Validation of Data Streams using Time Series Forecasting

Philsy Baban
Databases and Information Systems Group
TU Ilmenau, Germany
philsy.baban@tu-ilmenau.de

ABSTRACT

In an IoT environment, sensors capture data about the product usage, environmental conditions, etc., in regular intervals of time. This data is analyzed to gather information about the current situations or business environments that help make future decisions. However, the quality of data received from the sensor is poor due to sensor failures and malfunctions. In this imperfect data, some of the failures like manipulated data are not easily identifiable. Therefore, we have to validate the correctness of the data received from sensors along with data pre-processing. Data validation tasks are based on the application areas, and this can be performed per tuple, per sensor / customer, etc. In this paper, we developed a framework to validate the correctness of data for the energy systems domain. In this framework, validation is performed in three levels: validation per customer, validation per location, and validation per context using time series forecasting. As a result, the quality of data is improved.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

Data quality, Data validation, IoT data streams, Time series forecasting

1. INTRODUCTION

Nowadays, people, devices, infrastructures, and sensors can continuously communicate and exchange data. Thus, a vast amount of data is generated during the communication. This massive amount of data, called big data, provides information about customer needs, service quality, prediction and prevention of risks, etc. In the IoT paradigm, to collect data from different fields such as environmental data, geographical data, astronomical data, and logistic data, sensors are embedded into various devices and deployed. Statistical reporting, monitoring of systems and data, and forecasting are some of the big data applications.

In the big data era, data quality is far from perfect as data is generated from a wide variety of data sources. The data generated by IoT devices are not in a format ready for analysis. Because received data may have quality problems, such as data errors, missing information, inconsistencies, noise, redundancy, manipulated data, etc. For effective analysis, we need high-quality data. Otherwise, low data quality will lead to serious decision-making mistakes. Thus, the data's correctness is a crucial factor for the operation and reliability of a system. At the same time, data changes very fast, and if the system cannot process the data in real-time, then the data is invalid and outdated. Therefore, we need real-time processing engines to transform and filter what is to be stored since raw data storage is difficult. The selection of data quality elements will differ in different application environments.

In this paper, we consider the energy systems domain. Conventional electricity grids are inefficient and unreliable systems due to the issues such as low reliability, high outages, high greenhouse gas, and carbon emission, economics, safety, and energy security [8]. In order to solve these issues smart grid is proposed. A smart grid is a distributed intelligent energy system that enables the two-way flow of electricity and data.

In a smart grid, we can collect and analyze data acquired from transmission lines, distribution substations, and consumers to predict power supply and demand for power management. Energy demand prediction plays a significant role in the proper scheduling and operation of power systems [17]. Accurate energy forecasts can reduce energy costs such as maintenance and operation costs, enhance energy management, increase reliability and efficiency, and make better future development decisions. For energy demand forecasting, historical energy consumption data and relevant effect factors are required.

To improve the data quality, data must be pre-processed. During pre-processing, the system can identify and remove data quality problems such as missing data in a data record, inconsistent data, and data redundancy. Even after data pre-processing, the correctness of data cannot be assured. For this, we have to validate the data. This paper proposes an approach to validate energy consumption data using time series forecasting as a reference in real-time. The remaining part of this paper is organized as follows: Section 2 and 3 discuss some concepts about time series forecasting models and data integrity; Section 4 overviews the related

32^{nd} GI-Workshop on Foundations of Databases (Grundlagen von Datenbanken), September 01-03, 2021, Munich, Germany.

work in the area of data validation and energy forecasting models; Section 5 introduces our proposed data validation framework; Results are provided in section 6; Section 7 concludes the paper and proposes some future work.

2. DATA FAILURES

Data are a valuable asset that connects the cyber and physical worlds. High-quality data results in intelligent decisions. However, in the real world, data are often dirty. This is mainly due to harsh environments, interference, malicious nodes, network congestion, sensor breakdown, sensor malfunction, insufficient battery power, etc. As a result, the sensor fails to generate accurate data. Following are some of the main data failures in IoT application domains [11],

- Inconsistent data: Measured value may contain inconsistent data due to node failure or sensor malfunction.
- Dropped data: Some data records can be dropped or unavailable due to network congestion or may be due to some interference.
- Data duplication: Duplicate records can be received for processing. This is mostly due to malicious nodes, sensor malfunction, or maybe due to insufficient battery power.
- Manipulated data: Measurements made by the sensors were altered, thereby compromising the data integrity. For example, meter readings are altered in the energy systems domain due to this data integrity attack. During this attack, the attacker aims to modify the data measured by the sensor in four general ways [2],

  1. Modify the data by adding noise to the original measured value.
  2. Modify the data with the historical sensor measurements from the user.
  3. Modify the actual data by erroneous data but still clinically plausible.
  4. Modify the data with the data from another user.

  Also, manipulated data are not easily identifiable. This is a severe threat to grid operations, such as energy loss, incorrect energy forecasting of energy, uneconomical or even catastrophic decisions, etc. Therefore, the correctness of data has to be assured before data processing for the better performance of the system.

Since unreliable data, dropped data and duplicate data are observable and identifiable, removing these data can be performed during pre-processing. In this paper, we discuss how to validate the correctness of data. In section 4, we discuss methods that can be used for data validation.

3. RELATED WORK

[4] discusses an approach to detect sensor malfunction with the combined use of Continuous Wavelet Transform (CWT) and image analysis techniques. In this method, the CWT scalogram obtained from the test signal is compared with the scalogram obtained from the same signal's historical data. This method provides better performance than PCA (Principal Component Analysis) based approach and binary SVM classifier for data validation. Using this approach, different types and intensities of the sensor malfunctions from energy production plants can be identified. The main limitation is that using this method, sensor malfunctions due to drifts cannot be identified because of the regularity of the signal.

In [15] proposed VortoFlow, a domain-based data stream validation model using domain-specific modeling language called Vorto DSL to describe the characteristics of IoT devices declaratively. In this model, validation rules are derived from pre-defined models at run-time, and it can perform automated data validation. This approach captures the validity ranges for the online validation of data streams. The main restriction of this model is limited dimensions. For example, it did not support the temporal context.

[18] provides a centralized data validation algorithm to estimate the missing data and data outliers. When missing data or an outlier is identified, this algorithm tries to estimate accurate data by considering the temporal and spatial correlation between nearby sensors. The main drawback of this algorithm is that it considers only the errors like missing data and outliers.

All the approaches mentioned above perform only one level of validation. Even though the approach [18] considers the temporal and spatial context, it cannot validate the data's correctness. If we validate the correctness of data by considering different aspects or dimensions, this will increase data reliability. Therefore, we aim to validate the correctness of data by considering multiple levels of validation like per tuple, per context, etc.

4. DATA VALIDATION

As the data volume increases, quality decreases. Recently, this big data is used as a basis for many crucial business decisions. Thus, the correctness of data is essential. Therefore, before performing the actual processing or analysis of data, validation must be performed along with the pre-processing or data cleaning. This can reduce the errors in the data to a certain extent.

The data has to be validated by considering different aspects since manipulated data are not easily identifiable. Also, data validation tasks performed in the system are based on the application areas. Data validation tasks can be divided into different levels with a growing degree of complexity from one level to another, including more and more information. Following are the different validation levels [7],

- **Validation level 0**: In this level, the format and file structure of the data record is validated. For example, the completeness of each record is validated.
- **Validation level 1**: Here, consistency within the elements in each data record is checked. For example, check data in fields like identifier or year is not negative.
- **Validation level 2**: In this level, validate the data received from the same source or sensor. For example, validate data received from the same sensor or customer using time series forecasting.
- **Validation level 3**: In this level, consistency of data record is assured based on the comparison of the data record with other files in the same domain, for example, validation of data receiving from the same location.

- **Validation level 4**: In this level, plausibility checks are performed for each data record to the data in the different domain by the same provider or context. For example, data is validated with respect to seasons in Germany.

- **Validation level 5**: In this level, validate the data record to the data from a different provider. For example, in energy systems domain validation data receiving from Europe.

Data validation methods can be used either for data correction or faulty data detection[14]. Detection of faulty data can be performed using forecasting. Forecasting is required in many application domains. For example, in the energy systems domain, forecasting is required to predict the future energy demand. In addition, the validity of the data can be performed by comparing the similarity of the data. In the following sections, we discuss the forecasting methods and methods used to calculate the similarity.

4.1 Forecasting

Forecasting [10] is about predicting the future trends or demands based on the available information that includes past data and knowledge about any future event that might impact the forecasts. Depending on the application domain, short-term, medium-term, and long-term forecasting is possible. Forecasting methods can be classified into two as qualitative forecasting and quantitative forecasting. If there is no relevant data available to forecasts, then qualitative methods must be used. On the other hand, quantitative methods are used if the historical numerical data is available and a high probability of continuing the past trend in the future.

4.1.1 Qualitative forecasting

In qualitative forecasting is a judgemental method where forecasts are based on expert's knowledge, theories, and experience in the field who have seen the working and ware of economic changes that can occur every year. Market research and the Delphi method are the two standard methods for qualitative forecasting. This method is mainly used for long-term forecasting. Since this method is mainly opinion-based, the result can be inaccurate.

4.1.2 Quantitative forecasting

In quantitative forecasting, historical data and current data are used. Time series analysis and causal methods are the two types of quantitative forecasting. In causal methods, along with time-series data, factors that affect the business are also considered for forecasting. In time series analysis, past and current time series data are used. Time series data is a collection of equally spaced temporal data that consists of components such as patterns, cyclical changes, seasonal fluctuations, and irregular data. Time series forecasting can be classified as univariate and multivariate time series forecasting based on the number of variables used for forecasting. The selection of models depends on the availability of the past data, application domains, accuracy, costs, etc. Following are some of the models used in the energy systems domain [6],

- Exponential Smoothing: This model computes a weighted average of past observations, where recent observations have the higher weight in the forecast.

- Autoregressive Integrated Moving Average: It is a combination of autoregression and moving average model with differencing. In the autoregression model, forecasting is done based on the linear combination of past values. In the moving average model, past forecast errors are used for forecasting. ARIMA model does not consider seasonal trends for forecasting. ARIMA models that are capable of modeling seasonal data are called SARIMA models.

- Neural Network Models: In this model, the neural network can learn and identify the direct connections, patterns, and trends in the time-series data that are difficult to portray. This model is mainly used for non-linear data. There are mainly three layers: an input layer, an output layer, and a hidden layer or intermediate layer. Bayesian Neural Network, K-Nearest Neighbour regression, Support Vector Regression, Recurrent Neural Network, Long Short-term Memory, etc., are mainly used neural network models.

- Hybrid Model: To improve accuracy, some forecasting models are combined to form hybrid models. ARIMA-ANN and SARIMA-SVM are some examples of hybrid models.

4.2 Similarity Measures for Time Series Data

The similarity of the time series data is measured mainly for clustering and classification [12]. Following are some of the algorithms used for distance calculation.

- Euclidean Distance [5]: It is the shortest path between two points on time series that occur simultaneously. This method cannot be used when the series are out of sync.

- Dynamic Time Warping (DTW) [13]: This algorithm measures similarity between two given time-dependent sequences under certain restrictions. Initially, this algorithm was used for speech recognition. This algorithm is mainly used for temporal sequences with varying lengths and speeds.

5. PROPOSED FRAMEWORK

In this section, we discuss the proposed framework. The integration of communications network to power grid results in a reliable and more flexible Smart Grid [9]. As a result, energy meters are now sensors that can send data continuously. At the same time, the quality of data is far from perfect. Consequently, analysis of this data results in making poor decisions or predictions. The presence of incorrect data in a data record is mainly due to attacks and failures. Attackers aim to alter the measurements made by the smart meters (sensors) with fictitious data that is plausible or not but not accurate. Identifying the incorrect data in a data record is not easy. There we need an estimate or prediction to check whether the data is correct or not. In order to check the presence of manipulated in the sensor data, we perform data validation.

In each application domain, many factors were influenced during data generation. In the energy systems domain, energy consumption in a residential building depends on factors such as the building area, the total number of people staying in the building, customer behavior, weather condition,

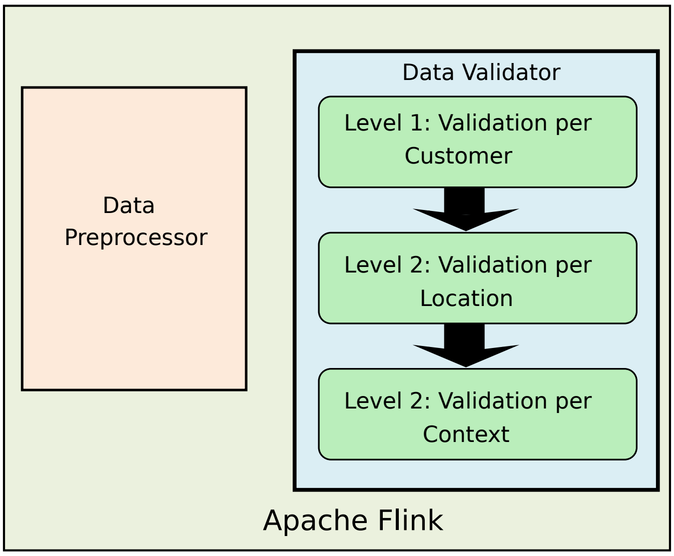
Data Validator
Level 1: Validation per Customer
Level 2: Validation per Location
Level 2: Validation per Context
Data Preprocessor
Apache Flink

Figure 1: Proposed system architecture

working life, etc. Thus, the validation of data by considering all the above conditions in a single step is difficult. Like we mentioned earlier, data validation can be performed at different levels. Therefore, we developed a framework that performs data validation in three levels. In our framework, the validation level 0 and 1 are performed as a part of preprocessing. Therefore, the first level in our framework will be the validation of data based on customer behavior. In the second level, we validate the data by considering the weather conditions and other exogenous factors. Finally, the validation is performed based on the general consumption behavior. For example, energy consumption on weekends will be high compare to weekdays. Also, consumption during the daytime will be less compared to nighttime. Therefore, we can consider most of the factors that affect the energy consumption pattern for validation using this framework. Figure 1 shows the developed framework.

Figure 1 shows the stream processing engine with data pre-processor and data validator. When the data record arrives, the data preprocessor checks whether the incoming record contains missing data, inconsistent data, or duplicate data. If the preprocessor did not find any issues mentioned above, the record is forwarded to the data validator. In data validator, validation is performed in three levels. The data record will be forwarded to the next level only if the record is valid. Otherwise, the validation process is aborted, and data in the record is considered invalid. In *Level 1*, our system checks that the data in the input record is in the predicted value range. In *Level 2* , the system checks that the data is spatially correlated with the data received from the same locality. The geographical data is stored in a database to perform the spatial validation, while the incoming data records contain only the device id, measured date, time, and meter value. The format of the input record will be discussed in detail in the next section. In *Level 3* , the system checks the data is valid to the day of a week(seasons.) If the data is valid in all three checks, then we consider this data as valid. These validity levels are implemented in Apache Flink. In Apache Flink, real-time validation is performed.

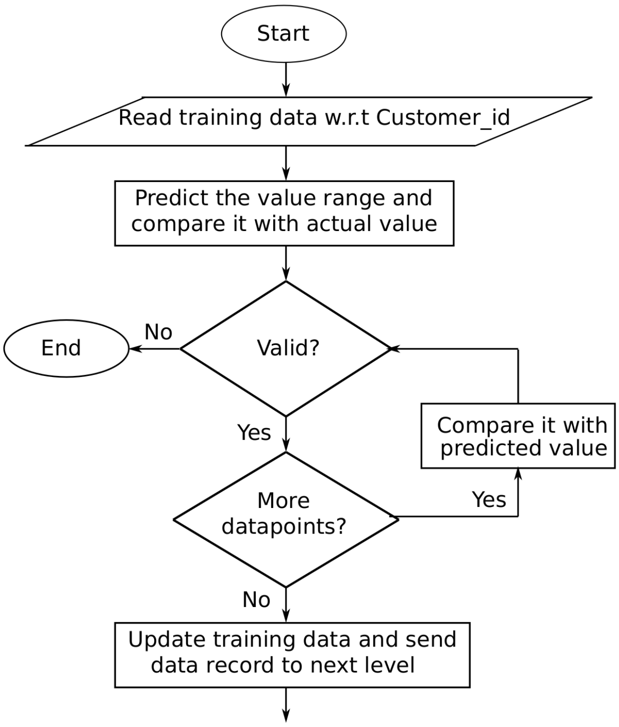
Start
Read training data w.r.t Customer_id
Predict the value range and compare it with actual value
Valid?
No
End
Yes
More datapoints?
Yes
Compare it with predicted value
No
Update training data and send data record to next level

Figure 2: Process flow of level 1

The algorithm that we used to perform three validation will be discussed in the following sections.

5.1 Data Validation per Customer

In a data validator, the first level of data validation is performed for each customer. Because the energy consumption pattern of each customer is different. This is mainly due to the customer behavior, space used by the customer, number of occupants, his/her working conditions, etc. Thus data should be validated for each customer. For this, previous consumption patterns are used for analysis and forecast future consumption. Time series forecasting is performed, and the training dataset contains only two fields as follows,

- ds: this field contains date and time with 30 minutes intervals. It is in format YYYY-MM-DD HH:MM: SS.
- values: energy consumed every 30 minutes with respect to date and time is stored in this field.

Figure 2 show the process flow for level 1. After pre-processing, the data record is sent to the first layer of the data validator. Based on the *customer id*, training data is selected and perform the training and prediction. This tool, along with the predicted values, predicts the maximum and minimum for the expected value. Thus, based on this range, our system checks the meter reading is valid or not. During prediction, this model predicts the following 12 values. Because in our input record, we receive 12 meter reading with a time interval of 30 minutes. Since this tool predicts 12 values, we do not have to perform a prediction for each time in the input record. If the record is valid, then the expected value is added to the training data to predict future time series.

For this validation, an open-source tool called Prophet developed by Facebook. Prophet [16] is a forecasting tool that automatically detects change points in a time series.

5.2 Data Validation per Location

In level 2, data validation of the data record is performed concerning the location. Each location is different. For example, people living in cities do not have the same consumption patterns as those living in villages. Nevertheless, the energy consumption pattern for all customers belongs to the same region will be similar as people from the same location have the same weather conditions and other exogenous conditions. Also, changes in the weather conditions will be reflected in the consumption pattern. As a result, this similarity can be used to validate data records receiving from the same region. The similarity between the time series obtained from the same location is calculated using the euclidean distance.

We calculated the minimum and maximum Euclidean distance for each location and stored it in a database from the historical data. Therefore, when an input record arrives, we calculate the euclidean distance for each data point and check the calculated value is in the range or not. If the data is not in the range, then that record is considered invalid and ends the validation process.

5.3 Data Validation per Context

In level 3, data is validated concerning a context. We can consider different contexts like weeks, months, or seasons. For example, we have different consumption in each season. The energy consumption pattern in winter is not the same as the pattern in winter. Due to less daylight and cold weather, energy consumption will be high compared to other seasons in winter. On the other side, energy consumption will be less in summer due to more daylight and hot weather. Likewise, in the case of the week, energy consumption will be less on weekdays compared to the weekend. In our framework, we consider week, where consumption pattern varies for each day of the week. Therefore every week, we can find a similar pattern on weekends and others on weekdays. In this level, we find the euclidean distance between the data points in a record.

As we mentioned in the previous level, we calculate the minimum and maximum Euclidean distance for each weekday and store it in a database. Here we calculate the euclidean distance for 12 data points together and compare it with the calculated value. If the data record is valid, then the validated data is added to training data. If the data record is valid in all three layers, then it is forwarded to the next steps for actual processing.

6. RESULTS

In this section, we discuss the accuracy of results obtained in each layer. For this, Ausgrid[1] dataset is used as the data source. In this dataset, data collected from 300 solar customers on a domestic tariff for the period starting from 1 July 2010 to 30 June 2013. In this dataset, each record consists of following fields,

- *Customer ID*: contains customer data and its value ranges from 1 to 300

- *Postcode*: store the location of the customer. It is a four digit code. For eg:2076

- *Generator Capacity*: records the solar panel capacity of each customer

- *Consumption Category*: it is a two-letter code like *GG* for energy generation and *GC* for consumption. This code is used to show whether meter value is consumption or generation.

- *Date*: is in DDMMMYYYY format.

- *0:30,1:00 ....,00:00*: fields from *0:30* to *0:00* contains energy consumed or generated in every 30 minutes. There are 48 fields in total for storing the meter value.

- *Row Quality*: it shows whether the data is actual value or an estimate.

From this dataset, data streams are generated with each record contains customer id, time at which stream is generated, 12 meter readings, and corresponding measured time and status of each meter value. The time interval between this 12 meter reading is 30 minutes, and status denotes whether the measured value is valid or not.

Our framework is used for validation in real-time. In table 1 show the Root Mean Squared Error (RMSE) and Mean Squared Error (MSE) of different models for different models considered for level 1 validation (validation per customer). In table 1, the Prophet model provides better results compared to LSTM and ARIMA model. In our framework, we used the Prophet model. For validation, we need a data range. The Prophet model's main advantage is that it predicts the upper bound and lower bound of the data to be predicted. Also, it predicts the lower and upper bound for daily consumption and weekly consumption that can be used for further validation of data.

Models	RMSE	MSE
Prophet	0.32	0.10
LSTM	0.33	0.11
ARIMA	0.34	0.11

Table 1: Comparison of RMSE and MSE of different models for level 1 validation

For level 1 validation, training data has to be updated in regular intervals of time. Also, data validation is performed in real-time. Therefore our system should provide better performance. For level 1, Prophet takes only less than 30 seconds for training and prediction. At the same time, the ARIMA model takes more than 30 minutes, and LSTM takes more than 15 minutes for training and prediction of the same data. Since we receive data every 3 hours from a customer,it is possible to perform the prediction and store these values for the validation of next set of values. As a result, latency can be reduced to a certain extend. In level 1, we validate the data with respect to the consumer's consumption pattern where in level 2 and level 3 we considers the external factors effecting consumption pattern such as weather, time, season, etc. For level 2, and level 3 validation, euclidean distance is measured while DTW takes more time to find the shortest path between the time series data points.

7. CONCLUSION

In the age of big data, high-quality data is a prerequisite to perform analysis. Otherwise, analysis of low-quality data results in serious decision-making mistakes [3]. Therefore, data quality is a critical factor for efficient analysis. Data quality issues like missing data, inconsistent data and redundant data are observable and identifiable. At the same time, some of the data quality issues like the correctness of data are not easily identifiable. In this paper, we propose a framework to validate data's correctness in the energy systems domain. In our system, data validation is performed in three levels: validation per customer, validation per location, and validation per context by considering the factors such as customer behavior, weather conditions, etc.

In our framework, data validation is performed using the time series forecasting models. Our framework is a hybrid model of Prophet and vector autoregression, where Prophet is univariate, and vector autoregression is a multivariate time forecasting model. Currently, data validation is performed on the energy consumption domain. As a next step, the energy generation domain will also be integrated into this framework. Also, this framework will be extended for other domains.

8. ACKNOWLEDGMENTS

This work is funded by the BMBF (Bundesministeriums für Bildung and Forschung) under grant 01IS18074A.

9. REFERENCES

[1] Ausgrid. Ausgrid - solar home electricity data. *URL https://www.ausgrid.com.au/Industry/Our-Research/Data-to-share/Solar-home-electricity-data*, 2021.

[2] H. Cai and K. K. Venkatasubramanian. Detecting data manipulation attacks on physiological sensor measurements in wearable medical systems. *EURASIP Journal on Information Security*, 2018(1):1–21, 2018.

[3] L. Cai and Y. Zhu. The challenges of data quality and data quality assessment in the big data era. *Data science journal*, 14, 2015.

[4] F. Cannarile, P. Baraldi, P. Colombo, and E. Zio. A novel method for sensor data validation based on the analysis of wavelet transform scalograms. *International Journal of Prognostics and Health Management*, 9(1):002, 2018.

[5] C. Cassisi, P. Montalto, M. Aliotta, A. Cannata, and A. Pulvirenti. Similarity measures and dimensionality reduction techniques for time series data mining. *Advances in data mining knowledge discovery and applications'(InTech, Rijeka, Croatia, 2012,*, pages 71–96, 2012.

[6] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah. A review on time series forecasting techniques for building energy consumption. *Renewable and Sustainable Energy Reviews*, 74:902–924, 2017.

[7] E. V. Foundation. Methodology for data validation 1.1 revised edition 2018. https://ec.europa.eu/eurostat/cros/system/files/ess_handbook_methodology_for_data_validation_v1.1_-_rev2018_0.pdf.

[8] A. Ghasempour. Optimized advanced metering infrastructure architecture of smart grid based on total cost, energy, and delay. In *2016 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–6, 2016.

[9] A. Ghasempour. Internet of things in smart grid: Architecture, applications, services, key technologies, and challenges. *Inventions*, 4(1):22, 2019.

[10] R. J. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[11] A. Karkouch, H. Mousannif, H. Al Moatassime, and T. Noel. Data quality in internet of things: A state-of-the-art survey. *Journal of Network and Computer Applications*, 73:57–81, 2016.

[12] A. Kianimajd, M. Ruano, P. Carvalho, J. Henriques, T. Rocha, S. Paredes, and A. Ruano. Comparison of different methods of measuring similarity in physiologic time series. *IFAC-PapersOnLine*, 50(1):11005–11010, 2017. 20th IFAC World Congress.

[13] M. Müller. Dynamic time warping. *Information retrieval for music and motion*, pages 69–84, 2007.

[14] I. M. Pires, N. M. Garcia, N. Pombo, F. Flórez-Revuelta, and N. D. Rodríguez. Validation techniques for sensor data in mobile health applications. *Journal of Sensors*, 2016, 2016.

[15] S. Pizonka, T. Kehrer, and M. Weidlich. Domain model-based data stream validation for internet of things applications. In *MODELS Workshops*, pages 503–508, 2018.

[16] F. Research. Prophet: forecasting at scale. *URL https://research.fb.com/prophet-forecasting-at-scale/*, 2021.

[17] F. Rodríguez, F. Martín, L. Fontán, and A. Galarza. Very short-term load forecaster based on a neural network technique for smart grid control. *Energies*, 13(19):5210, 2020.

[18] F. Sartori, R. Melen, and F. Giudici. Iot data validation using spatial and temporal correlations. In *MTSR*, 2019.