Adaptive Load Forecasting for Renewable-Rich Power Systems: Techniques, Evaluation, and Deployment

S Krishnan*, P Umasankar
Department of Electrical and Electronics Engineering, Mahendra Engineering College, Mallasamudram, Namakkal – 637503,
Email: pumaieee@gmail.com

Abstract—Renewable-rich distribution systems require load forecasts that may evolve over time with changing measurements and forecasts, operator input, tariffs, and control decisions. In this survey, that closed-loop task is treated as interactive load forecasting, with an overview of the literature published from 2015 to the present across different forecast horizons at residential, building, aggregator, and low-voltage feeder levels. The literature is organized according to a taxonomy based on representation, adaptation mechanism, uncertainty treatment, privacy architecture, and decision interface. Besides deployment evidence, deep temporal, graph, online, probabilistic, federated, and cost-oriented methods are compared. The main unresolved problem is the lack of causal consistency. The load being forecast is affected by activities pertaining to demand response, storage, and home-energy management. Consequently, there is a need for action-conditioned datasets, counterfactual assessments, calibrated online learning, and operational measures beyond mere average prediction error.

Index Terms— Load forecasting, concept drift, demand response, distribution systems, federated learning, probabilistic forecasting, smart grids.

I. Introduction

Load forecasting is changing from a periodic planning assessment to a real-time part of distribution-system control. Rooftop photovoltaic (PV) generation, batteries, electric vehicles (EVs), heat pumps, dynamic tariffs, and automated home-energy management systems (HEMS) cause rapid, partly controllable shifts in measured net load. Many of these changes were smoothed at the transmission scale. However, at a home, building, microgrid, or low-voltage feeder, one charging event may dominate the next. Smart-meter and/or weather conditions, the status of individual devices, operator knowledge, or market instructions relevant to forecasts are thus subject to change, so the forecasts need to be adjusted accordingly. This paper considers such a closed-loop process as interactive load forecasting. Classical reviews are sound when dealing with point forecasting or distributional/probabilistic forecasting.

Two key developments are the Global Energy Forecasting Competition (GEFCom), which demonstrated the utility of common datasets and scoring rules, and the main concepts of probabilistic load forecasting (PLF) established by Hong and Fan [1], [2]. Subsequent reviews compiled single/hybrid, convolutional, and recurrent models [3], [4]. Another recent review explained explainability and interpretability for machine-learning load forecasting [5]. These studies are useful, but the compartmentalizing question is often which model gives the best prediction of a historical value. Together, they do not discuss how information for intervention is received; how models learn from failover sets when they drift; how uncertainty is communicated to a controller; or how models remain in operation after a controller changes demand. The literature is growing for each of these strands, but there has not been a full synthesis of the closed operational loop.

This gap is important because passive prediction and operational forecasting have different requirements. A model trained with past weather and load data assumes that future load has the same form as past load. Demand response (DR), battery dispatch, and home-energy management challenge that assumption. A forecast can lead to an action, which can lead to an observed change in load that is

Journal of Intelligent Engineering and Informatics, Volume 1, Issue 1, Pages 40-52
Corresponding author. E-mail address: pumaieee@gmail.com
Received 9 July 2026; **revised** 19 August 2026; **accepted** 29 August 2026; **published** 06 September 2026.

then provided to the forecaster. When a planned action is not included in the conditioning variables, the update may treat a deliberate change as random drift. Low error on a historical test split does not indicate that the model will also be useful in this feedback-loop process.

The key challenge, therefore, is causal consistency under demand changes caused by the actions taken: the forecast should allow inherent load shifts to be separated from shifts induced by the decision that used the forecast. The review encompasses day-ahead, intraday, and minute-ahead forecasts for low-voltage (LV) and medium-voltage (MV) feeders, aggregators, buildings, microgrids, and households in peer-reviewed publications from 2015 to July 2026. The target may be gross demand or net demand with behind-the-meter generation and storage. At least one of the included methods must support one or more of the following interactive functions: high-frequency updating, explicit uncertainty, spatial/hierarchical reconciliation, transfer/personalization, privacy-preserving collaboration, explanation, cost-aware training, or a direct control interface.

Outside the boundary are long-term capacity forecasting, forecasting based only on electricity prices, forecasting of renewable generation alone, state estimation, and non-intrusive electricity load monitoring (NILM) without a forecasting role. The evidence base consists of thirty anchor studies from IEEE and leading energy and forecasting journals. Searches included “load forecasting” combined with residential, net load, online, concept drift, distribution shift, probabilistic, graph, federated, privacy, demand response, control, and cost-oriented terms.

This results in a three-fold taxonomy. The representation layer poses the question: What structure is learned—calendar and weather effects, temporal dynamics, appliance composition, or spatial and hierarchical relations? The adaptation layer addresses how the model changes through online learning, drift-aware ensembles, transfer and personalization, or federated updates. The decision layer addresses “what is delivered” and “what is used”: point forecasts, quantiles and scenarios, explanations, baselines, and cost/action-conditioned prediction. These layers are connected in a closed loop, as illustrated in Figure 1, and the evidence, evaluation, and implementation readiness are compared in Tables 2–4. The synthesis identifies four priorities. First, future datasets should document the action, when it occurred, and whether it was executed correctly, rather than only the resulting meter trace.

Second, uncertainty calibration needs to be maintained after online updating and at the tails for reserve/flexibility decisions. Third, the costs of privacy and communication have to be measured alongside accuracy. Fourth, operational regret and counterfactual tests have to be incorporated into forecast evaluation. These directions shift load forecasting from a purely predictive exercise to a tangible element of reliable renewable-rich power-system operation. They also provide a shared foundation for utilities, aggregators, and researchers to determine whether a reported forecasting gain can pass the test of a live control loop.

II. Survey Scope and Review Method

A. Exact Boundaries

The boundary listed in Table 1 is used for this survey. The word ‘interactive’ does not mean that a person has to edit every prediction. It means that forecasting is part of a continuous cycle of information–decision–observation. Although a human signal is one possibility, automated tariff, controller, battery, or demand-response signals are equally important. This definition deliberately makes the boundary narrower than all smart-grid forecasting and broader than online neural-network training.

The target is also distinguished from the inputs in Table 1. PV generation can be included as an input or as part of net load and can also be included in a solar-generation forecasting paper if solar generation is forecast in combination with demand. Similarly, appliance identification is not treated as the final objective when appliance models supply features that enhance an overall forecast [6].

Table 1. Scope boundary for interactive load forecasting

Dimension	Included	Excluded
System level	Household, building, campus, aggregator, microgrid, LV/MV feeder	National long-term adequacy and generation-expansion studies
Horizon	Minutes ahead, intraday, day ahead	Multi-year demand growth and capacity planning
Target	Gross load, net load, flexibility-adjusted demand, DR baseline	Price alone, renewable output alone, state-estimation variables
Interaction	New measurements, online updates, operator or user context, tariffs, distributed energy resource (DER) states, scheduled actions, forecast-error feedback	One-time offline fitting with no operational update or decision link
Method families	Statistical baselines, temporal deep learning, graph/hierarchical, probabilistic, adaptive, transfer, federated, explainable, decision-aware	NILM or anomaly detection unless it supplies a forecasting feature

B. Search and Synthesis Procedure

The search was designed as a focused integrative review. IEEE Xplore and major publisher indexes were searched for the stated period. Broad terms located surveys and benchmark papers, which in turn led to paired searches for emerging families, such as “residential load forecasting AND graph,” “load forecasting AND concept drift,” “probabilistic load forecasting AND temporal correlation,” “federated load forecasting AND privacy,” and “load forecasting AND demand response baseline.” Candidate records were retained when the target and horizon matched those in Table 1 and when the method supported an interactive function.

Method coding was used in the synthesis instead of publication order. Forecast target, system scale, temporal resolution, input context, output type, adaptation rule, privacy setting, validation protocol, and deployment evidence were coded for each study. The coding revealed dependencies that a model-only taxonomy fails to capture. A method can, for instance, be an offline point model, an online adaptive model, or a local model within a federated system. It is therefore classified in a two-step process: first by its learned structure and second by the mechanism through which it becomes interactive. Review papers [1]–[5] provide the field baseline, while other papers are considered representative evidence in their respective methodological groupings. A statistical meta-analysis is not intended because of the lack of alignment in datasets, aggregation levels, time horizons, and measures.

III. Theoretical Foundation of Interactive Forecasting

A. Forecast Target in a Renewable-Rich Feeder

The measured quantity at a customer or feeder is often net load rather than uncontrolled demand. A simple accounting identity is

$$L_t^{\text{net}}=L_t^{\text{gross}}-P_t^{\text{PV}}+P_t^{\text{ch}}-P_t^{\text{dis}}, \tag{1}$$

where photovoltaic production reduces measured demand, battery charging increases it, and battery discharge reduces it. The same sign convention can be applied to electric-vehicle charging, curtailed load, and exported power. Several terms are decision variables. When a storage unit or flexible demand is scheduled, a future net-load value depends not only on weather and past load but also on the operation of the storage unit or flexible demand.

Let H_t denote the history available at issue time t, $\mathbf{x}_{t+1:t+H}$ the known or forecast exogenous context, and $\mathbf{a}_{t:t+H}$ the planned tariff, dispatch, or demand-response actions. An interactive multi-horizon quantile forecast is

$$\hat{\mathbf{y}}_{t+1:t+H}^{(q)}=f_{\theta_t}(\mathrm{H}_t,\mathbf{x}_{t+1:t+H},\mathbf{a}_{t:t+H},q). \tag{2}$$

The parameter subscript t allows the model to change online. The action argument is the main distinction from passive forecasting. If actions are unknown, the model should state a policy or scenario for them instead of silently treating them as noise.

B. Interaction as a Closed Operational Loop

Figure 1 shows the proposed closed-loop taxonomy. Data and context flow into a forecasting engine that provides point, probabilistic, or explanatory output in response to user actions through a decision interface; as the grid changes, so does the realization of the predicted physical outcome; and, as a result of detected errors, the models are recalibrated, personalized, and/or updated in the forecasting engine. What is new is that information on the effect of the action on demand needs to be fed back into the forecast model through the causal-consistency gate.

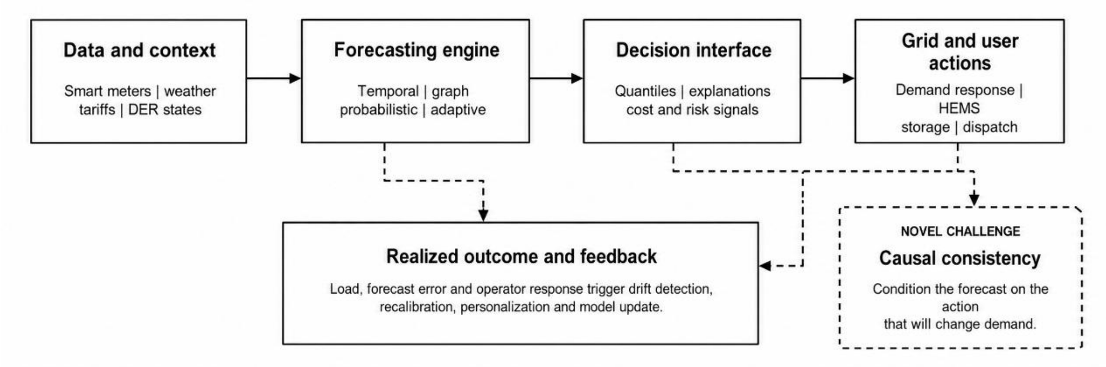

Figure 1. Closed-loop taxonomy of interactive load forecasting

One common update is stochastic online learning,

$$\theta_{t+1}=\theta_t-\eta_t\nabla_\theta \,\mathrm{L}\left(y_{t+1},f_{\theta_t}(\mathrm{H}_t,\mathbf{x},\mathbf{a})\right), \tag{3}$$

where η_t controls how quickly recent evidence replaces older structure. A large step adapts quickly but may forget seasonal behavior; a small step is stable but slow after an abrupt tariff or occupancy change. Drift detectors and online ensembles address this stability–plasticity trade-off by updating model weights, retraining selected components, or introducing a new learner [7]–[10].

C. Calibration and Decision Frameworks

Renewable-rich operation needs distributions, not only averages. A quantile model is commonly trained with pinball loss,

$$\rho_q(e)=\max\{qe,(q-1)e\}, \qquad e=y-\hat{y}^{(q)}. \tag{4}$$

Prediction intervals/scenarios can be calculated from the quantiles, but nominal coverage is not enough. Intervals should be calibrated according to horizon, season, aggregation level, and operational regime. Temporal dependence is also important because a storage controller sees a trajectory, not independent intervals over time [11]. Even a seemingly well-calibrated set of independent quantiles can result in implausible ramps. There is a distinction between prediction quality and decision quality.

A good interactive objective might involve a combination of point error, probabilistic score, operational regret and cost of resources:

$$\mathrm{J}=\lambda_p\mathrm{L}_{\mathrm{point}}+\lambda_q\mathrm{L}_{\mathrm{prob}}+\lambda_r\mathrm{R}_{\mathrm{op}}+\lambda_c\mathrm{C}_{\mathrm{compute,comm}}. \tag{5}$$

The weights need to be determined according to the use case. If the system is designed for a home display, the balance might be more toward interpretability and light processing, while for a feeder reserve scheduler, the emphasis might be more on tail calibration and low operational regret. Cost-oriented forecasting directly addresses downstream cost [12], but this objective should not reward a biased forecast that is safe for one tariff regime and unsafe under another tariff regime.

D. Causal Consistency

Suppose an operator selects action a because of a forecast. The observed target is then the potential outcome $Y(a)$, while the load under a different action a' is unobserved. The operational question is

$$\Delta_{t,\mathrm{h}}(a,a')=\mathrm{E}[Y_{t+\mathrm{h}}(a)-Y_{t+\mathrm{h}}(a')|\mathrm{H}_t]. \tag{6}$$

When action assignment is based on the same forecast features, ordinary train/test splits estimate neither term. This leads to policy confounding and leakage of policy feedback. The problem is exemplified by demand-response baseline models, which provide information on the event trajectory but estimate what demand would have been if the event had not occurred [13]. A causally consistent system should be able to capture the actions performed, approximate compliance, distinguish between independent changes and interventions, and assess counterfactual or intervention performance. This requirement does not call for another neural architecture to be added to the passive benchmark; rather, it links the theoretical target to the future research agenda.

IV. Methodological Classification

Examples of this condensed taxonomy are provided in Table 2. Model families are not mutually exclusive. They denote the main function of a method in the operational loop. A method may produce quantiles from a graph network, federatively train an adaptive recurrent model, or optimize a federated model to reduce household costs. The columns therefore indicate the source of value, interactive mechanism, typical evidence, main deployment constraint, and readiness.

Table 2. Method taxonomy and deployment readiness

Method family	Structure or output learned	Interactive mechanism	Representative evidence	Main deployment constraint	Readiness
Calendar/statistical and support-vector regression (SVR) baselines	Seasonal, weather, lag, and event relations	Rapid refit; transparent user corrections	Residential granularity [14]; DR baseline [13]	Limited nonlinear transfer under DER-driven regimes	High for bounded tasks
Deep temporal and hybrid	Long- and short-memory patterns across horizons	Recurrent state update; pooled or multitask learning	Long short-term memory (LSTM) [15], pooling recurrent neural network (RNN) [16], hybrid review [4], appliance multitask [6]	Compute, opacity, and sensitivity to drift	Medium–high
Transfer and meta-learning	Shared representation plus customer adaptation	Few-shot personalization for new meters	Individualized transfer/meta-learning [17]	Negative transfer and support-set quality	Medium
Graph and hierarchical	Spatial, electrical, or similarity relations	Neighbor/context updates; coherent aggregation	Residential graph forecasting [18]	Graph construction, topology change, meter alignment	Medium
Probabilistic and scenario	Quantiles, intervals, densities, trajectories	Recalibration; risk-aware decision interface	Ensemble [19], pinball LSTM [20], temporal correlation [11], quantile combination [21], input uncertainty [22]	Tail data and joint calibration	Medium–high
Online and drift-aware	Time-varying parameters or ensemble weights	Error-triggered continuous adaptation	Adaptive RNN [7], RNN–autoregressive integrated moving average (ARIMA) ensemble [8], online probabilistic combination [9], online quantile ensemble [10]	Stability, delayed labels, rollback safety	Medium
Federated and privacy-preserving	Global, clustered, hierarchical, or personalized parameters	Distributed update without raw meter pooling	Edge federated learning (FL) [23], RNN FL [24], clustered FL [25], privacy FL [26]–[28], cost-personalized FL [29]	Communication, heterogeneity, privacy accounting	Medium
Decision- and action-aware	Forecast tuned to operating cost, baseline, or control action	Controller feedback and action conditioning	Cost-oriented loss [12], DR baseline [13], forecast-plus-model predictive control (MPC) [30]	Policy confounding and limited field counterfactuals	Early–medium

A. Conventional Forecasting Baselines

Operational constraints due to interactivity mean that calendar, weather, autoregressive, regression, and support-vector models are still needed. A model that can be refitted in seconds, audited by the operator, and rolled back after a bad update is probably more useful than a slightly more accurate network. Residential evidence indicates that both forecast granularity and calendar effects are significant contributors to error and therefore affect forecast comparisons. Baselines for office-building demand response have also been computed using SVR [13], where stability and traceability are as important as average error.

These methods provide a realistic lower threshold. Their characteristics align with known causes, and prediction intervals can be derived using residual, quantile, or conformal procedures. They are less effective when multiple flexible devices generate nonlinear interactions, the historical record is brief, or relationships change drastically. An interactive system does, however, need to retain a seasonal-naïve and a regularized linear or tree baseline. Without them, the measured value of a complex adaptation step can be confounded by a simple calendar effect.

B. Deep and Hybrid Prediction Methods

Deep temporal models capture nonlinear relationships among consecutive (lagged) load, weather, and calendar time series. Kong et al. [15] proposed a widely used short-term residential demand model based on deep recurrent learning. Before recurrent prediction, Shi et al. [16] demonstrated how cross-series structure can compensate for volatile individual traces. Several reviews already cover convolutional neural network (CNN)–LSTM and hybrid architectures [3], [4]; however, a model does not require great depth to be interactive. Reusable state and modular updating provide the interactive value. A recurrent state can store the most recent observation and does not need to be rebuilt to include all features. A small output head can be adjusted to a new home while the encoder remains frozen. In a hybrid model, a stable seasonal component is retained and only a nonlinear residual is updated.

The addition of application-aware multitask learning provides another useful inductive bias: representations shared among related appliance behaviors can also be shared across task heads [6]. This can enhance explanation and adaptation when a heat pump or vehicle enters operation. Silent extrapolation is the main risk. Random shuffling can leak seasonality and customer identity, while lengthy training windows can delay a network's response to new tariffs. Deep models also tend to produce confident values outside the data blocks on which they were trained. Thus, deployment needs uncertainty layers, explicit missing-data tests, a fallback model, and time-ordered backtests. Model size should be justified by measurable cost or risk reduction, not merely by a slight reduction in normalized root-mean-square (RMS) error.

C. Transfer and Meta-Learning

New meters and changing occupants create a cold-start problem. The process starts with a model learned from similar customers and fine-tunes it using a small profile of local history. Meta-learning also involves learning an initialization that can be adapted with a small number of examples. Lee and Rhee incorporated these concepts into a framework for creating individualized short-term load forecasts [17]. This family is interactive; that is, user-specific evidence alters a shared prior without requiring full retraining. Personalization can be valuable when weather sensitivity and lifestyle are similar across multiple households, even when the households vary in size and appliance inventory. It can reduce the data required for a new customer and limit computation on an edge device. Negative transfer is its main weakness. Simply averaging load may connect households with different flexibility or PV characteristics. A safe design should expose the source cohort, track whether adaptation improves a local baseline, and allow a customer model to reject the shared initialization. Personalization should also be assessed after occupancy or technology changes, not solely during a held-out month under stationary conditions.

D. Graph-Based Forecasting Approaches

Spatial and hierarchical models explicitly recognize one aspect of the problem that sequence models do not account for: loads depend on geography, topology, and building type; they also depend on aggregation level. A graph neural network (GNN) is a type of artificial neural network (ANN) that models meters or zones as nodes and represents how information flows between them. A spatial–temporal graph formulation for short-term residential forecasting was proposed by Lin, Wu, and Boulet [18]. Another hierarchy is provided by bottom-up appliance and customer models with reconciliation constraints that refer to feeder totals [6]. In an interactive system, the graph structure can be used to target updates. Adaptation can be limited to relevant neighbors because a local outage, weather cell, or event may affect only a connected subset of meters. Hierarchical reconciliation can also physically reconcile forecasts for customers, transformers, and feeders. The problem is that a convenient similarity graph might not be causal or electrically meaningful. A tariff change or photovoltaic installation may require updates to correlation edges, and the topology of the distribution network may not be fully known. Studies should compare physical, geographic, learned, and dynamic graphs; account for graph-construction costs; and test the consequences when meters appear, disappear, or move between feeder groups.

E. Probabilistic and Temporally Coherent Forecasting

Probabilistic forecasting provides the decision layer with quantiles, intervals, densities, or scenarios. The basic tutorial and GEFCom work established the main ideas of scoring and calibration [1], [2]. Since then, hybrid ensembles have been used to produce deterministic and probabilistic low-voltage forecasts [19]. Pinball-loss-guided LSTMs for directly targeting household quantiles [20] and nonparametric quantile combination for limiting dependence on any single model [21] are also closely related. Weather or contextual uncertainty can be propagated into volatile residential forecasts through input modeling [22]. Recalibration is the main interactive operation. If recent coverage is below its nominal level, the system can widen intervals, modify quantiles, or reweight ensembles. Temporal correlations also need to be preserved. Lemos-Vinasco, Bacher, and Møller developed online household models that explicitly account for temporal dependence [11].

This is important for storage and reserve scheduling because independent interval samples can generate impossible trajectories. Three limitations remain. First, there are few observations in the tails of the household distribution, so a seemingly precise value for the 0.99 quantile may be unstable. Second, poor conditional coverage can be concealed during peak demand, extreme weather, and demand-response events. Third, many papers report forecast scores but fail to demonstrate how the distribution affects a decision. A good evaluation should report pinball loss or a continuous ranked probability score, together with empirical coverage, interval width, multistep dependence, and operational cost on the same rolling test.

F. Online Adaptation and Drift-Aware Ensembles

Online methods are updated as soon as an observation or a short batch becomes available. Fekri et al. introduced a smart-meter forecasting method based on an online adaptive recurrent network [7]. Other work combines a recurrent learner with an ARIMA learner in an online ensemble under concept drift [8]. Recent literature includes online combinations for probabilistic forecasts and quantiles under concept shift or distribution shift [9], [10]. Such studies explicitly address nonstationarity and constitute a well-defined methodology for interactive forecasting. Normalization of recent data, adjustment of a final layer, fine-tuning across the entire network, changes in combination weights, introduction of a specialist model, and adjustment of output intervals are all possible forms of adaptation.

The selection should respond to the type of change. A regime model may require only parameter updates during a gradual weather change, whereas installation of a new device or a sudden tariff change may require a new regime model. Error-based drift detectors help, but errors do not appear until after the forecast horizon and can be distorted by missing meter data. Operational safeguards for updates are infrequently assessed. An update can disrupt a model that previously performed well and degrade subsequent performance. A deployable system should therefore include shadow scoring before an update, bounded parameter changes, versioned rollback, and a rule for returning to a stable baseline. It should also account for exogenous drift and intervention effects. Otherwise, a planned battery discharge might cause "adaptation"—in other words, teach the model an incorrect uncontrolled-load pattern.

G. Federated and Privacy-Preserving Collaboration

Federated learning transmits model changes rather than raw meter readings. In this regard, early work proposed a federated-learning approach with edge computing for electrical load forecasting [23]. Distributed smart-meter data are then used to train a shared forecaster using recurrent federated models [24]. Clustering is a technique for dealing with non-independent household data by creating clusters of related hyperparameters or behaviors [25]. Privacy-preserving, hierarchical, and distributed variants introduce secure aggregation, layered coordination, or richer attack models [26]–[28]. More recently, personalized federated learning has linked the quality of local forecasts to home-energy cost [29]. This family is interactive at two time scales. Local models change with new household observations, and local updates are periodically combined with others on a server. A regular weighted aggregation is

$$\theta^{(r+1)} = \sum_{k=1}^{K} \frac{n_k}{\sum_j n_j}\, \theta_k^{(r+1)}, \tag{7}$$

where n_k is the local sample count. Simple sample weighting can be wrong when large but atypical customers dominate, so clustered, hierarchical, or personalized aggregation is often preferable.

Keeping data "local" is not a complete privacy guarantee. Gradients can leak information, secure aggregation does not protect against all inference attacks, and differential privacy introduces an accuracy trade-off. Forecast error should be reported together with communication rounds, client availability, energy use, privacy parameters, and attack tests. Asynchronous software updates and versioning are also important for field feasibility. Most studies still emulate clients on a central machine; this remains an algorithmic evaluation rather than an edge deployment.

H. Action-Oriented Prediction Models

Providing explanations can make interaction explicit by showing why specific weather, calendar, or device features influenced a prediction. Baur et al.'s review differentiates between built-in and post-hoc model interpretability [5]. A useful explanation should be stable under small input variations and related to an action, such as a peak warning associated with heat sensitivity, vehicle charging, or missing data. When there is no operational question, feature importance may provide confidence but not support a decision. Cost-oriented forecasting configures the model to account for the downstream consequence of forecast error. Zhang, Wang, and Hug found that optimal forecasting in this case may not minimize a symmetric statistical loss but may instead reduce operating cost [12]. This idea was combined with personalized federated learning for home-energy management by Barja-Martínez et al. [29].

Forecast-plus-control studies also show how prosumer projections are considered in MPC of storage to make a feeder dispatchable [30]. These methods are more 'closed loop' than accuracy-only studies. They still do not fully condition on actions. A controller typically requests a forecast and determines storage or demand-response actions. When the forecaster does not receive the selected action, it forecasts a blend of past policies. The counterfactual issue is particularly apparent when demand-response baselines are used [13]. The next step is a joint, but still modular, design: forecast autonomous demand, estimate action response with uncertainty, sum autonomous demand and action response to obtain an action-conditioned net-load distribution, and store sufficient data to assess alternative policies. This design is more auditable than an end-to-end model that provides less transparency about the intervention represented within a latent state.

V. Evaluation Techniques and Deployment Evidence

A. Statistical and Probabilistic Metrics

A single error measure is not sufficient for the scope in Table 1. Mean absolute error (MAE) is easy to understand and less sensitive to peaks than root-mean-square error (RMSE). RMSE is sensitive to large errors and is highly scale dependent. Mean absolute percentage error (MAPE) is unstable near zero and should not be used when net load may contain zeros; in such cases, MAE or scaled errors may be more useful for customer comparisons. Proper scores, such as pinball loss or the continuous ranked probability score (CRPS), together with empirical coverage and interval width [1], [2], are needed to evaluate probabilistic forecasts. Calibration should account for horizon and operating regime.

Each evaluation question is linked to evidence and a common failure in Table 3. The table illustrates why one average score cannot identify the "best model." An interactive model may be accurate at the time of an update but less accurate afterward; it may also be expensive to communicate and potentially unsafe for a controller.

Table 3. Evaluation framework for interactive forecasting

Evaluation question	Recommended measures	Required protocol	Frequent failure
Is the point forecast accurate?	MAE, RMSE, normalized or scaled error, peak error	Rolling-origin, time-ordered testing by horizon	Random split or one favorable season
Is uncertainty useful?	Pinball loss, CRPS, coverage, width, reliability by regime	Recalibrate only with past data; test joint trajectories	Reporting coverage without sharpness or tail checks
Does adaptation help?	Pre/post-drift error, detection delay, recovery time, forgetting	Inject or identify abrupt and gradual shifts; retain old-regime test	Updating on future labels or treating an action as drift
Does the decision improve?	Operating cost, imbalance, reserve shortfall, comfort violation, regret	Compare decisions using identical information and constraints	Claiming value from forecast error alone
Is deployment feasible?	Runtime, memory, energy, communication bytes/round, failure rate	Edge or realistic hardware; missing and delayed data tests	Central simulation labeled as deployment
Is privacy supported?	Privacy budget, attack success, utility loss, client dropout resilience	Threat model plus secure/federated implementation details	Equating local data storage with formal privacy
Is the forecast causally consistent?	Policy-specific error, baseline bias, treatment-effect error, off-policy regret	Log actions, propensity/compliance, and counterfactual assumptions	Omitting the action selected from the forecast

B. Validation Protocols

Rolling-origin evaluation best matches repeated operation: train on the past, forecast the next horizon, reveal only available outcomes, update, and continue. The protocol should cover seasonal changes, rare peaks, missing intervals, delayed weather, topology/customer changes, and intervention windows. Selecting a model based on the entire test year leaks information about the future regime. For online methods, hyperparameters and drift thresholds need to be set using a preceding validation stream. Results should be reported for each series as well as in aggregate for household and feeder studies. Personalization can reveal customers whose forecasts worsen after personalization, which is reflected in the average.

In federated studies, the participating clients should be varied and/or non-independent data should be excluded or substituted; in graph studies, nodes should be deleted or reassigned. Probabilistic models should be tested not only on sampled marginal quantiles but also on sampled trajectories. Decision-aware studies should present both a statistical score and a downstream score so that readers can determine whether the attained cost improvement is due to genuine information or an overly narrow loss function for the specific tariff.

C. From Simulation to Implementation

Table 4 distinguishes between what is promised by the algorithm and what is realistic for implementation. A field-ready method must have a live data path, causal ordering, update safeguards, and measured computation and communication. Central replay of smart-meter files can provide rigorous simulation, but it cannot test meter outages, delayed labels, privacy governance, or controller feedback. Table 4 does not indicate that simulation provides “weak evidence.” Wherever possible, simulation should be the initial test for uncommon or hazardous situations. The distinction is whether a paper substantiates the forecasting solution, the software design, or the overall claim of operational readiness. These represent different maturity levels and should be reported separately.

Table 4. Comparative evidence and implementation realism

Method/example	Typical evidence in reviewed work	What is already realistic	What remains mainly simulation-based
Statistical/SVR baseline [14], [13]	Historical household/building replay	Fast central processing unit (CPU) inference, auditability, bounded retraining	Counterfactual baseline validation during changing policies
Deep temporal/hybrid [15], [16], [6], [19]	Public or utility smart-meter backtests	Batch inference at aggregator; mature software	Safe continuous fine-tuning on constrained devices
Transfer/meta-learning [17]	Cross-customer adaptation experiments	Cold-start initialization and small local heads	Long-term monitoring of negative transfer after lifestyle change
Graph/hierarchical [18]	Constructed spatial/similarity graphs	Feeder or campus aggregation where topology is known	Dynamic topology, node churn, and electrical constraint integration
Online/drift-aware [7]–[10]	Sequential replay with artificial or observed shifts	Rolling updates and ensemble reweighting	Rollback, delayed labels, cyber faults, and field governance
Probabilistic [11], [20]–[22]	Quantile and interval backtests	Quantile interfaces for risk-aware scheduling	Joint tail calibration during interventions
Federated/private [23]–[29]	Multiple emulated clients, occasional edge framing	Local training and server aggregation prototypes	Large asynchronous fleets with measured privacy/energy cost
Forecast plus control [12], [13], [30]	Cost simulation, DR baseline, MPC case study	Decision-coupled objectives and receding-horizon control	Action-conditioned learning with off-policy field evaluation

VI. Critical Analysis

However, the most feasible near-term architecture is not a single large model. It is a stable statistical baseline, a compact temporal learner, residual or ensemble updates, calibrated quantiles, and a logged decision interface. Monitoring and rollback capabilities are possible for each part. Both statistical models and SVR are already practical for buildings and demand-response baselines because they are fast and auditable [13], [14]. Deep recurrent models are also deployable at an aggregator, although fine-tuning the whole network on edge devices is more difficult to manage [15], [16], [7]. The most straightforward approach to adaptation at the operational level is through online ensembles.

They can reweight a trusted baseline and an NLL without discarding either one, and several recent probabilistic and quantile-combination methods directly mitigate distribution shift [8]–[10]. Their published evidence has yet to progress beyond sequential replay. Field operation adds delayed labels, meter faults, update rollback, and actions that resemble drift. From a spatial perspective, graph methods can improve spatial coherence, but the improvement relies on a meaningful graph under topology and customer changes [18]. Marginal coverage is not sufficient for reserve, storage, and flexibility decisions that require probabilistic methods.

Temporally correlated scenarios are more realistic than independent intervals [11]. Although federated learning is appealing when meter data cannot be centralized, client simulations are often centralized, and communication, client-dropout, and privacy-accounting costs are often underestimated [23]–[28]. Federated solutions that address personalization and cost are attractive, but they remain at the advanced-prototype level rather than utility scale [29]. Decision-aware approaches appear strongest, whereas causal evidence remains weakest.

Cost-oriented loss and forecast-plus-MPC can improve simulated operating results [12], [30], and SVR baselines support bounded real-life applications [13]. None fully solve the feedback problem because the selected action can change the forecast target. Deployable progress therefore depends less on another architecture benchmark and more on action logs, counterfactual tests, safe online deployment, and common operational metrics. Claims should distinguish algorithm validation, system prototypes, shadow operation, and closed-loop field-control evidence.

VII. Open Challenges & Future Directions

An action-conditioned benchmark is the first priority. This requires correlation of meter time, weather time, tariff time, storage time, demand-response command time, user-override time, compliance time, and forecast-issue time. Benchmark protocols should state identification assumptions and allow off-policy or matched counterfactual evaluation while recognizing that randomized actions will be rare. Second, safety controls are needed in online learning. Drift detection should distinguish among autonomous drift, sensor failures, cyber manipulation, and deliberate intervention. Confidence gates, shadow evaluation, versioned rollback, and tests for catastrophic forgetting should be included in update procedures. Probabilistic calibration has to be validated as soon as possible after any accepted update, particularly for net-load reversals and rare peaks. Third, privacy claims must be quantified consistently.

Communication volume, threat models, privacy budgets, client-dropout behavior, client energy expenditure, and accuracy loss should be reported for federated studies. Graph and hierarchical methods should address topology changes and reconciliation among customer, transformer, and feeder distributions. Finally, prediction should be linked to operation for evaluation. Statistical metrics, comfort/service violations, compute, communication, calibration, and policy regret should be reported in shared tasks. At least shadow-mode trials are required to validate causal ordering and latency over multiple seasons, while digital twins can test unsafe events. A human component also needs to be investigated: operators should be able to annotate events or reject an update if that helps make better decisions later, and the system should record whether this benefited the operator.

VIII. Conclusion

Interactive load forecasting should be understood as a closed-loop operation rather than simply a new label for deep learning. The literature offers promising building blocks: temporal and graph models for volatile demand, quantiles and scenarios for risk modeling, online ensembles for drift, federated learning for distributed data, and cost-oriented models for control. Integration is not yet complete. Most evidence is based on historical replay, and few studies condition predictions on demand-response, storage, or home-energy actions taken from the forecast. This survey contributes four elements: a bounded definition, a three-layer taxonomy, an evaluation framework, and a deployment-realism comparison. Its principal finding is direct: reliable development now requires causal action logs, counterfactual testing, calibrated safe updating, and operational regret, rather than merely average error.

References

[1] T. Hong and S. Fan, “Probabilistic electric load forecasting: A tutorial review,” *International Journal of Forecasting*, vol. 32, no. 3, pp. 914–938, 2016, doi: 10.1016/j.ijforecast.2015.11.011.

[2] T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, and R. J. Hyndman, “Probabilistic energy forecasting: Global Energy Forecasting Competition 2014 and beyond,” *International Journal of Forecasting*, vol. 32, no. 3, pp. 896–913, 2016, doi: 10.1016/j.ijforecast.2016.02.001.

[3] A. Al Mamun, M. Sohel, N. Mohammad, M. S. H. Sunny, D. R. Dipta, and E. Hossain, “A comprehensive review of the load forecasting techniques using single and hybrid predictive models,” *IEEE Access*, vol. 8, pp. 134911–134939, 2020, doi: 10.1109/ACCESS.2020.3010702.

[4] K. Ullah, M. Ahsan, S. M. Hasanat, M. Haris, H. Yousaf, S. F. Raza, R. Tandon, S. Abid, and Z. Ullah, “Short-term load forecasting: A comprehensive review and simulation study with CNN-LSTM hybrids approach,” *IEEE Access*, vol. 12, pp. 111858–111881, 2024, doi: 10.1109/ACCESS.2024.3440631.

[5] L. Baur, K. Ditschuneit, M. Schambach, C. Kaymakci, T. Wollmann, and A. Sauer, “Explainability and interpretability in electric load forecasting using machine learning techniques—A review,” *Energy and AI*, vol. 16, Art. no. 100358, 2024, doi: 10.1016/j.egyai.2024.100358.

[6] S. Wang, X. Deng, H. Chen, Q. Shi, and D. Xu, “A bottom-up short-term residential load forecasting approach based on appliance characteristic analysis and multi-task learning,” *Electric Power Systems Research*, vol. 196, Art. no. 107233, 2021, doi: 10.1016/j.epsr.2021.107233.

[7] M. N. Fekri, H. Patel, K. Grolinger, and V. Sharma, “Deep learning for load forecasting with smart meter data: Online adaptive recurrent neural network,” *Applied Energy*, vol. 282, Art. no. 116177, 2021, doi: 10.1016/j.apenergy.2020.116177.

[8] R. K. Jagait, M. N. Fekri, K. Grolinger, and S. Mir, “Load forecasting under concept drift: Online ensemble learning with recurrent neural network and ARIMA,” *IEEE Access*, vol. 9, pp. 98992–99008, 2021, doi: 10.1109/ACCESS.2021.3095420.

[9] C. Cao, Y. He, Y. Zhou, and S. Wang, “An online probabilistic combination framework for power load forecasting under concept-drifting scenarios,” *Applied Energy*, vol. 399, Art. no. 126518, 2025, doi: 10.1016/j.apenergy.2025.126518.

[10] D. Qin, X. Wu, D. Sun, Z. Liang, and N. Zhang, "Load forecasting under distribution shift: An online quantile ensembling approach," *Applied Energy*, vol. 401, Art. no. 126812, 2025, doi: 10.1016/j.apenergy.2025.126812.

[11] J. Lemos-Vinasco, P. Bacher, and J. K. Møller, "Probabilistic load forecasting considering temporal correlation: Online models for the prediction of households' electrical load," *Applied Energy*, vol. 303, Art. no. 117594, 2021, doi: 10.1016/j.apenergy.2021.117594.

[12] J. Zhang, Y. Wang, and G. Hug, "Cost-oriented load forecasting," *Electric Power Systems Research*, vol. 205, Art. no. 107723, 2022, doi: 10.1016/j.epsr.2021.107723.

[13] Y. Chen, P. Xu, Y. Chu, W. Li, Y. Wu, L. Ni, Y. Bao, and K. Wang, "Short-term electrical load forecasting using the support vector regression model to calculate the demand response baseline for office buildings," *Applied Energy*, vol. 195, pp. 659–670, 2017, doi: 10.1016/j.apenergy.2017.03.034.

[14] P. Lusis, K. R. Khalilpour, L. Andrew, and A. Liebman, "Short-term residential load forecasting: Impact of calendar effects and forecast granularity," *Applied Energy*, vol. 205, pp. 654–669, 2017, doi: 10.1016/j.apenergy.2017.07.114.

[15] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019, doi: 10.1109/TSG.2017.2753802.

[16] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—A novel pooling deep RNN," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 5271–5280, 2018, doi: 10.1109/TSG.2017.2686012.

[17] E. Lee and W. Rhee, "Individualized short-term electric load forecasting with deep neural network based transfer learning and meta learning," *IEEE Access*, vol. 9, pp. 15413–15425, 2021, doi: 10.1109/ACCESS.2021.3053317.

[18] W. Lin, D. Wu, and B. Boulet, "Spatial-temporal residential short-term load forecasting via graph neural networks," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5373–5384, 2021, doi: 10.1109/TSG.2021.3093515.

[19] Z. Cao, C. Wan, Z. Zhang, F. Li, and Y. Song, "Hybrid ensemble deep learning for deterministic and probabilistic low-voltage load forecasting," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 1881–1897, 2020, doi: 10.1109/TPWRS.2019.2946701.

[20] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Applied Energy*, vol. 235, pp. 10–20, 2019, doi: 10.1016/j.apenergy.2018.10.078.

[21] Y. He, C. Cao, S. Wang, and H. Fu, "Nonparametric probabilistic load forecasting based on quantile combination in electrical power systems," *Applied Energy*, vol. 322, Art. no. 119507, 2022, doi: 10.1016/j.apenergy.2022.119507.

[22] G. Xie, X. Chen, and Y. Weng, "Input modeling and uncertainty quantification for improving volatile residential load forecasting," *Energy*, vol. 211, Art. no. 119007, 2020, doi: 10.1016/j.energy.2020.119007.

[23] A. Taïk and S. Cherkaoui, "Electrical load forecasting using edge computing and federated learning," in *Proc. IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6, doi: 10.1109/ICC40277.2020.9148937.

[24] M. N. Fekri, K. Grolinger, and S. Mir, "Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks," *International Journal of Electrical Power & Energy Systems*, vol. 137, Art. no. 107669, 2022, doi: 10.1016/j.ijepes.2021.107669.

[25] N. Gholizadeh and P. Musilek, "Federated learning with hyperparameter-based clustering for electrical load forecasting," *Internet of Things*, vol. 17, Art. no. 100470, 2022, doi: 10.1016/j.iot.2021.100470.

[26] J. D. Fernández, S. P. Menci, C. M. Lee, A. Rieger, and G. Fridgen, "Privacy-preserving federated learning for residential short-term load forecasting," *Applied Energy*, vol. 326, Art. no. 119915, 2022, doi: 10.1016/j.apenergy.2022.119915.

[27] Y. He, F. Luo, M. Sun, and G. Ranzi, "Privacy-preserving and hierarchically federated framework for short-term residential load forecasting," *IEEE Transactions on Smart Grid*, vol. 14, no. 6, pp. 4409–4423, 2023, doi: 10.1109/TSG.2023.3268633.

[28] Y. Dong, Y. Wang, M. Gama, M. A. Mustafa, G. Deconinck, and X. Huang, "Privacy-preserving distributed learning for residential short-term load forecasting," *IEEE Internet of Things Journal*, vol. 11, no. 9, pp. 16817–16828, 2024, doi: 10.1109/JIOT.2024.3362587.

[29] S. Barja-Martínez, F. Teng, A. Junyent-Ferré, and M. Aragüés-Peñalba, "Personalized federated learning with cost-oriented load forecasting for home energy management systems," *IEEE Transactions on Industry Applications*, vol. 61, no. 1, pp. 1410–1419, 2025, doi: 10.1109/TIA.2024.3462668.

[30] F. Sossan, E. Namor, R. Cherkaoui, and M. Paolone, "Achieving the dispatchability of distribution feeders through prosumers data driven forecasting and model predictive control of electrochemical storage," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 4, pp. 1762–1777, 2016, doi: 10.1109/TSTE.2016.2600103.