



ISSN : 2339 - 1871

BETRIK BESEMAH TEKNOLOGI INFORMASI & KOMPUTER

Editor Office : Pusat Penelitian & Pengabdian Pada Masyarakat
(PPPM) ITPA

Phone : 0857-9716-9578

email : betriktpa@itpa.ac.id

Evaluasi Komparatif *Ridge*, *SVR*, *Extratrees* Untuk Prediksi Konsumsi Energi *IoT Smarthome* Dengan *lag-1*

Sri Hartati^{1*}, Rusidi², Erlita Sulistiati³

STEM,Manajemen Informatika , Universitas Mahakarya Asia,Baturaja,Indonesia^{1*}

STEM,Teknik Informatika , Universitas Mahakarya Asia,Baturaja,Indonesia²

STEM,Teknik Informatika , Universitas Mahakarya Asia,Jakarta,Indonesia³

Sur-el : *hartatiakmi1984@gmail.com¹, rusidi081@gmail.com², erlita14@gmail.com³

Penulis Korespondensi: Sri Hartati, hartatiakmi1984@gmail.com

Abstrak: Penelitian ini mengevaluasi secara komparatif Ridge Regression, *Support Vector Regression* (*SVR*, *RBF*), dan *ExtraTrees Regressor* terhadap *baseline lag-1 (persistence)* untuk *short-term load forecasting (STLF)* konsumsi energi *IoT smart home* beresolusi 1 jam. Urgensi riset terletak pada kebutuhan prediksi beban yang akurat guna efisiensi operasional, *peak shaving*, dan *demand response* rumah tangga. Metode yang digunakan menekankan replikabilitas: *resampling* 1 jam, imputasi celah pendek, rekayasa fitur minimalis (fitur kalender dan *lag*), serta pembagian berbasis waktu 70%/15%/15% (train/valid/test). Tiga model dibandingkan dengan pelaporan MAE, RMSE, dan R^2 ; RMSE dijadikan metrik utama karena peka terhadap lonjakan (*spike*). Hasil menunjukkan pada validasi *ExtraTrees* unggul (RMSE 1,012, R^2 0,854) melampaui *baseline lag-1* (RMSE 1,507, R^2 0,677). Namun pada test *baseline lag-1* paling akurat (RMSE 1,457, R^2 0,808, MAE 0,900), sedangkan *ExtraTrees* paling mendekati namun belum melampaui (RMSE 1,482, R^2 0,802, MAE 1,119). *SVR* dan Ridge menurun nyata di test (RMSE > 2,0; R^2 < 0,65). Temuan ini menegaskan kuatnya *persistence* pada horizon satu jam dan perlunya fitur musiman/eksogen serta *penalaan hiperparameter* untuk menurunkan kesalahan besar. Kontribusi praktis riset ini adalah *pipeline Colab* yang ringkas dan replikabel. Pengembangan ke depan meliputi *walk-forward validation*, *model residual* terhadap *lag-1*, penambahan *lag* harian/mingguan dan variabel cuaca, serta uji signifikansi perbedaan error.

Kata kunci : *Baseline lag-1*, *ExtraTrees*, *IoT*, *Prediksi beban*, *Ridge Regression*, *Support Vector Regression (SVR)*, *Smart home*

Abstract : This study presents a comparative evaluation of Ridge Regression, Support Vector Regression (SVR, RBF), and ExtraTrees Regressor against a lag-1 (persistence) baseline for short-term load forecasting (STLF) of IoT smart-home energy consumption at an hourly resolution. The research is motivated by the need for accurate forecasts to enable operational efficiency, peak shaving, and residential demand response. The workflow emphasizes reproducibility: hourly resampling, short-gap imputation, minimalist feature engineering (calendar and lag features), and a time-based split of 70%/15%/15% (train/validation/test). Models are compared using MAE, RMSE, and R^2 , prioritizing RMSE due to its sensitivity to spikes. Results show that on validation ExtraTrees performs best (RMSE 1.012, R^2 0.854) and surpasses the lag-1 baseline (RMSE 1.507, R^2 0.677). However, on the test set the lag-1 baseline is most accurate (RMSE 1.457, R^2 0.808, MAE 0.900), while ExtraTrees is the closest yet does not surpass it (RMSE 1.482, R^2 0.802, MAE 1.119). SVR and Ridge degrade markedly on test (RMSE > 2.0; R^2 < 0.65). These findings highlight the strength of persistence at a one-hour horizon and the need for seasonal/exogenous features and hyperparameter tuning

Received: 22-08-2025 | Accepted: 26-08-2025 | Published Online: 30-08-2025

All author: Sri Hartati, Rusidi, Erlita Sulistiati

to reduce large errors. Our practical contribution is a concise, Colab-based pipeline. Future work includes walk-forward validation, residual modeling over lag-1, adding daily/weekly lags and weather variables, and significance testing of error differences.

Keywords : Baseline lag-1, ExtraTrees, IoT, Load prediction, Ridge Regression, Support Vector Regression (SVR), Smart home

1. PENDAHULUAN

Gelombang transformasi digital beberapa tahun terakhir mengubah cara masyarakat memproduksi, mengakses, dan memaknai data dalam keseharian, termasuk data energi rumah tangga. Data energi kian mudah dikumpulkan dan dianalisis melalui ekosistem *Internet of Things (IoT)* di *smart home*. Dimulai dari *smart meter*, *smart plug*, pengendali HVAC, hingga sensor lingkungan yang memantau konsumsi listrik secara *near real-time*[1]. Di tingkat tata kelola, inisiatif dan kerangka *open energy data* memperkuat akses terhadap data energi yang terstandar dan terdokumentasi dengan baik sehingga penelitian replikabel dapat dilakukan lintas institusi dan lintas negara[2]. Pada saat yang sama, ketika data lapangan sulit diperoleh atau bersifat sensitif, muncul perangkat pembangkit data sintetis yang realistis untuk *smart home* yang memungkinkan ilmuwan data menguji pendekatan pemodelan di skenario terkontrol tanpa melanggar privasi.

Kebutuhan akan prediksi konsumsi listrik jangka pendek *short-term load forecasting (STLF)* muncul sebagai isu strategis dalam pengelolaan energi rumah tangga dan jaringan distribusi karena berkaitan langsung dengan efisiensi biaya, *demand response*, dan mitigasi *peak shaving* yang bekerja pada horizon jam ke depan[3]. Dalam bidang ini, keluarga model *deep learning* berbasis *long short-term memory (LSTM)* banyak diadopsi untuk menangkap non-linearitas dan pola temporal beresolusi tinggi, memperlihatkan kinerja yang kompetitif pada beban listrik jangka pendek di beragam pengaturan data[4]. Tinjauan dan studi terapan juga menegaskan bahwa rancangan dan penerapan STLF yang tepat termasuk pemilihan horizon, fitur, dan protokol validasi memiliki dampak langsung pada keandalan operasi energi dari level utilitas hingga bangunan.

Di luar arsitektur model, kualitas dan strategi fitur menjadi penentu utama keberhasilan STLF: pendekatan *feature mining* bertahap pada deret beban membantu mengekstrak pola laten yang tidak tersaji eksplisit pada sinyal asli, sehingga model memperoleh *inductive bias* yang tepat untuk memprediksi konsumsi pada horizon pendek. Pada skala agregat, alternatif yang lebih *physics-informed* seperti *Dynamic Mode Decomposition (DMD)* juga menunjukkan kemampuan memetakan dinamika beban dari jangka pendek hingga menengah dengan biaya komputasi relatif efisien, sehingga layak dipertimbangkan sebagai pembandingan metodologis. Selain itu, pendekatan *hybrid* modern yang mengombinasikan berbagai mekanisme pemodelan menawarkan ruang perbaikan akurasi, terutama ketika kompleksitas pola beban menuntut kapasitas representasi yang lebih kaya daripada model tunggal[5].

Integrasi modul peramalan ke dalam penjadwalan operasi energi menutup *loop* dari analitik menuju aksi: prakiraan yang lebih akurat terbukti meningkatkan kualitas keputusan operasional sehingga efisiensi

dan keandalan sistem energi ikut terdongkrak[6]. Di sisi keandalan model jangka panjang, kebutuhan akan *explainability* dan deteksi *concept drift* menjadi sama pentingnya dengan akurasi organisasi perlu memahami mengapa model memberikan prediksi tertentu serta kapan perubahan distribusi data membuat kinerja model merosot agar intervensi (penalaan ulang, *retraining*, atau pembaruan fitur) bisa dilakukan tepat waktu [7]. Dalam lanskap metode, studi yang memadukan rekayasa sinyal dan pembelajaran statistik modern juga terus bermunculan memanfaatkan transformasi domain dan *regularization* untuk menekan galat sambil menjaga efisiensi komputasi sebagai alternatif yang lebih ringan dibanding arsitektur *deep learning* besar [8].

Pada sisi arsitektur jaringan, kombinasi *convolutional neural network (CNN)* dan LSTM berlapis kerap dipakai untuk menangkap pola musiman harian sekaligus perubahan transien yang cepat pada deret beban, memperkuat kemampuan model menggeneralisasi lintas hari dan minggu [9]. Kerangka *hybrid deep learning* bertingkat memperlihatkan kinerja kompetitif pada berbagai horizon dan tingkat agregasi permintaan, terutama ketika fitur kalender dan *lag* diintegrasikan dengan sinyal historis [10]. Di luar beban murni, studi pada keluaran *photovoltaic (PV)* terdistribusi menegaskan pentingnya konteks kalender-meteorologi dan *feature engineering* yang memperhatikan interaksi cuaca terhadap profil energi, sehingga pembelajaran dari ranah PV dapat menginspirasi praktik peramalan beban residensial [11].

Dimensi manusia tak kalah penting, bagaimana pengguna memiliki persepsi dan memanfaatkan data energi memengaruhi tingkat keterlibatan mereka dalam praktik hemat energi, sehingga desain antarmuka, transparansi, dan kejelasan rekomendasi menjadi bagian integral dari solusi *smart home* berbasis data [12]. Dalam konteks Indonesia, diterimanya layanan digital energi misalnya aplikasi utilitas untuk pembacaan meter dan transaksi menggambarkan kesiapan pengguna berinteraksi dengan ekosistem analitik energi. Temuan ini memperkuat urgensi penelitian yang menghubungkan akurasi prediksi dengan kegunaan praktis di tingkat rumah tangga. Pada tataran terapan, pengembangan aplikasi pemantauan dan deteksi anomali berbasis *IoT* di lingkungan lokal menunjukkan arah evolusi menuju sistem rumah tangga cerdas yang dapat diawasi dan disesuaikan secara *real-time*, sehingga hasil-hasil peramalan dapat diorkestrasi menjadi tindakan operasional[13].

Dari perspektif ilmiah, fokus STLF untuk smart home bukan sekadar mengejar akurasi tertinggi, melainkan merancang protokol evaluasi yang adil, replikabel, dan berguna—meliputi pembagian berbasis waktu, pelaporan multi-metrik (MAE, RMSE, R^2), serta perbandingan ketat dengan baseline sederhana[14]. Tantangan umum meliputi heterogenitas perangkat (meter, plug, sensor) dan kualitas/frekuensi data, missing values yang perlu imputasi hati-hati agar tidak over-smoothing, serta *concept drift* yang menuntut pemantauan dan *explainability*[15]. Dalam praktiknya, baseline lag-1/seasonal-naïve kerap kuat untuk horizon 1 jam pada periode stabil, sehingga keuntungan model modern sering hanya tipis kecuali saat terjadi spike; karena itu RMSE relevan ketika lonjakan mahal secara operasional dan pelaporan MAE–RMSE– R^2 di validation dan test sangat dianjurkan[16]. Di sisi lain, deep learning tidak selalu terbaik untuk semua konteks: model ringan seperti Ridge, SVR (RBF), dan ExtraTrees sering memberi rasio kinerja–kompleksitas yang baik, lebih ramah latency/memory, dan relatif lebih mudah dijelaskan—relevan untuk

integrasi pada edge device[17]. Atas dasar itu, studi ini membandingkan Ridge, SVR, ExtraTrees terhadap baseline lag-1 dengan protokol berbasis waktu dan pelaporan multi-metrik yang dapat diulang.

2. METODOLOGI PENELITIAN

Metodologi penelitian menguraikan rancangan metodologis untuk mengevaluasi *short-term load forecasting* (STLF) konsumsi energi *IoT smart home*. Alur kerja mencakup pemilihan *dataset Electricity consumption dataset of a local energy cooperative* (Portugal, 2022–2023, CC BY 4.0), pra-pemrosesan (sinkronisasi waktu, *resampling* 1 jam, imputasi celah pendek), rekayasa fitur (kalender dan *lag* terutama lag-1 sebagai *baseline*), serta pembagian berbasis waktu (train–valid–test). Tiga model Ridge, SVR (RBF), *ExtraTrees* dibandingkan terhadap *baseline lag-1* memakai metrik MAE, RMSE, R². Seluruh implementasi direplikasi di *Google Colab* dengan penyimpanan otomatis tabel dan grafik hasil.

2.1 Desain dan Metode

Penelitian ini menggunakan desain evaluatif-komparatif berbasis waktu untuk *short-term load forecasting* (STLF) pada data konsumsi energi *IoT smart home*. Protokol mengutamakan keterulangan (*reproducibility*) dan keadilan evaluasi melalui:

1. Pra-proses transparan: normalisasi format waktu, *resampling* ke resolusi 1 jam, imputasi celah pendek saja (≤ 3 jam) agar tidak terjadi *over-smoothing*.
2. Rekayasa fitur minimalis namun informatif: fitur kalender (*hour, dayofweek, month, is_weekend*) serta lag-1 (untuk baseline dan fitur), dengan opsi lag-24 dan rolling-24 jika data.
3. Pemisahan berbasis waktu: 70% *train* → 15% *valid* → 15% *test* (urutan kronologis, tanpa *shuffling*) guna mencegah *data leakage* temporal. Split: {'train': 8139, 'valid': 1744, 'test': 1745}.
4. *Baseline* kuat: lag-1 (*persistence*) sebagai tolok ukur minimum yang harus dilampaui model.
5. Pelaporan multi-metrik: MAE, RMSE (metrik utama karena peka *spike*), dan R² pada *valid* dan *test*.

2.2 Data & Akuisisi

Dataset utama adalah *Electricity consumption dataset of a local energy cooperative* (Portugal, 2022–2023, CC BY 4.0)—dua CSV (konsumsi multi-rumah & cuaca) digabung berdasarkan *timestamp* dan kemudian di-*resample* ke 1H. File hasil gabungan diunggah ke Colab (mis. *cooperative_merged.csv*) dan diproses otomatis sesuai skrip. Jumlah kolom konsumsi yang dijumlah adalah 172.

Rumus singkat target:

$$y_t = \sum_{i=1}^m c_{i,t}, y_{t-1} = \sum_{i=1}^m c_{i,t-1}, \dots \quad (1)$$

Keterangan : m = banyak kolom.

2.3 Pra-pemrosesan

Jumlah baris awal rentang tanggal, jumlah, *missing* adalah 11652 sedangkan jumlah sampel efektif setelah fitur adalah 11628, seperti digambarkan pada Gambar 1.

```
Fitur: ['hour', 'dayofweek', 'month', 'is_weekend', 'lag1', 'lag24h', 'roll24h']  
Sampel setelah fitur: 11628
```

Gambar 1. Sampel efektif setelah fitur

1. Parsing waktu & penyamaan tipe *numeric*.
2. *Resampling* 1 jam dan imputasi celah pendek (≤ 3 jam) menggunakan *interpolation* agar kurva tidak terlalu mulus; celah panjang dibiarkan kosong dan terbuang saat pembentukan fitur.
3. Target: agregat konsumsi (penjumlahan beberapa *meter* rumah) pada setiap *timestamp* 1 jam.

2.4 Rekayasa Fitur

Kalender: *hour*, *dayofweek*, *month*, *is_weekend*. Lag: *lag_1* (fitur inti & *baseline*), opsional *lag_24* dan *roll_24* untuk tren harian. Pendekatan *feature mining* bertahap pada deret beban terbukti meningkatkan prediksi STLF [Zha 2024, [18]].

2.5 Model

Seluruh model diimplementasikan via *scikit-learn*

1. *Ridge Regression* — *regresi linear* dengan *L2-regularization* untuk menstabilkan estimasi saat fitur berkorelasi; efektif sebagai *strong linear baseline*
2. *Support Vector Regression* (SVR, RBF) — meminimalkan kesalahan di dalam ϵ -tube sambil memaksimalkan margin; *kernel RBF* memetakan hubungan nonlinier lokal
3. *ExtraTrees Regressor* — *ensemble* pohon keputusan dengan *randomized splits* dan *bagging*; cenderung memiliki varians lebih rendah serta tahan terhadap nonlinieritas
4. *Baseline* (lag-1) *benchmark* wajib di studi peramalan musiman/berketergantungan kuat

2.6 Hipotesis

- 1 H1: Dengan fitur kalender dan *lag* minimalis, Ridge dan SVR (RBF) menurunkan RMSE dan meningkatkan R^2 dibanding *baseline lag-1*, karena mampu menangkap hubungan linear ter-regularisasi (*Ridge*) dan *nonlinear* lokal (SVR) yang tidak tertangkap oleh *persistence*
- 2 H2: *ExtraTrees* berpotensi kompetitif, tetapi tanpa *tuning* mendalam dapat tertinggal pada data beresolusi 1 jam yang relatif mulus dengan *spike* sporadic.

2.7 Rancangan Analisis & Implementasi

1. Muat data \rightarrow *merge* konsumsi & cuaca \rightarrow *set index* waktu danurut kronologis.
2. *Resampling* 1H + imputasi pendek (≤ 3 jam) \rightarrow buang baris dengan *NaN* tersisa pada saat membuat fitur.
3. Bangun fitur: kalender + *lag_1* (+ *lag_24*, *roll_24* jika tersedia).
4. Split waktu: 70% *train*, 15% *valid*, 15% *test*.
5. Latih model Ridge dan SVR dalam *pipeline* dengan *StandardScaler* (*with_mean=False*) untuk kestabilan numerik dan *ExtraTrees* dengan *n_estimators* ≈ 300 , *random_state=42*.

6. Evaluasi: hitung MAE, RMSE, R^2 pada *valid & test*; bandingkan terhadap baseline lag-1 (prediksi = lag_1).
7. Visualisasi: kurva aktual vs prediksi pada *test* dan Permutation Importance (Top-10 fitur) untuk model terbaik; simpan ke outputs/.
8. Kendali mutu: korelasi y_{test} vs lag_1 (indikasi kekuatan baseline), pemeriksaan *drift* sederhana (gap kinerja valid→test), dan memastikan tidak ada kebocoran (lag dihitung dengan shift(1)).

2.8 Alat & Reprodusibilitas

Eksperimen dijalankan pada *Google Colab* (Python 3.x) dengan pustaka: pandas, numpy, matplotlib, scikit-learn. Semua aset (tabel evaluasi, grafik) disimpan otomatis ke direktori outputs.

3. HASIL DAN PEMBAHASAN

3.1 Hasil

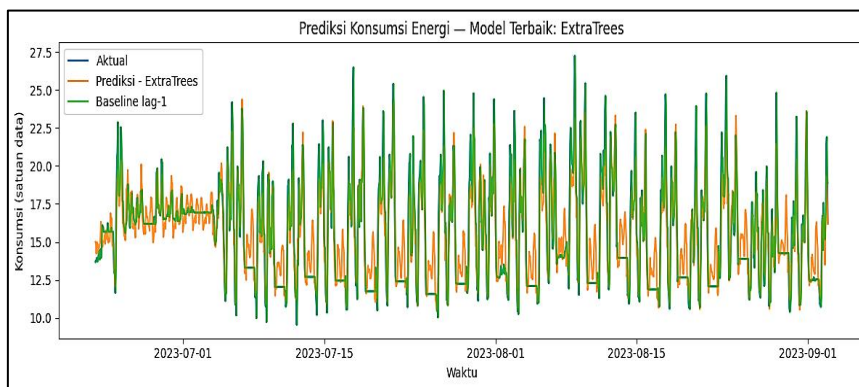
Evaluasi dilakukan pada pembagian waktu 70% (*train*), 15% (*valid*), dan 15% (*test*). Tabel berikut merangkum metrik utama yang diperoleh dari *notebook* Colab (format disederhanakan dari hasil ekspor):

Tabel 1. Hasil evaluasi

<i>index</i>	<i>Model</i>	<i>MAE_</i> <i>Valid</i>	<i>RMSE</i> <i>Valid</i>	<i>R2_</i> <i>Valid</i>	<i>MAE_</i> <i>Test</i>	<i>RMSE</i> <i>Test</i>	<i>R2_</i> <i>Test</i>
0	<i>Baseline (lag-1)</i>	1.061	1.507	0.677	0.9	1.457	0.808
1	<i>ExtraTrees</i>	0.777	1.012	0.854	1.119	1.482	0.802
2	<i>SVR_RBF</i>	0.848	1.187	0.799	1.471	2.023	0.631
3	<i>Ridge</i>	0.877	1.209	0.792	1.557	2.092	0.605

1. Temuan kunci pada *test set*:
 - a. *Baseline lag-1* adalah terbaik (RMSE 1.457; R^2 0.808; MAE 0.900).
 - b. *ExtraTrees* paling mendekati baseline, tetapi masih sedikit lebih buruk: RMSE naik +1,7% (1.482 vs 1.457), R^2 turun -0,006, MAE naik +24,3% (1.119 vs 0.900).
 - c. SVR (RBF) dan Ridge turun signifikan dibanding baseline:
 - 1) SVR: RMSE +38,9% (2.023 vs 1.457), R^2 0,631 (-0,177 vs baseline).
 - 2) Ridge: RMSE +43,6% (2.092 vs 1.457), R^2 0,605 (-0,203 vs baseline).
2. Temuan pada *validation*:
 - a. *ExtraTrees* unggul jelas (RMSE 1.012; R^2 0.854) dan mengalahkan baseline (RMSE 1.507; R^2 0.677) dengan selisih $\approx 33\%$ pada RMSE.
 - b. SVR dan Ridge juga mengalahkan baseline di *validation*, tetapi keunggulan tidak terbawa ke *test*.

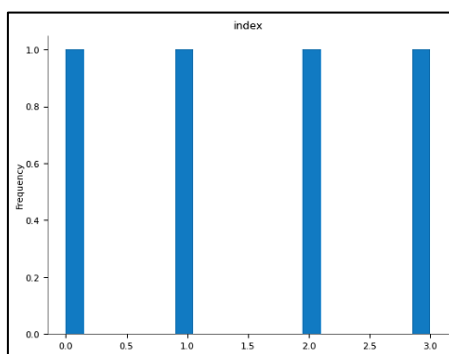
Gambar 2. menunjukkan kurva aktual (biru) dibandingkan prediksi *ExtraTrees* (oranye) dan baseline *lag-1* (hijau) pada horizon 1 jam untuk periode Juni–September 2023. Secara umum, *ExtraTrees* mengikuti kontur harian konsumsi dengan baik, namun cenderung lebih halus dan sedikit *underestimate* pada puncak (*spike*), sedangkan baseline *lag-1* sering menempel ketat pada nilai aktual pada periode stabil tetapi kurang responsif saat terjadi perubahan mendadak. Visual ini melengkapi tabel metrik dengan memperlihatkan bagaimana tiap pendekatan menangani musiman harian dan lonjakan beban.”



Gambar 2. Grafik Prediksi Konsumsi Energi

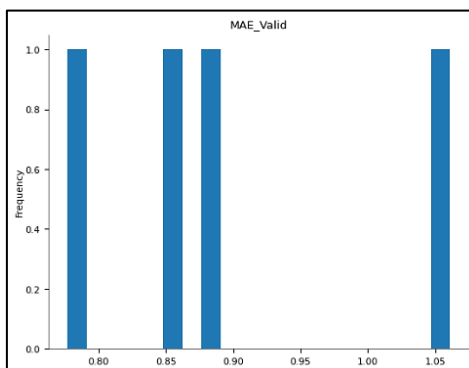
Selanjutnya disajikan distribusi metrik evaluasi pada tahap validasi untuk empat model (*Baseline lag-1*, *ExtraTrees*, *SVR_RBF*, dan *Ridge*). Setiap histogram memperlihatkan sebaran nilai per metrik—MAE, RMSE, dan R^2 —sehingga pembaca dapat melihat rentang dan posisi relatif antar-model secara visual. Karena jumlah model hanya empat, puncak histogram muncul sebagai beberapa batang terpisah; visual ini melengkapi tabel metrik dengan menegaskan bahwa RMSE terkecil dan R^2 tertinggi di validasi dicapai oleh *ExtraTrees*, sementara *baseline* berada di sisi sebaliknya.

Distribusi jumlah entri per model; memastikan setiap model berkontribusi satu observasi pada tiap metrik. Seperti grafik yang ada pada gambar 3.



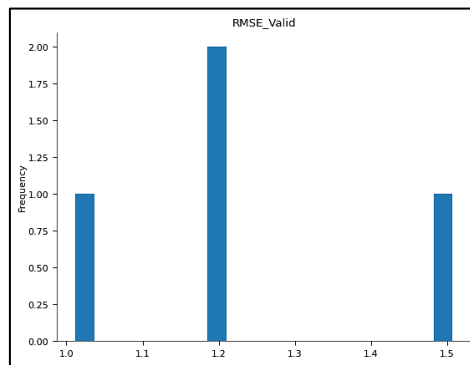
Gambar 3. Index

Distribusi MAE pada validasi; semakin ke kiri semakin baik. Rentang 0,78–1,06, dengan *ExtraTrees* berada pada nilai terendah. Seperti grafik yang ada pada gambar 4.



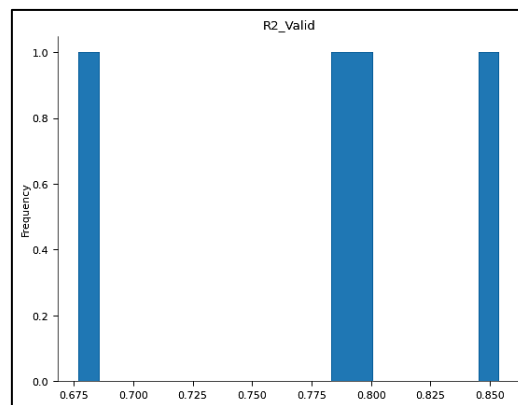
Gambar 4. MAE_Valid

Distribusi RMSE pada validasi; semakin ke kiri semakin baik. Rentang 1,01–1,51, dan *ExtraTrees* menunjukkan RMSE terkecil. Seperti grafik yang ada pada gambar 5.



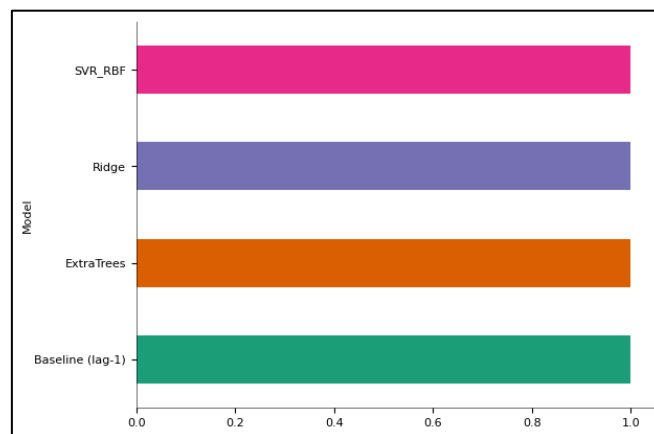
Gambar 5. RMSE_Valid

Distribusi R^2 pada validasi; semakin ke kanan semakin baik. Rentang 0,677–0,854, dengan ExtraTrees mencapai nilai tertinggi. Seperti grafik yang ada pada gambar 6.



Gambar 6. R2_Valid

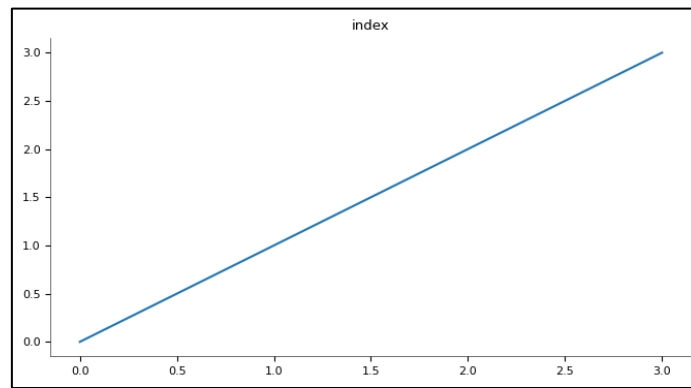
Panel ringkas yang menampilkan empat model yang dievaluasi (panjang batang = 1 menandakan satu observasi per model), berfungsi sebagai legend pendamping. Seperti grafik yang ada pada gambar 7.



Gambar 7. Categorical distributions

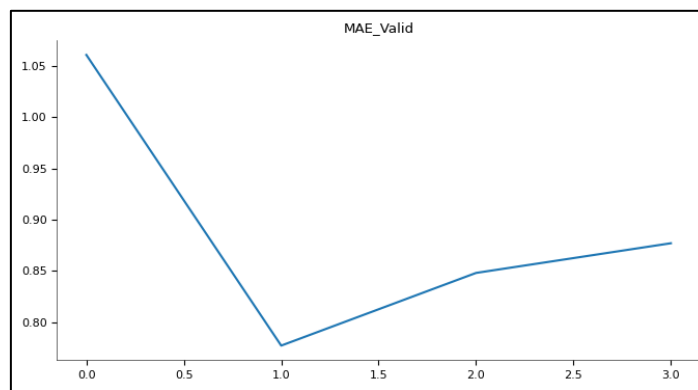
Rangkuman kinerja pada tahap validasi untuk empat model. Metrik yang ditampilkan adalah MAE, RMSE (semakin kecil semakin baik), dan R^2 (semakin besar semakin baik). Sumbu-X 'index' menunjukkan urutan model: 0=Baseline, 1=ExtraTrees, 2=SVR_RBF, 3=Ridge.

Pada gambar 8 Menunjukkan **urutan indeks model (0–3)** sebagai penanda pemetaan yang dipakai pada gambar-gambar berikut. Grafik ini bukan ukuran kinerja, melainkan **acuan identitas model**: 0=Baseline, 1=ExtraTrees, 2=SVR_RBF, 3=Ridge.



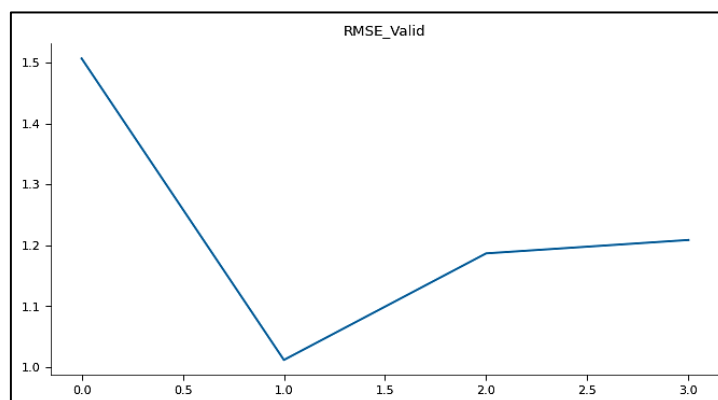
Gambar 8. index (pemetaan model)

Pada gambar 9 Menampilkan nilai MAE pada validasi terhadap indeks model. Terlihat penurunan tajam dari Baseline (~1,06) ke ExtraTrees (~0,78) sebagai yang terbaik, disusul SVR_RBF (~0,85) dan Ridge (~0,88). Ini mengindikasikan ExtraTrees menghasilkan kesalahan rata-rata terendah pada data validasi.



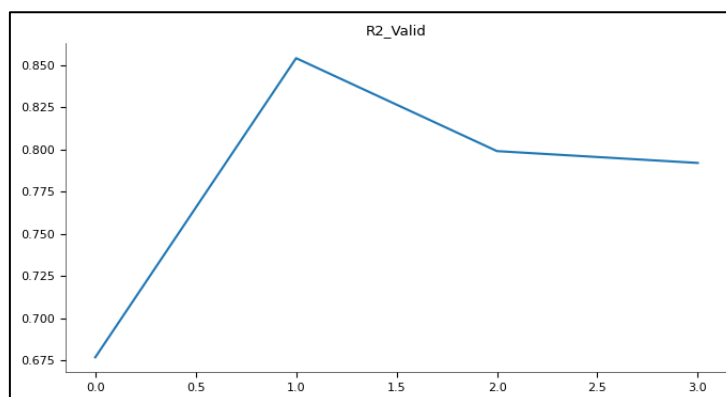
Gambar 9. MAE_Valid per model

Pada gambar 10 Memperlihatkan RMSE pada validasi. ExtraTrees (~1,01) kembali menjadi terkecil dibanding SVR_RBF (~1,19), Ridge (~1,21), dan Baseline (~1,51), menegaskan keunggulannya dalam menekan kesalahan besar (spike) di tahap validasi.



Gambar 10. RMSE_Valid per model

Pada gambar 11 Menggambarkan daya jelaskan (R^2) pada validasi. ExtraTrees (~0,85) mencapai R^2 tertinggi, disusul SVR_RBF (~0,80) dan Ridge (~0,79), sementara Baseline (~0,68) terendah. Hasil ini konsisten dengan temuan MAE/RMSE bahwa ExtraTrees paling baik di validasi.



Gambar 11. R2_Valid per model

3.2 Pembahasan

Penafsiran temuan kuantitatif pada Tabel Hasil dan Gambar perbandingan dengan merujuk langsung pada tujuan riset serta hipotesis H1–H2. Fokus bahasan adalah (i) menjelaskan mengapa baseline *lag-1* masih unggul pada *test set* meski beberapa model lebih baik di *validation*, (ii) mengurai kesenjangan valid→test yang muncul—apakah dipicu *overfitting* ringan atau *concept drift*—dan (iii) memetakan posisi relatif Ridge, SVR (RBF), dan ExtraTrees terhadap metrik utama RMSE beserta pelengkap MAE dan R². Pembahasan juga menyoroti karakteristik tiap model, implikasi praktisnya untuk skenario *smart home*, serta rekomendasi peningkatan (penambahan fitur musiman/eksogen, *tuning* terarah, dan validasi *walk-forward*) agar kinerja dapat melampaui baseline pada horizon satu jam.

Pada *validation*, ExtraTrees terbaik (RMSE 1,012; R² 0,854), melampaui baseline *lag-1* (RMSE 1,507; R² 0,677). Namun pada *test*, baseline justru paling akurat (RMSE 1,457; R² 0,808; MAE 0,900), sementara ExtraTrees hanya mendekati (RMSE 1,482; R² 0,802; MAE 1,119). SVR (RBF) dan Ridge turun tajam di *test* (RMSE > 2,0; R² < 0,65) (lihat Tabel Hasil dan Gambar perbandingan).

1. Kekuatan baseline pada horizon 1 jam.

Konsumsi rumah tangga menunjukkan autokorelasi jam-ke-jam tinggi, sehingga tebakan **persisten (lag-1)** sangat kompetitif pada periode stabil. Inilah sebabnya baseline tetap unggul di *test* meskipun model ML menang di *validation* (Gambar kurva aktual-prediksi).

2. Kesenjangan valid→test (generalization gap).

Semua model melemah saat pindah ke *test*, paling besar pada SVR dan Ridge. Indikasinya: (a) **overfitting ringan** pada jendela validasi; dan/atau (b) **concept drift** (perubahan cuaca/perilaku penghuni) yang tak tertangkap oleh **fitur minimalis** (kalender + beberapa *lag*). Hal ini menjelaskan mengapa keunggulan ExtraTrees di *validation* tidak sepenuhnya terbawa ke *test*.

3. Stabilitas ExtraTrees.

ExtraTrees paling stabil lintas *split*: unggul di *validation* dan di *test* hanya terpaut ~1–2% RMSE dari baseline. Ini menunjukkan ansambel pohon adaptif pada nonlinieritas ringan, tetapi belum cukup menangkap lonjakan (*spike*) tanpa fitur musiman/eksogen atau *tuning* lebih dalam.

4. MAE vs RMSE.

Baseline memiliki MAE terkecil (0,900), menandakan performa baik pada periode “normal”. Sekaligus, *baseline* juga memberi RMSE terkecil di *test*, menandakan model ML belum menurunkan kesalahan besar—tujuan utama ketika RMSE dijadikan metrik utama.

5. Implikasi dan arah peningkatan (ringkas).

Tambahkan lag musiman (*lag_2...lag_6, lag_24, lag_168*), fitur siklik (*sin/cos jam & hari*), indikator weekend/holiday, dan variabel eksogen (cuaca) bila tersedia; lakukan *tuning* terarah (SVR: *C, γ, ε*; ExtraTrees: *n_estimators, max_depth, min_samples_leaf, max_features*; Ridge: *alpha*); gunakan walk-forward validation; dan pertimbangkan model residual:

$$e_t = y_t - \hat{y}_t, \quad \hat{y}_t = y_{t-1} + \hat{e}_t \quad \dots\dots (2)$$

Strategi ini sering efektif untuk melampaui *baseline* pada horizon 1 jam.

Pada konfigurasi saat ini, *baseline lag-1* tetap benchmark terkuat di *test*; ExtraTrees adalah kompetitor terdekat namun belum unggul; SVR dan Ridge memerlukan fitur & penalaan tambahan. Temuan ini menegaskan pentingnya protokol berbasis waktu, pelaporan multi-metrik, dan *baseline* persisten dalam evaluasi STLF *IoT smart home*.

4. KESIMPULAN

4.1 Kesimpulan

Penelitian ini mengevaluasi Ridge, SVR (RBF), dan ExtraTrees terhadap *baseline lag-1 (persistence)* untuk tugas short-term load forecasting (STLF) konsumsi energi *IoT smart home* (resolusi 1 jam; dataset koperasi energi Portugal, 2022–2023). Protokol berbasis waktu (train–valid–test berurutan), fitur minimalis (kalender + *lag*), dan pelaporan multi-metrik (MAE, RMSE, R²) diterapkan agar adil dan replikabel.

Hasil utama:

1. Validasi: *ExtraTrees* terbaik (RMSE 1.012, R² 0.854).
2. Test: *baseline lag-1* paling akurat (RMSE 1.457, R² 0.808, MAE 0.900); *ExtraTrees* terdekat namun belum melampaui (RMSE 1.482, R² 0.802, MAE 1.119); SVR dan Ridge turun nyata (RMSE > 2.0, R² < 0.65).

Implikasi terhadap hipotesis: H1 (*Ridge & SVR > baseline*) ditolak pada *test set*; H2 (*ExtraTrees kompetitif namun dapat kalah tanpa tuning*) terdukung sebagian. Secara praktis, pada horizon 1 jam dengan fitur minimalis, *persistence* tetap menjadi benchmark yang kuat; model ML mulai unggul bila tersedia informasi tambahan dan penalaan yang memadai, terutama untuk menurunkan kesalahan besar (*spike*) yang tercermin pada RMSE. Batasan utama: satu domain data, fitur eksogen terbatas, belum ada *hyperparameter tuning* ekstensif, dan evaluasi masih *hold-out* tunggal.

4.2 Saran

Studi lanjutan berikutnya berpeluang menghadirkan model yang stabil, akurat, dan melampaui baseline pada kondisi riil *smart home*, sekaligus tetap efisien dan mudah direplikasi. Agar kinerja konsisten melampaui baseline dan lebih siap operasional, kami menyarankan :

1. Pengayaan fitur: lag_2...lag_6, lag_24, lag_168 (7×24h), fitur siklik (*sin/cos* jam & hari), indikator akhir pekan/libur, serta variabel cuaca (suhu, kelembapan, hujan) jika tersedia.
2. Strategi pemodelan:
 - a. Model residual terhadap lag-1
 - b. Coba boosting (LightGBM/XGBoost/CatBoost) dan regularisasi linear lain (Lasso/Elastic Net).
 - c. Ensemble sederhana (rata-rata berbobot atau stacking) untuk mengatasi *spike*.
3. Penalaan & validasi: *tuning* terarah (SVR: C, γ , ϵ ; ExtraTrees: n_estimators, max_depth, min_samples_leaf, max_features; Ridge: alpha) dan walk-forward validation (rolling origin) agar validasi merefleksikan kondisi *test*.
4. Evaluasi lanjut: uji signifikansi (mis. Diebold–Mariano), serta peramalan probabilistik (quantile/*pinball loss*, CRPS) guna mendukung keputusan peka risiko.
5. Explainability & monitoring: Permutation Importance/SHAP, serta deteksi concept drift dan *data-quality checks* untuk menjadwalkan *re-training*.
6. Generalisasi: uji lintas rumah tangga/area lain dan multi-horizon (1h, 3h, 6h, *day-ahead*); kaji aspek latensi & memori untuk kemungkinan deployment di edge.

5. UCAPAN TERIMA KASIH

Dengan penuh rasa syukur, penulis mengucapkan terima kasih kepada Tim Redaksi Jurnal Betrik (Besemah Teknologi Informasi dan Komputer) Institut Teknologi Pagar Alam yang telah mendukung, memfasilitasi dan memberikan kesempatan untuk penerbitan artikel ini.

DAFTAR RUJUKAN

- [1] F. Monteiro et al., "Electricity consumption dataset of a local energy cooperative," *Data Brief*, vol. 54, Jun. 2024, doi: 10.1016/j.dib.2024.110373.
- [2] D. V. Fernandes and C. S. Silva, "Open Energy Data — A regulatory framework proposal under the Portuguese electric system context," *Energy Policy*, vol. 170, Nov. 2022, doi: 10.1016/j.enpol.2022.113240.
- [3] F. Liu and C. Liang, "Short-term power load forecasting based on AC-BiLSTM model," *Energy Reports*, vol. 11, pp. 1570–1579, Jun. 2024, doi: 10.1016/j.egyr.2024.01.026.
- [4] W. Waheed, Q. Xu, M. Aurangzeb, S. Iqbal, S. H. Dar, and Z. M. S. Elbarbary, "Empowering data-driven load forecasting by leveraging long short-term memory recurrent neural networks," *Heliyon*, vol. 10, no. 24, Dec. 2024, doi: 10.1016/j.heliyon.2024.e40934.
- [5] G. Yan, J. Wang, and M. Thwin, "A new Frontier in electric load forecasting: The LSV/MOPA model optimized by modified orca predation algorithm," *Heliyon*, vol. 10, no. 2, Jan. 2024, doi: 10.1016/j.heliyon.2024.e24183.

- [6] G. Huang, M. Tan, Z. Meng, J. Yan, J. Chen, and Q. Qu, "Optimizing hydropower scheduling through accurate power load prediction: A practical case study," *Heliyon*, vol. 10, no. 7, Apr. 2024, doi: 10.1016/j.heliyon.2024.e28312.
- [7] C. Samarajeewa et al., "An artificial intelligence framework for explainable drift detection in energy forecasting," *Energy and AI*, vol. 17, Sep. 2024, doi: 10.1016/j.egyai.2024.100403.
- [8] Y. Wang, S. Sun, G. Fathi, and M. Eslami, "Improving the Method of Short-term Forecasting of Electric Load in Distribution Networks using Wavelet transform combined with Ridgelet Neural Network Optimized by Self-adapted Kho-Kho Optimization Algorithm," *Heliyon*, vol. 10, no. 7, Apr. 2024, doi: 10.1016/j.heliyon.2024.e28381.
- [9] C. Wang, X. Li, Y. Shi, W. Jiang, Q. Song, and X. Li, "Load forecasting method based on CNN and extended LSTM," *Energy Reports*, vol. 12, pp. 2452–2461, Dec. 2024, doi: 10.1016/j.egy.2024.07.030.
- [10] M. S. Iqbal, M. Adnan, S. E. G. Mohamed, and M. Tariq, "A hybrid deep learning framework for short-term load forecasting with improved data cleansing and preprocessing techniques," *Results in Engineering*, vol. 24, Dec. 2024, doi: 10.1016/j.rineng.2024.103560.
- [11] M. M. Husein, E. J. Gago, B. Hasan, and M. C. Pegalajar, "Towards energy efficiency: A comprehensive review of deep learning-based photovoltaic power forecasting strategies," Jul. 15, 2024, Elsevier Ltd. doi: 10.1016/j.heliyon.2024.e33419.
- [12] R. Prabowo, E. S. Negara, H. Saksono, and P. H. Saksono, "Evaluasi Penggunaan Aplikasi PLN Mobile Kota Palembang menggunakan Technology Acceptance Model (TAM)."
- [13] F. Cellina, E. Lobsiger-Kägi, D. Wemyss, G. Profeta, and P. Granato, "Households in energy transition: Promoting household energy-sufficient routines via app-based peer-to-peer interaction," *Environ Innov Soc Transit*, vol. 52, Sep. 2024, doi: 10.1016/j.eist.2024.100868.
- [14] H. Liu, Z. Li, C. Li, L. Shao, and J. Li, "Research and application of short-term load forecasting based on CEEMDAN-LSTM modeling," *Energy Reports*, vol. 12, pp. 2144–2155, Dec. 2024, doi: 10.1016/j.egy.2024.08.035.
- [15] C. Samarajeewa et al., "An artificial intelligence framework for explainable drift detection in energy forecasting," *Energy and AI*, vol. 17, Sep. 2024, doi: 10.1016/j.egyai.2024.100403.
- [16] M. Wood, E. Ogliari, A. Nespoli, T. Simpkins, and S. Leva, "Day Ahead Electric Load Forecast: A Comprehensive LSTM-EMD Methodology and Several Diverse Case Studies," *Forecasting*, vol. 5, no. 1, pp. 297–314, Mar. 2023, doi: 10.3390/forecast5010016.
- [17] Z. Chen, F. Xiao, F. Guo, and J. Yan, "Interpretable machine learning for building energy management: A state-of-the-art review," Feb. 01, 2023, Elsevier Ltd. doi: 10.1016/j.adapen.2023.100123.
- [18] W. Zha, Y. Ji, and C. Liang, "Short-term load forecasting method based on secondary decomposition and improved hierarchical clustering," *Results in Engineering*, vol. 22, Jun. 2024, doi: 10.1016/j.rineng.2024.101993.