ORIGINAL RESEARCH **Open access**

Explainable Drift Detection for Energy Consumption Forecasting Using Energy Embeddings

Victor Moreno[1*], Daniela Ruiz[2], Andres Salazar[1]

Abstract

Precisely estimating energy use is vital for lowering running expenses, reaching net-zero carbon targets, and building sustainable future urban environments. Although Artificial Intelligence (AI) models are commonly applied in Building Science to analyze and predict energy trends, relying entirely on these methods to estimate demand solves only part of the problem. Shifts in energy consumption patterns can cause these AI tools to become inaccurate, ultimately resulting in flawed decisions and actions. Although methods for detecting such shifts exist, current research lacks a dependable system that can both spot these deviations and explain them with practical insights. Therefore, this study introduces an AI architecture designed for energy forecasting that features explainable drift detection to tackle these issues. This new system utilizes energy embeddings, a highly efficient dimensional model housed within a data warehouse, and a scalable cloud setup to detect shifts effectively while providing clear explanations. The system was tested using actual data from a multi-campus, mixed-use higher education institution located in Victoria, Australia. Experimental outcomes demonstrate the system's proficiency in spotting concept drift, updating forecasts, and explaining these variations using energy embeddings.

Keywords Building energy consumption forecasting, Explainable drift detection, Energy embedding, Dimensional modeling

*Correspondence:
Victor Moreno
victor.moreno@ub.edu
[1] Department of AI for Electrical Power Systems, Faculty of Engineering, University of Barcelona, Barcelona, Spain
[2] Department of Energy Informatics, School of Engineering, Autonomous University of Madrid, Madrid, Spain

Introduction

The growing implementation of smart grid systems in commercial spaces and institutions has facilitated the use of Artificial Intelligence for eco-friendly power management [1, 2]. In this area, a main objective is forecasting energy usage trends across different facilities and meters. Forecasting power needs is vital, as it enables preemptive resource allocation to reduce overall usage. By projecting upcoming energy requirements, businesses can spot peak usage times early and take specific actions to lessen the impact. Such steps may involve fine-tuning HVAC operations, modifying lighting timers, and introducing energy-saving tools in high-usage zones [3, 4]. Projecting energy needs offers many advantages by precisely estimating upcoming usage trends [5]. This knowledge empowers companies to make smarter choices and act preemptively, resulting in substantial financial savings and smoother operations. Decreasing expenses is a primary advantage. When armed with accurate energy projections, companies can gauge their future requirements more precisely, giving them leverage to secure better rates from utility providers and arrange contracts that match their actual usage patterns.

This is an open access article distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0), which permits non-commercial use, distribution, reproduction, and adaptation in any medium or format, provided that appropriate credit is given to the original author(s) and source, a link to the license is provided, and any changes made are indicated. Adapted material must be distributed under the same license.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Even with the extensive creation and application of AI forecasting models in smart city [6, 7] and smart grid [8, 9] environments, these systems frequently operate under the assumption that the training data's distribution will perfectly mirror the live data encountered in real-world use. For instance, predictive systems relying on meteorological information might lose their precision due to abrupt weather shifts or unprecedented disruptions like the COVID-19 outbreak. Such occurrences triggered unexpected facility shutdowns, altering the energy usage patterns away from the data the algorithms initially learned. Practical disruptions within energy networks, including the sudden breakdown of electrical equipment, can trigger unanticipated surges in power demand, further altering the data's distribution. These shifts can cause major departures from the original energy projections, forcing manual updates such as feeding the algorithms new data if the new trends continue. Consequently, it is vital to spot any departures from standard energy use estimates [10]. Once these deviations are spotted, understanding their underlying triggers is critical for improving decision-making, underscoring the need for an explainable method.

Existing studies offer various explanations for the concepts of explainability and interpretability [3]. It is helpful to highlight two primary distinctions. Interpretability is defined as the extent to which a person can grasp a system's output relying entirely on its structural design, needing no outside context. Here, the model's structure is key: outputs are clear based on the internal links between variables and data points. Conversely, explainability is the degree to which an individual can grasp the elements driving a model's output by relying on extra, external context. This is generally accomplished by examining a completely trained model after the fact, rather than relying on its built-in architecture. Thus, when developing an interpretable method for finding data shifts, researchers can choose one of two paths: crafting an inherently interpretable detection model, or delivering an explainable method that supplies surrounding context. By connecting a detected shift to historical occurrences, we can provide contextual insight into the reasons behind the deviation. Therefore, this article introduces the creation and structuring of an AI architecture tailored for transparent drift detection in energy forecasting, utilizing past data to supply necessary context. We then test this architecture in an actual, multi-campus, mixed-use higher education environment. The primary contributions of this study include:

1. The implementation of change point detection inside energy prediction algorithms, allowing for the accurate pinpointing of major alterations or deviations.

2. An innovative method for explainable drift detection that relies on energy embeddings derived from time-series power usage data.

3. A dimensional structure designed to record information tied to drift events, making it easier to analyze the detected deviations thoroughly.

4. An AI architecture that incorporates interpretable drift detection, tested in an actual multi-campus power forecasting environment at La Trobe University.

Earlier studies have underscored the importance of understanding concept drift and identifying it [11], especially regarding time-series data where spotting change points—the exact instances when shifts happen—is vital. Yet, this issue remains underexplored in energy usage prediction, a field heavily impacted by variances and deviations. To close this gap, our study presents a method utilizing a change point detection algorithm tailored to spot energy shifts. A primary highlight of our study is offering possible explanations for these discovered shifts by examining historical instances of similar trends. We accomplish this by leveraging energy embeddings pulled from past power usage logs. We propose a dimensional model that thoroughly captures drift-centric information, providing deeper insights into identified shifts through these energy embeddings. By merging these features, our proposed system delivers a comprehensive strategy for transparent drift identification in energy forecasts, ultimately aiding smarter decision-making in power management.

Related Work

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

The precision of energy use predictions has advanced considerably thanks to the deployment of smart meters, which supply continuous 15-minute interval usage data, alongside the evolution of advanced forecasting methods. Early approaches relied on statistical techniques, utilizing regression models applied to historical data and verified by contrasting predicted outcomes against actual values [5, 12]. For instance, Liu *et al*. [13] applied an extensive Vector Autoregression (VAR) algorithm to predict crucial meteorological factors across 61 US cities by leveraging time lag variables. Furthermore, López *et al*. [14] introduced a Seasonal Autoregressive Integrated Moving Average (ARIMA) system for brief load predictions to optimize electrical distribution operations. Even though these statistical methods are straightforward and require minimal computational power, they proved inadequate for handling massive datasets and identifying complex nonlinear patterns. Consequently, Support Vector Machines (SVMs), predominantly utilized for classification, were introduced to the energy sector. Specifically, Wang *et al*. [15] leveraged SVMs to infer users' socio-demographic profiles based on their energy consumption. Support Vector Regression (SVR) techniques, which build upon SVMs, were similarly integrated into energy prediction models. Meng *et al*. [16] demonstrated this through a case study forecasting China's power usage with SVR algorithms. Random Forest algorithms are also frequently applied in this field. Li *et al*. [17] utilized a Random Forest approach to enhance the precision of daily corporate power usage forecasts. While these techniques are highly capable, particularly for point predictions, they struggle with extended data sequences. To resolve these shortcomings, deep learning strategies have been adopted. Ahmad and Chen [18] presented a multi-tiered intelligent energy prediction system utilizing two distinct deep learning architectures. Similarly, Ishaq *et al*. [19] developed an ensemble deep learning framework combining convolutional and bi-directional long-short-term memory networks to forecast energy requirements and usage.

While diverse forecasting techniques show great promise in projecting power usage trends, they face specific hurdles. A major obstacle for these systems is concept drift, meaning the target variable's statistical characteristics change over time [20]. Tackling concept drift is essential for preserving the dependability and precision of energy predictions. Such drifts can happen because of shifts in unobservable, unmeasurable underlying factors [21]. As a result, machine learning algorithms trained on earlier data can output incorrect forecasts when faced with these concept drifts, rendering them unsuitable for practical use. Overcoming this requires preemptive actions, leading to an increased focus on drift detection methodologies. The Drift Detection Method (DDM) [22], one of the initial algorithms in this space, was built to manage the online error frequency of a system. In this method, training samples are introduced in sequence, and the model evaluates new inputs based on its current state. From a statistical standpoint, the error rate drops as long as the data distribution is stable, but rises when the distribution shifts. Various adaptations of this approach have emerged, such as the Early Drift Detection Method (EDDM) [23], Heffding's inequality-based Drift Detection Method (HDDM) [24], and the Fuzzy Windowing Drift Detection Method (FW-DDM) [25], which utilizes fuzzy time intervals to categorize data points into distinct concepts accurately. Another approach focusing on error rates is Adaptive WINdowing (ADWIN) [26], which identifies error rate fluctuations by evaluating recent data windows against the total data window.

Beyond error-centric approaches, drift detection models focused on data distribution utilize distance metrics to measure differences between baseline historical data and incoming data. For example, Statistical Change Detection for multidimensional data (SCD) [27] employs density testing to check whether new data points originate from the same underlying distribution as the original dataset. Researchers have also introduced Principal Component Analysis-based change detection structures for multidimensional data flows [28], along with versions utilizing symbolic architectures [29]. A further classification of drift detection models relies on multiple hypothesis testing. Two frequent techniques here include parallel multiple hypothesis testing [30] and hierarchical hypothesis testing [31]. Even with the heavy emphasis on algorithms to spot drift, minimal effort has gone toward investigating and clarifying the actual drifts once found. Webb *et al*. [32] emphasize the necessity of quantitative profiling of drifts in marginal distributions for proper drift analysis. Hinder *et al*. [33] clarify concept drift using contrasting examples and distinct spatial feature shifts depicted by typical cases grounded in counterfactual explanations.

A primary shortcoming of the aforementioned methods is their failure to account for the temporal flow intrinsic to drift data, as well as their tendency to ignore latent data characteristics. Consequently, it is vital to examine research on vector embeddings for time-series data, as this provides a viable path to address these issues. Studies have shown that

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

embedding strategies successfully encapsulate both hidden traits and the time-based dynamics of sequence data. In an early example, Hayashi *et al*. [34] embedded time-series data into vector spaces utilizing Dynamic Time Warping (DTW) distances. Other research has highlighted transforming time sequences to generate vector formats via Fourier transforms [35] and utilizing Kernels to achieve higher-dimensional representations [36]. Although these earlier techniques depend on established transformation rules, contemporary strategies suggest unsupervised approaches to deduce vector embeddings from time sequences [37]. Franceschi *et al*. [38] created an encoder utilizing causal dilated convolutions to extract generalized time-series embeddings. Yue *et al*. [39] designed a unique system for capturing time-series representations across different semantic tiers through hierarchical contrastive learning based on augmented contextual perspectives.

The detection and evaluation of concept drift have been successfully implemented across numerous fields and industries. Žliobaitė [40] provides an extensive analysis of various real-time monitoring and control use cases [41, 42]. Within the energy sector, managing concept drift is especially vital, particularly for systems utilizing weather and chronological inputs where distribution shifts are inevitable. The literature details several efforts to construct architectures for predicting energy use and associated tasks. For example, Gökçe and Gökçe [43] showcase a model-centric, comprehensive system framework that unifies IT operations during the planning, building, and active phases of structures to streamline building management. Hong *et al*. [44] introduce a microgrid energy management architecture designed around the Energy Internet, tested in China. Plageras *et al*. [45] present a cloud-based structure for consolidating remote sensor deployments, enabling centralized data governance. Despite these advancements, the existing literature lacks a fully integrated framework combining concept drift identification with practical application in energy use prediction, creating an opportunity to introduce our architecture and its real-world implementation.

The Proposed Artificial Intelligence Framework for Explainable Drift Detection

The proposed AI architecture for transparent drift identification comprises five distinct modules, as illustrated in **Figure 1**. Incoming data from smart meters, weather updates, and event logs are gathered and stored in a central data warehouse. This data warehouse initially aggregates the smart meter readings, temporal event records, and weather details, arranging them into a dimensional structure via an ETL procedure, as detailed in Moraliyage *et al*. [46]. The forecasting component accesses this data warehouse to produce energy consumption predictions. In contrast, the energy embedding component simultaneously creates embeddings derived from the usage data and logs them into a vector database. The drift detection unit reviews the energy predictions and flags any suspicious drifting behavior. It then records the drift metadata back into the data warehouse, which activates the drift analysis component. This final module leverages both the metadata stored in the warehouse and the energy embeddings from the vector database to enable explainable drift identification. **Algorithm 1** summarizes the overarching process of the proposed system, running from initial data aggregation to the final drift evaluation. The core elements—forecasting, drift detection, energy embedding, drift analysis, and the data warehouse—are detailed in the following subsections.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Data Collection:
$D_{\text{raw}} \leftarrow \text{Collect}(\text{Data from smart meters, logs, weather})$
Data Warehousing:
$D_{\text{warehouse}} \leftarrow \text{Store}(D_{\text{raw}})$
Energy Consumption Forecasting:
$D_{\text{model}} \leftarrow \text{Training}(D_{\text{warehouse}})$
$D_{\text{forecast}} \leftarrow \text{Inference}(D_{\text{model}})$
Generating Energy Embeddings:
$D_{\text{actual}} \leftarrow \text{RetrieveConsumptionData}(D_{\text{warehouse}})$
$\mathcal{E} \leftarrow \text{GenerateEnergyEmbedding}(D_{\text{actual}})$
$\text{Store}(\mathcal{E}, \text{vector database})$
Drift Detection:
$C \leftarrow \text{Compare}(D_{\text{forecast}}, D_{\text{actual}})$
$\mathcal{M}_{\text{drift}} \leftarrow \text{DetectDrift}(C)$
Drift Analysis:
$\mathcal{E}_{\text{similar}} \leftarrow \text{RetrieveSimilar}(\mathcal{E}, \text{vector database})$
$\mathcal{M}_{\text{meta}} \leftarrow \text{RetrieveMetaData}(\mathcal{E}_{\text{similar}})$
$\text{Analyze}(\mathcal{M}_{\text{drift}}, \mathcal{E}_{\text{similar}}, \mathcal{M}_{\text{meta}})$

Algorithm 1. Explainable drift detection framework

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

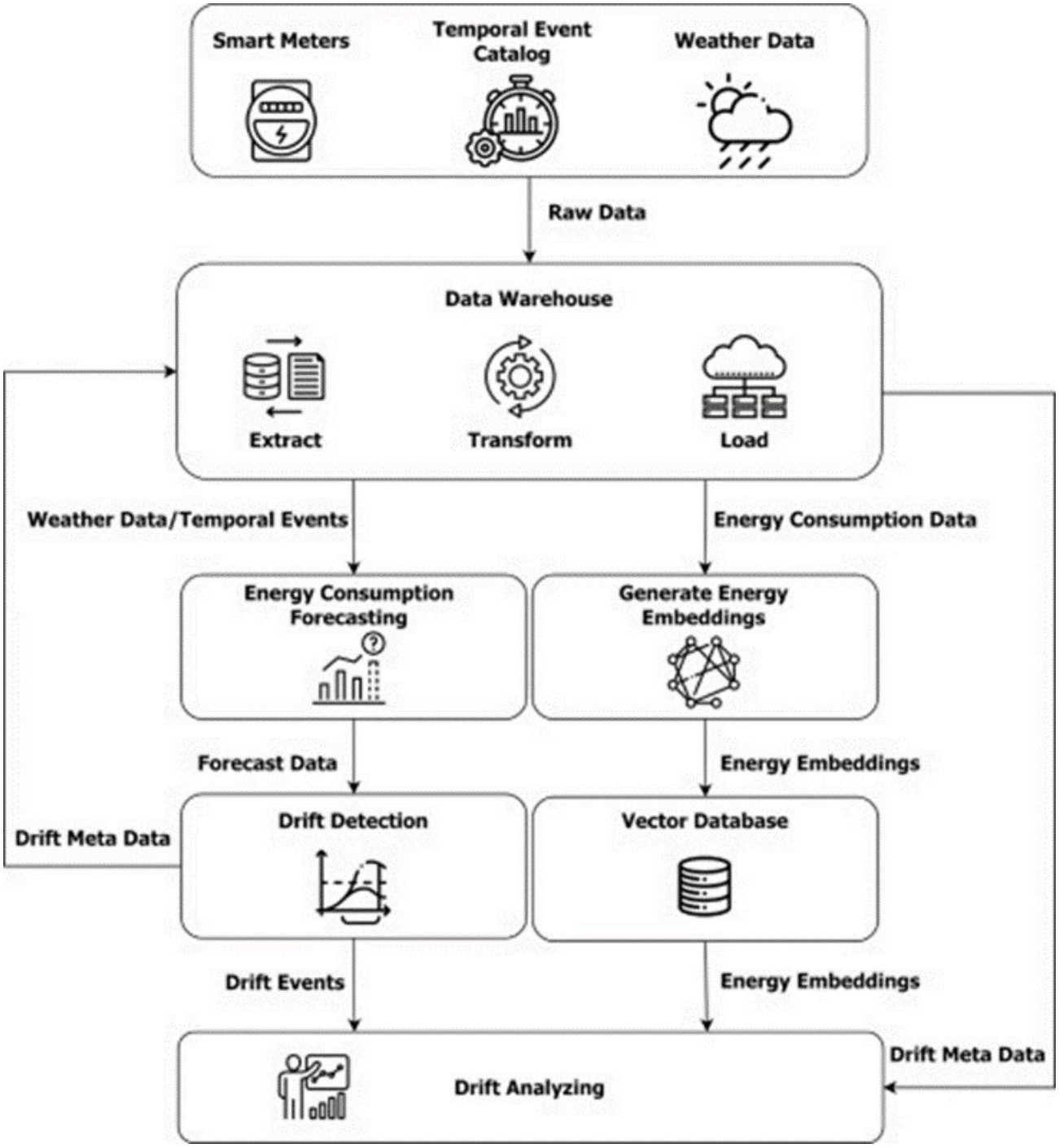


Figure 1. Modular composition of the proposed Artificial Intelligence framework for explainable drift detection in energy consumption forecasting.

Data warehouse

The data warehouse serves as the initial repository for power usage data pulled from smart meters, alongside event logs and weather data. When entering the drift detection stage, all information tied to the spotted drifts is routed back into the data warehouse. This continuous process builds up knowledge, consistently improving drift analysis. The accumulation of more information over time yields a robust analysis. In addition to the metadata kept in the vector database, this data is structured across multiple dimensions utilizing a dimensional model to ensure optimized retrieval. Whenever a drift is

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

spotted daily, its corresponding traits across various dimensions are logged in this format. The suggested dimensional framework, illustrated in **Figure 2**, features a central fact table named FactDrift connected to five distinct dimensions: DimDate, DimEvent, DimEnergy, DimClimate, and DimBuilding.

· DimDate: Contains calendar-specific details regarding when the drift took place, noting if the event coincided with a holiday, an academic semester, or an examination period (applicable in a university environment). This dimension allows for querying and segmenting drift data across different time blocks.

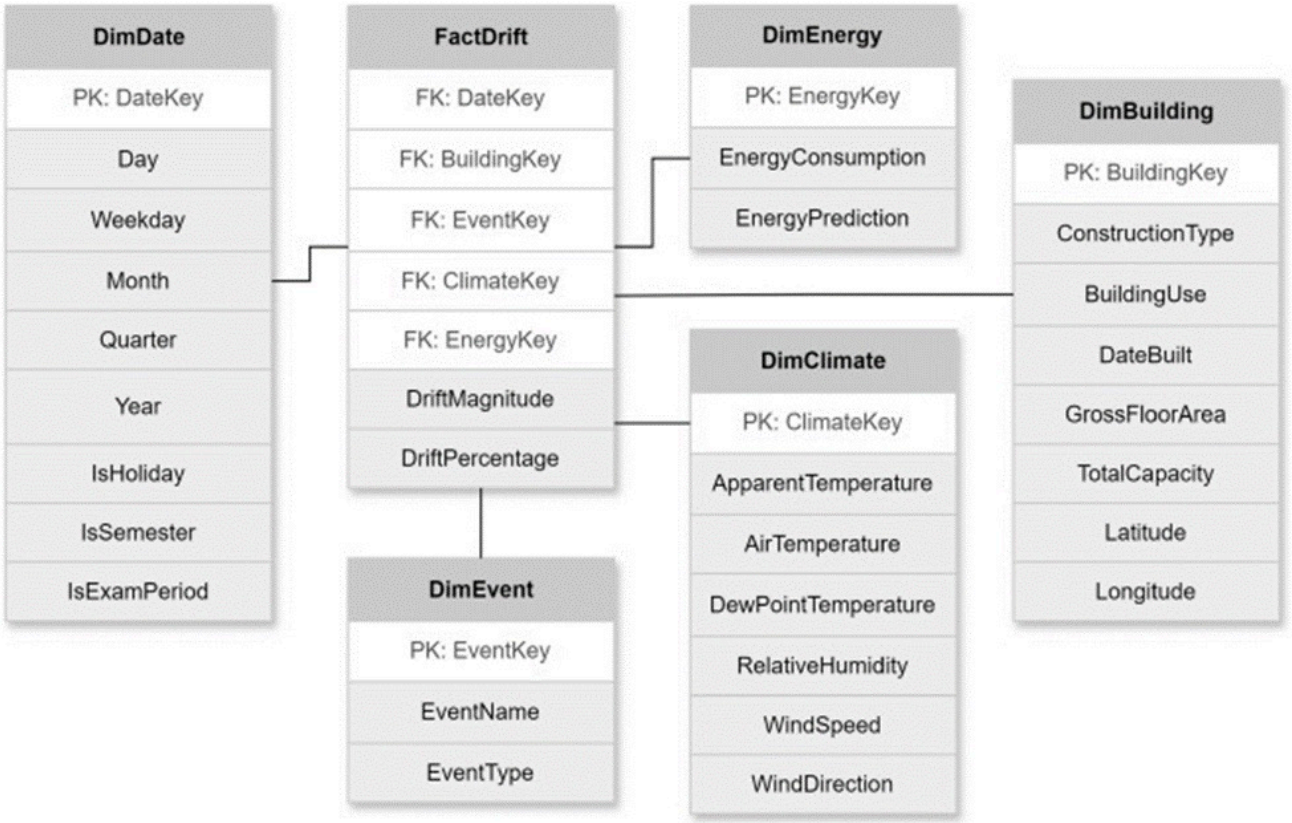
· DimEvent: Records explicit interventions that happened on the day the drift occurred, including actions like LED bulb swaps, HVAC adjustments, or building closures. This is vital for linking observable drifts to manual interventions within the system.

· DimEnergy: Houses the power consumption figures alongside the predictive outputs from the forecasting algorithms, linking directly to the discrepancies spotted in the drifts.

· DimClimate: Logs the daily weather data to pinpoint if adverse or unusual weather patterns might have triggered the recorded drifts.

· DimBuilding: Compiles various attributes of the specific facility where the drift took place, covering metrics like construction style, intended use, age, floor space, location, and capacity. These details enable a comprehensive review of drifts in relation to different building attributes.

The FactDrift table retains the foreign keys from all dimensions, along with the drift magnitude and percentage facts. These quantitative metrics help gauge the impact of the drifts and support evaluation across different thresholds.



© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Figure 2. Dimensional model for storing drift data in the proposed framework for explainable drift detection. (Primary Key (PK): The identifier that uniquely identifies a record within a dimension or fact; Foreign Key (FK): The identifier that establishes a relationship between a dimension and a fact.).

The forecasting module

Constructed using gradient-boosted decision trees, the forecasting module capitalizes on their rapid, efficient training capabilities and their inherent capacity to evaluate feature importance within a trained model autonomously. Feature importance assigns a metric reflecting the utility of each variable during the tree-building process; frequently utilized attributes that drive critical splits receive a higher relative score.

This evaluation is performed explicitly across every dataset attribute, facilitating direct comparison and ranking. Within an individual decision tree, importance is determined by measuring the enhancement in the metric of performance—specifically Mean Squared Error (MSE) here—yielded by an attribute's split, factored by the volume of samples handled by that node.

These values are subsequently averaged across the full tree ensemble to deliver an all-encompassing assessment of each variable's role in the overall predictive accuracy. The total importance metric for feature j across the entire gradient-boosted tree ensemble can be written as:

$$I_j = \frac{1}{T}\sum_{t=1}^{T}\sum_{s\in S_{j,t}} \Delta I_{s,t} \quad (1)$$

Where $S_{j,t}$ represents the collection of nodes within the tth tree where feature j determines a split, and $\Delta I_{s,t}$ signifies the performance metric enhancement gained from splitting node s inside tree t.

Because power consumption is tightly bound to meteorological factors, integrating weather inputs like temperature and ambient humidity is crucial for generating precise forecasts. Furthermore, operational schedules (such as building opening and closing hours) significantly dictate energy consumption trends. As a result, our framework's models explicitly integrate these variables during estimation.

The predictive models deployed in this framework rely on XGBoost regressors. These models incrementally construct their objective functions via loss function minimization [47]. This objective combines empirical training loss with a regularization penalty. The training loss component $\sum_i l\,(\hat{y}_i, y_i)$ quantifies the deviation between estimated $\hat{y}_i$ and true y_i targets to guarantee a precise fit. Meanwhile, the regularization element $\sum_k \Omega\,(f_k)$—defined by $\Omega\,(f) = \gamma T + \frac{1}{2}\lambda \sum w_k^2$ —restricts model complexity to mitigate overfitting. Within this formulation, γT penalizes the total count of terminal leaves T, whereas the $\frac{1}{2}\lambda \sum_k w_k^2$ term penalizes excessive leaf weights w_k via L2 regularization. Such a design successfully balances fitting precision with structural simplicity. The aggregate objective loss function $L\,(\phi)$ is formulated as:

$$L(\phi) = \sum_i l(\hat{y}_i, y_i) + \gamma T + \frac{1}{2}\lambda \sum_k w_k^2 \quad (2)$$

Drift detection

The main objective of drift detection is identifying unpredicted shifts in variables governing consumption behaviors that cause departures from established norms. Because these anomalies require model recalibration, a preemptive identification strategy is vital. Consequently, an intelligent drift detection mechanism to spot these baseline shifts is indispensable.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

To implement this, actual consumption figures are paired with baseline energy forecasts. The variations between these real values and expectations are structured as a sequential time series and supplied to the Pruned Exact Linear Time (PELT) change point detection algorithm [48] to isolate the exact moments of divergence.

Change point detection centers on discovering specific positions in a dataset where fundamental statistical traits shift. Let us define a sequential time series $x_{1:n}$ = (x_1, x_2, …, x_n) containing m distinct change points expressed as integers bounded by 0 and n - 1, designated as $\tau_{1:m}$ = (τ_1, τ_2, …, τ_m), where \tau_1 = 0 and τ_m = n- 1. Based on these boundaries, the dataset is partitioned into m + 1 distinct intervals, where the i-th interval encompasses the corresponding segment elements $x_{\tau_{i-1}+1:\tau_i}$.

Identifying the ideal configuration of change points requires minimizing a designated cost function, such as Huber loss, quadratic error loss, or negative log-likelihood. Consequently, the optimization problem can be mathematically framed as:

$$\underset{0\le\tau_1<\cdots<\tau_m\le n-1}{\arg\min}\left(\sum_{i=1}^{m+1} C(x_{\tau_{i-1}+1:\tau_i})+\beta\right) \quad (3)$$

In this expression, *C* (.) denotes the selected cost metric, and \beta represents the penalty cost assigned for each added change point.

The goal is to minimize total cost, striking an equilibrium between precise sequence partitioning and the penalty constraints. In the absence of a penalty, the routine would generate excessive change points—particularly within noisy sets—causing severe overfitting. Implementing a per-change-point penalty discourages superfluous partitions, helping distinguish genuine transitions from stochastic variations or background noise. Thus, the penalty coefficient \beta regulates the trade-off between model fit quality and parameter complexity.

An online strategy is used when deploying the PELT algorithm for energy data drift discovery. Daily at closing, usage records covering the preceding m days are gathered and run through the algorithm. This establishes a moving window spanning the most recent m days of data, within which change-point detection assessments are executed. Any structural changes identified within this trailing m-day frame (where 0 \le m \le n) are pulled, flagging a prospective drift in the underlying energy baseline. This prompts an alert for administrative action, either routing to human review or transitioning directly into the drift analysis workflow. The programmatic logic for this variation-based drift tracking is detailed in **Algorithm 2.**

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Input: Size of consumption data period n days
Input: Size of analysis window m days
Input: Energy consumption data $c_{1:n} = (c_1, c_2, ..., c_n)$ over n days
Input: Energy forecast data $f_{1:n} = (f_1, f_2, ..., f_n)$ over n days
Input: Penalty β against over fitting
Input: Cost function C
Output: Identified change points $\tau_{1:m}$
Initialize: $m \leftarrow 0$, change points $\tau \leftarrow \{0\}$, error values $x_{1:n} = (c_1 - f_1, c_2 - f_2, ..., c_n - f_n)$
procedure CHANGE POINT DETECTION
    $\tau_{\text{end}} \leftarrow n - 1$
    $S \leftarrow \{0, n-1\}$ ▷ Set of potential change points
    while not converged **do**
        $[\tau_{\text{new}}, \text{cost}] \leftarrow FindChangePoint(x_{1:n}, S)$
        if cost + β < CurrentCost(S) **then**
            $S \leftarrow S \cup \{\tau_{\text{new}}\}$
        else
            break ▷ No improvement in cost
        end if
    end while
    $\tau_{1:m} \leftarrow \text{Sort}(S)$
    return $\tau_{1:m}$
end procedure
function FINDCHANGEPOINT($x_{1:n}, S$)
    minimize: $\sum_{i=1}^{m+1} C(x_{\tau_{i-1}+1:\tau_i}) + \beta$
    subject to: $0 \leq \tau_1 < \tau_2 < \cdots < \tau_m \leq n-1$
    Calculate the cost for each possible change point
    return change point with minimum cost
end function

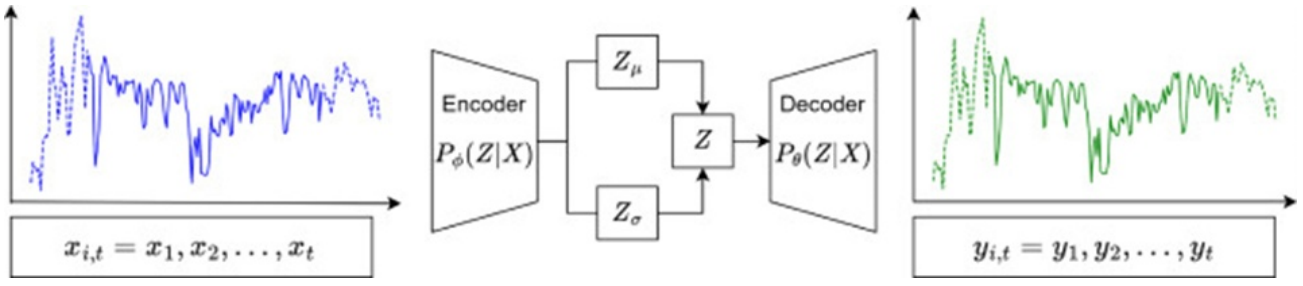
Algorithm 2. Drift detection in energy consumption data

Energy embedding

Energy embeddings are derived via a Variational Autoencoder (VAE) framework that features an encoder-decoder pair, as diagrammed in **Figure 3**. Continuous day-to-day power consumption metrics over a specified duration are treated as a sequential time series. The encoder maps this input through multiple fully connected layers, using nonlinear activations and dropout for regularization. It translates the input observations into a reduced latent space characterized by a mean vector (\mu) and a log-variance vector (logvar).

A reparameterization step is leveraged to extract samples from this latent space while maintaining network differentiability during backpropagation. Following this, the decoder attempts to rebuild the original sequence from the sampled latent coordinate. Given a temporal sequence of duration t, designated as $x_{i,}$ = x_1, x_2, …, x_t, given an encoder *Enc*, a decoder *Dec*, and an output reconstructed sequence $y_{i,t}$ = y_1, y_2, …, y_t, the routine proceeds through the following phases:

- Encoding μ, $logvar$ = Enc($x_{i,t}$), where \mu denotes the mean, and $logvar$ signifies the log-variance.
- Reparameterization: z = reparametrize(μ, $logvar$), where z represents the latent sample vector.
- Decoding: $y_{i,}$ = Dec(z)



© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Figure 3. Energy embeddings model.

Model optimization relies on a dual-component objective function:

· KL divergence loss: Forces the latent distribution to match a standard Gaussian distribution to encourage compact, semantically rich tracking.

· Reconstruction: Penalizes discrepancies between the original inputs and the generated outputs.

Minimizing this combined error guides the network to preserve granular details of the source material. Ultimately, the target energy embeddings are acquired by routing fragments of consumption time-series data through the encoder, capturing the resulting fixed-length latent vectors. These vectors act as dense, highly descriptive signatures of consumption behavior, facilitating robust quantitative comparison via appropriate metric indices.

Vector database

Leveraging the energy embedding system yields high-dimensional vector representations across diverse intervals and facilities. Standard relational and NoSQL storage systems, however, are poorly suited for managing this type of information. Because their native indexing schemes cannot efficiently accommodate multidimensional structures, they suffer from slow lookup speeds and weak similarity search throughput. Horizontal scaling presents another hurdle, with both database varieties experiencing steep performance drops as dataset volumes and dimensionality scale upward.

Additionally, legacy systems lack the specialized query operations needed for workflows like k-nearest neighbors (k-NN) lookups, which are indispensable for high-dimensional spatial data analysis. To circumvent this, a dedicated vector database is implemented to hold these multidimensional embedding values. This deployment guarantees ultra-fast similarity lookups and significantly accelerates matching operations.

The innate horizontal scalability of vector databases is a critical asset for capturing a continuous stream of time-series embeddings. Furthermore, these modern engines support co-storing vectors with their contextual metadata. This capability is exceptionally beneficial for tracking operational events—like HVAC adjustments or facility closures—that might explain anomalies in consumption behaviors during the specific time blocks mapped to those vectors.

Drift analysis

Throughout the drift analysis stage, the core emphasis centers on isolating identified anomalies in energy forecasts to diagnose their potential root causes. The workflow kicks off by isolating a historical usage window that spans the precise time frame where the anomaly occurred. A corresponding embedding vector is then produced for this designated sequence via the trained encoder module. This numerical representation distills the high-dimensional energy usage profile recorded during the drifted window.

Next, this newly minted embedding is evaluated against historical embeddings representing multiple consumption profiles across various buildings and dates stored within the vector repository. Pulling the nearest neighboring vectors out of storage allows the system to identify past behaviors that strongly mirror the anomalous sequence. This comparative search leverages the database's advanced querying framework, optimized to process complex vectors with minimal latency.

After the most comparable records are extracted, their accompanying metadata is investigated. These data logs contain records of notable structural or environmental shifts, such as climate control adjustments, facility occupancy fluctuations, or machinery faults. Inspecting this context makes it possible to precisely identify the probable root factor triggering the observed deviation in the consumption forecast.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Experiments

This section presents the real-world deployment of the introduced AI architecture across the multi-campus environment of a higher education institution. Driven by La Trobe University's "Net Zero Carbon Emissions Program," the initiative targets eliminating the institution's carbon footprint to reach net zero emissions by 2029, while boosting energy efficiency and maximizing asset utilization. The technical backbone facilitating this target is the La Trobe Energy AI Platform (LEAP). Detailed architectural layouts for LEAP can be reviewed in Mills *et al*. [49], and its corresponding datasets have been released publicly for replication and subsequent studies [46, 50].

Experimental setting

The system infrastructure was launched on the Microsoft Azure Cloud environment as illustrated in **Figure 4**, with the complete source code hosted publicly on GitHub (https://github.com/CDAC-lab/Drift-Detection).

Key components of the operational deployment include:

· Service architecture: Every distinct service is executed as an Azure Function. Isolated message queues orchestrate and trigger specific operations, including energy baseline training (forecast model), energy baseline forecasting, drift detection, and energy embedding tasks.

· API integration: Two secure REST API endpoints allow administrators to manually prompt the baseline training and forecasting processes via POST requests. These requests include unique building identifiers and the targeted time windows.

· Efficiency optimization: If a single request contains multiple building identifiers, parallel queue messages are dispatched. This concurrent execution boosts overall efficiency and prevents individual Azure Function instances from becoming overloaded.

Data flows systematically through the following pipeline:

· Data sourcing and storage: Training data is pulled from a centralized data warehouse configured with a dimensional schema. Once trained, models are preserved within an Azure Blob Storage container.

· Automated forecasting: Energy consumption projections are generated on a daily schedule, initiated by a dedicated Azure Function that runs every day at 1 AM. This routine pushes individual messages for each active facility, kicking off the predictive service. The resulting forecasts are saved as files and transferred to a separate blob storage container. Utilizing blob storage decouples the predictive component from other functionalities across the LEAP platform.

· Drift and embedding execution: Following a similar pattern, the drift detection and embedding routines operate daily, prompted by messages routed to their respective queues via the same 1 AM function trigger.

· Drift analysis lookups: The drift analyzer service is called via API endpoints. It processes incoming parameters—specifically the building identifier and the timeframe of the observed variations—and queries the vector database to return the five closest historical time sequences. This output is then passed to the web interface for visual comparison.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press



Figure 4. Energy consumption forecasting framework implemented on Azure Cloud platform.

Results

Energy consumption forecasting models

To capture the granular nuances of energy usage across La Trobe University's varied structural environments, smart meter data recorded at 15-minute intervals was used to train the predictive models. Given the clear operational variations between facilities, standalone models were tailored for each distinct building.

· Model specifications: The framework utilizes XGBoost regression models. Input features combine raw smart meter historical data, weather indicators (temperature and humidity), and a range of temporal indicators (such as year, week, month, public holidays, academic semesters, and examination blocks).

· Training dataset: The training data spans from January 1, 2018, onward, with power consumption serving as the target variable to be estimated from the remaining dataset attributes.

· Testing and metrics: During validation, the trained XGBoost models output projected power usage metrics in kWh. Model accuracy was rigorously measured using a 10-fold cross-validation approach.

The predictive performance across primary building configurations was evaluated using Mean Absolute Percentage Error (MAPE) and Root Mean Squared Error (RMSE), summarized in **Table 1:**

· Teaching building: This category demonstrated the lowest overall errors in both MAPE and RMSE metrics. This strong performance is heavily tied to its highly predictable and regular energy consumption routines.

· Library: This facility recorded the maximum RMSE value. The higher absolute error stems from its massive aggregate power draw, driven by high occupant density and an extensive inventory of electrical systems. Conversely, its relatively low MAPE score confirms the model's high utility, as the errors remain small relative to the massive scale of actual consumption.

· Sports centre: This zone yielded the highest MAPE score, primarily due to volatile and irregular operational timetables that disrupt consumption patterns. Nevertheless, its low RMSE value indicates that the total volume of energy consumed by the facility remains modest.

Table 1. Evaluation results for energy consumption forecasting models of major building types.

Building	Mean absolute percentage error (MAPE)	Root mean square error (RMSE)
Teaching building	7.39	3.60

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Library	9.86	11.66
Sports centre	16.63	2.88
Office building	11.11	5.34
Mixed-use building	11.98	4.54

Drift detection

The drift tracking module analyzes both raw smart meter records and energy forecasting outputs dating from January 1, 2018. As illustrated by the sample scenario in **Figure 5**, the PELT algorithm parses the discrepancies between real-world usage and baseline predictions to pinpoint change points, isolating sustained deviations from standard operations.

In the provided demonstration, the model successfully highlights three critical change points:

1. May 2022
2. July 2022
3. November 2022

The tracking routine accurately isolated these distinct transitions, which exhibited varying levels of divergence from the projected consumption baseline. Crucially, the algorithm successfully filtered out short-lived anomalies (visible as sharp spikes in the orange plot), validating its ability to overlook transient noise and prioritize lasting structural shifts over time.

To validate operational accuracy, performance was benchmarked against known historical events at the university that altered normal power demands (e.g., LED lighting retrofits, full building shutdowns, and HVAC reconfigurations). Usage profiles covering six-month blocks both containing and lacking these operational shifts were processed via PELT to track change-point accuracy. The resulting precision, recall, and F1-score metrics are compiled in **Table 2.**

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

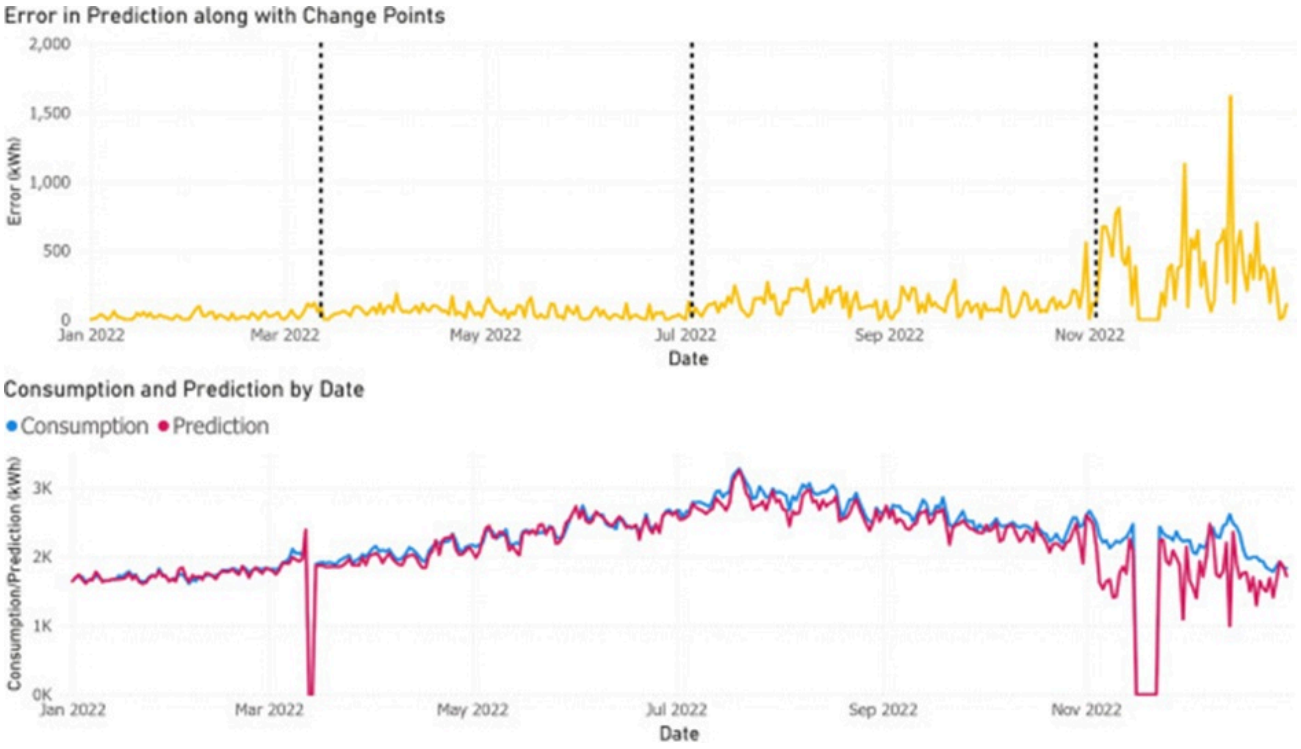


Figure 5. Visualization of drift detection in energy consumption forecasting.

Analyzing performance across different facility environments revealed the following trends:

· Precision vs. recall: Across nearly all facility categories (excluding the sports center), precision fell below recall scores. Because the system is optimized to identify any variation from standard baselines, it inherently flags unlogged operational changes. This results in a higher volume of false positives than false negatives, lowering precision but improving recall. These unlogged discoveries remain highly valuable for surfacing unrecorded system adjustments, enabling proactive interventions to normalize consumption.

· Sports centers: This category yielded the highest overall accuracy. Its major operational variations are highly apparent, and it experiences fewer naturally occurring fluctuations in background consumption.

· Offices and classrooms: These settings demonstrated lower metric scores due to brief, transient shifts—such as rapid HVAC updates—which the algorithm occasionally misclassified despite their non-permanent nature.

The diagnostic capability of the model is further validated in **Figure 6**, which presents individual ROC graphs alongside a consolidated confusion matrix:

· Each facility's ROC curve tracks well above the diagonal random-guess threshold (the dashed line), confirming the framework's effectiveness.

· The strong leftward skew toward the y-axis signifies a robust true positive rate paired with a minimal false positive rate. This layout is driven by the vast majority of true negatives relative to false negatives, as illustrated in the accompanying confusion matrix.

Table 2. Evaluation results for drift detection of major buildings.

Empty cell	F1-Score	Recall	Precision

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Library	0.63	0.86	0.66
Teaching building	0.57	0.80	0.44
Sports centre	0.91	0.83	1.0
Office building	0.58	0.78	0.47
Mixed-use building	0.87	0.99	0.77

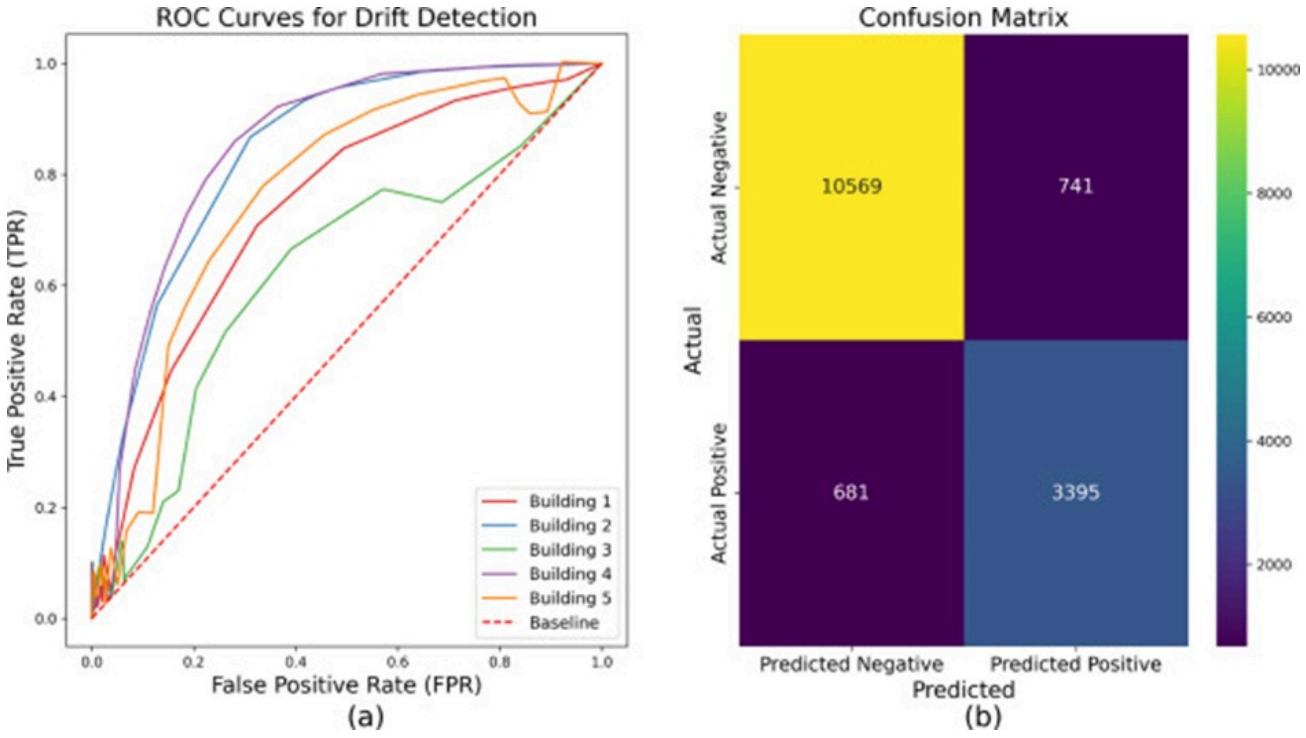


Figure 6. (a) ROC curves in drift detection for five different buildings; (b) Confusion matrix for drift detection across all buildings.

Energy embeddings

To extract high-value insights from power utilization behaviors, smart meter records from La Trobe University spanning from January 1, 2018, were thoroughly examined. The source dataset aggregated streams from 114 active meters, partitioned into distinct time-series blocks of 30 consecutive days. This processing pipeline generated a total of 121,431 training data samples.

Before training, data was normalized via standard scaling, and the optimization process was executed using a batch size of 128 data points. The configuration details of the Variational Autoencoder (VAE) include:

· Encoder design: Built with two linear hidden layers containing 512 and 300 dimensions, respectively, both utilizing ReLU activation functions.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

· Decoder design: Configured to ingest the 300-dimensional latent representation and reconstruct an output vector scaled to a dimension of 30, matching the initial input footprint.

· Embedding generation: The target energy embeddings comprise 300-dimensional vectors collected from the encoder's terminal linear layer.

To facilitate visualization, these multidimensional vectors were projected down into a 2D coordinate space via t-SNE mapping, as shown in **Figure 7**. Within this visualization, data points are labeled according to building categorization. The distinct, visible groupings confirm that the vector embeddings—constructed strictly from raw consumption data—successfully isolate and differentiate the unique energy signatures of various facility types.

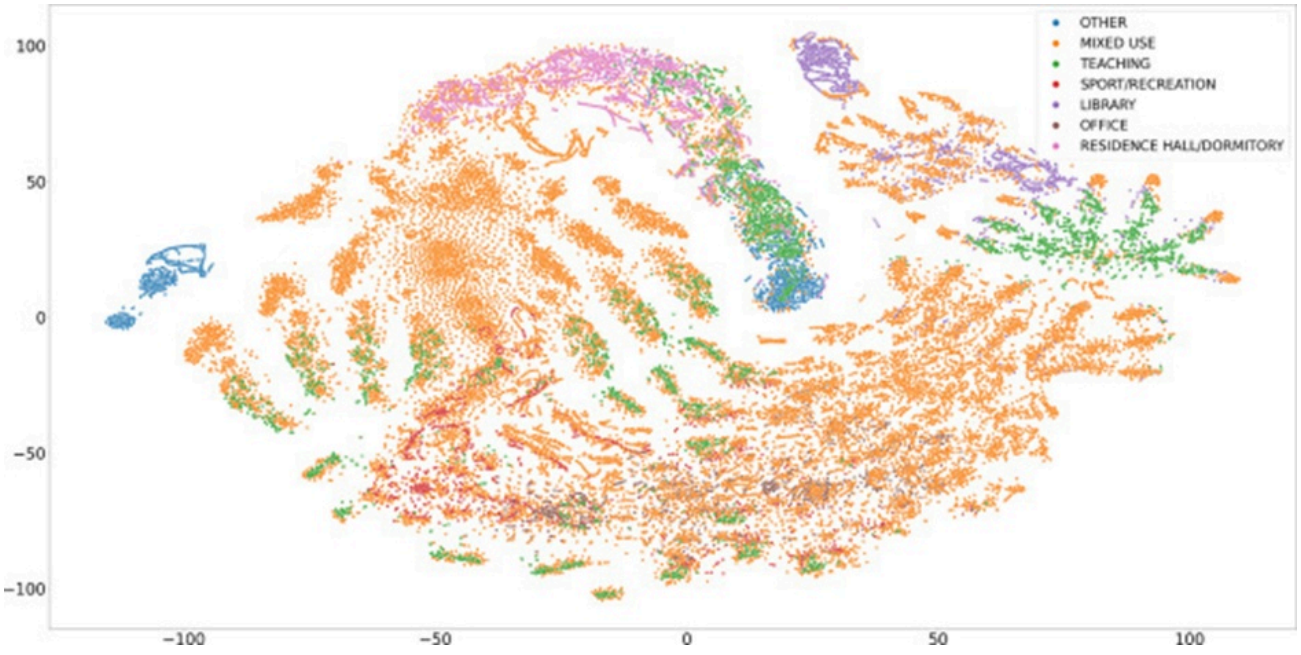


Figure 7. Energy embeddings labeled by type of building.

Drift analysis

The experimental environment for drift analysis evaluated how effectively the generated embeddings could locate matching anomalous patterns. The 300-dimensional energy embeddings were indexed within an instance of the Pinecone vector database, utilizing cosine similarity as the core distance metric.

Each indexed vector was paired with descriptive metadata, recording:

· Calendar date and timestamp

· Unique building identifier and operational category

· Building construction details

· An indicator marking the presence of a drift

· Documented building events during that specific timeframe

When given a new 30-day energy time-series sequence, the system runs the data through the pre-trained encoder to generate an embedding vector. This vector is then queried against the repository to extract the most statistically similar

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

historical intervals, as demonstrated by the case study in **Figure 8.**

In **Figure 8,** the upper graph plots the continuous energy consumption of a chosen facility from June 2020 to February 2021. A specific 30-day block, marked in solid red on this top chart, serves as the query input. The encoder parses this segment to create the query vector used to scan the database. The lower stack of graphs illustrates the resulting matching energy segments from various buildings over the same timeline, highlighting the closest corresponding periods in solid colors. These results are ordered sequentially from top to bottom, descending from the highest-confidence match to the lowest. The filled regions on the blue plots indicate that a comparable power usage pattern occurred during those dates, mimicking the baseline traits of the query sequence.

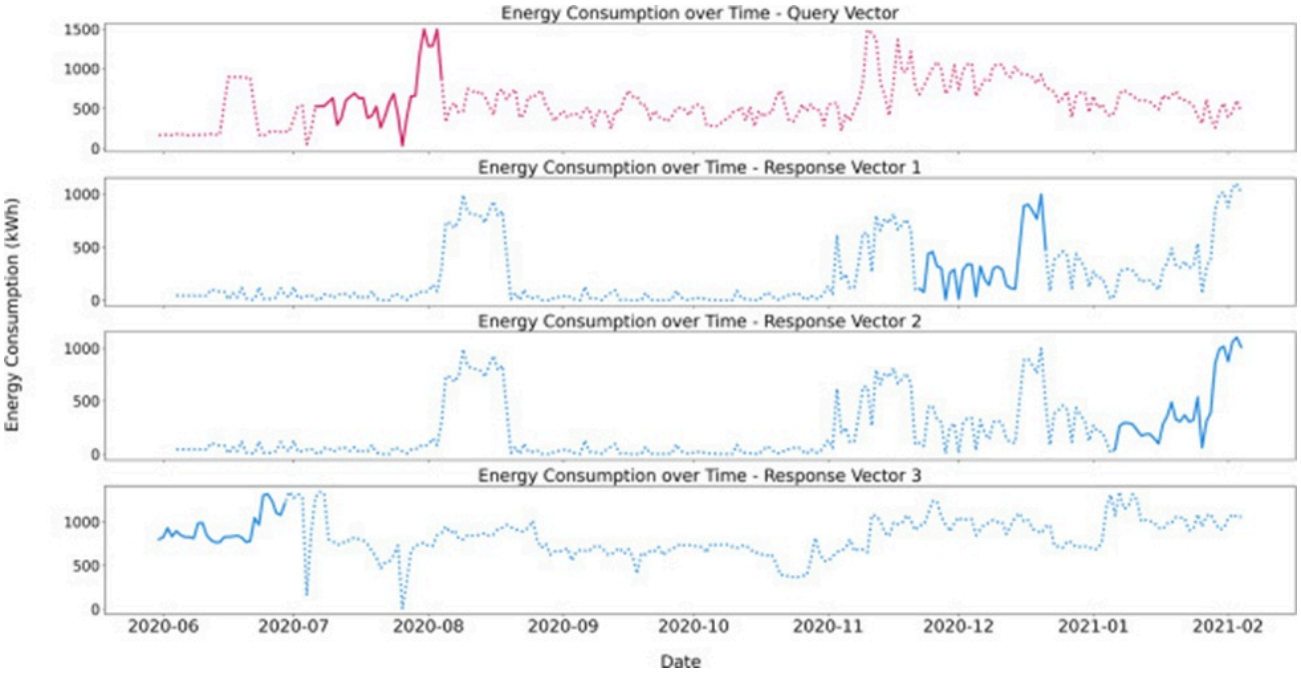


Figure 8. An example of retrieving the closest energy vectors using vector embeddings. (Top) Energy time series sequence used to query. (Bottom) Three closest energy time series sequences returned as the result.

To test the framework's precision in tracking down similar behaviors across different anomaly-inducing incidents, three separate categories of events were chosen: building shutdowns, HVAC adjustments, and LED lighting upgrades. These occurrences, spanning five distinct building types, were audited across the timeline from January 1, 2018, to October 1, 2023; for every specific pairing of event and building style, an operational window covering the incident was captured. The vector engine was then queried to pull the five, ten, and fifteen most similar results. The resulting metrics for precision, recall, and Top-K accuracy are documented in **Table 3.**

Operational takeaways from these trials indicate:

· Building shutdowns: This event type achieved the highest detection accuracy. The sudden, massive drops in energy demand alter the consumption profile so thoroughly that the resulting embeddings are highly distinct and easily matched.

· HVAC adjustments: Minor and short-lived consumption shifts, such as climate control adjustments, yielded lower matching accuracy because their impact on the broader consumption signature is far more subtle.

· Libraries and teaching facilities: Variations within these structures were captured with superior precision. Their highly consistent daily operating schedules and dense occupant traffic create predictable baselines, making sudden

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

deviations highly conspicuous during active events.

· Office buildings: This category proved the most difficult to match accurately. Substantial consumption variance among separate properties within the same asset class makes cross-building pattern matching less reliable.

Despite these inherent challenges, embeddings extracted purely from power usage trends demonstrate strong efficacy in surface-matching comparable operational anomalies. This allows facilities teams to diagnose the hidden triggers behind unexplained power spikes or drops. Ultimately, the embedding space successfully pairs similar usage trends, providing an effective diagnostic tool for uncovering the roots of unmapped consumption drift.

Table 3. Top-K accuracy (%) evaluation for drift matching across different building types and events. B1: Library, B2: Mixed Use Building, B3: Office Building, B4: Accommodation, B5: Teaching Building.

Empty cell	HVAC/LED			Climate			Shutdown		
	K = 15	K = 10	K = 5	K = 15	K = 10	K = 5	K = 15	K = 10	K = 5
B1	37.33	51.00	80.00	88.00	90.00	90.67	66.00	80.00	96.00
B2	34.00	35.00	58.00	46.67	48.00	52.00	65.33	68.00	76.00
B3	74.67	86.00	96.00	31.33	41.00	68.00	48.67	56.00	60.00
B4	30.00	41.00	62.00	82.67	84.00	90.00	63.33	71.00	74.00
B5	58.67	66.00	76.00	53.33	55.00	62.00	66.00	75.00	82.00

Results and Discussion

This research introduces an all-encompassing Artificial Intelligence (AI) architecture designed for identifying drift within power demand projections. To optimize the energy consumption prediction pipeline, the architecture coordinates several core elements: a forecasting module, an embedding module, a drift detection component, a vector storage system, and a drift analysis tool. Driven by the expansion of intelligent power grids, the vast availability of real-time usage metrics has paved the way for leveraging AI methodologies to anticipate power requirements. Many initiatives have sought to harness this wealth of information for predictive modeling, using frameworks ranging from basic machine learning algorithms to sophisticated deep learning architectures [17-19].

Nevertheless, deploying these configurations effectively in operational environments hinges heavily on the consistency of the underlying data profiles used during training and inference. In actual practice, power consumption data distributions are prone to shifts, often triggered by unpredicted meteorological shifts or manual modifications to electrical grids, which demand recurrent model updates. Consequently, we highlight the critical importance of drift identification techniques to smartly flag discrepancies between predicted baselines and real-world observations, a capability central to our architecture. Once these variations are surfaced, guiding operators on corrective actions becomes imperative, a process requiring clear contextualization of the deviations. Our methodology utilizes energy embeddings derived from usage sequences to reveal hidden behavioral configurations, matching them against indexed vectors inside a vector database to isolate comparable historical events. This technique embeds interpretability directly into the drift tracking workflow, empowering operators to diagnose the underlying triggers of noted forecasting deviations.

Although various past frameworks [43, 44] have embedded power demand prediction components tailored for micro-grids, an all-inclusive system that pairs drift tracking with diagnostic explainability has been absent. The architecture introduced here fills this academic void, delivering a critical utility for energy projection infrastructures. A series of empirical tests were executed across every functional block of the system, utilizing real-world usage data gathered from

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

the smart-grid network spanning La Trobe University's multiple campuses. The introduced energy prediction engine delivered encouraging outcomes when mapping demand across distinct facilities, achieving robust precision, especially within structures characterized by stable behavioral profiles.

The drift identification routine operates as a binary classification system, evaluating whether a specific temporal marker indicates an operational drift based on utilization metrics. Evaluation indices—namely precision, recall, and F1-score—highlighted superior tracking efficiency in structures experiencing enduring deviations relative to alternative sites. A qualitative inspection of the generated energy embeddings surfaced clear groupings, proving their diagnostic value in pairing analogous drift states. Finally, the performance of the drift evaluation tool was tested empirically by extracting data from the vector repository for specific target incidents to verify its capacity to retrieve corresponding historical patterns. These trials demonstrated strong accuracy across the board, with elevated performance observed when an incident provoked substantial deviations from typical baseline utilization.

It is important to acknowledge that despite offering an all-inclusive methodology for interpretable drift discovery in power demand forecasting, this framework presents certain constraints. The drift tracking process flags anomalies by detecting departures from past baselines of typical operation. Nonetheless, when assigning explicit root causes to these discovered variations, our system relies entirely on an existing master log of events, which poses a constraint.

Subsequent research should explore integrating diverse external information streams to uncover potential operational triggers outside the current event registry, yielding a more comprehensive understanding of why drift occurs. Modifying the underlying algorithm to identify and adjust for permanent shifts in usage behaviors could bolster its resilience, avoiding the erroneous labeling of structural transitions as temporary drifts. Furthermore, incorporating multidimensional time-series inputs along with supplementary predictive variables might refine the identification of slow, evolving drifts and allow for the proactive anticipation of impending irregular operations.

Conclusion

In this study, we presented an AI-driven architecture for interpretable drift tracking in power utilization forecasting. The system integrates six modules: load forecasting models, drift tracking, energy embeddings, a vector repository, drift diagnostic tools, and a data warehouse. It utilizes a structural change point identification routine to flag anomalies in the predictive models alongside a variational autoencoder network configured to synthesize dense usage embeddings. These embeddings successfully abstract the spotted variations while serving as a framework to clarify their foundational triggers. The real-world viability of this system was demonstrated and benchmarked through its live rollout across the multi-campus environment of La Trobe University.

Our strategy uses a specialized vector database to archive behavioral embeddings, enabling rapid similarity retrieval during diagnostics. Pairing these dense vectors with historical meta-records organized in a dimensional format inside the data warehouse reinforces the reliability and holistic nature of our strategy. Practical tests using field data from La Trobe University highlight the capacity of our tracking system to pinpoint variations away from baseline energy estimations. We illustrated how these derived energy signatures enable a granular comprehension of the tracked anomalies, offering meaningful context regarding the specific details of the underlying operational triggers.

In summary, this research addresses the challenge of concept drift within energy forecasting engines by delivering a well-rounded strategy that combines anomaly detection with explainability routines via vector embeddings. The practical deployment and auditing at La Trobe University confirm both the real-world operational readiness and overall effectiveness of the predictive framework established through this research.

Abbreviations

AI = Artificial Intelligence

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

ETL = Extract-Transform-Load

HVAC = Heating, Ventilation and Air Conditioning

k-NN = k-nearest neighbors

KL = Kullback–Leibler

LEAP = La Trobe Energy AI Platform

MAPE = Mean Absolute Percentage Error

PELT = Pruned Exact Linear Time

ReLU = Rectified Linear Unit

REST = Representational State Transfer

RMSE **=** Root Mean Squared Error

ROC = Receiver Operating Characteristic

t-SNE = t-Distributed Stochastic Neighbor Embedding

Nomenclature

β Penalty parameter

ΔIs, The improvement in the performance measure from the split at node s in tree t

Ij Feature importance score for feature j in the entire ensemble of gradient-boosted decision trees

C(.) Cost function

Dec Decoder Model

Enc Encoder Model

$logvar$ log-variance vector

μ mean vector

$x_{1:n}$ a time series of data containing n elements x1, x2, ..., xn

Acknowledgements

This work is supported by the Department of Climate Change, Energy, the Environment and Water of the Australian Federal Government, as part of the International Clean Innovation Researcher Networks (ICIRN) program, grant number ICIRN000077.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 05 Dec 2022 Revised: 10 Feb 2023 Accepted: 27 Feb 2023
Published online: 15 June 2024

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at https://creativecommons.org/licenses/by-nc-sa/4.0/.

References

1. Li J, Herdem MS, Nathwani J, Wen JZ. Methods and applications for artificial intelligence, big data, Internet of Things, and blockchain in smart energy management. Energy AI. 2023;11:100208.

2. De Silva D, Sierla S, Alahakoon D, Osipov E, Yu X, Vyatkin V. Toward intelligent industrial informatics: A review of current developments and future directions of artificial intelligence in industrial applications. IEEE Ind Electron Mag. 2020;14(2):57-72.

3. Baur L, Ditschuneit K, Schambach M, Kaymakci C, Wollmann T, Sauer A. Explainability and interpretability in electric load forecasting using machine learning techniques–A review. Energy AI. 2024;100358.

4. Yan B, Hao F, Meng X. When artificial intelligence meets building energy efficiency, a review focusing on zero energy building. Artif Intell Rev. 2021;54(3):2193-220.

5. De Silva D, Yu X, Alahakoon D, Holmes G. Semi-supervised classification of characterized patterns for demand forecasting using smart electricity meters. In: 2011 International Conference on Electrical Machines and Systems. IEEE; 2011. p. 1-6.

6. Nallaperuma D, De Silva D, Alahakoon D, Yu X. Intelligent detection of driver behavior changes for effective coordination between autonomous and human driven vehicles. In: IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society. IEEE; 2018. p. 3120-5.

7. Nawaratne R, Alahakoon D, De Silva D, Yu X, Rathnayake P, Hewage C, et al. Hierarchical two-stream growing self-organizing maps with transience for human activity recognition. IEEE Trans Ind Inf. 2019;16(12):7756-64.

8. Somu N, MR GR, Ramamritham K. A deep learning framework for building energy consumption forecast. Renew Sustain Energy Rev. 2021;137:110591.

9. Bonetto R, Rossi M. Machine learning approaches to energy consumption forecasting in households. arXiv:1706.09648. 2017.

10. Sifat MMH, Choudhury SM, Das SK, Ahamed MH, Muyeen S, Hasan MM, et al. Towards electric digital twin grid: Technology and framework review. Energy AI. 2023;11:100213.

11. Agrahari S, Singh AK. Concept drift detection in data stream mining: A literature review. J King Saud Univ-Comput Inf Sci. 2022;34(10):9523-40.

12. Ahmad T, Chen H, Guo Y, Wang J. A comprehensive overview on the data driven and large scale based approaches for forecasting of building energy demand: A review. Energy Build. 2018;165:301-20.

13. Liu Y, Roberts MC, Sioshansi R. A vector autoregression weather model for electricity supply and demand modeling. J Mod Power Syst Clean Energy. 2018;6(4):763-76.

14. López JC, Rider MJ, Wu Q. Parsimonious short-term load forecasting for optimal operation planning of electrical distribution systems. IEEE Trans Power Syst. 2019;34(2):1427-37.

15. Wang Y, Chen Q, Gan D, Yang J, Kirschen DS, Kang C. Deep learning-based socio-demographic information identification from smart meter data. IEEE Trans Smart Grid. 2019;10(3):2593-602.

16. Meng Z, Sun H, Wang X. Forecasting energy consumption based on SVR and Markov model: A case study of China. Front Environ Sci. 2022;10:883711.

17. Li C, Tao Y, Ao W, Yang S, Bai Y. Improving forecasting accuracy of daily enterprise electricity consumption using a random forest based on ensemble empirical mode decomposition. Energy. 2018;165:1220-7.

18. Ahmad T, Chen H. Deep learning for multi-scale smart energy forecasting. Energy. 2019;175:98-112.

19. Ishaq M, Kwon S, Kim D, Kim Y, Kim S, Lee J, et al. Short-term energy forecasting framework using an ensemble deep learning approach. IEEE Access. 2021;9:94262-71.

20. Lu N, Zhang G, Lu J. Concept drift detection via competence models. Artif Intell. 2014;209:11-28.

21. Liu A, Song Y, Zhang G, Lu J. Regional concept drift detection and density synchronized drift adaptation. In: IJCAI International Joint Conference on Artificial Intelligence. 2017.

22. Gama J, Medas P, Castillo G, Rodrigues P. Learning with drift detection. In: Bazzan ALC, Labidi S, editors. Advances in Artificial Intelligence – SBIA 2004. Berlin: Springer; 2004. p. 286-95.

23. Baena-García M, del Campo-Ávila J, Fidalgo R, Bifet A, Gavalda R, Morales-Bueno R. Early drift detection method. In: Fourth International Workshop on Knowledge Discovery from Data Streams. Vol. 6. Citeseer; 2006. p. 77-86.

24. Frias-Blanco I, del Campo-Ávila J, Ramos-Jimenez G, Morales-Bueno R, Ortiz-Díaz A, Caballero-Mota Y. Online and non-parametric drift detection methods based on Hoeffding's bounds. IEEE Trans Knowl Data Eng. 2015;27(3):810-23.

25. Liu A, Zhang G, Lu J. Fuzzy time windowing for gradual concept drift adaptation. In: 2017 IEEE International Conference on Fuzzy Systems. IEEE; 2017. p. 1-6.

26. Bifet A, Gavalda R. Learning from time-changing data with adaptive windowing. In: Proceedings of the 2007 SIAM International Conference on Data Mining. SIAM; 2007. p. 443-8.

27. Song X, Wu M, Jermaine C, Ranka S. Statistical change detection for multi-dimensional data. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2007. p. 667-76.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

28. Qahtan AA, Alharbi B, Wang S, Zhang X. A PCA-based change detection framework for multidimensional data streams. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2015. p. 935-44.

29. Osipov E, Kahawala S, Haputhanthri D, Kempitiya T, De Silva D, Alahakoon D, et al. Hyperseed: Unsupervised learning with vector symbolic architectures. IEEE Trans Neural Netw Learn Syst. 2024;35(5):6583-97.

30. Alippi C, Boracchi G, Roveri M. Just-in-time classifiers for recurrent concepts. IEEE Trans Neural Netw Learn Syst. 2013;24(4):620-34.

31. Yu S, Wang X, Príncipe JC. Request-and-reverify: Hierarchical hypothesis testing for concept drift detection with expensive labels. arXiv:1806.10131. 2018.

32. Webb GI, Lee LK, Petitjean F, Goethals B. Understanding concept drift. arXiv:1704.00362. 2017.

33. Hinder F, Artelt A, Vaquet V, Hammer B. Contrasting explanation of concept drift. In: 30th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN). 2022.

34. Hayashi A, Mizuhara Y, Suematsu N. Embedding time series data for classification. In: Perner P, Imiya A, editors. Machine Learning and Data Mining in Pattern Recognition. Berlin: Springer; 2005. p. 356-65.

35. Zhou T, Ma Z, Wen Q, Wang X, Sun L, Jin R. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In: Proceedings of the 39th International Conference on Machine Learning. PMLR; 2022. p. 27268-86.

36. Mikalsen KØ, Bianchi FM, Soguero-Ruiz C, Jenssen R. Time series cluster kernel for learning similarities between multivariate time series with missing data. Pattern Recognit. 2018;76:569-81.

37. Kleyko D, Osipov E, De Silva D, Wiklund U, Alahakoon D, Li H, et al. Integer self-organizing maps for digital hardware. In: 2019 International Joint Conference on Neural Networks (IJCNN). IEEE; 2019. p. 1-8.

38. Franceschi JY, Dieuleveut A, Jaggi M. Unsupervised scalable representation learning for multivariate time series. Adv Neural Inf Process Syst. 2019;32.

39. Yue Z, Wang Y, Duan J, Yang T, Huang C, Tong Y, et al. TS2Vec: Towards universal representation of time series. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 36, No. 8. 2022. p. 8980-7.

40. Žliobaitė I, Pechenizkiy M, Gama J. An overview of concept drift applications. In: Japkowicz N, Stefanowski J, editors. Big Data Analysis: New Algorithms for a New Society. Cham: Springer; 2016. p. 91-114.

41. Moreira JPCLM. Travel time prediction for the planning of mass transit companies: A machine learning approach [PhD thesis]. Porto: Universidade do Porto; 2008.

42. Chamishka S, Madhavi I, Nawaratne R, Alahakoon D, De Silva D, Chilamkurti N, et al. A voice-based real-time emotion detection technique using recurrent neural network empowered feature modelling. Multimedia Tools Appl. 2022;81(24):35173-94.

43. Ufuk Gökçe H, Umut Gökçe K. Integrated system platform for energy efficient building operations. J Comput Civ Eng. 2014;28(6):05014005.

44. Hong B, Zhang W, Zhou Y, Chen J, Xiang Y, Mu Y. Energy-internet-oriented microgrid energy management system architecture and its application in China. Appl Energy. 2018;228:2153-64.

45. Plageras AP, Psannis KE, Stergiou C, Wang H, Gupta BB. Efficient IoT-based sensor BIG data collection–processing and analysis in smart buildings. Future Gener Comput Syst. 2018;82:349-57.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press

46. Moraliyage H, Mills N, Rathnayake P, De Silva D, Jennings A. UNICON: An open dataset of electricity, gas and water consumption in a large multi-campus university setting. In: 2022 15th International Conference on Human System Interaction (HSI). IEEE; 2022. p. 1-8.

47. Chen T, Guestrin C. XGBoost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016. p. 785-94.

48. Killick R, Fearnhead P, Eckley IA. Optimal detection of changepoints with a linear computational cost. J Am Stat Assoc. 2012;107(500):1590-8.

49. Mills N, Rathnayaka P, Moraliyage H, De Silva D, Jennings A. Cloud edge architecture leveraging artificial intelligence and analytics for microgrid energy optimisation and net zero carbon emissions. In: 2022 15th International Conference on Human System Interaction (HSI). IEEE; 2022. p. 1-7.

50. Wimalaratne S, Haputhanthri D, Kahawala S, Gamage G, Alahakoon D, Jennings A. UNISOLAR: An open dataset of photovoltaic solar energy generation in a large multi-campus university setting. In: 2022 15th International Conference on Human System Interaction (HSI). IEEE; 2022. p. 1-5.

© 2024 The Author(s). | Published by Institute of Smart Energy Systems Press