is the only customer predicted to be on Phase B while the remainder of the customers labeled on that transformer are predicted to be on Phase A. Looking at the figure, all customers, 1-4, are labeled as being on the transformer plotted in red. However, looking at the Google Earth version of the model, clearly customers 2-4 are connected to the adjacent, Phase A transformer plotted in blue, leaving just customer 1, connected to the Phase B transformer that it was originally labeled on in the dataset version of the model. What we see here is the clustering algorithm predicting the correct phase labels, and another version of the utility model that reflects the predicted situation. It appears that the utility corrected this error in the model at some point between the two versions. This further confirms the clustering phase predictions.
6.2 Unmarked Transformer

Looking at Figure 45, the original utility labels are on the right and the labels verified in street view are on the left. The customers in orange and the customer in yellow were all labeled as being connected to transformer 70, however the customers in orange were
predicted to be on Phase B and the customer in yellow was predicted to be on Phase A. Investigation of Google Earth and street view show that there is a transformer, plotted in red, that does not appear on the utility model. The transformer in red serves the orange customers and transformer 70 serves the yellow customer.

![Google Earth view of an unmarked transformer](image)

6.3 Unlabeled Solar Power

The customer plotted in yellow in Figure 45 clearly has solar power in the Google Earth satellite image, but, in the utility model, this customer is not marked as having solar
power. The Google Earth satellite imagery is from October of 2016 which is included in the time period covered by the dataset.

Figure 46 - Possible three-phase customer

Figure 47 shows a customer plotted in yellow that may actually be a three-phase customer instead of the Phase A (red lines) that is shown in the figure. It is next to the substation shown in red. However, the most correlated customers to the customer plotted in yellow contain customers of more than one phase which is unusual when plotting the most correlated customers. The hypothesis is that this is actually a three-phase customer rather than a single-phase customer. Street view is inconclusive, but the customer in yellow is adjacent the three-phase lines coming from the substation.
6.4 Unknown Model Errors

There are six customers out of the 22 which I have left as ‘inconclusive, but near model issues’. Figure 48 shows an example of this. Five of the six customers are shown here in yellow (there are two customers under one of the markers), the utility labeling shows the whole street, four transformers, to be on Phase A. However, most of the street contains unlabeled houses. There are power lines to those homes, and they clearly have power, however there are no meter numbers in the GIS utility model. That means there is an error in the model, at least meaning that information is missing, if not other errors as well. It’s unclear based on the available information what the topology actually looks like on this street, and the street view is not helpful in this case. However, I believe it’s
reasonable to hypothesize that there are model errors in play at this location. The other one out of the 6 that is inconclusive is also near issues in the model.

![Figure 48 - Example of clear issues in the utility model](image)

6.5 Unknown Customers Locations

There is an interesting extension of the concepts applied to analyze topology errors discussed in the previous section. In the description of the dataset, recall that Feeder 1 has 41 customers that are labeled by the utility as being ‘unknown’, they have customers numbers, receive power, are billed monthly, voltages are recorded, etc.; however, they are missing from the utility electrical model so they do not have phase labels and geographical coordinates (so they are not in the Google Earth model). From Figure 21 those 41 customers were assigned predicted phase labels in roughly equal proportions for the three phases. The correlation coefficient methodology described above was applied to these customers in an attempt to locate them within the feeder.
Two examples are described below showing a proof of concept that this methodology could be further developed and used to locate these types of customers within the feeder.

Looking at Figure 49, the top 5 most correlated customers were labeled as being on this transformer, labeled transformer 80 in the figure. The top four are plotted in yellow, and the fifth is not present in the Google Earth version of the model. The green customers are also on that transformer, but not in the top most correlated to the unknown customer. There are two unlabeled homes, plotted in red, that are connected to transformer 80 as well based on the street view images. The hypothesis is that the unknown customer is one of those two residences and likely the fifth most correlated is the other one.
Figure 50 shows an analogous situation. The top 2 most correlated customers are plotted in yellow. There is an adjacent home plotted in red that is unlabeled. Again, the hypothesis is that the unknown customer is that house.

These examples show that this method could potentially be developed to assist in locating these unknown customers within the model.
7. Future Work

The clear next step with this research is to compare the Spectral Clustering algorithm with other state-of-the-art methods. For example testing the results against the constrained k-means algorithm as used in [11], the Constrained Multi-Tree algorithm as used in [24], and the supervised learning techniques as used in [18].

Another interesting aspect of future work would be to implement a multi-step process of identifying customers labeled as being on incorrect transformers as a first step and then applying the phase identification as a second step. The pairwise correlation coefficient analysis shows promise as a methodology to identify incorrect transformer labels, and here it is used as a post-clustering analysis technique, however in practice it would be more effectively used as a pre-processing step prior to phase identification. Similarly, a step in the process of identifying and locating the ‘unknown’ customers on the feeder would also aid in further analysis involving the feeder and dataset. One could foresee a sequence of steps to correct these errors in series.

Both the Constrained Multi-Tree algorithm and the constrained k-means algorithm for phase identification incorporate knowledge of the underlying topology into the algorithm. This takes the form of ‘must-link’ constraints for the constrained k-means, for example customers on the same transformer must have the same phase. One of the interesting results from this research is the discovery of the other types of topology errors as a result of doing the phase identification followed by the topology validation.
and correlation coefficient methodology. This shows one of the risks of simply incorporating the topology information without verification that the model is accurate. Another aspect of future work might be to experiment with incorporating the topology information into the phase identification after correcting some of the topology errors that were found.

To effectively validate these results applying these methodologies to a fully field verified system would be ideal. However, obtaining that type of data from real world distribution systems of sufficient complexity as to be broadly applicable remains difficult.
8. Conclusion

Spectral clustering correctly identified the phase connections of customers and detected many other issues in the utility model. Although the utility had spent significant effort calibrating the model for Feeder 1, results show that the model likely contained errors for approximately 10% of customers. After spectral clustering is applied, there is an 80% reduction in model errors and uncertainty based on correctly identifying the phase connections of service transformers, as verified by Google Street View. Additionally, it was found that spectral clustering could be combined with other analytic techniques to identify and correct errors other than phase connections, such as pairing between meters and transformers, missing transformers, etc. Overall, 91% of the customers’ predicted phase labels matched the original utility labeling, ~7% of the customers’ phase labels were predicted to be incorrect in the utility model and were validated using topology validation and street view, and ~2% of customers were strongly indicated to be other types of errors in the utility model. Uncertainty in the electrical power system models can be a major limitation in implementing smart grid technology and designing the grid of the future. The proposed spectral clustering algorithm, with the sliding window ensemble, has demonstrated the ability dramatically reduce the uncertainty in the phase labels of the utility model. This research also demonstrates the potential of using machine learning to leverage the big data produced by AMI meters to positively
impact the power systems industry and enable utility companies to better calibrate their distribution system models.
9. References


