Appendix A. Abbreviations List

The following abbreviations are used in this manuscript:

ACF	Analysis Of The Auto-correlation Function
AMI	Advanced Metering Infrastructure
ANN	Artificial Neural Network
ARIMA	Auto-regressive Integrated Moving Average
AUC	Area Under the Curve, PR-AUC precision-recall AUC
UCI	UC Irvine Machine Learning Repository
ObSM	Observer Smart Meter
CNN	Convolution Neural Networks
DL	Deep Learning
FA-XGBoost	Firefly-Algorithm-based Extreme-Gradient-Boosting
HNTLD	Hybrid NTL Detection
HNTLDS-ObSM	Hybrid NTL Detection Scheme with Observer SM
HNTLDSWObSM	Hybrid NTL Detection Scheme without Observer SM
MCC	Matthew Correlation Coefficient
ML	Machine Learning
MODWPT	Maximum Overlap-Decomposition and Packet Transform
NTL	Non-Technical Loss
PACF	Partial Autocorrelation Function
RF	Random Forest
RMSE	Root Mean Square Error
RUSBoost	Random Under-Sampling Boosting
SG	Smart Grid
SGCC	State Grid Corporation Of China
SM	Smart Meter
VGG	Visual Geometry Group
WADCNN	Deep and Wide CNN

Appendix B. ARIMA predicts SM coefficients using OBSM and SMs data.

To obtain the coefficient $c_{k,i}$ and calculate approximate function, from $E_j = \sum_{i=1}^{n} e_{i,j}$ we know that $E_j = \sum_{i=1}^{n} e_{i,j} = \sum_{i=1}^{n} f_i(r_{i,j}) = \sum_{i=1}^{n} \sum_{k=1}^{m} c_{k,i} r_{i,j}^k$. It can be rewritten as Equation (B.1).

$$E_j = \sum_{i=1}^{n} \sum_{k=1}^{m} r_{i,j} c_{k,i} \tag{B.1}$$

$$c_{k,i} = \frac{e_{k,i}}{r_{k,i}} \tag{B.2}$$

The value of $c_{k,i}$ calculated using Equation (B.2) $c_{k,i}$ ideally equals to 1 as the SM_i reading $r_{k,i}$ and ObSM corresponding $e_{k,i}$ reading are equal. A small difference with the accepted ε range value can be ignored. The value of ε can be calculated using Equation (B.3).

$$\left| e_{k,i} - r_{k,i} \right| < \varepsilon \tag{B.3}$$

The accepted value of ε is typically 0.02, and if it is greater than this value, the SM is considered malfunctioning or, more specifically, an NTL fraud.

The values of $r_{i,j}$ and E_j in Equation (B.1) are known, but the values of coefficients $c_{k,i}$ are unknown. We will measure $n * m$ samples to obtain the coefficient $c_{k,i}$ value.

ObSM readings E_j represented through an array E_T using Equation (B.4).

$$\mathbf{E_T} = (E_1, E_2, E_1, E_3, E_4, ..., E_m)^T \tag{B.4}$$

All individual coefficients as mentioned in Equation (B.2), may be represented using vector C Equation (B.5) where each c_i is the sum of all the broken down E_j readings to the assumed $e_{k,j}$ reading to c_i, where $c_i = \sum_{k=1}^{n} \frac{e_{k,i}}{r_{k,i}}$ for the j^{th} ObSM.

$$\mathbf{C} = (c_1, c_2, c_3, \cdots, c_n)^T \tag{B.5}$$

All SMs readings are represented through matrix R Equation (m, n).

$$\mathbf{R} = \begin{bmatrix} r_{1,1} & \cdots & r_{1,j} & \cdots & r_{1,n} \\ r_{2,1} & \cdots & r_{2,j} & \cdots & r_{2,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_{i,1} & \cdots & r_{i,j} & \cdots & r_{i,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_{m,1} & \cdots & r_{m,j} & \cdots & r_{m,n} \end{bmatrix} \tag{B.6}$$

Equation (B.7) shows that the j^{th} ObSM reading E_j when no fraud or tempered SM is there, its value is the sum of multiplying R_j^{th} row by the C_j^{th} column.

$$\mathbf{E_T} = \boldsymbol{R} \times \mathbf{C} \tag{B.7}$$

As we already know, E_T and R from each measurement, R^{-1} can be obtained by taking the inverse of R. Putting E_T and R^{-1} in Equation (B.8) and calculate vector C of size n. Here n represents the number of SMs.

$$\mathbf{C} = \boldsymbol{R^{-1}} \times \mathbf{E_T} \tag{B.8}$$

where C is a vector of size n and (n) is the number of SMs. The logic will work as follows:

Calculate the result of Equation (B.8). If all C vector values are close to 1, it is considered that all the n SMs are non-fraudulent. On the other hand, if some of the entries in the C vector are far from 1, then at least one fraudulent SM exists. To identify fraudulent rows, The expression in equation B.9 needs to be calculated for each row i:

$$Diff_i = (\sum_{j=1}^{n} r_{i,j}) - E_j \tag{B.9}$$

If $Diff_i <= \epsilon$ (where ϵ is in accepted deviation, and it usually is 0.02), then the row's readings are normal non-fraudulent rows of SM readings. While If $Diff_i > \epsilon$, then the row i is a fraudulent row of SM readings.

From the above, the SMs data is divided into two parts: $Before\_fraud\_data_i$ from row sample 1 to $i-1$, while the other subset is $Suspected\_fraudulent\_data_i$ from the row i till the end.

We proceed then to use the $Before\_fraud\_data_i$ subset to FIT the ARIMA Model and to use the generated ARIMA Model to forecast ARIMA predictions where each entry should be the equivalent readings in the suspected ObSM $e_{i,j}$ readings subset, as follows:

For each SM (j) column in the SM readings, R do:

1) Calculate the mean μ and the standard deviation σ of the SM readings in $Before - fraud - data\_i$.

2) Using Chebyshev's Theorem, Equation (B.10) the fraud threshold is configured.

$$P(|R_i - \mu| > h\sigma) < \frac{1}{h^2} \tag{B.10}$$

At least $1 - 1/h^2$ data from a sample R_i must fall within h standard deviations from the mean. We calculate h that we can consider as the threshold that if it is exceeded for continuous readings, then the SM will be fraudulent; from now on, for simplicity, we consider the 3σ as the fraud threshold.

The next step is to generate ARIMA_FIT. It is an ARIMA model from the SM readings in $Before\_fraud\_data_i$ using the method explained in Section Appendix C.

Appendix C. ARIMA process

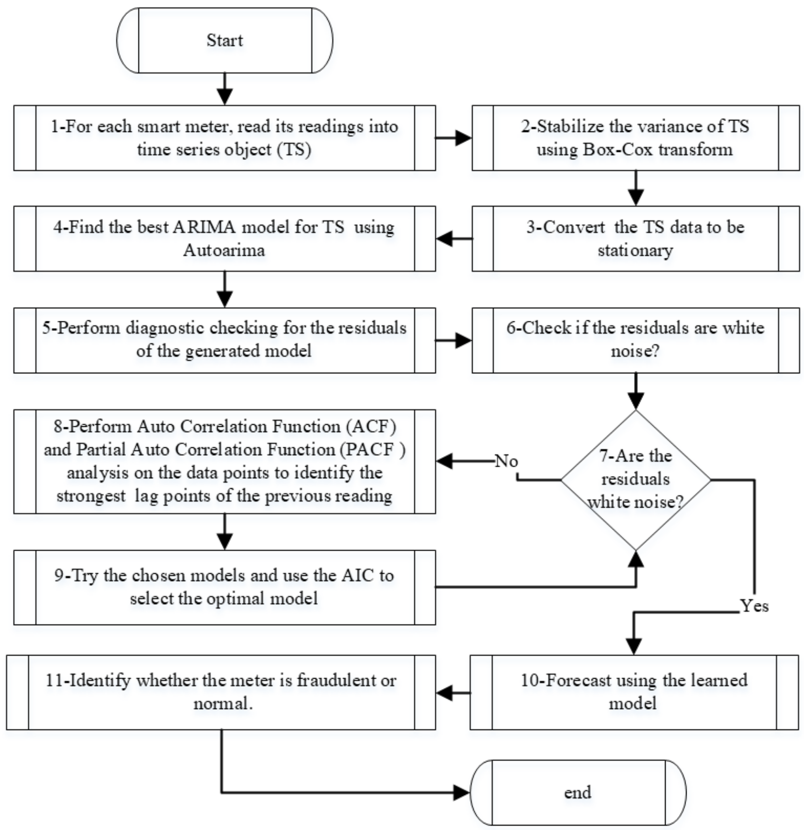

Figure C.1: Fraud Detection using ARIMA Prediction

To forecast using ARIMA, the following steps are performed for all the SM's reading data:

1. Take the i-th SM data into Time Series object TS and illustrate it.
2. Perceive TS to discern the temporal-series trend, seasonality, and randomness patterns.
3. Make TS stationary by utilizing Box-Cox transform. Continue to differentiate TS data until it becomes stationary.
4. Utilize auto.arima() function to find the best ARIMA model for TS to generate the ARIMA forecasting model.

5. Carry out diagnostic checking. The first step is to draw a time series plot of residuals. Then, three plots, "A-Standardized Residuals Plot", "B-ACF Residuals Plot", and "C-Plot defining the P-Values of Box-Cox statistics" are produced. The last step is the Port-Manteau test of the residuals. Then, the model can be considered a good fit depending on the plot values.
6. Examine the residual value for "white noise". White noise is a term used to describe time series that have no autocorrelation. If the answer is "No", it will reiterate the ACF/PACF analysis of the most recent k data-series values to detect the most significantly linked values, which will be employed in the ARIMA model's parameters. If the answer is "Yes", then we create the forecasts. Finally, the forecasted values will be compared with the related SM reading to identify whether they are fraudulent or non-fraudulent.

Appendix D. Sample data the used Datasets

A sample of ten SMs' non-fraudulent usage data and the corresponding reported data of the monitoring ObSM is listed in Tables D.1, taken from the UCI Dataset. Individual SM data is presented in the first ten columns, while ObSM data is in the last column. This is because there are no fraudulent SMs in this table. That is, the sum of all SMs data equals ObSM data.

Table D.1: 20 Samples from 10 Non-fraudulent SMs Monitored by One Observer SM

SM 1	**SM 2**	**SM 3**	**SM 4**	**SM 5**	**SM 6**	**SM 7**	**SM 8**	**SM 9**	**SM 10**	**Observer SM**
4560	3300	6780	1900	2870	3200	4510	2800	4320	5100	39340
5120	2560	4560	2440	1900	4500	5700	2120	5700	6700	41300
3230	3500	6780	3900	1670	5100	4340	2000	5200	6400	42120
6780	2900	5990	1890	2560	6700	5800	5300	6880	6890	51690
6430	3000	5300	2120	4980	7450	6400	3050	4990	4800	48520
1670	4300	5440	2800	5880	4560	6340	3560	4670	5120	44340
2430	2700	6330	3000	2200	1400	5300	3650	4770	5700	37480
7890	2300	4600	1900	2560	5700	4670	3670	2120	1440	36850
8700	3400	6120	2650	2780	6120	6400	3100	5670	5870	50810
4400	1400	3670	2990	2330	1670	4600	3540	5600	4660	34860
5200	2800	4780	2110	5430	3120	5600	3000	5900	4670	42610
3560	2900	5340	2440	5280	4230	5670	2890	4900	4220	41430
2450	4500	5200	2660	3760	5230	5800	2400	2220	2780	37000
4560	3890	5780	2770	4670	5400	6200	2670	3450	3990	43280
3670	2700	6400	3120	4980	6400	6900	2670	3560	6450	46850
3560	4100	5990	3450	4330	4780	6770	1990	4110	5980	45060
2340	3330	6900	3200	3780	1560	5440	2500	4560	5840	39450
5320	3990	5550	3320	4550	9530	6780	5700	4880	6330	55950
3240	3980	5090	1990	2670	4500	7000	3890	4670	5300	42330
4670	2900	4200	2550	4670	8670	7100	3600	5450	8120	51930

A sample of ten SMs with thirty fraudulent and non-fraudulent usage data and the corresponding reported data of the monitoring ObSM is listed in Tables D.2, taken from the UCI Dataset. Individual SMs data is presented in the first ten columns, while ObSM data is in the last column. This table contains fraudulent data for some SMs. That is, the sum of all SMs data is not equal to ObSM data.

Sample SMs Data when ObSM data is unavailable

A sample of n SMs data is shown in Table D.3, taken from dataset SGCC Dataset. The first column is consumer no, the second column is fraudulence flag (1 mean fraudulent), and the rest are day-wise consumption. This table contains both fraudulent and non-fraudulent data.

Table D.2: 30 Samples from 10 Fraudulent SMs Monitored by One Observer SM

SM 1	**SM 2**	**SM 3**	**SM 4**	**SM 5**	**SM 6**	**SM 7**	**SM 8**	**SM 9**	**SM 10**	**Observer SM**
2110	3300	4710	1500	3000	8230	4500	2800	4600	5300	40050
1950	2600	5330	2330	1530	8550	5600	4494	5100	5600	43084
1800	3500	6600	2560	1990	9000	4800	2000	5200	6100	43550
2220	2900	6780	1890	2560	9980	5120	5000	6780	6990	50220
2300	3000	4990	2120	4560	8770	6000	3050	4900	4670	44360
1570	3700	6100	2890	5660	8990	6340	3500	4880	5120	48750
2500	2700	6230	3000	2200	7990	5300	3600	4780	5330	43630
2300	3200	4780	1900	2560	8230	4990	3670	2100	5900	39630
2880	3400	5330	2440	2890	8450	4890	3090	5800	5810	44980
1930	3100	3550	2670	2330	8650	4600	3550	5700	4890	40970
2120	2500	4880	2110	5330	8100	4500	2990	5550	4550	42630
2300	2900	4940	2230	5280	7450	5670	2890	4900	4200	42760
2000	2900	5120	2660	3600	7810	5800	2700	2100	2600	37290
1750	4000	5440	1990	4670	9110	6200	2560	3600	3900	43220
1820	2700	5980	3120	4880	9300	6900	2670	3700	6230	47300
2740	2760	5770	3450	4330	8560	6770	1900	4100	6430	46810
2450	3330	4890	3330	3990	8440	5440	2500	4800	5810	44980
2530	3670	5220	3320	4550	9400	6450	3060	4900	6300	49400
2320	3980	5540	1970	2550	9910	6340	3560	4780	5900	46850
2780	2990	5900	2550	3890	8760	6100	3880	5600	5660	48110
2300	3450	6780	1770	3200	7770	5990	3660	1200	1900	38020
3450	4320	7120	1800	3560	7650	5870	3400	1340	1980	40490
5320	2450	8120	2890	4120	8010	5770	3080	2450	200	42410
4340	3090	7890	3120	4030	7990	5050	2090	2980	4500	45080
2630	4670	6120	3450	2120	71.2	6120	4090	2100	5230	43650
3120	5120	599.4	3200	2670	68.9	4890	3900	3210	6120	45780
3090	3100	580.5	4300	3400	67.7	4990	3890	4500	5900	46390
1980	3980	513.0	3780	3080	60.8	5340	3100	4090	4980	42110
2890	4560	650.7	3880	3030	61.3	5220	2030	1900	4870	41740
2770	3900	670.5	3200	3980	74.5	5090	2450	2890	800	39980

Table D.3: Sample Data from SGCC Dataset

CONS_NO	**FLAG**	**01/01/2014**	**...**	**6/9/2016**	**7/9/2016**	**8/9/2016**	**9/9/2016**
D64..	1	0.91	1.36	...	0	0.54	2.22
A9E..	1	17.23	78.73	...	47.71	38.28	28.31
356..	1	2.77	3.54	...	5.31	8.86	6.31
EE3..	1	8.05	10	...	17.56	17.71	11.68
004..	1	15.1	15.06	...	14.74	16.71	7.94
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
660..	0	0	0	...	4.81	4.72	5
939..	0	2.35	3.78	...	3.69	1.99	2.95

Appendix E. Performance metrics formulas

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{E.1}$$

$$Specificity = \frac{TN}{(TN + FP)} \tag{E.2}$$

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \tag{E.3}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{E.4}$$

$$F1 - Score = 2 * \frac{Precision * Specificity}{(Precision + Specificity)} \tag{E.5}$$