Received: June 16, 2023. Revised: July 19, 2023. 438

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Electricity Theft Detection System Using Cloud Computing and Deep Learning Techniques

Saja Abdul-hamza[1] Laith A. Abdul-Rahaim[1] Sarmad Ibrahim[1]*

[1]*Department of Electrical Engineering, University of Babylon, Babylon, Iraq*
* Corresponding author's Email: sarmad.ibrahim@uobabylon.edu.iq

Abstract: Electricity theft is a widespread problem with significant economic implications. In this paper, we present an efficient system to detect electricity theft using cloud computing and deep learning techniques, specifically convolutional neural networks (CNNs) due to its ability to effectively extract features from electrical data. To address the challenge of imbalanced data, we tested various data preprocessing techniques. The adaptive synthetic (ADASYN) technique was selected to handle data imbalance. The evaluation of our CNN model demonstrates its effectiveness in accurately detecting incidents of theft, where our model achieved an accuracy of 97.22%, with a precision of 97% and a recall of 99.9%. The system was tested under real-life conditions and proved to be effective. Furthermore, cloud computing techniques have greatly facilitated the dissemination of CNN models by providing storage for data and a computational space for executing the model, as well as presenting the results to the authorities for electricity theft detection. The use of cloud servers has greatly simplified the distribution and utilization of CNN models in the context of electricity theft detection.

Keywords: Cloud computing, Deep learning, Electricity theft, Technical losses.

1. Introduction

In modern times, there is extensive research and development taking place in the field of the smart grid (SG) to enhance efficient power control, management, and monitoring. An SG is a network of electrical and information and communication technology (ICT) systems that aim to provide effectiveness, reliability, and efficiency in power control and monitoring through the use of ICT advancements [1]. SGs enable the monitoring of electricity usage and ensure that electricity consumption aligns with the system load. It achieves this through bidirectional communications and self-healing capabilities for transmission, energy generation, distribution, and power observation, all of which can be remotely controlled [2]. Compared to traditional grid systems, effectively managing a large number of smart grids requires an approach that is scalable, efficient, reliable, and secure. To optimize the performance of the SG system, it is essential to expand communication networks and ensure decentralized data storage for efficient data management. Recent research indicates that integrating cloud computing into the smart grid system can significantly enhance its overall performance. Cloud computing is a rapidly emerging technology that enables remote data storage, monitoring, and control from any location globally. By incorporating cloud computing, the SG can benefit from remote access, real-time data control, monitoring, and regardless of location and time. This is made possible through appropriate communication interfaces, establishing cloud computing as a vital component of the next-generation SG approach [3, 4].

Machine learning (ML) and deep learning (DL) play a key role in identifying patterns and producing valuable information [5]. ML and DL can handle massive amounts of data as well as find relationships and patterns that humans might not be able to easily detect. However, the quality of the data is crucial for building valuable models. Poor quality data can lead to inaccurate predictions and decisions, which can have negative consequences. Therefore, it is

important to preprocess the data before feeding it into the machine learning algorithm. This preprocessing stage involves tasks such as cleaning the data, transforming it into a suitable format, and enriching it with additional features or attributes [6].

Cloud-based technologies make it possible to deploy ML and DL models at scale to enhance the capabilities of the SG. DL, a powerful form of machine learning, has demonstrated its effectiveness in various applications like image classifications and language processing. Lately, DL has gained increasing attention for its potential applications in SG applications, including load prediction, fault identification, and security [7].

The increasing demand for electric power has resulted in increased electrical losses, encompassing both technical loss (TL) and non-technical loss (NTL). TL is attributed to inefficiencies in electricity transmission and distribution, whereas NTL arises from theft or self-consumption [8].

Electricity larceny is a significant problem for utility companies globally, as it makes up a significant percentage of their overall losses. The annual losses due to non-technical losses amount to approximately 89.3 billion U.S. dollars for electricity companies worldwide [9]. Therefore, it is crucial to take this issue seriously by developing and implementing a smart system that can effectively address non-technical losses in a low-cost and efficient manner.

Our system utilizes cloud computing techniques to monitor electricity consumption and detect larceny. By leveraging a cloud server, we ensure secure storage and remote analysis of data, enabling real-time monitoring from anywhere.

To enhance larceny detection efficiency, we have implemented a CNN model. CNN's ability to understand complex patterns and identify unusual behaviors makes it an excellent solution for theft detection. The model is trained using historical data to identify abnormal load patterns that may indicate electricity larceny.

Our paper focuses on effective data preprocessing, particularly in handling missing values using techniques such as 'linear interpolation' and 'fillna'. We also address data imbalance using various methods, achieving excellent results compared to the existing literature in electricity larceny detection despite its modernity.

The integration of cloud computing and deep learning improves the system's overall performance, reliability, security, and efficiency in handling large volumes of data. Additionally, utilizing cloud servers simplifies deployment and reduces training and execution time.

The remaining sections of the paper are structured as follows. Section 2 provides a review of different techniques employed to address the issue of electricity theft. Section 3 presents a comprehensive description of the proposed system, including its architecture, components, and functionalities. The proposed larceny detection technique, which utilizes CNN, is explained in detail in section 4. The obtained results are discussed in section 5. Finally, section 6 concludes the paper by summarizing the main findings.

2. Related works

Over the past few years, numerous researches have been carried out to develop effective systems for detecting electricity theft. These studies have explored various approaches and techniques to enhance the accuracy and efficiency of theft detection models. In this section, we will discuss some notable works that have contributed to the field.

In 2019, Hasan et al. proposed a system for detecting electricity theft utilizing a hybrid CNN-LSTM deep learning model. This model combines the strengths of CNN and LSTM networks. CNN is responsible for extracting features and classification, while LSTM is specifically designed to handle time-series data. To handling of class imbalance problems, SMOTE was employed. The model achieved an impressive overall accuracy of 89% [10].

In 2020, Adil et al. introduced a model that combines LSTM and bat-based random under sampling boosting (RUSBoost) techniques for the effective handling of class imbalance problems. The RUSBoost technique specifically addresses the issue of imbalanced data by undersampling the majority class while boosting the minority class. On the other hand, the bat algorithm is employed for parameter tuning to optimize the performance. The evaluation metrics of the proposed model demonstrate its effectiveness, with an F1-score of 96.1%, a precision of 88.9%, a recall of 91.09%, and a ROC-AUC of 87.9%. While the proposed model outperformed alternative techniques, the paper highlights its sensitivity to changes in input data. This suggests that the model's performance may vary when applied to different datasets or when the data undergoes significant changes [11].

In 2020, Mujeeb et al. presented an enhanced model for electricity theft detection that incorporates the differential evaluation random under sampling boosting (DE-RUSBoost) classifier. This classifier is optimized using the differential evaluation (DE) meta-heuristic optimization algorithm. The proposed method achieved a high accuracy of 96%, and a false

detection rate of 0.004 [12].

In 2020, Chen et al. introduced a novel approach called ETD-DBRNN for detecting electricity theft. This approach utilizes deep bidirectional recurrent neural networks (DBRNN) to effectively analyze time-series data. By combining the strengths of both DRNN and Bi-RNN, the proposed method captures both the internal characteristics and external correlations within the data. The DBRNN model achieves impressive performance metrics. The accuracy of the model is recorded as 97.44%, a precision of 95.70%, a recall of 99.74%, and the F1-Score 92.09% [13].

In 2021, Pereira et al. used CNN for detecting instances of electricity theft. The researchers addressed the challenge of unbalanced class distributions in the dataset by employing several techniques. These techniques include random undersampling, cost-sensitive learning, random oversampling, synthetic minority oversampling technique, K-medoids-based undersampling, and cluster-based oversampling. The study concludes that the random oversampling technique (ROS) performs the best in terms of area under the curve (AUC), with a value of 0.6714 [14].

In 2023, A. Nawaz et al. propose a novel ensemble model called CNN-XGB for detecting electricity theft in smart grids. This model combines a convolutional neural network (CNN) with an extreme gradient boosting (XG-Boost) classifier to achieve highly accurate results. The CNN component extracts essential features from both one-dimensional (1-D) and two-dimensional (2-D) electricity consumption data, while the XG-Boost classifier further enhances the prediction accuracy. The CNN-XGB model has achieved 92% accuracy in detecting electricity theft [15].

In 2022, Lucas et al. employed a BiGRU-CNN artificial neural network, which combines the strengths of bidirectional gated recurrent unit (Bi-GRU) and CNN architectures. This hybrid approach allows the model to extract long-term temporal correlations through the Bi-GRU layer and capture local trends using the CNN layer. The model was evaluated utilizing various metrics. The accuracy of the model is reported as 0.929, the precision as 0.885, the recall as 0.801 the F1-Score 0.841, and the ROC AUC of 0.966 [16].

In 2020, Syed et al. developed an LSTM-based electricity theft detection model. The model is capable of capturing long-term dependencies in data sequences. The proposed model achieves an accuracy of 92.69%. The authors conclude that their proposed methodology is simpler and can be applied to various SG applications [17].

In 2023, Youngghyu et al. present a model that tackles the challenge of imbalanced data by generating synthetic electricity theft data with the same characteristics as real data utilizing VAE-GAN (variational autoencoder-generative adversarial network). Once the balanced dataset is obtained, a CNN model is employed as the detector. The reported model's performance values for precision, true positive rate, F1-score, and MCC are 0.925, 0.909, 0.905, and 0.834, respectively [18].

In 2021, Ouamane conducted research on detecting fraudulent behavior in power usage. The study involved the development of deep learning-based models utilizing both 1D-CNNs and 2D-CNNs. The 1D-CNNs model achieves an accuracy of 94.52%, and an AUC value of 77.66%. On the other hand, the 2D-CNNs model did not perform as well as the 1D-CNNs model. The accuracy of the 2D-CNNs model was 93.14% [19].

Table 1. Performance comparison of electricity larceny detection techniques

Ref	Techniques applied	Data Balancing Techniques	Accuracy	Precision	Recall	AUC	Dataset
10	CNN-LSTM	SMOTE	89%	94.04%	87%	-	SGCC [20]
11	LSTM	RUSBoost	87.9%	88.9%	91.09%	-	SGCC
12	DE-RUSBoost	RUSBoost	96%	90.2%	-	-	SGCC
13	DBRNN	-	97.44%	95.70%	99.74%	-	Irish Smart Energy Trials [21]
14	CNN	ROS	78.61%	-	-	-	SGCC
15	CNN-XGB	-	92%	92%	54%	54%	SGCC
16	BiGRU-CNN	-	92.9%	88.5%	80.1%	96.6%	SGCC
17	LSTM	-	92.6941	-		-	SGCC
18	CNN	VAE-GAN	94%	92.5%	90.9%	-	SGCC
19	1D-CNN 2D-CNN	-	94.52% 93.29%	-		77.66% 72.72%	SGCC
Our system	CNN	ADASYN	97.22%	96.8%	99.9%	97.22%	SGCC

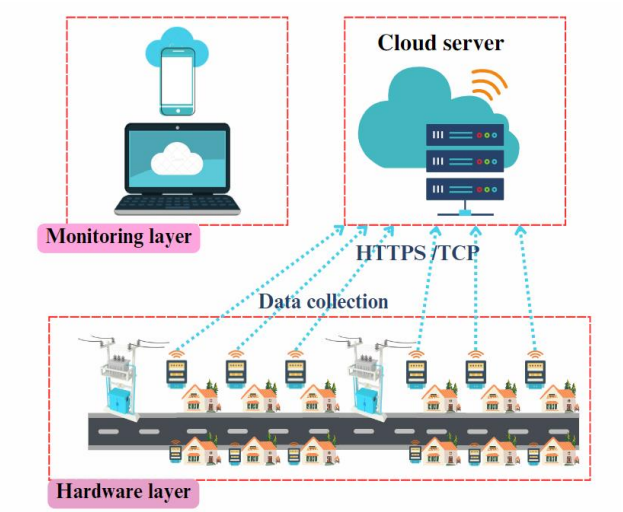

Figure. 1 The proposed system framework

Looking at the results achieved by our model compared to the other models listed in Table 1, in terms of various metrics, our model outperforms the alternative methods. These results indicate that our system is more effective and reliable in detecting instances of electricity theft.

3. Proposed system

The purpose of our system is to monitor electricity usage for consumers and detect electricity theft efficiently using cloud computing and deep learning techniques. Implementing this system requires the deployment of smart meters to collect data, which constitutes the device layer. This data is then transmitted to the cloud server layer, where it is stored and analyzed. A DL approach is employed to detect theft, which will be discussed in detail in section 4. The analysis results are displayed in a web-based application in the monitoring layer. Fig. 1 illustrates the framework of the proposed system.

The hardware layer is represented by a set of smart meters equipped in residential areas for monitoring electrical usage and collecting data. It incorporates an Arduino Mega Wi-Fi R3 module, which manages communication between the energy meter and the LCD, as well as facilitates data transmission to the cloud server layer.

The PZEM-004T energy meter is responsible for measuring electrical parameters such as voltage, current, power, and energy consumption. It accurately captures these parameters and sends the data to the Arduino Mega Wi-Fi R3 module, which in turn sends it to the cloud server. To provide users with a clear understanding of their electrical usage, the hardware layer incorporates an LCD. Fig. 2 illustrates the architecture of the smart meter.

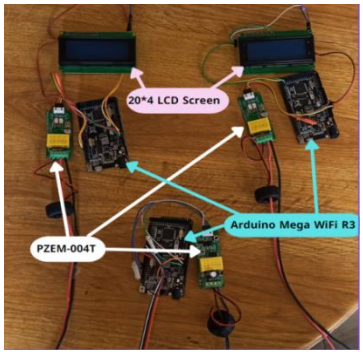

Figure. 2 The architecture of the smart meter

The cloud server layer plays a crucial role in the proposed system as it serves as the foundation for storing, managing, and analyzing data gathered from the device layer. In our system, we have chosen to implement an infrastructure as a service (IaaS) model, which grants the organization ownership and operation of its cloud infrastructure. This allows for enhanced control and security over the system's data, as well as the ability to scale and adapt to changing data management requirements.

In the cloud server layer, the collected data from the smart meters is stored and analyzed using DL techniques to detect electricity theft and identify abnormal load patterns.

The monitoring layer is represented by a web-based application that enables authorities to monitor the real-time electricity usage of consumers and views the results of larceny analysis. This application has been developed using the ASP.NET Core framework and is hosted on the cloud server, allowing users to access it from anywhere and at any time.

4. The proposed larceny detection technique

In this section, we will provide a detailed explanation of the theft detection technique employed in the system. The system utilizes DL, more specifically, CNNs to detect theft.

4.1 Convolutional neural networks (CNNs)

CNNs are a specialized type of artificial neural network primarily used for processing data arranged in a grid-like structure, such as images or videos. These networks employ a mathematical operation called convolution, which combines two functions to generate a third function representing the modified version of one function by the other [10]. Compared to traditional classification methods, have the capability to capture and understand more intricate non-linear relationships in data. This allows them to achieve better generalization performance, meaning they can effectively classify new, unseen data

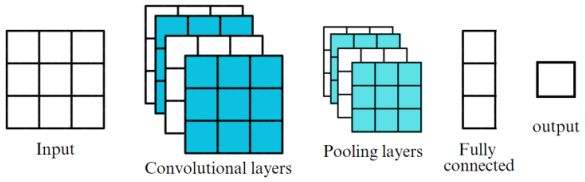

Figure. 3 The CNN general architecture

accurately [22]. CNN networks' main goal is to extract features using a kernel, which is like a filter that slips over the input and performs a convolution process. This process results in the creation of a resource map. By using different kernels, we can obtain diverse resource maps, we can obtain diverse resource maps, which are then incorporated to create the convolution layer's output [23]. CNN architecture is made up of three types of layers: convolutional layers, pooling layers, and fully connected layers. Convolutional layers are particularly crucial for capturing the essential features of electrical theft patterns [22]. Fig. 3 shows the general arrangement of these layers that are commonly used to construct a CNN.

The behavior of the layers in the CNN can be described using Eqs. (1) and (2) [16].

$$y_i = f_i(x_i w_i + b_i) \tag{1}$$

$$\dot{y} = max(y_{i,j}) \tag{2}$$

Where each convolution and max-pooling layer takes an input x_i and produces an output y_i and is associated with an activation function f_i. The behavior of the convolution layer is defined by the offset vector b_i and weights w_i, which are used to compute the output y_i through Eq. (1). The max-pooling layer then takes the highest value of the outputs, producing the output y' through Eq. (2) [16].

4.2 Dataset overview

The dataset used to train the model is sourced from Kaggle [20] and is a smart grid dataset provided by the state grid corporation of China (SGCC). It contains data on the electricity consumption of consumers from January 1, 2014, to October 31, 2016. The dataset consists of 42,372 electricity customers, of which the first 3615 have been identified as having committed electricity theft, while the remaining customers have been found to be honest.

4.3 Data prepossessing

SG technology has revolutionized electricity generation, distribution, and consumption. However, the data collected by SGs is often messy and error-prone, making it challenging to analyze and interpret [24]. Data preprocessing plays a vital role in addressing these issues and involves the following steps:

a. **Data cleaning:** One important aspect of data preprocessing is removing duplicate rows, which can impact the accuracy of the analysis and modeling results.

b. **To fill missing values (NaN)**, the 'linear interpolation' technique is utilized, estimating values based on surrounding data points [17]. Sequential missing values up to a maximum of two can be filled. Because linear interpolation may not accurately predict values. Any remaining missing values are then filled using the 'fillna' function, replacing NaN values with '0' to ensure dataset completeness. The mathematical model for linear interpolation can be represented as follows: Consider a set of data $(x_1, x_2, \ldots, x_n)$ and we want to estimate the value of x_i between x_{i-1} and x_{i+1}. We can approximate the value of $f(x_i)$ as follows:

$$f(x_i) = \begin{cases} \frac{x_{i-1}+x_{i+1}}{2}, x_i \in NaN, \begin{matrix}(x_{i-1}),\\(x_{i+1})\end{matrix} \notin NaN \\ x_{i-1}, \begin{matrix}(x_i)\\(x_{i-1})\end{matrix} \in NaN, \quad x_{i+1} \notin NaN \\ x_{i+1}, \begin{matrix}(x_i)\\(x_{i+1})\end{matrix} \in NaN, \quad x_{i+1} \notin NaN \\ x_i, \qquad\qquad x_i \notin NaN \end{cases} \tag{3}$$

c. **Outlier treatment:** is a technique used to handle extreme values that can have a significant impact on analysis. Outliers are data points that differ significantly from other data points in a dataset and can affect the accuracy of models. By assuming that an outlier point is any value in a given row that exceeds three standard deviations from the mean and should be replaced with the threshold value [19]. For each row in the dataset, the mean (m) and standard deviation (st) are calculated. Then, any value in the row that exceeds the threshold of $(m + 3st)$ is replaced with this threshold value.

d. **Normalization:** is a technique used to transform the data values to a similar range. The most commonly used normalization technique is the min-max scaling, where the values are scaled to a range between 0 and 1. The min-max scaling is applied to a SG dataset, to ensure that all features contribute equally to the analysis

[16]. The mathematical formula that represents this is:

$$f(x) = \frac{(x - x_{min})}{(x_{max} - x_{min})} \quad (4)$$

Where x : is a data point in a column, x_{min} : is the minimum value in that column, x_{max} : is the maximum value in that column, $f(x)$: is the normalized value between 0 and 1.

4.4 Handling data imbalance

In machine learning, it is common to encounter data imbalance, which refers to a notable disparity in the number of cases across different classes. In the case of detecting electricity theft in SG data, the majority class is "no larceny" with 36677 instances and the minority class is "larceny" with only 3579 instances. This can cause the CNN model could lean towards the majority class, which will negatively affect the performance of the minority class. In order to solve this issue, several techniques were tested, which are:

a. Adaptive synthetic sampling (ADASYN):

This technique generates synthetic instances of the minority class in regions of the feature space where the density of minority instances is low while keeping the density of majority instances unchanged. ADASYN is effective when the distribution of data is highly skewed and focuses more on those minority instances that are harder to learn [25].

b. Random under-sampling (RUS):

This technique arbitrarily deletes samples from the majority class until the dataset is balanced. RUS can be effective when the majority class has numerous samples and removing some of them does not significantly impact the performance of the model. [26].

c. Random over sampling (ROS):

This technique creates new samples by randomly replicating available samples in the minority class until the number of instances for each class is balanced. ROS is a simple and effective way to handle the problem of imbalanced datasets and can enhance the performance of ML models [26].

d. Synthetic minority over-sampling technique (SMOTE):

This technique generates new synthetic instances of the minority class by interpolating between existing instances. SMOTE selects a minority class instance and then selects one or more of its k nearest neighbors. New instances are generated along the line segments joining the selected minority instance and its selected neighbors [27].

e. SMOTETomek:

The SMOTETomek technique is a blend of two methods, namely SMOTE and Tomek links. Tomek links refer to pairs of samples belonging to different classes that are in close proximity. These pairs of samples can sometimes cause noise in the data, leading to a decrease in model performance. Tomek links are identified and removed from the dataset to enhance the accuracy of the model. The SMOTE technique is first used to oversample the minority class and generate synthetic samples. Then, Tomek links are used to remove any noisy samples from the dataset [28].

4.5 Data splitting

Data splitting is implemented before training the model which involves dividing our dataset into two subsets: the training set and the test set. The purpose of this is to utilize the training set for model training and subsequently evaluate the model's performance on unseen data by employing the test set. We used a test size of 20% of the data and the remaining 80% was used as the training set.

4.6 Proposed CNN model

CNNs can identify specific features of the consumption data that are indicative of electricity theft, such as sudden spikes or drops in usage. The ability to identify these patterns makes CNNs a powerful tool for detecting electricity theft. The CNN model is arranged as:

Convolutional layers: These layers apply filters to the input data and extract relevant features. In our model, we have two convolutional layers, each having 64, 7 filters, and a kernel size respectively. ReLU is used as an activation function, which helps to introduce non-linearity into the model and speed up the convergence of the model during training.

Dropout layer: This layer is added after the convolutional layers to prevent overfitting. Dropout randomly sets a fraction of the input units to zero during training, which helps to reduce over-reliance on certain features and encourages the model to learn more robust representations.

Flattening layer: The output of the dropout layer is then flattened, which converts the 3D output of the convolutional layers into a 1D input that can be passed through a fully connected layer.

Fully connected layer (FCL): This layer has 32 neurons and a ReLU activation function. The FCL's purpose is to perform feature extraction on the flattened output of the convolutional layers.

Dropout layer: Another dropout layer is added

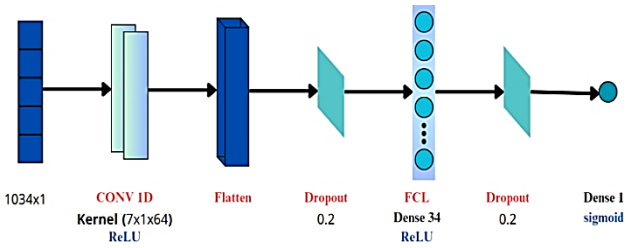

Figure. 4 Architecture of the proposed CNN model

		Actual value: Positive	Actual value: Negative
Predicated value	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Positive (TP)

Figure. 5 Confusion matrix form

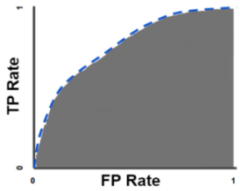

Figure. 6 Area under the curve (AUC) [18]

before the final output layer

The output layer consists of only one neuron that utilizes a sigmoid activation function. This function generates a probability score ranging from 0 to 1, which signifies the probability of a sample being a thief.

4.7 Evaluation metrics

ML and DL model performance are evaluated using evaluation metrics. Various evaluation metrics may be used, and each one is suitable for various use cases.

a. **A confusion matrix** summarizes the predicted and actual classes of a model's predictions on a given dataset. The matrix has two dimensions, the actual class (rows) and the predicted class (columns). The four cells of the matrix are populated based on the classification results: true negatives (TN), true positives (TP), false negatives (FN), and false positives (FP) [12].

b. **Accuracy** is determined by calculating the ratio of rightly categorized instances to the total number of instances [13].

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (5)$$

c. **AUC** stands for "Area Under the Curve". It is the measure of Separation between the positive and negative classes. An AUC of 1 indicates an ideal classifier [12].

d. **The F1 score** is a metric that considers both precision (P) and recall (R) [13]. It is defined as:

$$F_1 score = \frac{2 \times PR}{P + R} \quad (6)$$

Where precision (P) measures the accuracy of positive predictions made by a model. It is calculated as:

$$P = \frac{TP}{TP + FP} \quad (7)$$

And recall (R) measures the ability of a model to correctly identify positive instances. It is calculated as:

$$R = \frac{TP}{TP + FN} \quad (8)$$

4.8 Implementing a CNN model on cloud server

After training the CNN model, the best model is saved and deployed on the cloud server using ASP.NET Core. A user interface is also created to display the analyzed results, providing authorities with convenient monitoring capabilities.

To utilize the CNN model, new data is received from the device layer and stored in a database on the cloud server. The data is then analyzed using CNN, allowing the model to process and interpret the information to detect instances of electricity theft.

5. Discussions and results

5.1 Handling data imbalance

Table 2 illustrates the effects of using different methods to handle imbalanced data on the original data. The "Original" column represents the original dataset without balancing. The techniques used are RUS, ROS, SMOTE, SMOTETomek, and ADASYN. Table 2 contains the number of consumers with no larceny, the number of consumers with larceny, and the total number of consumers.

5.2 CNN model evaluation

Table 3 presents the results of training the CNN model for 10 epochs on the dataset after applying several techniques to address the class imbalance. When we applied data balancing techniques, we

Table 2. Distribution of consumers with and without larceny using different data imbalance techniques

Data Balancing Techniques	Original	RUS	ROS	SMOTE	SMOTETomek	ADASYN
No. of cons with no larceny	36677	3579	36677	25705	29370	25705
No. of cons with larceny	3579	3579	36677	12852	14685	25795
Total cons	40256	7158	73354	38557	44055	51500

Table 3. Performance of CNN model with various data balancing techniques

Metrics	Original dataset	RUS	ROS	SMOTE	SMOTETomek	ADASYN
Training accuracy	99.34%	78.06%	98.35%	99.49%	96.60%	99.34%
Validation accuracy	93.63%	76.61%	98.37%	94.70%	94.89%	97.73%
Test accuracy	93.18%	79.61%	98.33%	94.70%	97.33%	97.22%
F1-Score for positive class (larceny)	57.61%	78.21%	98.35%	95.91%	96.49%	97.14%
F1-Score for negative class (No larceny)	96.29%	75.65%	98.55%	92.45%	96.34%	97.30%
AUC	73.96%	78.06%	98.35%	95.27%	96.60%	97.22%
Precision	67.2%	79.1%	96.9%	88.3%	92.9%	96.8%
Recall	50.4%	74.3%	99.9%	97%	96.9%	99.9%

observed improvements in accuracy and other metrics compared to the original dataset without any processing.

RUS technique showed improvements specifically in the F1-score for the positive class and the AUC when dealing with severe class imbalance. The original model might have been biased towards the majority class, leading to lower F1-scores and AUC. RUS rebalances the class distribution, allowing the model to focus on learning from the positive class, thereby improving its ability to identify positive instances and distinguish them from negative ones. However, this technique may result in a decrease in overall accuracy as it involves sacrificing information by removing instances from the majority class.

On the other hand, when applying ROS, we achieved the highest results among the techniques. However, these high results might be attributed to overfitting, where the model memorizes the repeated instances without truly understanding the underlying patterns. Consequently, the model shows good results for the available dataset but may struggle when applied to new data.

The remaining three techniques, namely SMOTE,

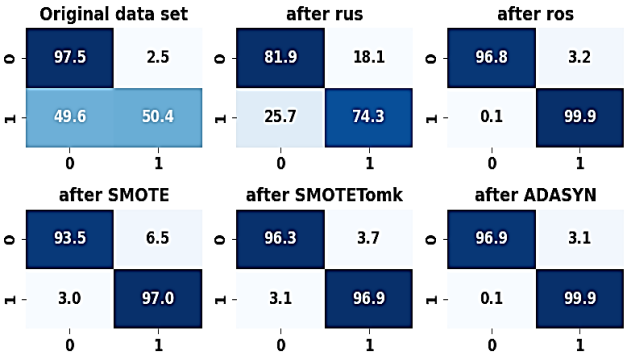

Figure. 7 The confusion matrices of the CNN model with various data balancing techniques

SMOTETomek, and ADASYN, demonstrated promising performance in addressing the class imbalance. However, based on the provided results, ADASYN showed comparable or slightly better performance across different metrics, including accuracy, F1-scores, and AUC. ADASYN can be considered a reliable choice as it effectively handles class imbalance. The performance of the CNN model can be easily understood through confusion matrices which showcase the impact of various techniques on its predictions for minority and majority classes, as

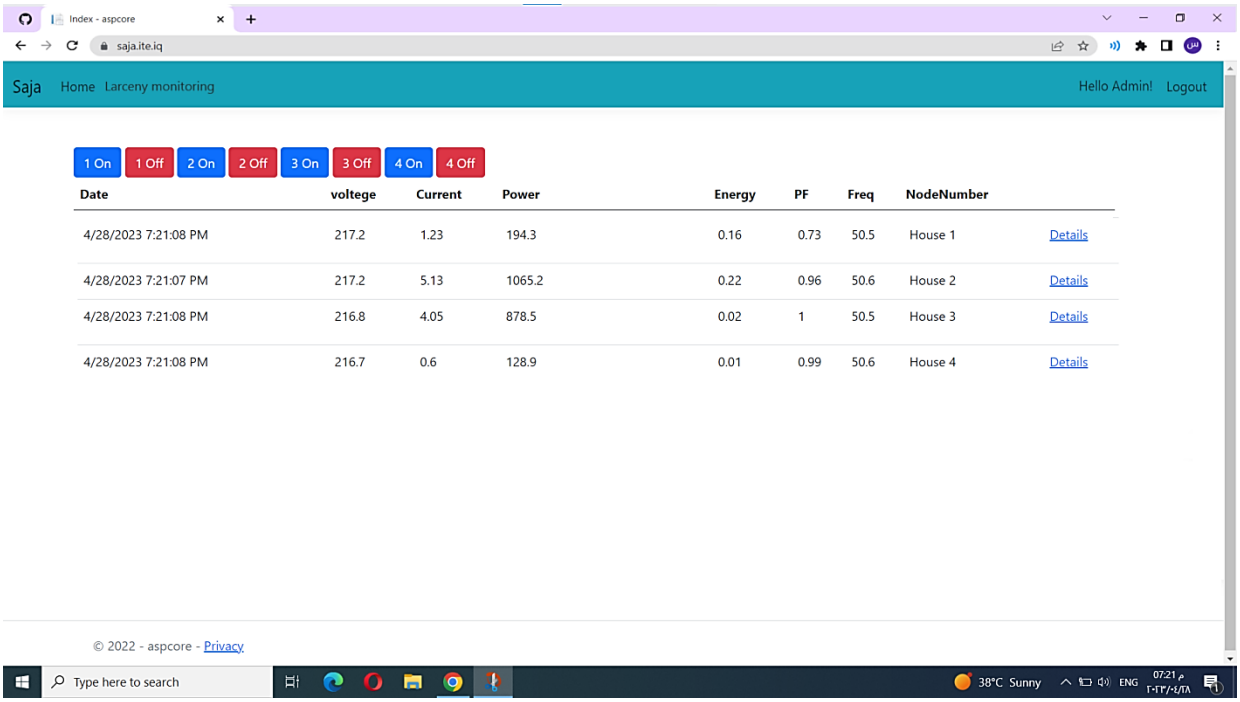

Figure. 8 Electric usage monitoring interface

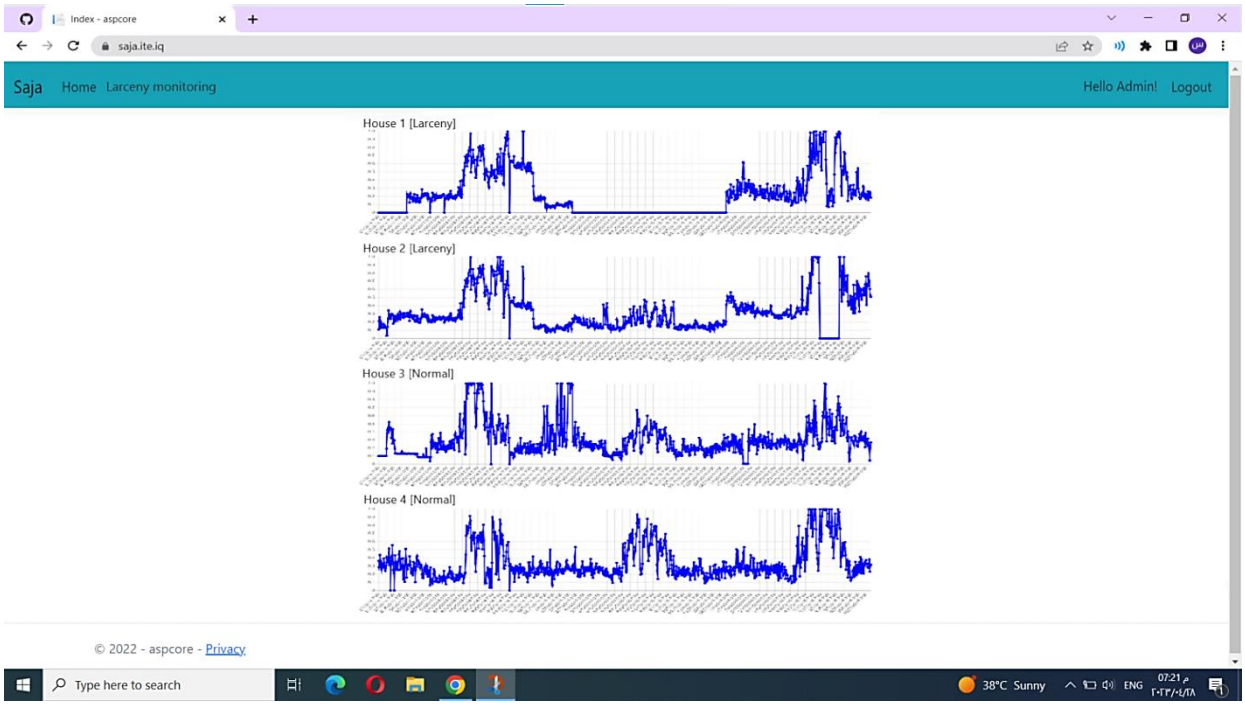

Figure. 9 Larceny monitoring interface

depicted in Fig. 7.

5.3 The results of CNN analysis

The system includes two interfaces: the electric usage monitoring interface and the larceny monitoring interface.

The system was tested in a real-world scenario, as shown in Figs. 8 and 9. Fig. 9 illustrates the load pattern of the thief consumers (first two houses), which was detected by the CNN model, and the normal load pattern of houses 3 and 4.

6. Conclusion

In this paper, we have presented a system that detects electricity theft using cloud computing and deep learning techniques. The evaluation of our

model has demonstrated its effectiveness in accurately detecting theft incidents. With an accuracy of 97.22%, precision of 97%, and recall of 99.9%, our model has showcased high levels of accuracy and reliability in identifying electricity theft. Through our research, we have addressed the challenge of imbalanced data by utilizing the ADASYN (Adaptive Synthetic) technique, which has significantly improved the performance of our detection model. Furthermore, the deployment of cloud computing technologies has played a vital role in facilitating the dissemination of our CNN model. The cloud server has provided storage capabilities, processing environments, and a platform for executing the model.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The manuscript has been prepared by the 1st author while the review and editing have been performed by the 2nd and 3rd authors.

References

[1] S. Bera, S. Misra, and J. Rodrigues, "Cloud Computing Applications for Smart Grid: A Survey", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26, No. 5, pp. 1477-1494, 2015.

[2] J. Popeangă, "Cloud computing and smart grids", *Database Systems Journal*, Vol. 3, No. 3, pp. 57-66, 2012.

[3] M. Shubbar, L. A. Rahaim, and A. Hamad, "Larceny Revelations of Electric Energy with Cloud Computing", In: *Proc. of* 2021 *International Conference on Advance of Sustainable Engineering and its Application (ICASEA)*, Wasit, Iraq, pp. 77-82, 2021.

[4] P. Siano, "Demand response and smart grids—A survey", *Renewable and Sustainable Energy Reviews*, Vol. 30, pp. 461-478, 2014.

[5] T. Kotsiopoulos, P. Sarigiannidis, D. Ioannidis, and D. Tzovaras, "Machine Learning and Deep Learning in smart manufacturing: The Smart Grid paradigm", *Computer Science Review*, Vol. 40, pp. 100341-100377. 2021.

[6] E. Hossain, I. Khan, F. U. Noor, S. Sikander, and M. Sunny, "Application of Big Data and Machine Learning in Smart Grid, and Associated Security Concerns: A Review", *IEEE Access*, Vol. 7, pp. 13960-13988, 2019.

[7] M. Massaoudi, H. A. Rub, S. Refaat, I. Chihi, and F. Oueslati, "Deep Learning in Smart Grid Technology: A Review of Recent Advancements and Future Prospects", *IEEE Access*, Vol. 9, pp. 54558-54578, 2021.

[8] F. Savian, J. Siluk, T. Garlet, F. Nascimento, J. Pinheiro, and Z. Vale, "Non-technical losses: A systematic contemporary article review", *Renewable and Sustainable Energy Reviews*, Vol. 147, pp. 1-13, 2021.

[9] M. Jeffin, G. Madhu, A. Rao, G. Singh, and C. Vyjayanthi, "Internet of Things Enabled Power Theft Detection and Smart Meter Monitoring System", In: *Proc. of International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, pp. 0262-0267, 2020.

[10] D. Hasan, R. Toma, A. Nahid, M. Islam, and J. Kim, "Electricity Theft Detection in Smart Grid Systems: A CNN-LSTM Based Approach", *Energies*, Vol. 12, No. 17, pp. 3310-3328, 2019.

[11] M. Adil, N. Javaid, U. Qasim, I. Ullah, M. Shafiq, and J. Choi, "LSTM and Bat-Based RUSBoost Approach for Electricity Theft Detection", *Applied Sciences*, Vol. 10, No. 12, pp. 4378-4399, 2020.

[12] S. Mujeeb, N. Javaid, R. Khalid, M. Imran, and N. Naseer, "DE-RUSBoost: An Efficient Electricity Theft Detection Scheme with Additive Communication Layer", In: *Proc. of ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, pp. 1-6, 2020.

[13] Z. Chen, D. Meng, Y. Zhang, T. Xin, and D. Xiao, "Electricity Theft Detection Using Deep Bidirectional Recurrent Neural Network", In: *Proc. of the 22nd International Conference on Advanced Communication Technology (ICACT)*, Phoenix Park, Korea (South), pp. 401-406, 2020.

[14] J. Pereira and F. Saraiva, "Convolutional neural network applied to detect electricity theft: A comparative study on unbalanced data handling techniques", *International Journal of Electrical Power & Energy Systems*, Vol. 131, pp. 107085-107092, 2021.

[15] A. Nawaz, T. Ali, G. Mustafa, S. U. Rehman, and M. Rashid, "A novel technique for detecting electricity theft in secure smart grids using CNN and XG-boost", *Intelligent Systems with Applications*, Vol. 17, p. 200168, 2023.

[16] L. D. Soares, A. D. S. Queiroz, G. López, E. C. Franco, J. L. Lezama, and N. M. Galeano, "BiGRU-CNN Neural Network Applied to Electric Energy Theft Detection", *Electronics*, Vol. 11, No. 5, pp. 693-706, 2022.

[17] D. Syed, H. A. Rub, S. Refaat, and L. Xie, "Detection of Energy Theft in Smart Grids using Electricity Consumption Patterns", *IEEE*

International Conference on Big Data (Big Data), Atlanta, GA, USA, pp. 4059-4064, 2020.
[18] Y. Sun, J. Lee, S. Kim, J. Seon, S. Lee, C. Kyeong, and J. Kim, "Energy Theft Detection Model Based on VAE-GAN for Imbalanced Dataset", *Energies*, Vol. 16, No. 3, pp. 1109-1112, 2023.
[19] O. FERIAL, "Fraud detection using deep learning Detecting energy consumption fraud using deep learning", *Archives.univ-biskra.dz*, 2021.
[20] [online] Available: https://www.kaggle.com/datasets/sreen28g10/electricity-theft-detection
[21] [online] Available: http://www.ucd.ie/issda/data/commissionforenergyregulationcer/.
[22] X. Gong, B. Tang, R. Zhu, W. Liao, and L. Song, "Data Augmentation for Electricity Theft Detection Using Conditional Variational Auto-Encoder", *Energies*, Vol. 13, No. 17, pp. 4291-4305, 2020.
[23] R. Madhure, R. Raman, and S. Singh, "CNN-LSTM based Electricity Theft Detector in Advanced Metering Infrastructure", In: *Proc. of 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, pp. 1-6, 2020.
[24] L. Feng, S. Xu, L. Zhang, J. Wu, J. Zhang, C. Chu, Z. Wang, and H. Shi, " Anomaly detection for electricity consumption in cloud computing: framework, methods, applications, and challenges", *EURASIP Journal on Wireless Communications and Networking*, Vol. 2020, No. 1, pp. 1-12, 2020.
[25] M. Zakariah, S. A. A. Qahtani, and M. S. A. Rakhami, "Machine Learning-Based Adaptive Synthetic Sampling Technique for Intrusion Detection", *Applied Sciences*, Vol. 13, No. 11, pp. 6504-6535, 2023.
[26] K. Akyol and Ü. Atila, "Comparing the Effect of Under-Sampling and Over-Sampling on Traditional Machine Learning Algorithms for Epileptic Seizure Detection", *Academic Platform - Journal of Engineering and Science*, Vol. 8, No. 2, pp. 279-285, 2020.
[27] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique", *J. Artif. Intell. Res.*, Vol. 16, pp. 321-357, 2002.
[28] Z. Wang, C. Wu, K. Zheng, X. Niu, and X. Wang, "SMOTETomek-Based Resampling for Personality Recognition", *IEEE Access*, Vol. 7, pp. 129678-129689, 2019.