International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)
ISSN: 2305-0543 (Online), ISSN: 2411-6173 (Print)
Vol. 11 (42), Oct. 2021, pp. 5050-5058

Detecting Electric Power Theft in Households Using Hidden Markov Models

* Erwin Normanyo
Department of Electrical and Electronic Engineering,
Faculty of Engineering,
University of Mines and Technology
Tarkwa, Ghana
enormanyo@umat.edu.gh

Muniratu Kassim
Department of Electrical and Electronic Engineering,
Faculty of Engineering,
University of Mines and Technology
Tarkwa, Ghana
kassimmuniratu19@gmail.com

***Abstract*— Electric power theft is a severe problem affecting the energy sector of a nation and continues to propagate and the energy sector tends to lose a chunk of its revenue resulting in higher tariffs for consumers innocent and guilty alike. Since electrical energy is widely used across the world and its demand is increasing rapidly, generation of electric power has to be increased or the already available one, economised. An effective system for detecting electric power theft is therefore needed so that when implemented, will either provide the nation with adequate funds to increase generation when households pay for energy that is consumed, or effectively utilise what is already available when households economise their energy consumption. This paper provides a means whereby power consumption of consumers can be monitored and recorded over a period of time. Based on that, the Hidden Markov Model (HMM) was used to predict the consumption with the deviations between real and predicted values giving indications of theft or otherwise. Appropriate sanctions can then be meted out to perpetrators of electric power theft. This way, higher tariffs are not imposed on innocent consumers and consumers will pay for the right amount of energy consumed.**

Keywords—Electricity tariffs, Households, Hidden Markov Model (HMM), Power consumers, Power theft detection

I. INTRODUCTION

Electrical energy is one of the most commonly used forms of energy in the world [1]. Electric power reaches consumers after generation through transmission substations, where some industries get their power or to the distribution substations where it is distributed to households. During these processes, there are some losses such as technical and non-technical losses. Technical losses are system losses mainly by heat dissipation caused by I^2R losses and others such as improper joints. Non-technical losses are caused by inaccurate meters, metering errors and tampering with meters.

According to Sakyi (2019) [1], in developing economies, a staggering USD 64.7 billion are lost each year to non-technical losses mostly due to electric power theft. The study concluded that the rest of the world aside South Africa, Brazil, India and China (which are associated with huge electric power theft), records a loss of USD 31.3 billion as a result of electric power theft on annual basis. This makes electricity the third most stolen commodity aside credit cards and cars. Electric power theft is not only rampant in developing countries but also in developed countries such as the USA and Canada. A loss of USD 6 billion was estimated in the USA in 2010 [2]. In 2011, the Electricity Company Ghana (ECG) reported that 30% of power supplied is lost through electric power theft and other illegal activities. Research conducted by the ECG in mid-2018 showed that 11,890 out of 250,616 (4.744%) electricity meters inspected had been tampered with [1].

Most of the aforementioned problems have been largely mitigated in recent years. Pre-paid meters were introduced in Ghana to help reduce the non-technical losses but they seem to have failed [3]. Even after its introduction before 2019, cases of electric power theft are still on the rise. Other designs such as Outlier Detection Algorithms (ODA) [4], Global System for Mobile Communication (GSM) based power theft detection [5] and Artificial Neural Networks (ANNs) [6] seem to have some limitations.

Though these designs seem promising, the ANN factors technical losses as well which contaminates the results and makes it difficult to differentiate electric power theft from I^2R losses for instance. The GSM technology-based electric power theft detection fails to factor in bypass of the meter and the ODA requires noise to be removed from the system as it may distort and blur non-theft cases from the theft cases. In this work, Hidden Markov Model (HMM) is proposed for electric power theft detection as it is able to deal with shortcomings of the other approaches reported in the literature.

II. RELATED WORKS

A lot of research has been conducted into detecting electric power theft. However, these works either have a problem with accuracy or ignoring other aspects of electric power theft. Table 1 provides the contribution and limitations of related works.

Article History:
Received Date: Apr. 17, 2021, Accepted Date: Jun. 02, 2021, Available Online: Oct. 01, 2021

TABLE 1 A TABULATION OF RELATED PUBLICATIONS ON ELECTRIC POWER THEFT DETECTION

SN	Author	Title of Paper	Contribution	Limitation
1.	Gu *et al.* (2017) [2].	Comparison of Machine Learning Techniques for the Detection of Electricity Theft.	Comparison between four machine learning techniques: LR, KNN, SVM and ANN.	The mathematical model is only suitable for simple methods.
2.	Yeckle and Tang (2018) [4].	Detection of Electricity Theft in Customer Consumption Using Outlier Detection Algorithms.	ODAs validation. ODAs application in Advanced Metering Infrastructure (AMI) in a feasibility study.	Design complexity: Requires noise removal prior to application.
3.	Nabil *et al.* (2018) [7].	Deep Recurrent Electricity Theft Detection in AMI Networks with Random Tuning of Hyper-parameters.	A generalised RNN based detection using hidden gated recurrent unit layers where hyper-parameter tuning was carried out for improved detector performance.	A sub-optimal solution due to random search usage in optimization stage.
4.	Ullah *et al.* (2018) [8].	A Prediction Mechanism of Energy Consumption in Residential Buildings Using Hidden Markov Model.	HMM used for accurate prediction of energy consumption and compared with SVM, ANN and Classification and Regression Trees (CART).	The HMM algorithm cannot be used in individual households.
5.	Huang *et al.* (2018) [6].	Energy Theft Detection via Artificial Neural Networks.	Historical data usage to train ANNs to detect electric power theft from incoming meter readings.	Technical losses cannot be differentiated from non-technical losses due to use of each consumer's historical data.
6.	Lydia *et al.* (2019) [9].	Detection of Electricity Theft based on Compressed Sensing	Accurate detection at a minimum cost using small number of sensors	Not applicable to distribution systems due to the small number of current sensors.
7.	Wu *et al.* (2018) [10].	AdaBoost-SVM for Electrical Theft Detection and GRNN for Stealing Time Periods Identification.	Improved classification performance. GRNN predicts energy consumption and identifies theft intervals	High relative error between actual consumption and predicted consumption.
8.	Sekhar and Puviarasi (2018) [11].	Simulation of GSM Based Power Theft Detection Using Proteus	Meter tampering or bypass detection using an embedded IR device to flag a PIC controller.	Use of a mobile number for GSM communication: Not suitable for deployment in utility companies.
9.	Mucheli *et al.* (2019) [12].	Smart Power Theft Detection System.	Electric power theft detection based on distributor box located between electric pole and energy meter	Tapings can be made before the distributor box therefore there can be bypass.
10.	Ask *et al.* (2018) [13].	Design and Simulation of Smart Prepaid-Postpaid Energy Meter with Alarm and Theft Control.	Automatic billing, data sharing between customer and utility company in prepaid or postpaid meter.	Consumer cannot receive real-time information that the buzzer gives for purchase of electricity notably in his/her absence.
11.	Ogu and Chukwudebe. (2017) [14].	Development of a Cost-Effective Electricity Theft Detection and Prevention System based on IoT Technology.	Utility company gets meter readings via the internet. Theft is as well visualised.	Only human interference is detected by infrared sensor
12.	Zheng *et al.* (2018) [15].	A Novel Combined Data-Driven Approach for Electricity Theft Detection.	High accuracy of detection due to application of Maximum Information Coefficient (MIC).	Extra cost goes into the purchase of observer meters
13.	Singh *et al.* (2018) [16].	Minimising Energy Theft by Statistical Distance based Theft Detector in AMI.	Use of Jensen-Shannon distance, Hellinger distance and Cumulative Distribution Function based distance	Malicious energy consumption data needed for training the algorithm could not be readily obtained.
14.	Nabil *et al.* (2019) [17].	PPETD: Privacy-Preserving Electricity Theft Detection Scheme with Load Monitoring and Billing for AMI Networks.	Use of privacy preserving machine leaning model. Load monitoring and bill computation using dynamic pricing are achievable.	Slightly low detection rate due to security feature.
15.	Zhou *et al.* (2017) [18].	Energy Theft Detection in Multi-Tenant Data Centers (MTDC) with Digital Protective Relay (DPR) Deployment.	A premier work that enables theft detection in MTDCs making use of Minimum Covariance Determinant (MCD) based anomaly identification technique.	No learning feature in the face of theft activity variations by tenants causing MCD solution deterioration.

Electric power theft detection saw deployment of certain machine learning techniques such as SVM, LR, KNN [2], ANN [6], HMM [8], CS [9] and GRNN [10] in different works to detect electric power theft and in the case of HMM [8], power consumption. Other AI techniques namely, ODA [4], MCD [18], data-driven approaches [15], Jensen-Shannon distance, Hellinger distance and Cumulative Distribution Function (CDF) based distance [16] were also applied to aid in electric power theft detection. The non-AI based electric power theft detection methods such as GSM-based [11], distributor box inclusion on consumer lines [12], smart prepaid-postpaid energy meter [13] and IoT technology-based [14] detections were used as well. All the mentioned methods proved to be effective to some extent.

From the literature, it is clear that the non-AI based techniques are more disadvantageous whiles the AI-based mostly deal with supervised learning under machine learning and look more promising. Energy consumption however, has other factors that contribute to it. Factors such as weather condition and appliances used in each household are hidden factors. It is for this reason that this paper proposes an unsupervised learning technique namely, HMM to be able to factor in other external factors that go into energy consumption in the process of electric power theft detection. HMM has already been exploited in electric power theft detection but for residential buildings [8]. The algorithm used was the Baum-Welch algorithm which was unable to handle a large data size [19]. This research is therefore focused on households which are common in Ghana. The Viterbi algorithm was used instead since it handles large data better [19].

The contribution of this paper is the development of electricity theft detection system for consumers in households. The novelty of paper is the use of HMM and application of Viterbi algorithm in electricity theft detection.

The rest of the paper is organised as follows. Section 3 presents the methodology that comprises design concept, mathematical modelling and computer programming. The results and discussion are availed in Section 4 and Section 5 gives the conclusion.

III. METHODOLOGY

A. *Design Concept*

Fig. 1 gives a graphical representation of the electric power theft detection system.

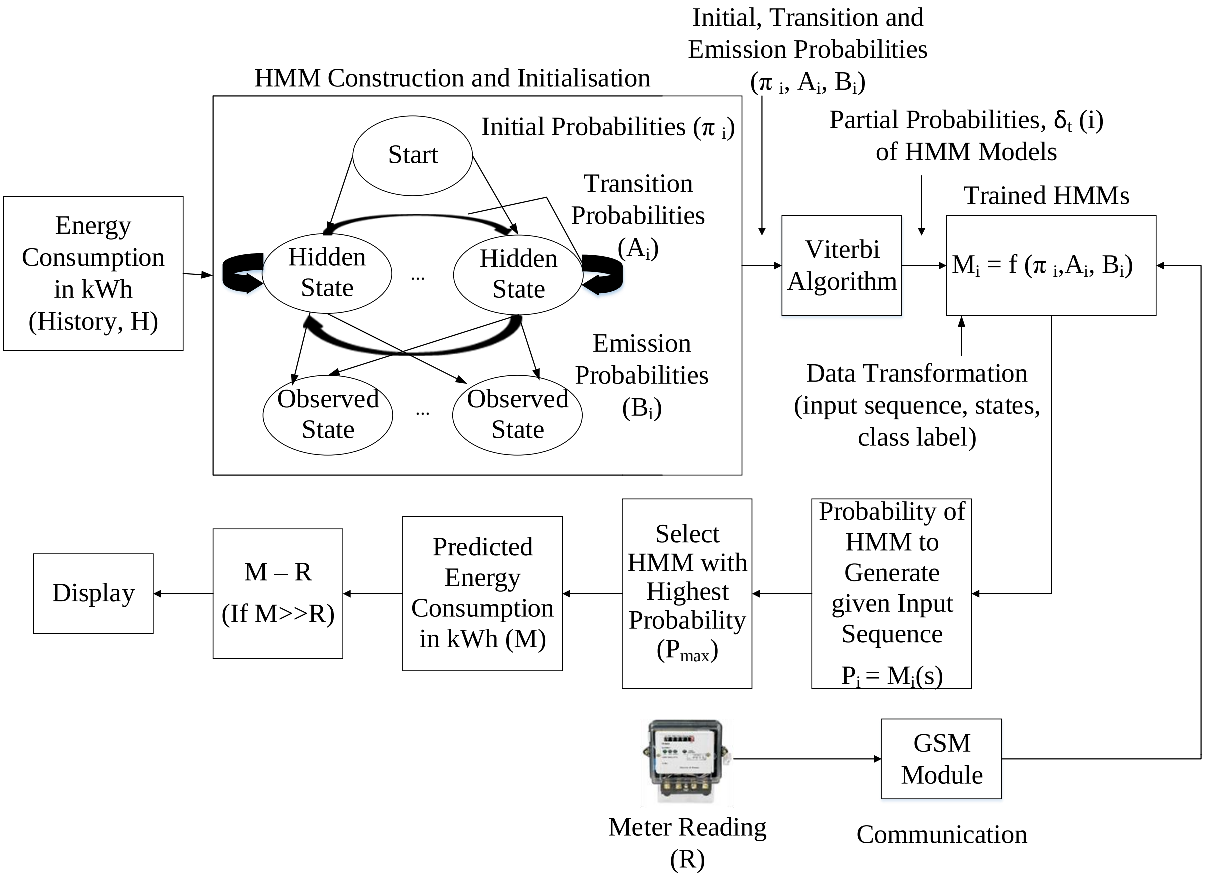

Initial, Transition and Emission Probabilities (π_i, A_i, B_i)
HMM Construction and Initialisation
Initial Probabilities (π_i)
Start
Transition Probabilities (A_i)
Hidden State
...
Hidden State
Emission Probabilities (B_i)
Observed State
...
Observed State
Energy Consumption in kWh (History, H)
Partial Probabilities, δ_t (i) of HMM Models
Viterbi Algorithm
Trained HMMs
$M_i = f(\pi_i, A_i, B_i)$
Data Transformation (input sequence, states, class label)
Probability of HMM to Generate given Input Sequence
$P_i = M_i(s)$
Select HMM with Highest Probability (P_{max})
Predicted Energy Consumption in kWh (M)
M – R
(If M>>R)
Display
Meter Reading (R)
GSM Module
Communication

Fig. 1. Design of hidden Markov model for electric power theft detection

Energy consumption data

Energy consumption data per minute for a particular household for two years was collected from “Kaggle” an online source, where date, time of the day, reactive power, active power, intensity and voltages have been provided. These data given in kilowatts (kW) are converted to kWh for use to program the model in Python 3.7 software with the help of an algorithm. For the purpose of this work, the Viterbi algorithm is used in the training of the model. The energy consumption data for twenty months are used to train the models whiles the remaining four months data are used to test the selected model for accuracy.

Hidden Markov model construction and initialisation

A number of models is constructed to be trained based on the various energy consumption values given. The model which gives the highest probability of energy consumption prediction is selected and used for the electric power theft detection. The HMMs are constructed to determine the initial

(π_i), transition (A_i) and emission (B_i) probabilities of each state to be able to implement the Viterbi algorithm in the training phase. Initial probability is the probability of the system to be in any hidden state at a given time. The transition probability is the probability of moving from one hidden state to another whereas the emission or observation probability is the probability of moving from one hidden state to an observed state. There is "n" number of hidden states and "k" number of observed states. The initial probability is set at one (1) for one hidden state and zero (0) for the others. One variable that is time of the day is considered for the hidden states. Variations in this variable gives a number of hidden states. With HMMs, since the model undergoes training whether or not the initial probability is accurate, it has negligible effect on the system. After series of iterations the model corrects itself since the initial probabilities change for every state.

Viterbi algorithm

The Viterbi algorithm named after its inventor, Andrew Viterbi is used to determine the most likely sequence of hidden states that generated an observation sequence. Given an observation sequence, in this case, the history of energy consumption, the Viterbi algorithm finds the most probable sequence of hidden states. The most probable sequence of hidden states is that combination that maximises the probability of the observed sequence given the hidden state combination [19].

Partial probabilities (δ) are calculated for each state in the model and partial best paths, depending on the path that gives the highest partial probability is determined as well. Partial probability, δ (i, t) is the maximum probability of all sequences ending at state i at time t, and the partial best path is the sequence which achieves this maximal probability [19]. Fig. 2 gives the manner in which the Viterbi algorithm works.

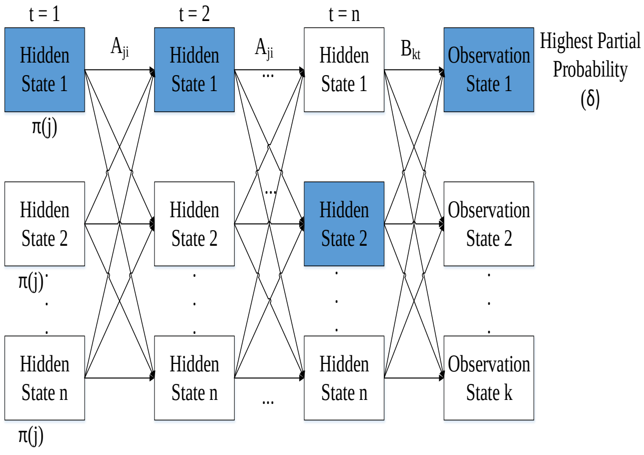


Fig. 2. Trellis diagram of the Viterbi algorithm

At t = 1, initial probability π(j) for each hidden state is defined. At t = 2, there is a transition from t = 1 (state j) to t = 2 (state i) therefore the transition probability is defined. However, there are transitions from hidden state 1 at t = 1 to hidden states 1, 2, …, n at t = 2, transitions from hidden state 2 at t = 1 to hidden states 1, 2, …, n at t = 2 and so on. Therefore, a transition matrix (A_{ji}) is obtained. At t = n (final iteration), an observation matrix (B_{kt}) is also obtained. Since there are a number of hidden states in the model, the partial probability is calculated at time t of each hidden state in the sequence and iterated till the current state has been reached. The overall best path is found by choosing the final state with the maximum partial probability and choosing its partial best path through the various hidden states. The blue boxes in Fig. 2 indicate the sequence of the overall best path with its associated partial probability.

The partial probability at hidden state t = 1 and t > 1 is given respectively by equation (1) and equation (2) [19].

$$\delta_1\,(i) = \pi(j) * B_{11} \tag{1}$$

$$\delta_t\,(i) = \max_j\,(\delta_{t-1}(j) * A_{ji}) * B_{kt} \tag{2}$$

where, δ_1 (i) is partial probability at first iteration, i is current hidden state, π(j) is initial probability, j is immediate past hidden state, B_{11} is observation probability at first iteration, δ_t (i) is partial probability at current hidden state, t is time at current hidden state in hours, $\max_j$ (δ_{t-1}(j) * A_{ji}) is select highest result of the product of δ_{t-1}(j) and A_{ji} at each t-1, δ_{t-1}(j) is partial probability at immediate past hidden state, t-1 is time at immediate past hidden state in hours, A_{ji} is transition probability, B_{kt} is observation probability and k is observation state.

The significance of the partial probabilities and partial best path is to determine the sequence of hidden states that give an observation sequence. Based on this knowledge given a set of input observation sequence data (testing data), each model will be able to generate the given input sequence and its associated probability. The predicted model that gives a higher probability will be selected.

Meter reading

The meter reading in this case is the four months energy consumption data that has been saved for testing to determine whether or not the prediction is correct. This data undergoes data transformation, that is a sequence of observations is generated to be fed into the model to give an energy consumption prediction. This is put into classes based on the total energy consumption of each state.

GSM module

The Global System for Mobile Communications (GSM) module which is to be incorporated into the energy meter during implementation is used to communicate meter readings to the utility company where the algorithm for the electric power theft detection system is located. In this design, the meter reading is the already available test data therefore it is directly connected to the trained HMMs block where input observation sequences of hidden states are generated.

Trained hidden Markov models block

The trained HMMs block takes its input from both the Viterbi algorithm and GSM module block. The partial probabilities and partial best paths from the Viterbi algorithm block that is fed into the trained HMMs block serve as a basis for the prediction. Based on the partial best paths that the algorithm is able to determine, sequences of the hidden states of the test data are obtained for each model. Partial probabilities are then calculated for the test data and the path or sequence with the highest partial probability is chosen as the right prediction.

Input sequence generation block

Each trained HMM generates an input sequence based on the initial input observation sequence that was fed to it from the test data. For each input sequence, a class label is defined based on the total energy consumption for that input sequence.

Highest probability selection block

The partial probability associated with each input sequence based on the best path of its hidden states is calculated for the individual models. The model that gives the highest partial probability is selected as the most likely prediction.

Energy consumption prediction block

Since each model is constructed based on different energy consumption values, the energy consumption value of the model that gave the highest partial probability is the predicted energy consumption.

Electric power theft detection block

The actual testing data is compared with the predicted power consumption by the model. This is done with the help of a computer program which computes the mathematical difference between the test data and predicted power consumption. If the deviation of the actual testing data is far less than the predicted consumption, it indicates theft. A tolerance level of +5 kWh of the deviation is specified in the program.

Display

As said earlier, if the test data is found out to be far less than the predicted consumption, the program outputs a theft indication. This is done by displaying the magnitude of electric power stolen in Microsoft excel which includes the date and time of the day within which the electric power was stolen. This is best appreciated in the practical implementation of the design.

B. Mathematical Modelling

a. Modelling of the Constructed Hidden Markov Model

The constructed HMM is modelled as follows:

n = total number of hidden states

k = total number of observations

x_t = hidden state at time t

y_t = observation at time t

π_j = initial probability

$B_{kt} = P(y_t|x_i)$ = emission/observation probability for an observation at state k at time t

$A_{ji} = P(x_i|x_j)$ = transition probability from state j to state i

ψ_{nn} = transition/confusion matrix of probabilities of moving from a hidden state at time t-1 to a hidden state at time t

φ_{nk} = observation matrix of probabilities of moving from a hidden state to an observation

The transition matrix also known as the confusion matrix gives the probabilities of moving from one hidden state at time t-1 to the other t. After these probabilities have been obtained, time t now becomes t-1 for the next iteration till the loop ends. The HMM is constructed to obtain the initial, transition and partial probabilities. The initial probability (π_j) is given by equation (3) and equation (4) [8].

$$\pi_j[1] = 1 \qquad (3)$$

$$\pi_j[q] = 0; \qquad \forall q \in \{1, \dots, n\} \qquad (4)$$

Where, q is any hidden state aside the first within the total number of hidden states

The transition probability (A_{ji}) is given by equation (5) [8].

$$A_{ji} = 1/n_i; \qquad \forall j, i \in \{1, \dots, n\} \qquad (5)$$

The emission or observation probability (B_{kt}) is given by equation (6) [8].

$$B_{kt} = 1/\gamma; \qquad \forall t \in \{1, \dots, n\}, \forall k \in \{1, \dots, \gamma\} \qquad (6)$$

Where, γ is total number of different energy consumptions for each input sequence

b. Modelling of Viterbi Algorithm

As said earlier, the Viterbi algorithm is used to determine from an observation sequence the most likely sequence of hidden states that generated it. From the trellis diagram in Fig. 3.2, some parameters have been specified and defined. The Viterbi algorithm uses the concept of probabilities in determining the sequence of hidden states. The Viterbi algorithm can be modelled as follows:

The probability of reaching an intermediate state in the trellis from the initial state is given by equation (7) [19] [20].

$$\delta_1(i) = \pi(j) * B_{11} \qquad (7)$$

The probability of reaching the goal state after several iterations from the initial state through the trellis is given in equation (8) [19] [20].

$$\delta_t(\mathrm{i}) = \max_j (\delta_{t-1}(\mathrm{j}) * A_{ji}) * B_{kt} \qquad (8)$$

After δ_t (i) has been obtained, it is possible to backtrack to determine which preceding state at time t-1 generated δ_t(i) and subsequent ones. This gives the underlying sequence of hidden states. The determination is done by holding a back pointer (Φ) for each hidden state which points to the predecessor that optimally provokes the current [19]. The back pointer is given in equation (9) [19] [20].

$$\Phi_t(i) = argmax_j(\delta_{t-1}(\mathrm{j}) * A_{ji}) \qquad (9)$$

where, $\Phi_t(i)$ is the back pointer of state *i* at time t, *i* is current hidden state, t is time at current hidden state, argmax_j is mathematical operation that selects the state *j* which maximises the expression in the bracket, $\delta_{t-1}(j)$ is partial probability at predecessor state, t-1 is time at predecessor hidden state, *j* is predecessor hidden state and A_{ji} is transition probability. The back-pointer aids in generating the best path of the model with the highest partial probability.

c. Modelling of the Trained Hidden Markov Models Block

The modelling of the trained HMMs (M_i) is given as a function of the initial, transition and emission probabilities. The ith trained HMM is given in equation (10) [8].

$$M_i = f(\pi_i, A_{ji}, B_{kt}) \tag{10}$$

Where, M_i is ith trained HMM and π_i is initial probability at ith state (partial probability at predecessor state j)

d. Modelling of Input Sequence Generation Block

Each HMM regenerates the given input sequence as mentioned earlier. This is visualised by its associated partial probability. The modelling of the input sequence generation block is given in equation (11) [8].

$$P_i = \delta[M_i(s)] \tag{11}$$

where, P_i is probability of the ith input sequence generated by model *i*, δ [M_i (s)] is partial probability of M_i (s), M_i (s) is HMM that generated the given input sequence and s is input sequence.

e. Modelling of the Highest Probability Selection Block

The Viterbi algorithm works such that, the model that gives the highest probability associated with the given input sequence is selected. The probability of the model that gives the highest P_i is given as in equation (12) [8].

$$P_{\max} = \delta[M_i(s)] \tag{12}$$

Where, $P_{\max}$ is highest probability

f. Modelling of Predicted Energy Consumption Block

Each model has been constructed and trained based on various energy consumption values. The corresponding energy consumption of the model that gives the highest probability, $P_{\max}$ is the predicted energy consumption. Let the energy consumption associated with $P_{\max}$ to be E. The modelling of the predicted energy consumption block is given in equation (13).

$$\text{E}(\text{P}_{\max}) = \text{Predicted energy consumption} \tag{13}$$

Where, $E(P_{\max})$ is energy consumption associated with $P_{\max}$

g. Modelling of the Theft Detection Block

The theft detection block given as M – R, is as a result of the difference between the testing data (meter reading, R) and the predictions (M) given by the model. If predicted values are far greater than the test data, the system gives an output as theft. The modelling of the theft detection block is given in equation (14) and (15).

$$M = \text{E}(\text{P}_{\max}) \tag{14}$$

$$M - R = T \tag{15}$$

Where, T is positive or negative values from the deviation of R values from M values. T > 0 or M >> R indicates a case of theft whilst T < 0 or M << R indicates a non-theft case.

C. Computer Programming

The computer program is written using the Python 3.7 software. The Anaconda navigator software which contains the jupyter notebook IDE was used as the interface for the program. Anaconda contains various libraries such as “hmmlearn” and “matplotlib” which were imported for use in this program. The “hmmlearn” was used to train the model and the “matplotlib” was used to plot graphs of the results. The data which has been divided into three for every twenty-four hour is read by the program for the results to be obtained. This division is done to obtain results for morning (first eight hours), afternoon (second eight hours) and evening (third eight hours). These in real sense are the hidden states of the model. Fig. 3 gives a flowchart of electric power theft detection and Fig. 4 shows a section of the computer program used.

The following steps were undertaken in writing the computer program for electric power theft prediction.

Step 1: Start jupyter notebook and import libraries (hmmlearn, matplotlib.pyplot, numpy, csv, datetime, sys, time, warnings) necessary to run HMM program.
Step 2: Call energy consumption data into the program.
Step 3: Construct and initialise the HMM by creating initial, transition and emission probability classes and defining a range for the number of states of the model;
Step 4: Train the HMMs using viterbi algorithm.
Step 5: Import energy consumption data for last four months for testing the trained HMMs.
Step 6: Declare the HMM having the highest probability.
Step 7: Predict the energy consumption (M).

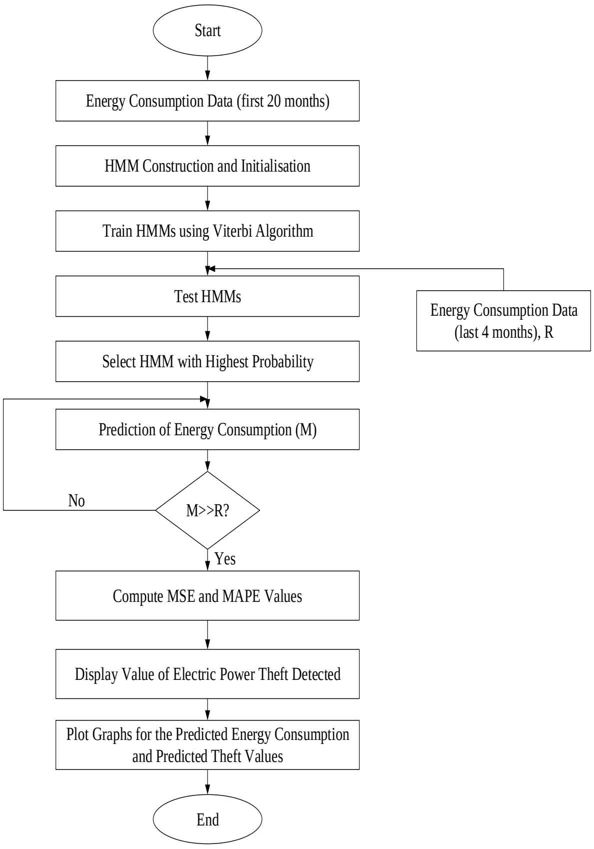

Start
Energy Consumption Data (first 20 months)
HMM Construction and Initialisation
Train HMMs using Viterbi Algorithm
Test HMMs
Energy Consumption Data (last 4 months), R
Select HMM with Highest Probability
Prediction of Energy Consumption (M)
No
M>>R?
Yes
Compute MSE and MAPE Values
Display Value of Electric Power Theft Detected
Plot Graphs for the Predicted Energy Consumption and Predicted Theft Values
End

Fig. 3. Flowchart of computer program of HMM for electric power theft prediction

From the program of Fig. 4, the last four months of the two-year data has been retained to be used in testing the model, the transitions between states is 55 and the program runs 10000 iterations to increase accuracy. The Viterbi algorithm used to determine the hidden states can be found in the HMM library and therefore a program was not written for it. It was implemented in this program to work based on the data fed into it. Other methods such as plotting of graphs, predicting power consumption, getting the predicted theft values, calculating Mean Square Error (MSE) and Mean Absolute Percentage Error (MAPE) were programmed to obtain a complete algorithm for electric power theft prediction. The MAPE is expressed as in equation (16) [21] and the MSE as equation (17) [22].

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{Y_t - X_t}{Y_t}\right| \qquad (16)$$

Where, Y_t is actual value at time t, X_t is predicted value at time t and n is total number of data points.

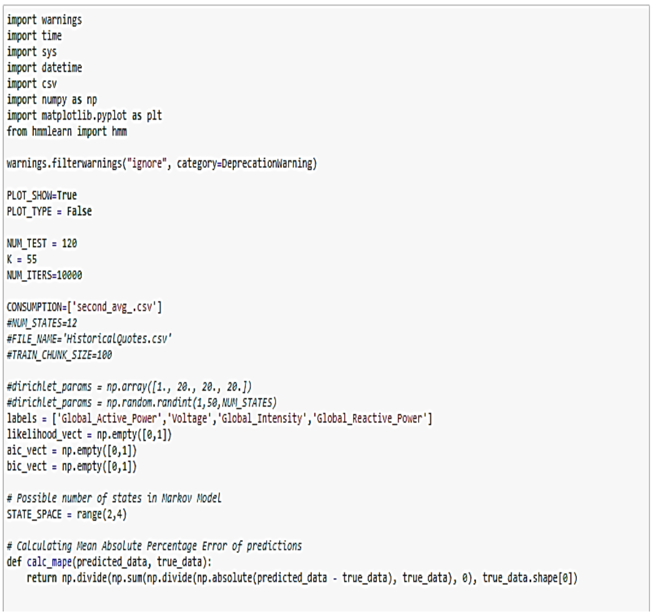
$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - X_i)^2 \qquad (17)$$

Where, Y_i is the i^{th} actual value, X_i is the i^{th} predicted value.

IV. RESULTS AND DISCUSSIONS

A. Results

Step 8: If predicted energy consiumption is far greater than meter reading data then compute MSE and MAPE values, else go to Step seven (7).
Step 9: Display value of electric power theft detected.
Step 10: Predicted energy consumption and the predicted theft values are then plotted.
Step 11: End.



```
import warnings
import time
import sys
import datetime
import csv
import numpy as np
import matplotlib.pyplot as plt
from hmmlearn import hmm

warnings.filterwarnings("ignore", category=DeprecationWarning)

PLOT_SHOW=True
PLOT_TYPE = False

NUM_TEST = 120
K = 55
NUM_ITERS=10000

CONSUMPTION=['second_avg_.csv']
#NUM_STATES=12
#FILE_NAME='HistoricalQuotes.csv'
#TRAIN_CHUNK_SIZE=100

#dirichlet_params = np.array([1., 20., 20., 20.])
#dirichlet_params = np.random.randint(1,50,NUM_STATES)
labels = ['Global_Active_Power','Voltage','Global_Intensity','Global_Reactive_Power']
likelihood_vect = np.empty([0,1])
aic_vect = np.empty([0,1])
bic_vect = np.empty([0,1])

# Possible number of states in Markov Model
STATE_SPACE = range(2,4)

# Calculating Mean Absolute Percentage Error of predictions
def calc_mape(predicted_data, true_data):
    return np.divide(np.sum(np.divide(np.absolute(predicted_data - true_data), true_data), 0), true_data.shape[0])
```

Fig. 4. Computer program of HMM for electric power theft prediction

The results obtained from the computer program for the system have been provided in this section. These results have been grouped into power consumption prediction and theft prediction.

a. Simulation Results on Power Consumption Prediction

Fig. 5 gives power consumption results for the last four months of the two-year data collected. Each day of the last four months has three predicted and actual power consumptions that is, morning, afternoon and evening. One point on the x-axis therefore indicates one time of the day. This means the first 3 points on the x-axis gives the 3 predictions for the first day, the next 3 points give 3 predictions for the following day and so on.

a. Simulation Results on Theft Prediction

Fig. 6 gives a graph of the predicted theft values. The absolute value of the difference between predicted and actual values have been plotted and a tolerance level of 0.5 was specified when plotting the graph. Table 2 gives the predicted electric power theft values in kWh, corresponding time of the day and the date that electric power was stolen for seven days. This result is displayed in an excel file when the program was ran.

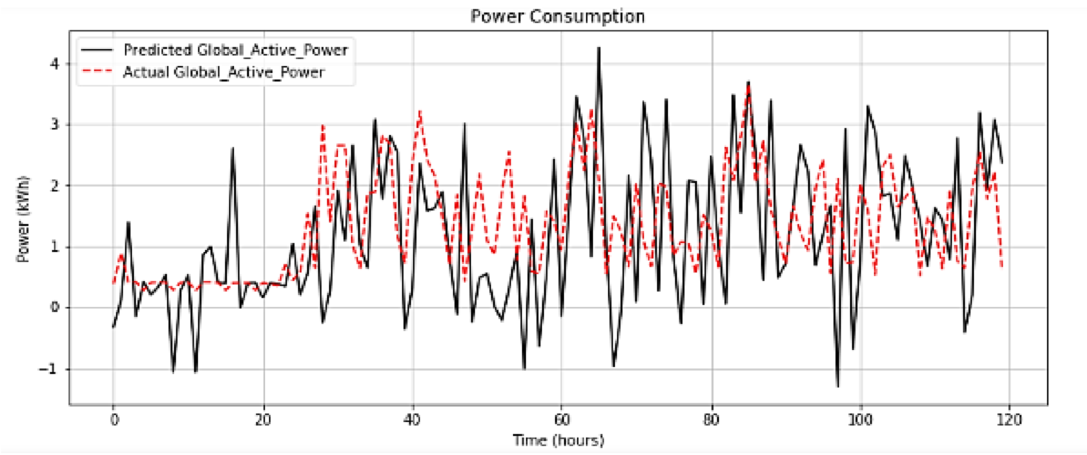


Fig. 5 A Graph of power consumption against time

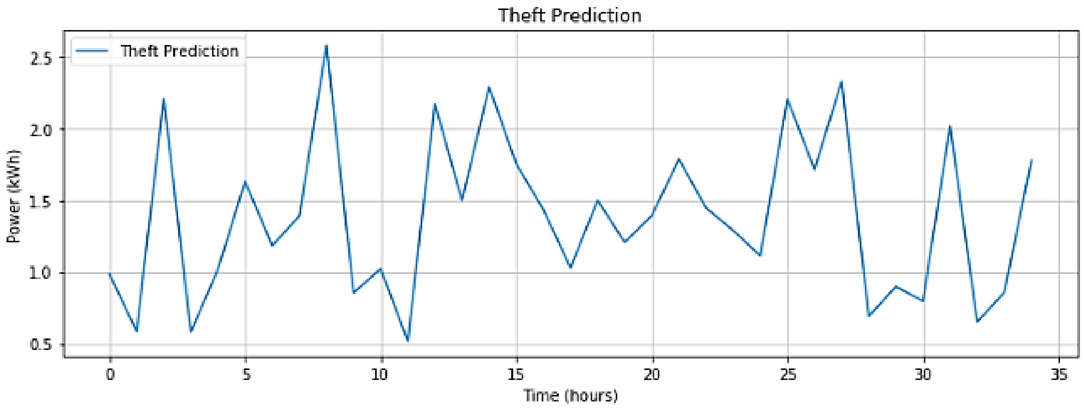
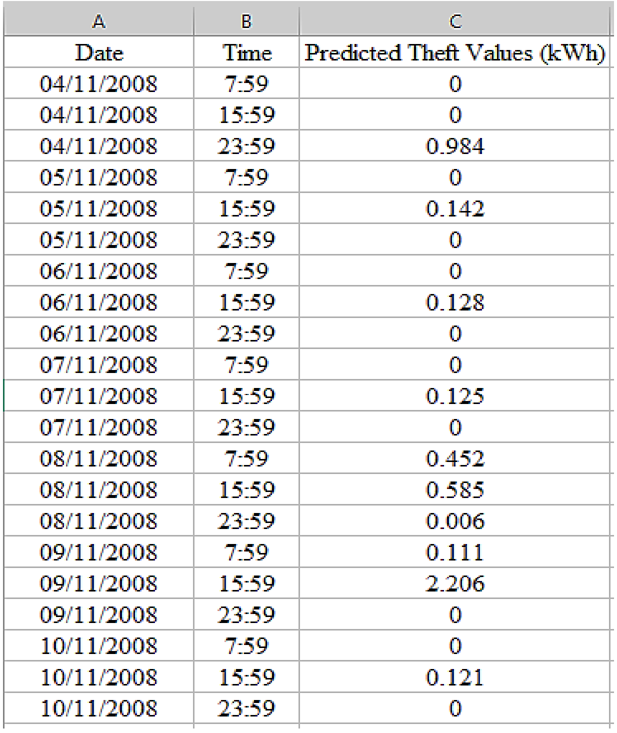


Fig. 6 A Graph of electric power theft prediction against time

TABLE 2 PREDICTED ELECTRIC POWER THEFT FOR SEVEN DAYS



A	B	C
Date	Time	Predicted Theft Values (kWh)
04/11/2008	7:59	0
04/11/2008	15:59	0
04/11/2008	23:59	0.984
05/11/2008	7:59	0
05/11/2008	15:59	0.142
05/11/2008	23:59	0
06/11/2008	7:59	0
06/11/2008	15:59	0.128
06/11/2008	23:59	0
07/11/2008	7:59	0
07/11/2008	15:59	0.125
07/11/2008	23:59	0
08/11/2008	7:59	0.452
08/11/2008	15:59	0.585
08/11/2008	23:59	0.006
09/11/2008	7:59	0.111
09/11/2008	15:59	2.206
09/11/2008	23:59	0
10/11/2008	7:59	0
10/11/2008	15:59	0.121
10/11/2008	23:59	0

b. Results on Mean Absolute Percentage Error and Mean Squared Error

The MAPE and MSE between the predicted and actual values were 1.055 and 10.5 respectively. The optimum number of states for the model was found to be 3.

B. Discussions

a. Discussion of Power Consumption Prediction

Based on the result presented in Fig. 5, it can be seen that the actual test data for the last four months has been presented in red and that of the prediction has been presented in black. It can be seen that the deviation of the prediction from the actual from time 0 - 20 is significant. This is because as said in previous chapters the HMM initially assigns an initial probability of "1" to one hidden state and "0" for others. This implies that the accuracy of the system at the beginning depends on how right the initial probability assignment is. However, after a series of iterations the model corrects itself since initial probabilities for each iteration get better and become more accurate.

b. Discussion on Theft Prediction

From Fig. 6, it can be seen that the system has been able to successfully predict electric power theft. Values below 0.5 have been considered to be non-theft cases and therefore have not been indicated on the graph. Values above 0.5 are indications of theft on the graph. Table 2 however gives an indication of theft and non-theft cases at each time of the day. As mentioned earlier each day has been divided into morning, afternoon and evening. Time 7:59 represents 0:00 to 7:59 (morning), time 15:59 represents 8:00 to 15:59 (afternoon) and 23:59 represents 16:00 to 23:59 (evening). Each time of the day has a corresponding theft or non-theft value. A value of 0 on Table 2 indicates a non-theft case and any other value indicates a theft case. A value of 0.5 (which indicates the tolerance level) has been subtracted from every value indicated on the table.

c. Discussion on Mean Absolute Percentage Error and Mean Squared Error

Whilst the MAPE gave the expected results. The MSE however, was a bit higher than expected. This was because, only one variable (time of the day) was used as hidden state for the model. With HMMs, the higher the number of variables, the more accurate the system is [19]. The MSE is therefore expected to reduce with an increase in variables.

V. CONCLUSION

The proposed electric power theft prediction system for households has been successfully realised in this paper based on HMM and Viterbi algorithm where a part of the data collected was used to represent meter readings. This means that when there is a bypass of the meter, the meter readings will be inconsistent and same for tampering as well. The system therefore, will be able to predict theft in any of these two cases when power consumption is predicted and compared with the readings. The MSE and MAPE results indicate that the system gives the expected results. However, the MSE result though a bit higher than expected can be improved by increasing the number of variables for the hidden states.

REFERENCES

[1] A. P. K. Sakyi, "A sustainable approach in curbing energy theft",*https://energynewsafrica.com/index.php/2019/04/20/a-sustainable-approach-in-curbing-electricity-theft-april/* Accessed: September 2, 2019, 2019.

[2] G., Gu, H., Bo, W. Qingsu, and D. Bo, "Comparison of machine learning techniques for the detection of the electricity theft", *Proceedings of the International Conference on Computer Technology, Electronics and Communication*, Beijing, China, pp. 925–928, 2018.

[3] O. Yakubu, N. C. Babu, and O. Adjei,, "Electricity theft: analysis of the underlying contributory factors in Ghana", *Energy Policy*, Vol. 123, No. 10, pp. 611–618, 2018.

[4] J. Yeckle, and B. Tang, "Detection of electricity theft in customer consumption using outlier detection algorithm", *Proceedings of the 1st International Conference on Data Intelligence and Security*, Mississippi State University, Mississippi State, USA, pp. 135–140, 2018.

[5] G. V. Tanuja, P. S. Neha, W. H. Kalyani, S. Raj, and M. Shrinath, "Power theft detection by using GSM", *International Journal of Advance Engineering and Research*, Vol. 5, No. 5, pp. 581–586, 2018.

[6] H. Huang, S. Liu, and K. Davis, "Energy theft detection via artificial neural networks", *Proceedings of the PES Innovative Smart Grid Technologies Conference*, Delft, Netherlands, pp. 721–726, 2018.

[7] M. Nabil, M. Ismail, M. Mahmoud, M. Shahin, K. Qaraqe, and E. Serpedin, "Deep recurrent electricity theft detection in AMI networks with random tuning of hyper-parameters", *Proceedings of the 24th International Conference on Pattern Recognition*, Beijing, China, pp. 740-745, 2018.

[8] I. Ullah, R. Ahmad, and D. Kim, "A prediction mechanism of energy consumption in residential buildings using hidden Markov model", *Energies*, Vol. 11, No. 2, pp. 1-20, 2018.

[9] M. Lydia, P. E. G. Kumar, and Y. Levron, "Detection of electricity theft based on compressed sensing", *Proceedings of the 5th International Conference on Advanced Computing and Communication Systems*, Coimbatore, India, pp. 995-1000, 2019.

[10] R. Wu, L. Wang, T. and Hu, "AdaBoost-SVM for electrical theft detection and GRNN for stealing time periods identification", *Proceedings of the 44th Annual Conference of the IEEE Industrial Electronics Society*, Shenzhen, China pp. 3073-3078, 2018.

[11] V. S. Sekhar, and R. Puviarasi, "Simulation of GSM-based power theft detection using Proteus", *Proceedings of the 4th International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics*, Chennai, India, pp. 332 – 335, 2018.

[12] N. K. Mucheli, U. Nanda, D. Nayak, P. K. Rout, S. K. Swain, S. K. Das, and S. M. Biswal, "Smart power theft detection system", *Proceedings of the Devices for Integrated Circuit Conference*, Kalyani, India, pp. 302-305, 2019.

[13] K. Ask, N. K. Singh, A. K. Singh, D. K. Singh, and K. Anand, "Design and simulation of smart prepaid-postpaid energy meter with alarm and theft control", *Proceedings of the 5th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics*, Allahabad, India, pp. 751-756, 2018.

[14] R. E. Ogu, and G. A. Chukwudebe, "Development of a cost-effective electricity theft detection and prevention system based on IoT technology", *Proceedings of the 3rd International Conference on Electro-technology for National Development*, Owerri, Nigeria, pp. 756-760, 2017.

[15] K. Zheng, Q. Chen, Y. Wang, and C. Kang, "A novel combined data driven approach for electricity theft detection", *IEEE Transactions on Industrial Informatics*, Vol. 15, No. 3, pp. 1809-1819, 2018.

[16] S. K. Singh, R. Bose, and A. Joshi, "Minimising energy theft by distance based theft detector in AMI", *Proceedings of the 24th National Conference on Communications*, New Delhi, India, pp. 556-560, 2018.

[17] M. Nabil, M. Ismail, M. Mahmoud, W. Alasmary, and E. Serpedin, "PPETD: Privacy-preserving electricity theft detection scheme with load monitoring and billing for AMI networks", *Special Section on Security, Privacy and Trust Management in Smart Cities*, Vol. 7, No. 3, pp. 96334-96348, 2019.

[18] Y. Zhou, Y. Liu, and S. Hu, "Energy theft detection in multi-tenant data centers with digital protective relay deployment", *IEEE Transactions on Sustainable Computing*, Vol. 3, No. 1, pp. 16-29, 2017.

[19] D. Kane, "Data science - Part XIII - Hidden Markov models", https:www.slideshare.net/mobile/DerekKane. Accessed: January 23, 2020, 2017.

[20] Anon., "Lecture 12: Algorithms for HMMs", *http://www.people.cs.georgetown.edu*. Accessed: February 25, 2020, 2017.

[21] G. Stephanie, "Mean Absolute Percentage Error", *https://www.statisticshowto.com/mean-absolute-percentage-error-mape/*. Accessed: February 10, 2020, 2017.

[22] Anon., "Mean Squared Error", *https://en.m.wikipedia.org/wiki/Mean_squared error*. Accessed: February 10, 2020, 2018.