Australasian Universities Power Engineering Conference, AUPEC 2020, Hobart, TAS, Australia, 29 November – 3 December 2020

Prediction of Customer Connectivity to Distribution Transformer via Application of Machine Learning Techniques to Smart-Meter Data

Ravin Navintha Tennakoon
School of Engineering
RMIT University
Melbourne, Australia
rntennakoon@gmail.com

Aruna Tharanga Kulathilaka
School of Engineering
RMIT University
Melbourne, Australia
arunakulathilaka@gmail.com

Raju Giri
School of Engineering
RMIT University
Melbourne, Australia
rajugiri11@outlook.com

Nameer Al Khafaf
School of Engineering
RMIT University
Melbourne, Australia
nkhafaf@gmail.com

Lasantha Meegahapola
School of Engineering
RMIT University
Melbourne, Australia
lasantha.meegahapola@rmit.edu.au

Mahdi Jalili
School of Engineering
RMIT University
Melbourne, Australia
mahdi.jalili@rmit.edu.au

***Abstract*— Accurate topology information is essential for the efficient operation of a power distribution network. However, due to the constant changes in the physical topology, maintaining this accurate information is a challenging task. This research attempts to solve a common predicament faced by distribution network service providers (DNSPs), by providing an accurate connectivity model of the power distribution network. This would provide an accurate map as to which customer is powered by which distribution transformer. The dataset used for this study includes the average RMS voltage values from 710 smart meters powered by five transformers, every 15 minutes for five months. A classification model that can predict the smart meter connectivity to its respective transformer with an average accuracy of 90% is obtained. Initially, similarity values are calculated using the Euclidean distance measure by comparing the voltage profiles of each smart meter with that of a selected reference meter from each transformer. Next, this data is used to train the classifier using the algorithm created. Finally, an unknown set of smart meter data is tested using the trained model to observe the prediction accuracy. The solution provided could be further developed by adding features such as phase identification.**

Keywords*— *Classification, distribution networks, Euclidean distance, similarity measures, smart-meter data.

I. Introduction

Topology identification in distribution networks is the process of mapping the connectivity between the customer meters and their corresponding transformers [1]. Accurate knowledge of this network topology is crucial for feeder-level optimisation tasks [2], monitoring grid operations, state-estimations [3], and fault location identification. [4]. Lack of an accurate map could further result in significant financial penalties for energy distributors.

There exist sometimes inaccuracies between the actual distribution network and the network topology that is in possession of the energy distributors. This could be due to changes to the connectivity model caused over time with repairs and maintenance work [5]. It has been understood that a large portion of power loss occurs at the distribution network. Therefore, it has become a priority to model the connectivity accurately for the efficient supply of power [6].

Different methods and techniques have been used for topology identification in the published literature. The correlation between multiple voltage measurements have been presented graphically in [7], where an algorithm incorporating mutual information-based identification handles meshed and tree networks. In another approach, the correlation factors of node voltage profiles (i.e., customer voltage profiles) are used to determine the network topology [8]. As the correlation coefficient increases, the connection between nodes becomes closer [8]. Alternatively, topology identification is also carried out based on state estimator results and measurements. In one such approach, an algorithm incorporating the status of switching devices was proposed [9].

The adaption rate of smart meters in the distribution network is increasing rapidly around the globe. Already one-third of the total meters in Australia [10] have switched to smart meters. They can digitally measure, record and report the energy consumption, and voltage profiles of users to its providers, which can be ultimately used for analytical purposes. Measurements from smart meters have been used for accuracy management of current calculations in low-voltage feeders [11]. They have also been primarily used in power distribution networks for monitoring purposes, such as monitoring voltage fluctuations [6].

This paper presents a two-step approach to determine the connectivity between smart customer meters and their respective transformers. First, the original dataset is divided for training (80%) and testing (20%) in order to ensure that the data undergoing testing is not exposed in training. Choosing a suitable similarity measure is of utmost importance since it depends on the nature of the dataset that will be used for the procedure. The attributes of the dataset, such as the order (number of dimensions) as well as the number of attributes, should be considered.

As the first step, Euclidean distance measure is used to calculate the difference between voltage variations of meters within the training dataset with selected references. These differences are then modified to generate a similarity matrix, which is fed into a classifier. In the second step, the classifier used is considered, since the output from the similarity calculation should be compatible with the input for the classifier. In this paper, the similarity values are trained by the cubic SVM classifier with 10-fold cross-validation to generate a suitable model fit to make predictions of unknown voltage variations. Afterwards, voltage variations of smart meters which are not exposed for training are introduced to the trained model, from which a prediction is made.

In this paper, Section II provides a review on numerous similarity measures and classification techniques. Which is helpful when narrowing down to the most suitable method. Section III provides a detailed analysis of the steps undertaken to construct a model from machine learning techniques incorporating the MATLAB toolbox, and Section IV provides the results obtained through testing. Finally, the conclusions of the study are summarised in Section V.

II. Similarity measures and Classification techniques

As mentioned in Section I, there are two main steps involved in the process of obtaining an accurate connectivity model. Several techniques related to these steps are discussed below.

A. *Similarity Measures*

The similarity is a real-value measure of how much alike (similar) two objects are. It is commonly described as a distance with dimensions representing features of the objects. This distance could be either high or low, depending on the similarity of the two objects in consideration. A small distance value indicates a high degree of similarity, while a large distance value indicates a low degree of similarity. The concept of similarity is highly subjective, and its outcome varies significantly based on the domain and application. This means that two objects can be very similar as well as very dissimilar based on the parameter by which its similarity is measured.

Therefore, the process of selecting a similarity measure should be done very carefully. Furthermore, normalising the relative values of each feature is of utmost importance, since the inability to do so could lead to one feature dominating the entire calculation. Few similarity indices have been analysed below.

1) Euclidean Distance

The basis of the Euclidean distance is the Pythagorean theorem which demonstrates that in a right-angled triangle, the square of its hypotenuse is equal to the sum of squares of the other two sides [12]. According to the definition of Euclidean distance, if p and q are two point placed in an n-dimensional Euclidean space, then the Euclidean distance (D) between p and q can be calculated using the following formula.

$$D_{(q,p)} = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} \quad (1)$$

$$D_{(q,p)} = \sum_{i=1}^{n} \sqrt{(q_i - p_i)^2} \quad (2)$$

In a single-dimensional context, the distance between any two points becomes the 'Absolute value' of the difference of their coordinates. In such an occasion, the above equation can be simplified to the following.

$$D_{(q,p)} = |q - p| = \sqrt{(q - p)^2} \quad (3)$$

2) Manhattan Distance

It is the distance between two points measured along axes at right angles. Simply, the sum of the horizontal and vertical distances between two points placed on a grid can be expressed as the Manhattan distance [13] between these two points. This metric is also known as rectilinear distance, taxicab metric, or City block distance. For two points $p_1 \equiv (x_1, y_1)$ and $p_2 \equiv (x_2, y_2)$, the Manhattan distance (*d*) can be calculated as shown below;

$$d = |x_1 - x_2| + |y_1 - y_2| \quad (4)$$

3) Jaccard Similarity

The Jaccard similarity measure focuses on calculating the similarity between objects, which are points in space, or vectors. For this particular similarity measure, the objects in consideration are sets [14]. By definition, Jaccard similarity is given by dividing the cardinality of the intersection of sets, by the cardinality of the union of sets. If A and B are two known sets, the Jaccard similarity could be obtained from the equation provided below;

$$J(A.B) = \frac{|A \cap B|}{|A \cup B|} \quad (5)$$

If two sets share all the members, the similarity would be 100%, while the similarity would be zero if they do not share any members. Thus, the higher the percentage, the higher the similarity between the two sets.

4) Cosine Similarity

The aim of cosine similarity is to determine the normalised dot product of two vectors [15]. As illustrated in Fig. 1, the approach here is to calculate the cosine value of the angle between these two vectors. As the determining factor is an angle, cosine similarity as defined in (6), only provides an understanding of the similarity between their orientation and not their magnitude.

$$\cos\theta = \frac{\mathrm{A} \cdot \mathrm{B}}{|\mathrm{A}||\mathrm{B}|} \quad (6)$$

B. *Classification Models*

Classification is the process of "predicting the class of given data points" [16]. Here, in the presence of a trained set of accurately identified data, unsupervised data could be clustered into groups, entirely by comparing the similarities from the known dataset. The algorithm that implements the classification is known as the classifier; it is often a

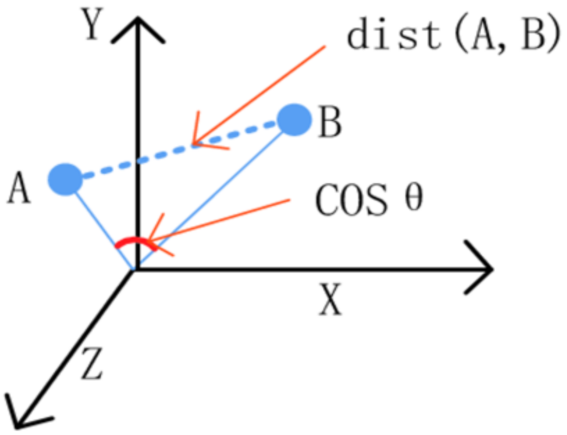

Fig. 1. Obtaining the cosine similarity between points A and B.

mathematical function which can classify unknown data into categories. Classifiers can be broadly categorised as discriminative vs generative, or lazy vs eager or as parametric vs non-parametric according to their various attributes. We have considered a number of classifiers in this study, and they are described below.

1) Support Vector Machine (SVM)

It is a discriminative type, machine learning algorithm which is commonly used for classification and regression algorithms. In the SVM algorithm, each point is represented as a data item within the n-dimensional space, where the value of each feature is the value of a specific coordinate [17]. SVM works by identifying the right hyper-plane which differentiates classes. Polynomial, linear, non-linear, radial basis function, etc. are some of the kernel functions used in the SVM algorithm.

2) Logistic Regression

It is another discriminative algorithm which predicts the probability that a given data entry belongs to a certain category [18]. It is used where the response variable is categorical. It could be applied when the response has only two values as well as when there are multiple possible values. This model has a relatively low variance, hence it is very efficient and accurate. However, accuracy could get minimised when many categorical features are present.

3) Random Forest

It is a method that has been developed by constructing a multitude of decision trees. This falls under the lazy learner category. It is mainly averaging out multiple deep decision trees which are being trained on different scenarios of the same training set and minimising the variance of it [19]. With this process, it dramatically enhances the performance and accuracy of the prediction, but also losses some extent of the interpretability.

Likewise, a wide variety of classifiers do exist and can be used in different methods based on the type of the inputs of the system as well as the complexity of the dataset. The accuracy of the classification process depends on the classifier used as well as the type of input used for the process. Therefore, it is important to analyse different classification procedures and to move forward with the most suitable technique by considering the dataset that is being used.

III. Methodology

In this research, a supervised machine learning algorithm is developed from residential smart meter voltage data which is correctly classified into five different transformers. This algorithm helps to predict the connectivity of unforeseen meter to its transformer. MathWorks' MATLAB is chosen as it is a robust platform that enables to manipulate and work with big matrices efficiently and smoothly. Fig. 2 shows the flow diagram of the methodology used in this work.

A. Similarity Calculation

The first and foremost task in approaching the transformer identification is to obtain a similarity measure between the voltage variation of a certain meter, with that of the reference with a transformer. The Euclidean distance measure is chosen as the most suitable similarity measure as it works well for a dataset with numeric attributes and treats all the dimensions equally. The initial algorithm developed is tested on a minimal dataset. It is purely done, to ensure whether the algorithm is performing as intended and to minimise other complexities that would arrive because of an enormous dataset.

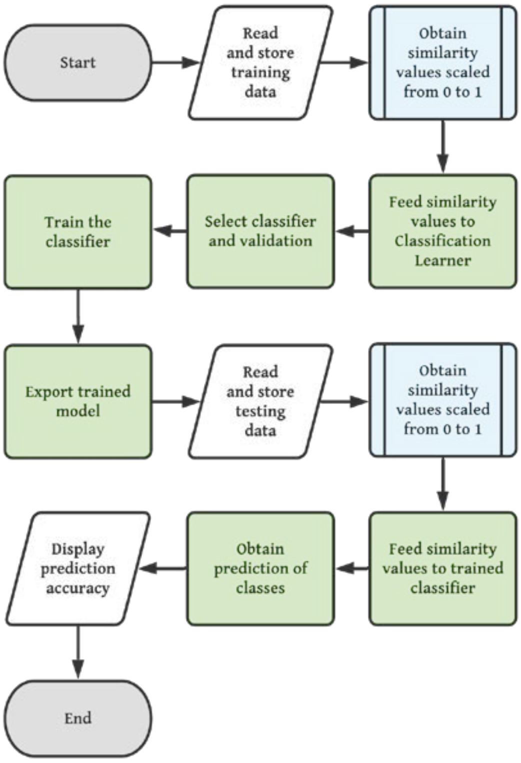
Start
Read and store training data
Obtain similarity values scaled from 0 to 1
Feed similarity values to Classification Learner
Select classifier and validation
Train the classifier
Export trained model
Read and store testing data
Obtain similarity values scaled from 0 to 1
Feed similarity values to trained classifier
Obtain prediction of classes
Display prediction accuracy
End

Fig. 2. Flow diagram of the overall methodology.

Later, 710 meters of data which records voltage variations every 15 minutes from 5 transformers over a 5-month period are added to the system. Next, the missing data within the dataset is tackled by replacing them with average voltage values corresponding to each transformer.

Similarly, a reference is created by averaging out the data of meters timeslot wise to calculate the similarity value for all five transformers. The similarity values for each of the meters with respect to the references from each transformer, are calculated using the formula for Euclidean distance. The references and meters are arranged together in a matrix for each transformer, and similarities are calculated using functions called “pdist (X)” and “square form”. The Euclidean distance between sets of items are calculated from these functions.

To ensure that the data used for testing is not exposed to the system during training, the entire dataset is divided into a training set (80%) and testing set (20%). The training set is solely used to train the respective classifier. In order to get a better understanding of the performance of the classifier as well as the impact of the dataset on the accuracy of the final prediction, three different simulation trials are carried out. For this, training and testing datasets are selected as from the original dataset as shown in Table I.

B. Training the Classifier

The next step is to train the selected classifier. This is done by feeding the similarity values calculated above, along with their known classes, to the classifier. Then, with the help of 10-fold cross-validation, the classifier would train itself with the assistance of corresponding features (similarity values) to the classes (transformers) [20]. Initially, a comparison of machine learning algorithms is performed by statistical comparisons of the accuracies of trained classifiers on the training datasets. The algorithm was executed using five different classifiers and the difference in accuracy for each pair was estimated. Then the generalisation error and the variance of the classifiers were calculated across all feasible training sets to obtain the results independent of the data partitioning.

The created similarity matrix along with the known transformer classes are fed to the classifier to perform automated training to search for the best classification model. Data is explored and experimented with different features, specified validation schemes, trained models, and the results are assessed. The test is run multiple times on the same dataset, with another random partitioning, and the number of identical outcomes is counted.

Besides exploring different types of models, alternative methods, such as feature selection, Principal component analysis (PCA), non-negative matrix factorisation are also tested to improve the prediction accuracy. Feature transformation is a form of dimensionality reduction, which reduces the complexity of data and makes it much easier to represent and analyse. Likewise, the validation errors after training multiple models are compared, and the best model is chosen to generate the predictions.

TABLE I. DATASETS CREATED FOR TRAINING AND TESTING

Simulation Trial	Training	Testing
A	Initial 80%	Remaining 20%
B	Remaining 80%	Mid 20%
C	Final 80%	Remaining 20%

C. Testing unknown datasets and predicting the transformers

Once the training is complete, the next task is to test and classify the unknown dataset. Firstly, the Euclidian distances for the voltage variations of the unknown dataset is calculated following the same similarity calculation procedure carried out previously. Then, the similarity measures were fed into the chosen cubic SVM classifier model to classify the meters to their respective transformers. Finally, depending on the number of correct predictions, an accuracy value is obtained. The same test is carried out multiple times to obtain an average accuracy.

Multiple tests have been carried out by using all three datasets to understand the effects of the dataset on the classifier performance as well as gain an insight as to how the prediction accuracy could be improved.

IV. RESULTS AND ANALYSIS

To select the most suitable similarity measure, the classifier is trained with cubic SVM for five different distance measures and the results are compared as shown in Table II. Here, Minkowski distance (p-norm) is used, where $p = 2$ gives a Euclidean distance, $p = 1$ provides a Manhattan distance, and as p approaches zero, it starts approximating a Hamming distance.

The results of different distance measures give dissimilar accuracies due to the different characteristics of each distance measure. For example, the datasets need to be vectors for the cosine similarity measure, and the Jaccard measure performs well when the data is in sets form. Both cosine and Jaccard similarities support the use of continuous as well as categorical variables, but not time-series data, as used in this study. Since the dataset used in this research study is a discrete-time series, the hamming distance measure is not appropriate, which is only useful for binary data strings.

After comparing the distance measures, the Euclidean distance measure shows acceptable results, as it is better for lower dimensional data, such as the dataset used in this research study. Table II results confirm the selection of the Euclidean measure as the most appropriate distance measure for this study.

Once the Euclidean distance of each meter is calculated from all five references, similarity measures are obtained, and interrelationship of the five transformers is analysed using Pearson correlation. It is carried out to select the class feature while carrying out a classification to design an accurate and better classifier model.

Pearson correlation matrix is shown in Fig. 3. It indicates that the transformer 1 is negatively correlated with the transformers 2, 4 and 5. Still, with transformer 3, it is positively correlated. A positive correlation indicates that if the similarity of the transformer 1 increases then the similarity of transformer 3 also increases and vice versa. Likewise, all the transformer similarity relation to each other are identified; its analysis is carried out to select the class feature while carrying out a classification to design an accurate and better classifier model.

Fig. 4 illustrates the scatter plot for similarities obtained against references of transformer 3 on x-axis and 4 on y-axis. This depicts the distribution of classes 1 to 5 based on the selected features. It is observed that classes 1 and 3 are separated from the rest. Similar distributions could be obtained based on other features to identify further classes.

The receiver operating characteristic (ROC) curve for class 2 is represented in Fig. 5. This shows the true positive rate (TPR) versus the false positive rate (FPR) for the trained classifier. A TPR of 0.97 indicates that the classifier correctly assigns 97% of observations. Similar ROC curves can be obtained for the remaining classes as well.

Once the distance measure was selected, to justify the selection of cubic SVM classifier, further testing was conducted. Here, several classifiers are compared using 10-fold cross-validation and the results are shown in Table III. According to Table III, it is visible that the Cubic SVM classifier gives the highest training accuracy. Therefore, the selection of cubic SVM as the primary classifier is justified.

After selecting the best distance measure and classifier, tests are carried out on the remaining 20% of the data, which are not exposed for training. Using the simulation trials, A, B and C, a training model is created, and the testing for the remaining 20% data is performed three times, respectively. Test results obtained for different datasets are presented in Tables IV, V and VI. The total accuracy is calculated based upon the average accuracies obtained from the above results. Fig. 6 shows the final average accuracy of the testing.

TABLE II. RESULTS OF DIFFERENT DISTANCE MEASURES OVER A DATASET.

Distance Measures	Training Accuracy
Euclidean	92.50 %
Manhattan	30.33 %
Jaccard	36.23 %
Cosine similarity	44.31 %
Correlation	27.92 %

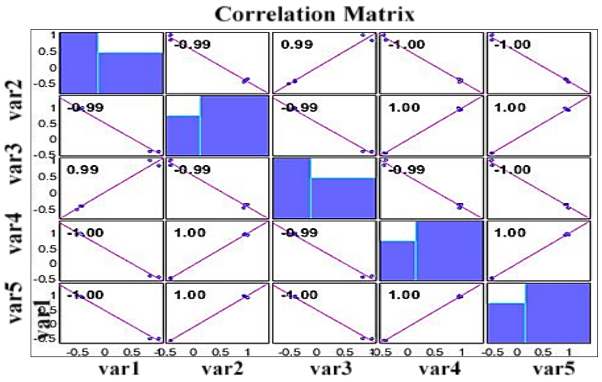

Fig. 3. Pearson correlation matrix.

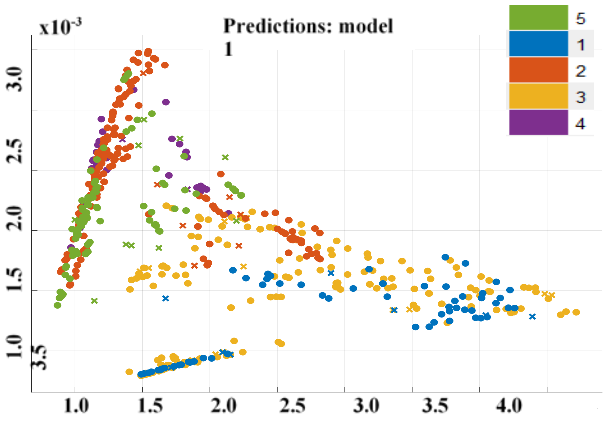

Fig. 4. Investigating features using a scatter plot.

TABLE III. CLASSIFIER TRAINING ACCURACY.

Classifier	Result Train Accuracy
Cubic SVM	92.5 %
Quadratic SVM	88.0 %
Fine Gaussian SVM	84.1 %
Weighted KNN	83.2 %
Ensemble Bagged Trees	79.3 %

TABLE IV. PREDICTIONS FOR TESTS CARRIED OUT ON DATASET A

Transformer	1	2	3	4	5	Average
Test 1 Accuracy (%)	63	96	96	89	64	81.6
Test 2 Accuracy (%)	63	96	96	89	82	85.2
Test 3 Accuracy (%)	68	96	96	89	82	86.2

TABLE V. PREDICTIONS FOR TESTS CARRIED OUT ON DATASET B

Transformer	1	2	3	4	5	Average
Test 1 Accuracy (%)	84	96	80	89	86	87
Test 2 Accuracy (%)	95	98	78	80	95	89
Test 3 Accuracy (%)	89	98	78	80	95	88

TABLE VI. PREDICTIONS FOR TESTS CARRIED OUT ON DATASET C

Transformer	1	2	3	4	5	Average
Test 1 Accuracy (%)	79	100	100	89	100	93.6
Test 2 Accuracy (%)	68	100	93	78	95	86.8
Test 3 Accuracy (%)	84	100	96	89	91	92.0

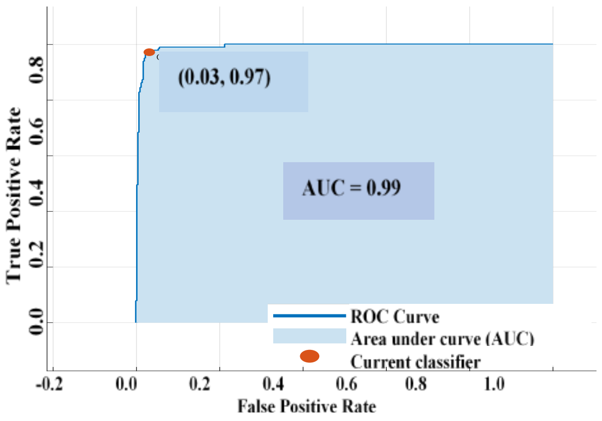

Fig. 5. ROC Curve for class 2.

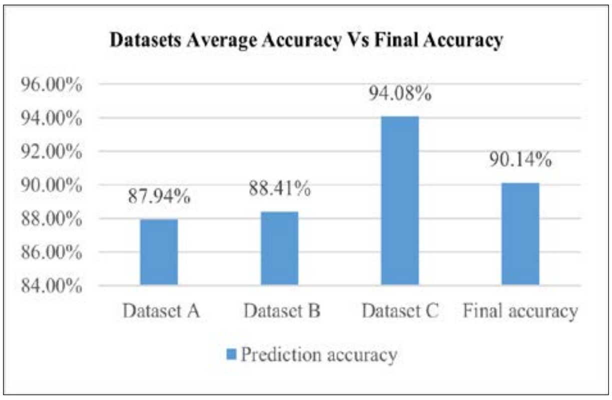

Fig. 6. The accuracy results.

As it is known that various parameters drive the machine learning, they mainly impact the results of the learning process. Proper parameter tuning can be obtained by finding the optimum value for these parameters to achieve a better result in future. Another technique called ensemble method, which can combine the outcome of multiple weak models and give a better result, can be used as well. However, this technique is more complicated than traditional methods and will be investigated in future studies.

V. Conclusion

In this research study, we have investigated the topology identification problem in low-voltage networks using smart meter data. We have proposed a machine-learning algorithm to classify customer smart meters to their respective transformers based on their voltage data. In the traditional approach, utility team members are dispatched for topology identification by examining the status of switches. Since this is costly and cannot be performed frequently, the proposed approach based on smart meter data is more cost effective and efficient. The main steps of the research study were identified to be calculating similarities, classification and prediction. Similarities were obtained by inverting and then scaling the calculated Euclidean distance values. Afterwards, cubic SVM classifier was trained by feeding the similarity values and finally, unknown voltage variations were fed to the trained classifier model to obtain a prediction on the connected transformer. Several simulations trials are performed and an average prediction accuracy of around 90% was obtained.

References

[1] W. Deng, Z. Zhang, J. Duan, X. Qiao, L. Zhu, and Y. Li, "Improved Topology Identification Algorithm of Distribution Network Mutual Information," *2019 IEEE Sustainable Power and Energy Conference (iSPEC)*, 2019.

[2] S. Taheri, V. Kekatos, and G. Cavraro, "An MILP Approach for Distribution Grid Topology Identification using Inverter Probing," *2019 IEEE Milan Powertech*, 2019.

[3] G. Cavraro and R. Arghandeh, "Power Distribution Network Topology Detection With Time-Series Signature Verification Method," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 3500-3509, Jul 2018.

[4] M. Farajollahi, A. Shahsavari, and H. Mohsenian-Rad, "Topology Identification in Distribution Systems Using Line Current Sensors: An MILP Approach," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1159-1170, Mar. 2020.

[5] S. J. Pappu, N. Bhatt, R. Pasumarthy, and A. Rajeswaran, "Identifying Topology of Low Voltage Distribution Networks Based on Smart Meter Data," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 5113-5122, Sep. 2018.

[6] G. Cavraro, V. Kekatos, and S. Veeramachaneni, "Voltage Analytics for Power Distribution Network Topology Verification," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 1058-1067, Jan. 2019.

[7] Y. Weng, Y. Z. Liao, and R. Rajagopal, "Distributed Energy Resources Topology Identification via Graphical Modeling," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2682-2694, Jul. 2017.

[8] W. P. Luan, J. H. Peng, M. Maras, J. Lo, and B. Harapnuk, "Smart Meter Data Analytics for Distribution Network Connectivity Verification," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1964-1971, Jul. 2015.

[9] G. N. Korres, N. M. Manousakis, "A State Estimation Algorithm for Monitoring Topology Changes in Distribution Systems," *2012 IEEE Power and Energy Society General Meeting*, 2012.

[10] "Falling behind 'down under." [Online]. Available: https://www.smart-energy.com/industry-sectors/smart-meters/falling-behind-down-under/#:~:text=Australia%20is%20behind%20the%20curve,a%20quarter%20of%20the%20country. [Accessed September 29, 2020].

[11] H. P. Schwefel, J. G. Rasmussen, R. L. Olsen, H. Ringgaard, N. Silva, and Ieee, "Using Smart Meter measurements to manage the accuracy of current calculations in LV feeders," *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (Smartgridcomm)*, 2019.

[12] C. E. Gartneer, "How to Find Euclidean Distance," [Online]. Available: https://sciencing.com/calculate-area-using-coordinates-8265405.html. [Accessed March 18, 2020].

[13] M. Goswami, A. Babu, and B. S. Purkayastha, "A comparative analysis of similarity measures to find coherent documents," *International Journal of Management, Technology and Engineering*, vol. 8, 2018.

[14] S. Glen. "Jaccard Index / Similarity Coefficient." [Online]. Available: https://www.statisticshowto.com/jaccard-index/. [Accessed Mar. 30, 2020].

[15] S. Prabhakaran. "Cosine Similarity – Understanding the math and how it works." Machinelearningplus. [Online]. Available: https://www.machinelearningplus.com/nlp/cosine-similarity/. [Accessed 2020].

[16] S. Asiri. "Machine learning classifiers," Towards data science. [Online]. Available: https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623. [Accessed 2020].

[17] "SVM Algorithm," [Online]. Available: https://www.educba.com/svm-algorithm/?source=leftnav. [Accessed September 14, 2020].

[18] Z. Bursac, C. H. Gauss, D. K. Williams, and D. W. Hosmer, "Purposeful selection of variables in logistic regression," *Source Code for Biology and Medicine*, vol. 3, no. 1, p. 17, 2008.

[19] A. Navlan. "Understanding Random Forests Classifiers in Python," 2020. [Online]. Available: https://www.datacamp.com/community/tutorials/random-forests-classifier-python. [Accessed: May 7, 2020].

[20] S. Yadav and S. Shukla, "Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification," *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, 27-28 Feb. 2016 2016, pp. 78-83.