https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

ML-IAM v1.0: Emulating Integrated Assessment Models With Machine Learning

Yen Shin[1], Changyoon Lee[2], Eunsu Kim[2], Junho Myung[2], Kiwoong Park[2], Jiheun Ha[3], Min-Young Choi[4], Bomi Kim[5, 6], Hyun W. Ka[1], Jung-Hun Woo[4], Alice Oh[2], and Haewon McJeon[3]

[1]School of Transdisciplinary Studies, KAIST, Daejeon, Korea
[2]School of Computing, KAIST, Daejeon, Korea
[3]Graduate School of Green Growth and Sustainability, KAIST, Daejeon, Korea
[4]Graduate School of Environmental Studies, Seoul National University, Seoul, Korea
[5]Environmental Planning Institute, Seoul National University, Seoul, Korea
[6]Department of Technology Fusion Engineering, Konkuk University, Seoul, Korea

Correspondence: Alice Oh (alice.oh@kaist.edu) and Haewon McJeon (hmcjeon@kaist.ac.kr)

Abstract. Integrated Assessment Models (IAMs) are essential tools for projecting future environmental variables under diverse environmental, economic, and technological scenarios. However, their computational intensity limits accessibility and application scope. We present ML-IAM v1.0, the first machine learning emulator trained on the IPCC AR6 Scenarios Database to replicate IAM functionality across diverse model families. Our best-performing model, XGBoost, achieves an R^2 of 0.97 against original IAM data, outperforming the more complex models Long Short-Term Memory (LSTM) and Temporal Fusion Transformer (TFT). ML-IAM v1.0 generates results for 2,000 scenarios in 50 seconds and can produce predictions for any IAM family. This enables rapid exploration of climate scenarios, complementing traditional IAMs with efficient, scalable computation.

1 Introduction

Integrated Assessment Models (IAMs) are computational frameworks that integrate economics, energy systems, land use, and climate science to project how different technological, policy, and socioeconomic scenarios affect future greenhouse gas (GHG) emissions and climate outcomes (Riahi et al., 2022). These models serve as essential tools for climate policy analysis, supporting the assessment of mitigation pathways and their economic implications across different policy contexts (Riahi et al., 2017). Despite their central role, IAMs face limitations: they are computationally intensive, with a single scenario taking up to hours to run (for example, GCAM 0.5–8 h, REMIND 3–7 h, MESSAGEix 3 min–4 h (Li et al., 2025)), sometimes failing to converge (IIASA Energy, Climate, and Environment (ECE) Program, 2024). Additionally, outcomes can vary across different IAM families due to their distinct modeling approaches and assumptions (Dekker et al., 2023).

Machine learning (ML) offers a promising pathway to address the computational limitations of IAMs through rapid, data-driven emulation (Watson-Parris et al., 2022; Li et al., 2025). This potential has been demonstrated across multiple domains of Earth system modeling. In atmospheric sciences, abundant observational data from resources like ERA5 reanalysis (Hersbach et al., 2020) and CMIP multi-model ensembles (Eyring et al., 2016) have enabled ML approaches to serve as alternatives

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

or complements for computationally expensive general circulation models (GCMs) and atmosphere-ocean coupled models (AOGCMs). Recent ML-based systems such as PACER (Saleem et al., 2024), GraphCast (Lam et al., 2023), ClimaX (Nguyen et al., 2023), and Aurora (Bodnar et al., 2025) achieve both rapid prediction and high accuracy by learning complex relationships directly from data. Similarly, ML emulators have shown success in land surface modeling, accelerating predictions of forest carbon stocks and fluxes in dynamic global vegetation models (Natel et al., 2025) and emulating crop yield responses to climate scenarios (Crane-Droesch, 2018). While the climate community has long relied on simplified emulators like MAGICC (Meinshausen et al., 2011), FaIR (Smith et al., 2018), and OSCAR (Gasser et al., 2017) to enable rapid scenario exploration at the cost of physical detail, these recent ML advances demonstrate the potential for achieving both speed and accuracy across Earth systems.

However, these climate-focused ML approaches are insufficient for assessing anthropogenic emissions, which are fundamentally driven by socioeconomic factors, industrial production, energy consumption, and policy interventions, rather than atmospheric dynamics (Peters et al., 2017). This domain of Human-Earth system modeling has traditionally been the purview of IAMs, which integrate economic theory with environmental dynamics (Calvin and Bond-Lamberty, 2018). Yet ML application to IAMs has been limited by data scarcity. Unlike how climate models benefit from high-resolution gridded observations covering the globe at subdaily to hourly intervals (Hersbach et al., 2020), IAM data exist only at coarse spatial aggregations (world, continental, or country level) with 5-10 year reporting intervals (Brian C. O'Neill et al., 2014). This limited spatiotemporal resolution has prevented the accumulation of the dense, large-scale training datasets that have driven ML advances in climate science. The 2022 release of the IPCC AR6 Scenarios Database, consolidating 3,131 scenarios from 188 model versions (Peters et al., 2023; Byers et al., 2022), finally provides a workable foundation for ML approaches in the IAM domain, despite these resolution constraints.

Despite recent advances, the application of ML to cross-family IAM emulation remains nascent. To our knowledge, only three studies have attempted related tasks. Takakura et al. (2020) developed emulators for the economic impacts of climate change, reproducing outputs from sectoral bio/physical impact models coupled with economic models—components that can be integrated within IAM workflows but do not emulate IAM mitigation pathways themselves. Li et al. (2025) leverage the AR6 Scenarios Database to develop a generative approach that creates plausible trajectories for IPCC temperature outcome categories (C1-C8). However, their model cannot predict emissions given socioeconomic input conditions, limiting practical applications for policy analysis. Holmes et al. (2024) developed an emulator for GCAM, demonstrating successful predictive emulation but limited to a single IAM family. While these studies advance the field, none address the core challenge of predictive emulation across diverse IAM families.

We present ML-IAM v1.0, the first ML-based emulator trained on the IPCC AR6 Scenarios Database to replicate IAM functionality across diverse model families (Figure 1). ML-IAM predicts GHG emissions and energy production from key socioeconomic variables such as GDP, population growth, and carbon price, learning patterns across over 95 model families rather than fitting to a single modeling framework. Our best-performing model, XGBoost, achieves an R^2 of 0.97 and can project 2,000 scenarios in 50 seconds. We further apply SHAP analysis to interpret how each model learns from inputs. ML-

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

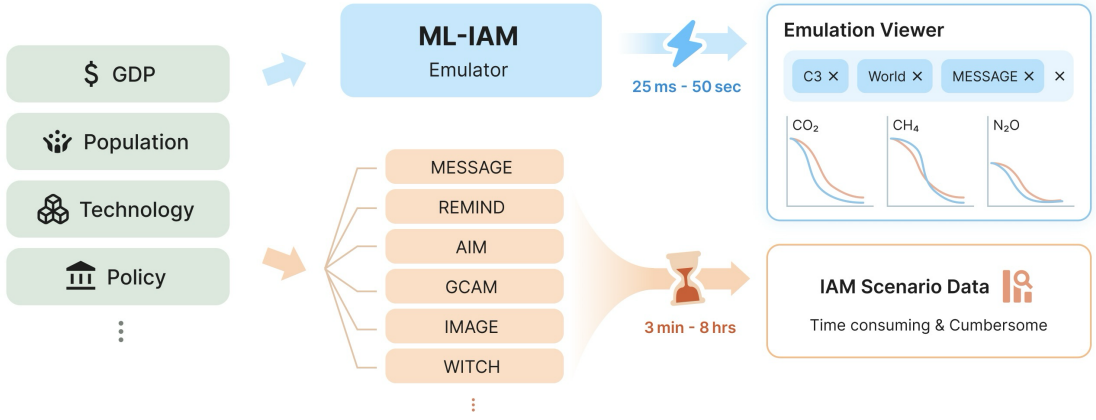


Figure 1. ML-IAM v1.0 emulates multiple IAM families from socioeconomic inputs (GDP, population, technology costs, policy variables), generating emission and energy projections in 25 milliseconds to 50 seconds compared to 3 minutes to 8 hours for traditional IAM runs. An interactive Emulation Viewer enables rapid scenario exploration across regions and model families.

IAM offers a computationally efficient complement to traditional IAMs, making IAM-emulated projections more accessible and customizable.

2 Methods

In this section, we outline the overall model training process, including the dataset used for model training (Section 2.1), the machine learning model selection (Section 2.2), and the training methodology (Section 2.3). Figure 2 provides an overview of the complete ML-IAM framework.

2.1 Dataset for Model Training

For model training, we use the IPCC AR6 Scenarios Database (Byers et al., 2022), which includes 3,131 scenarios from 188 modeling frameworks and over 95 model families, covering global, multi-regional aggregations, and country-level scales. As the largest standardized collection of IAM scenarios to date, its diversity in models, regions, and scenarios enables machine learning models to identify underlying patterns across different IAM frameworks rather than fitting to specific model characteristics. However, variables differ widely across models; therefore, reporting coverage varies dramatically—over half of the variables appear in fewer than 10% of scenarios. This inconsistency necessitates numerous processing decisions to create a coherent training dataset. Our data preparation pipeline involves: (1) systematic classification of input and output variables, (2) filtering data instances to ensure adequate variable coverage, and (3) determination of temporal boundaries and resolution.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

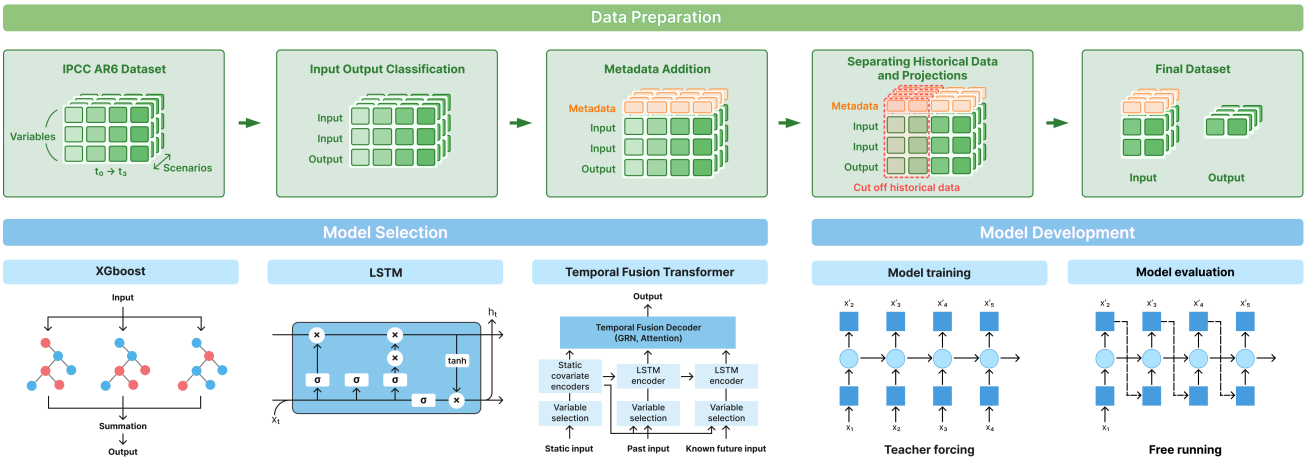


Figure 2. Framework of ML-IAM. **Data Preparation (top):** The pipeline processes the IPCC AR6 Scenarios Database through three stages: input/output variable classification, metadata addition (model family and region identifiers), and temporal boundary determination (filtering pre-base-year data). **Model Selection (left):** Three architectures are evaluated—XGBoost for tabular modeling with lag features, LSTM for sequential dependencies, and Temporal Fusion Transformer (TFT) for attention-based multi-horizon forecasting. **Model Development (right):** All models are trained using teacher forcing (feeding ground-truth previous values) and evaluated autoregressively (using self-generated predictions).

2.1.1 Input Output Classification

Unlike typical ML datasets, IPCC AR6 Scenarios Database variables are not pre-labeled as inputs or outputs, as their roles vary by modeling objective. Exogenous variables such as population, GDP, carbon prices, and capital costs consistently function as model inputs across IAMs. In contrast, endogenous variables such as emissions and energy production are typically calculated by IAMs based on these inputs and internal model dynamics. For ML-IAM, we use only exogenous variables as inputs for broader generalizability.

Exogenous IAM Variables. Proper classification is critical to prevent data leakage. If variables influenced by target outputs are incorrectly used as inputs (e.g., using coal consumption to predict emissions), models can achieve misleadingly high performance by exploiting shallow dependencies rather than causal mechanisms. We engaged domain experts to review the 539 most frequently reported variables (those present in >20% of scenarios), identifying 42 that consistently serve as exogenous drivers across IAMs.

Output Variables. Our implementation focuses on predicting nine key endogenous variables: energy production from six sources (coal, gas, nuclear, oil, solar, wind) and three GHG emissions (CO_2, CH_4, N_2O). However, the framework readily extends to any endogenous variable in the database. The complete list of input and output variables is provided in the Appendix Table A1.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

2.1.2 Metadata Addition

Beyond the exogenous IAM variables, we augment each data instance with model family and region identifiers as categorical inputs.

Model Family. We explicitly encode model family as a categorical input rather than treating model-specific behaviors as noise. This design choice serves two purposes. First, this improves predictive accuracy by capturing each IAM's characteristic response patterns. Second, it enables systematic bias correction: with predictions available across all model families, researchers can apply post-hoc techniques such as mixed-effects modeling to disentangle model-specific biases from underlying physical and economic relationships.

Region. For computational tractability, we treat regions independently, without modeling inter-regional interactions such as trade flows. This assumption is less restrictive given the dataset composition: 70% of data points represent aggregated regions (R10 to World level) where inter-regional dynamics are already embedded in the aggregation, while only 30% are country-level regions. Each region within a scenario thus constitutes a separate data point, allowing us to scale to the full geographic scope of IAM scenarios.

2.1.3 Separating Historical Data and Projections

IAMs initialize projections from a base year that separates the historical period (used for calibration and validation with observed data) from the future projection period (generated through model dynamics). Including historical observations in model training would contaminate the learning process by mixing empirical data with model-generated projections. Despite this critical distinction, the temporal boundary between these periods is not available as metadata in the AR6 Scenarios Database, requiring manual identification through external documentation review. To minimize manual effort, we first filter the dataset to remove records with missing outputs or with more than 60% of the selected inputs missing. For the remaining data, we retain only the projection period by removing pre-base-year historical observations. For models whose base years are publicly documented, we apply their specified cutoffs; otherwise, we conservatively use 2020 as the threshold. The manually collected base year data is available at Zenodo (Shin et al., 2025a).

For temporal resolution, we follow the predominant 5-year temporal intervals, with series reporting at coarser resolutions (e.g., 10-year intervals) treated as having missing intermediate values. All time series are truncated at 2100 for consistency.

The final dataset contains 23,365 time series, each representing distinct model-scenario-region combinations, providing sufficient scale and diversity for training.

2.2 Machine Learning Model Selection

We select ML model (hereafter, "model" is used for ML models, and "IAM" is used for Integrated Assessment Models) architectures based on the time-series and tabular characteristics of the task: XGBoost (Chen and Guestrin, 2016) among tabular models, LSTM (Hochreiter and Schmidhuber, 1997) and Temporal Fusion Transformer (Lim et al., 2021) among time-series models.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

2.2.1 Task Characteristics

IAMs operate through an autoregressive structure, iteratively computing system states at given time intervals from a base year (typically 2010, 2015, or 2020) through 2100—approximately 16-18 sequential steps. Each timestep builds upon the previous state with embedded physical, economic, and technological relationships to determine the subsequent state. This structure creates a unique prediction problem distinct from standard time-series forecasting or tabular regression. While the temporal evolution follows an autoregressive pattern, future states are primarily determined not by historical patterns but by the complex interplay of policy, technology, and socioeconomic covariates. This results in a hybrid prediction task with distinct characteristics:

Short horizon: Series contain only 16-18 timesteps per series, which are substantially shorter than typical deep learning time-series applications.

Short temporal dependencies: Each timestep primarily depends on the immediately preceding state plus current covariates. Models requiring or utilizing long context windows (e.g., 50-100+ timesteps) either cannot function with our 16-18-timestep sequences or waste computational resources processing irrelevant historical information.

Covariate-heavy: The prediction task depends critically on the interactions of hundreds of variables, more than temporal dependencies.

Multi-series heterogeneity: The model must generalize across diverse combinations of models, regions, and scenarios.

Non-periodic: Unlike traditional time-series domains, no seasonal patterns exist; instead, trajectories exhibit monotonic or structural shifts driven by policy interventions and technological transitions.

Multi-target: With potentially hundreds of output variables to predict, multi-target architectures offer significant computational efficiency over parallel single-target models.

This unique combination of characteristics positions IAM emulation at the intersection of tabular and time-series modeling. The task demands approaches that can leverage rich covariate information while respecting the autoregressive constraint. Consequently, we consider both tabular methods applied recursively and time-series architectures designed for short horizons.

2.2.2 Related Work and Applicability

IAM emulation requires both temporal modeling and rich covariate integration, positioning it closest to forecasting applications in economics, which share key characteristics: sparse time series with long horizons, structural breaks from policy interventions, and heavy dependence on covariates (Wang et al., 2021; Chen et al., 2025; Yoon, 2021; Liu et al., 2024; Yang et al., 2024). In this domain, tree-based ensembles (particularly gradient boosting) excel at capturing nonlinear dependencies in high-dimensional data (Anesti et al., 2024; Yoon, 2021; Yang et al., 2024), while LSTMs effectively model temporal dependencies and outperform traditional econometric methods (Zhang et al., 2022; Dong Thi Ngoc et al., 2025; Wang et al., 2021). Transformer architectures such as Temporal Fusion Transformers (TFT) show promise but remain less established (Han et al., 2023; Dong Thi Ngoc et al., 2025).

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

We also examined recent time-series advances but found most unsuitable for IAM characteristics. Long-horizon transformers (Autoformer (Wu et al., 2021), FEDformer (Zhou et al., 2022)) require 96-720 timesteps and exploit periodic patterns absent in our 10-20 step policy-driven sequences. Patch-based models (PatchTST (Nie et al., 2022)) assume temporal self-similarity incompatible with structural policy shifts. Time-series foundation models (TimesFM (Das et al., 2024), Chronos (Ansari et al., 2024), Moirai (Woo et al., 2024)) excel at pattern extrapolation but lack mechanisms for the heterogeneous covariates that drive IAM projections. See Appendix B for detailed model assessment.

2.2.3 Selected Models

We select three models with complementary strengths: XGBoost for nonlinear covariate interactions (Chen and Guestrin, 2016), LSTM for temporal dependencies (Hochreiter and Schmidhuber, 1997), and Temporal Fusion Transformer (TFT) for attention-based multi-horizon forecasting (Lim et al., 2021).

XGBoost follows the proven paradigm from macroeconomic forecasting, efficiently capturing feature interactions in high-dimensional covariate spaces. Its recursive approach with 1-2 lag features suits our limited temporal dependencies while providing interpretable feature attributions.

LSTM naturally handles IAM's autoregressive structure—each timestep depends on the previous state plus current covariates—leveraging its established success in covariate-driven forecasting without requiring the long contexts of more complex sequential models.

TFT extends transformers specifically for multi-variate, multi-horizon forecasting with mixed data types. Its variable selection networks identify relevant features from high-dimensional inputs, static covariate encoders maintain scenario-specific information, and gated networks integrate time-varying and known-future drivers—all aligned with IAM's covariate-heavy nature.

2.3 Model Settings

We configure the three models under consistent configurations for fair comparison, with careful attention to data splitting, temporal modeling strategies, and evaluation procedures.

2.3.1 Experimental Setup

Data Splitting. The dataset is split into training (80%), validation (10%), and testing (10%) sets at the scenario level. Each unique model-scenario-region combination (time series) is assigned entirely to one split, ensuring all time points from a given scenario remain together. This prevents temporal data leakage where models could learn from future observations of scenarios present in the training set.

Evaluation Metrics. Model performance is evaluated using coefficient of determination (R^2) and root mean squared error (RMSE), computed by pooling all test set predictions across scenarios and time points.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

2.3.2 Model Training Configuration

Hyperparameter Optimization. Hyperparameter optimization was conducted for all three models using random search with validation-based early stopping (detailed configurations in Appendix Table C1). For XGBoost, optimization targeted single-target RMSE across individual prediction tasks. Neural network models (LSTM and TFT) employed multi-target RMSE optimization, with TFT using multi-output loss aggregation across all nine target variables simultaneously, ensuring balanced optimization across outputs.

Missing Data Handling. The treatment of missing values differed across architectures due to their inherent capabilities. XGBoost natively handles missing values through its sparsity-aware split finding algorithm, allowing preservation of the original data structure without imputation. Both neural network architectures (LSTM and TFT) required explicit missing data handling through a unified dual strategy: (1) introducing binary missingness indicators for each input variable, and (2) imputing missing values with variable-specific median values calculated from the training set. This approach enables the models to distinguish between actual observations and imputed estimates while maintaining numerical stability during training.

Temporal Architecture. Temporal modeling strategies varied significantly across architectures. XGBoost incorporated lagged features, treating temporal dependencies as additional input features. The LSTM model implemented configurable sequence lengths (1-4 timesteps, optimized via hyperparameter search) with teacher forcing during training, where previous target values were incorporated as additional inputs alongside exogenous features. The TFT model utilized temporal windows with encoder lengths of 2 timesteps and variable prediction lengths, categorizing features into time-varying known reals, time-varying unknown reals, and static categoricals.

2.3.3 Autoregressive Inference

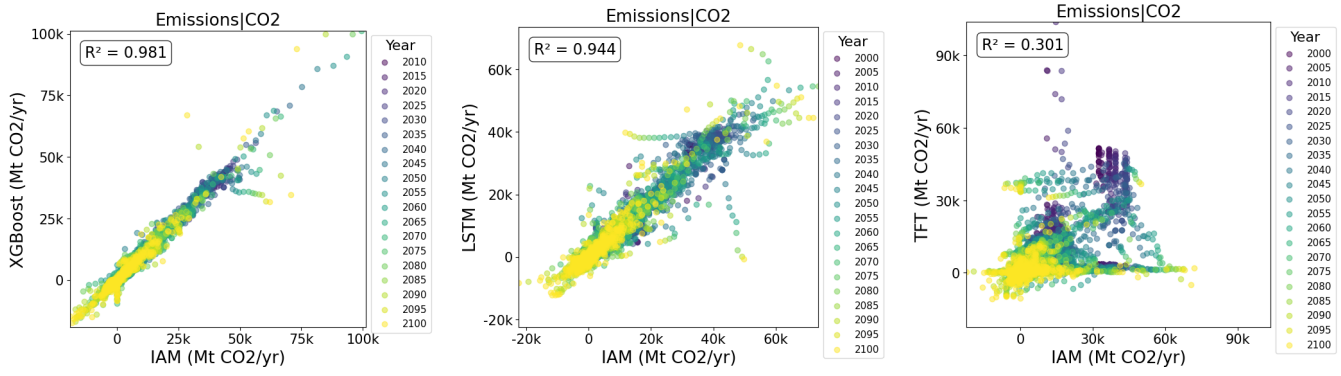
During training, all models use ground-truth values from previous timesteps: LSTM and TFT incorporate these through teacher forcing in their recurrent architectures, while XGBoost uses them as explicit lag features. However, when generating predictions for validation and test scenarios, all models operate autoregressively: after utilizing necessary historical context, the model predicts outputs for the next timestep, then uses its own predictions (rather than ground-truth values) as inputs for subsequent timesteps through 2100. This autoregressive generation process mirrors how traditional IAMs operate, iteratively computing future states based on current conditions and projected drivers.

Further details on hyperparameters and model configuration are provided in Appendix C.

2.4 Model Interpretability Analysis

To understand how models learn to emulate IAM outputs from inputs, we employ SHAP (SHapley Additive exPlanations) analysis (Lundberg and Lee, 2017). SHAP quantifies the contribution of each input feature to individual predictions by computing Shapley values from cooperative game theory, providing a unified measure of feature importance. We apply model-specific SHAP explainers to each architecture: TreeExplainer for XGBoost, which provides exact Shapley values by exploiting the

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026



(a) XGBoost (b) LSTM (c) TFT

Figure 3. ML model predictions versus original IAM CO_2 emissions for (a) XGBoost, (b) LSTM, and (c) TFT. Each point represents a test set prediction, colored by projection year. The variations in year range reflect differences in model architectures.

tree structure for efficient computation, and DeepExplainer for LSTM and TFT, which approximates Shapley values for neural networks using gradient-based attribution methods.

The resulting SHAP summary plots visualize feature importance using beeswarm plots, where features are ranked by their mean absolute SHAP values, and each point represents a single prediction instance. The color gradient represents feature values (red for high, blue for low), enabling interpretation of both the magnitude and directionality of feature effects on predictions.

3 Results

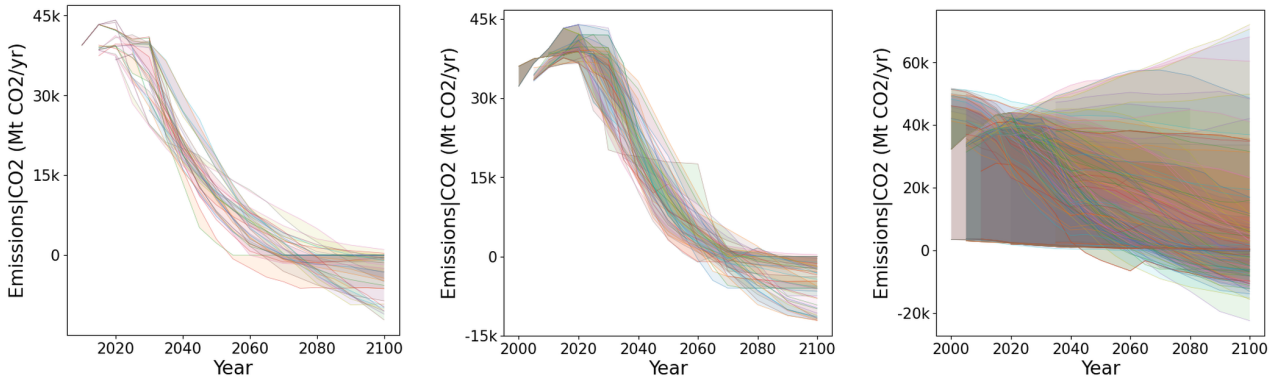
This section presents the performance evaluation of the three ML models in emulating IAM outputs (Section 3.1) and examines model interpretability through feature importance analysis (Section 3.2).

3.1 Performance Evaluation

Figure 3 reveals substantial performance differences across models. XGBoost achieves the highest accuracy ($R^2 = 0.981$) for CO_2 emissions with close alignment across all emission levels and time periods, followed by LSTM ($R^2 = 0.944$), which shows good correlation but increased variance. Despite extensive tuning, TFT performs poorly ($R^2 = 0.301$) with high prediction errors. This performance hierarchy holds consistently across all target variables (Figures S1–S3 in Supplementary Information). The models begin at different years due to varying temporal context requirements. LSTM starts at 2000 with optimal performance using no historical context, as its architecture inherently encodes sequence information. XGBoost and TFT start at 2010 and 2015, requiring 2 and 3 steps of context, respectively.

Beyond aggregated accuracy metrics, temporal trajectories of individual scenarios reveal how well each emulator (ML model) captures emission pathways dynamics. Figure 4 shows CO_2 emission trajectories for C3 scenarios (limit warming to 2°C (>67%)). XGBoost tracks IAM projections closely throughout the forecast horizon, while LSTM shows moderate

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026



(a) XGBoost (b) LSTM (c) TFT

Figure 4. CO_2 emission trajectories for C3: limit warming to 2°C (>67%) scenarios in the World region, comparing (a) XGBoost, (b) LSTM, and (c) TFT predictions against original IAM projections. Shaded areas between model predictions and IAM outputs indicate emulation error, with individual scenario trajectories shown as colored lines. Varying trajectory starting points reflect different base years across IAMs. XGBoost predictions exclude the first 10 years of each trajectory due to lag feature requirements.

deviations and TFT exhibits substantial divergence. Trajectories for other outputs are provided in Figures S4–S6 in Supplementary Information. To make the emulator more accessible and interpretable, we provide an interactive Emulation Viewer (https://mliam.dev/) for exploring emulation results across scenarios, regions, and model families (screenshots in Appendix Figure E1).

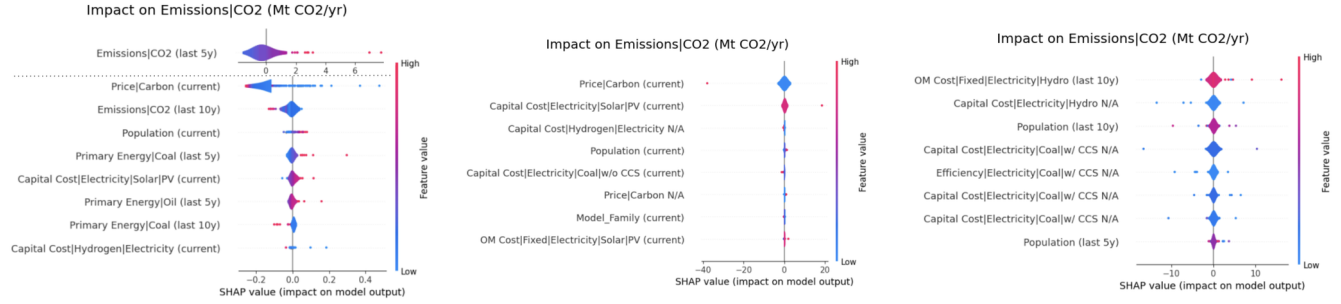
XGBoost's superior performance of average $R^2 = 0.97$ reflects the fundamental nature of IAM emulation: a covariate-heavy prediction task with short temporal sequences rather than complex long-term dependencies. Tree ensemble methods excel at handling the heterogeneous features and multi-source data that characterize IAM scenarios, whereas LSTM and TFT are optimized for problems requiring deep temporal pattern recognition.

Where traditional IAM runs can take hours and may fail to converge to a unique solution, ML-IAM with XGBoost generates 2,000 scenarios in 50 seconds with guaranteed convergence, and can generate predictions for any IAM family.

3.2 Model Interpretability

We interpret how each model learned from the inputs using SHAP analysis(Section 2.4). Figure 5 shows feature importance rankings for CO_2 emissions prediction across the three models. An interesting pattern is that all three models learn to use technology-specific variables and their missingness indicators as important features. This likely occurs because the categorical model family variable only provides labels without conveying characteristics of the model families or their vintages; the models appear to use these technology variables as proxies, as newer IAM versions tend to include technologies (e.g., hydrogen, geothermal) absent in older versions.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026



(a) XGBoost (b) LSTM (c) TFT

Figure 5. Feature importance analysis using SHAP values for CO_2 emissions prediction across (a) XGBoost, (b) LSTM, and (c) TFT models. Features are ranked by importance from top to bottom. In XGBoost, the top feature is shown with a different x-axis scale due to its substantially larger SHAP values. The horizontal position shows impact: points on the right push emissions higher, points on the left push emissions lower. Colors show the feature value itself: red/pink indicates high values, blue indicates low values. The width of each distribution shows how consistent the feature's impact is—narrow indicates consistent, wide indicates that the feature's impact varies depending on the context. "Current" refers to the prediction year, "last 5y" to 5 years prior, "last 10y" to 10 years prior.

XGBoost explicitly uses prior CO_2 emissions as lag features, unlike the time series models. Consequently, emissions from 5 and 10 years prior appear as dominant features. These two lags exhibit opposing effects—emissions from 5 years prior positively correlate with predictions, while emissions from 10 years prior negatively correlate—allowing the model to capture emission trends. Population and primary energy from the last 5 years emerge as secondary drivers.

LSTM shows all influential features from the current year, as this model was trained without explicit context—the optimal configuration for sequence length. The model's recurrent architecture still encodes temporal information internally. Carbon pricing dominates predictions, followed by technology costs for solar PV and hydrogen electricity, indicating the model learns through contemporaneous policy and technology signals.

TFT heavily weights seemingly arbitrary operational and capital cost variables from 10 years prior rather than fundamental emission drivers, explaining its poor performance. This suggests the model struggles to identify meaningful signals in the heterogeneous IAM feature space.

These patterns offer insights for future emulation development: the consistent importance of population, policy variables (carbon pricing), and technology costs across successful models suggests these are essential inputs, while the prevalence of missingness indicators as proxies highlights the need for explicit encoding of IAM characteristics.

4 Discussion

Our results establish ML-IAM as a reliable emulator of IAM data (R^2 = 0.970 with XGBoost). Reducing runtime from hours to seconds, ML-IAM transforms computationally intensive IAM projections into rapid scenario analysis, opening new possibilities for climate policy research and decision-making.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

The speedup from hours to seconds fundamentally changes how IAMs can be used in research and policy analysis. This efficiency enables previously impractical applications: controlled experiments varying single parameters while holding others constant, comprehensive sensitivity analysis across thousands of scenario variations, uncertainty quantification through Monte Carlo sampling, and interactive policy exploration with real-time feedback. Researchers can now optimize for multiple targets simultaneously—such as achieving specific temperature goals while maximizing sustainable development outcomes—through grid searches across millions of parameter combinations that would be computationally prohibitive with traditional IAMs.

Beyond computational advantages, ML-IAM addresses critical gaps in IAM accessibility and coverage. Many regions and research contexts lack comprehensive IAM results, forcing researchers to approximate using available scenarios that may not match their specific needs. ML-IAM's training across diverse model families enables the generation of model-agnostic scenarios for underrepresented regions and parameter combinations. Future extensions could enable ML-IAM to learn specific dynamics from individual models—such as COVID-19 impacts or rapid technological transitions captured by some IAMs but not others—and propagate these patterns across model families, potentially enriching the scenario landscape beyond what any single IAM provides.

The interactive accessibility of ML-IAM particularly benefits IAM data consumers across adjacent fields—land use, water resources, biodiversity, air pollution—and policymakers conducting preliminary scenario screening. Rather than selecting approximate matches from existing databases, these users can generate tailored scenarios matching their analytical needs. The substantial citation count of the AR6 Scenarios Database (over 200 citations) demonstrates the broad demand for such scenario data across disciplines, highlighting the potential impact of making IAM-emulated projections more accessible and customizable.

5 Conclusions

ML-IAM v1.0 provides a fast alternative to traditional IAMs, generating 2,000 scenarios in 50 seconds with $R^2 = 0.97$ accuracy. The emulator works across different IAM families and enables applications that require many scenario runs.

However, several limitations constrain ML-IAM's current scope. Our current ML-IAM treats regions independently, omitting inter-regional interactions such as trade flows and grid connections. Incorporating regional connectivity through architectures like graph neural networks could address this gap (Kipf, 2016). Additionally, our framework uses a fixed input variable set, requiring retraining for modifications. This limits adaptability when data availability varies across models or regions.

While ML-IAM itself does not remove model biases, it establishes the foundation for bias correction with mixed-effects modeling (Bates et al., 2015; Sigrist, 2022; Hajjem et al., 2014). By generating predictions across all model families for any given scenario—including those originally run with only a subset of models—ML-IAM creates the complete cross-family datasets necessary for mixed-effects modeling. Such statistical methods require balanced representation across models, which is often lacking in the original AR6 database, where scenarios may have sparse model coverage. Future work could apply mixed-effects models to these predictions to separate scenario effects from model biases, ultimately producing model-agnostic projections. Additional improvements to emulation could come from Physics-Informed Neural Networks (PINNs), which show

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

promise in economic forecasting (Alonso et al., 2023; Rani and Verma, 2025) and align naturally with IAMs' physics-based foundations (Raissi et al., 2019; Cuomo et al., 2022).

Beyond emulation, however, realizing machine learning's broader potential in climate policy requires addressing fundamental data limitations. We envision a future where ML learns from real-world observations rather than merely replicating IAM-generated outputs, enabling applications like nowcasting and near-term projections grounded in empirical data.

Three critical data gaps currently prevent this evolution. First, systematic reporting of input variables remains incomplete—over half appear in fewer than 10% of scenarios, and crucial policy inputs like carbon phase-out dates and technology subsidies are rarely included, as data aggregators primarily focus on output variables. Second, the field relies mostly on IAM-simulated data rather than real observations. The limited historical data (typically 1990-2020) cannot provide sufficient real-world grounding for ML models to learn patterns independent of IAM assumptions. Third, coarse temporal resolution (5-year intervals) and limited spatial resolution restrict the training data volume needed for complex architectures.

Addressing these challenges requires sustained, coordinated effort from the IAM community—improving variable reporting standards and expanding temporal-spatial coverage. As these foundations strengthen, machine learning can evolve from a computational tool into an independent assessment framework that complements conventional IAMs and accelerates progress through novel insights. ML-IAM represents a first step toward this vision, demonstrating how collaboration between modelers, data providers, and machine learning practitioners can advance climate policy research.

Code and data availability. The source code for ML-IAM v1.0 is permanently archived on Zenodo at https://doi.org/10.5281/zenodo.17390678 (Shin et al., 2025b). The supporting data files (base year mappings and input/output variable classifications) are archived separately at https://doi.org/10.5281/zenodo.17390113 (Shin et al., 2025a). The code is also available on GitHub at https://github.com/YenShin1891/ml-iam. The IPCC AR6 scenarios Database is available at https://doi.org/10.5281/zenodo.7197970 (Byers et al., 2022).

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Appendix A: Data and Preprocessing

Table A1. Variables of ML-IAM

Inputs			
1	Capital Cost|Electricity|Biomass|w/ CCS	23	Lifetime|Electricity|Geothermal
2	Capital Cost|Electricity|Biomass|w/o CCS	24	Lifetime|Electricity|Hydro
3	Capital Cost|Electricity|Coal|w/ CCS	25	Lifetime|Electricity|Nuclear
4	Capital Cost|Electricity|Coal|w/o CCS	26	Lifetime|Electricity|Solar|PV
5	Capital Cost|Electricity|Gas|w/ CCS	27	OM Cost|Fixed|Electricity|Biomass|w/ CCS
6	Capital Cost|Electricity|Gas|w/o CCS	28	OM Cost|Fixed|Electricity|Biomass|w/o CCS
7	Capital Cost|Electricity|Geothermal	29	OM Cost|Fixed|Electricity|Coal|w/ CCS
8	Capital Cost|Electricity|Hydro	30	OM Cost|Fixed|Electricity|Coal|w/o CCS
9	Capital Cost|Electricity|Nuclear	31	OM Cost|Fixed|Electricity|Gas|w/ CCS
10	Capital Cost|Electricity|Solar|CSP	32	OM Cost|Fixed|Electricity|Gas|w/o CCS
11	Capital Cost|Electricity|Solar|PV	33	OM Cost|Fixed|Electricity|Geothermal
12	Capital Cost|Electricity|Wind|Offshore	34	OM Cost|Fixed|Electricity|Hydro
13	Capital Cost|Electricity|Wind|Onshore	35	OM Cost|Fixed|Electricity|Nuclear
14	Capital Cost|Hydrogen|Electricity	36	OM Cost|Fixed|Electricity|Solar|CSP
15	Efficiency|Electricity|Biomass|w/ CCS	37	OM Cost|Fixed|Electricity|Solar|PV
16	Efficiency|Electricity|Biomass|w/o CCS	38	OM Cost|Fixed|Electricity|Wind|Offshore
17	Efficiency|Electricity|Coal|w/ CCS	39	OM Cost|Fixed|Electricity|Wind|Onshore
18	Efficiency|Electricity|Coal|w/o CCS	40	Population
19	Efficiency|Electricity|Gas|w/ CCS	41	Price|Carbon
20	Efficiency|Electricity|Gas|w/o CCS	42	Yield|Cereal
21	GDP|MER	43	Model Family
22	GDP|PPP	44	Region
Outputs			
1	Primary Energy|Coal	6	Primary Energy|Nuclear
2	Primary Energy|Gas	7	Emissions|CO2
3	Primary Energy|Oil	8	Emissions|CH4
4	Primary Energy|Solar	9	Emissions|N2O
5	Primary Energy|Wind		

Abbreviations: CCS = Carbon Capture and Storage; OM = Operations & Maintenance; MER = Market Exchange Rate; PPP = Purchasing Power Parity.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Appendix B: Machine Learning Model Selection

We review models from analogous domains (Section B1) and assess recent time-series architectures (Section B2) to justify our model selection.

B1 Evidence from Macroeconomic Forecasting

The task characteristics described in § 2.2.1 largely stem from the nature of macroeconomic variables. Consequently, macroeconomic forecasting for variables such as GDP, unemployment, and inflation shares more structural similarities with IAM emulation than IAM-adjacent fields like climate modeling. These tasks are typically constrained by short and sparse time series per entity (monthly to yearly intervals), long forecasting horizons (spanning decades), and a strong dependence on heterogeneous covariates (ranging from dozens to over 40 input variables) (Wang et al., 2021; Chen et al., 2025; Yoon, 2021; Liu et al., 2024; Yang et al., 2024). This alignment makes macroeconomic forecasting literature particularly informative for IAM emulation.

Tree-based ensemble models, including gradient boosting models such as XGBoost, are among the best-performing models for tabular data in real-world applications (Shwartz-Ziv and Armon, 2022) and are widely adopted in macroeconomic forecasting. These approaches effectively capture nonlinear dependencies and perform well in high-dimensional, structured data (Anesti et al., 2024; Chen et al., 2025; Yoon, 2021; Richardson et al., 2018; Yang et al., 2024; Goulet Coulombe, 2024).

Recurrent neural networks, particularly **Long Short-Term Memory (LSTM)** architectures and their hybrid variants, have also been prominent in economic forecasting. LSTM-based architectures frequently outperform traditional econometric models by modeling temporal dependencies and nonlinearities in economic cycles (Zhang et al., 2022; Dong Thi Ngoc et al., 2025; Dauphin et al., 2022; Wang et al., 2021; Oancea and Simionescu, 2024).

Some work also explores more advanced architectures such as **Temporal Fusion Transformers (TFT)** and other **attention-based models** (Han et al., 2023; Dong Thi Ngoc et al., 2025), but their application still remains limited.

Interestingly, some studies report competitive results from nonparametric methods such as **K-Nearest Neighbors (KNN)**, which leverage data self-similarity for short-term forecasts (Maccarrone et al., 2021). However, their predictive power deteriorates over longer horizons or when extrapolating beyond observed regimes.

B2 Recent Time-Series Forecasting Models

While macroeconomic forecasting validates certain model families, we also examined recent advances in time-series modeling to assess their applicability. Most state-of-the-art architectures, however, are designed for fundamentally different characteristics than those present in IAM data.

Long-horizon transformer models (Autoformer (Wu et al., 2021), FEDformer (Zhou et al., 2022), Crossformer (Zhang and Yan, 2023)) are explicitly engineered to exploit periodic structures across hundreds of timesteps. Autoformer's autocorrelation mechanisms and FEDformer's frequency-domain decomposition target seasonal patterns and long-range dependen-

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

cies—neither of which exist in IAM's 10-20 timestep sequences driven by policy interventions rather than cyclical dynamics. These models require context lengths (typically 96-720 steps) that far exceed IAM's temporal resolution.

Patch-based models (PatchTST (Nie et al., 2022)) decompose time series into local patches to capture recurring motifs, achieving strong performance on benchmarks with regular patterns (electricity demand, traffic flow, retail sales). However, this approach assumes local temporal similarity that does not hold for IAM projections, where trajectories exhibit structural shifts driven by discrete policy choices and technological transitions rather than continuous, self-similar patterns.

Time-series foundation models (TimesFM (Das et al., 2024), Chronos (Ansari et al., 2024), Moirai (Woo et al., 2024), Sundial (Liu et al., 2025)) have demonstrated impressive zero-shot forecasting capabilities by training on diverse temporal datasets. However, these models fundamentally treat all inputs as temporal tokens, using positional encodings and autoregressive generation. This architecture lacks explicit mechanisms to integrate the heterogeneous, domain-specific covariates (carbon prices, technology costs, policy stringency, socioeconomic drivers) that are the primary determinants of IAM outcomes. While these models excel when historical patterns are informative of future trajectories, IAM projections are driven by scenario assumptions about future conditions rather than extrapolation from past trends.

Appendix C: Model Architectures and Implementation

C1 Optimization and Loss Function

Hyperparameter optimization employed distinct strategies tailored to each architecture's characteristics. XGBoost used randomized search with 50 iterations across parameters, including learning rates (0.01-0.3), maximum depths (3-10), and regularization terms, optimizing individual RMSE for each target variable separately. LSTM optimization explored 48 parameter combinations using random parameter sampling, searching across hidden sizes (64-128), layer counts (2-3), dropout rates (0.1-0.2), sequence lengths (1-4), and dense layer configurations, with early stopping based on validation loss after 3 epochs of no improvement. TFT employed a similar random search methodology targeting transformer-specific parameters, including attention head counts, LSTM layer depths, and hidden dimensions, utilizing a multi-output loss aggregation framework to aggregate RMSE across all nine target variables simultaneously during optimization.

C2 Missing Data Handling

Missing value treatment reflected fundamental architectural differences in data processing capabilities. XGBoost leveraged its native sparsity-aware split finding algorithm (Chen and Guestrin, 2016), automatically directing missing values down optimal tree branches without requiring explicit imputation, preserving the original sparse data structure. Neural network architectures necessitated comprehensive preprocessing pipelines: both LSTM and TFT implementations employed dual-strategy missing data handling consisting of (1) systematic creation of binary missingness indicator variables (suffixed "_is_missing") for each feature, enabling models to learn missingness patterns, and (2) median imputation using training set statistics applied consistently across validation and test splits. The LSTM dataset constructor applied categorical encoding (converting to category

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Table C1. Hyperparameter Search Spaces and Optimization Configurations

Category	XGBoost	LSTM	TFT
Architecture Parameters			
Hidden/Tree Size	max_depth: [3, 4, 5, 6, 7, 8, 9, 10]	hidden_size: [64, 128]	hidden_size: [16, 32, 64]
Layers/Estimators	n_estimators: [50, 100, 200, 300]	num_layers: [2, 3]	lstm_layers: [1, 2]
Regularization	gamma: [0, 0.1, 0.2]	dropout: [0.1, 0.2]	dropout: [0.1, 0.2, 0.3]
	reg_alpha: [0, 0.01, 0.1]	dense_dropout: [0.0, 0.1]	
	reg_lambda: [1, 1.5, 2]	dense_hidden_size: [64, 128]	
Training Parameters			
Learning Rate	learning_rate: [0.01, 0.05, 0.1, 0.2, 0.3]	learning_rate: [0.01, 0.02]	learning_rate: [0.001, 0.01, 0.1]
Batch Size	–	batch_size: [64, 128]	batch_size: [64]
Weight Decay	–	weight_decay: [0.0, 1e-5]	–
Temporal Parameters			
Sequence Length	–	sequence_length: [1, 2, 3, 4]	encoder_length: [2]
Context Window	Lagged features (2 timesteps)	Variable (1-4 timesteps)	max_prediction_length: [13]
Optimization Configuration			
Search Method	RandomizedSearchCV	ParameterSampler	ParameterSampler
Search Iterations	50	48	20
CV Folds	5-fold	Validation split	Validation split
Early Stopping	–	5 epochs patience	3 epochs patience
Max Epochs	–	100	30
Loss Function	RMSE (per target)	MSE	RMSE (MultiLoss)
Scoring Metric	neg_root_mean_squared_error	val_loss	val_loss
Data Handling			
Missing Values	Native sparsity handling	Median imputation + indicators	Median imputation + indicators
Categorical Encoding	Native handling	Category codes	NaNLabelEncoder
Normalization	–	StandardScaler	GroupNormalizer
Feature Groups	Lagged features	All exogenous	Categorized by temporal type

codes) before imputation with a configurable mask value (-1.0), while TFT utilized specialized encoders for categorical variables and normalization techniques for numerical features, ensuring consistent vocabulary mapping across data splits.

C3 Temporal Context and Prediction Strategy

Temporal modeling approaches diverged significantly across architectures, reflecting their underlying computational paradigms. XGBoost transformed temporal dependencies into static feature representations through systematic lagging, creating additional input variables representing previous timestep values, effectively treating time series prediction as a supervised learning prob-

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

lem with engineered temporal features. The LSTM implementation employed sophisticated sequence modeling with configurable window lengths (1-4 timesteps searched during hyperparameter optimization), treating all features as exogenous inputs observed at each timestep, with optional teacher forcing (Williams and Zipser, 1989) incorporating previous target values as additional input channels during training phases. Sequence generation respected group boundaries (Model-Scenario-Region combinations), ensuring temporal windows never crossed experimental configurations. TFT utilized the most sophisticated temporal architecture, categorizing features into distinct temporal groups: time-varying known reals (Year, DeltaYears), time-varying unknown reals (economic indicators), static categoricals (Model, Scenario, Region), and time-varying known categoricals (missingness indicators). The encoder processed 2-timestep historical contexts while supporting variable prediction horizons, employing attention mechanisms to identify relevant temporal patterns. All architectures transitioned from teacher forcing during training to autoregressive prediction during inference, where models generated multi-step forecasts by iteratively using their own predictions as inputs for subsequent timesteps—essential for the extended projection horizons characteristic of integrated assessment modeling applications.

Appendix D: Extended Results

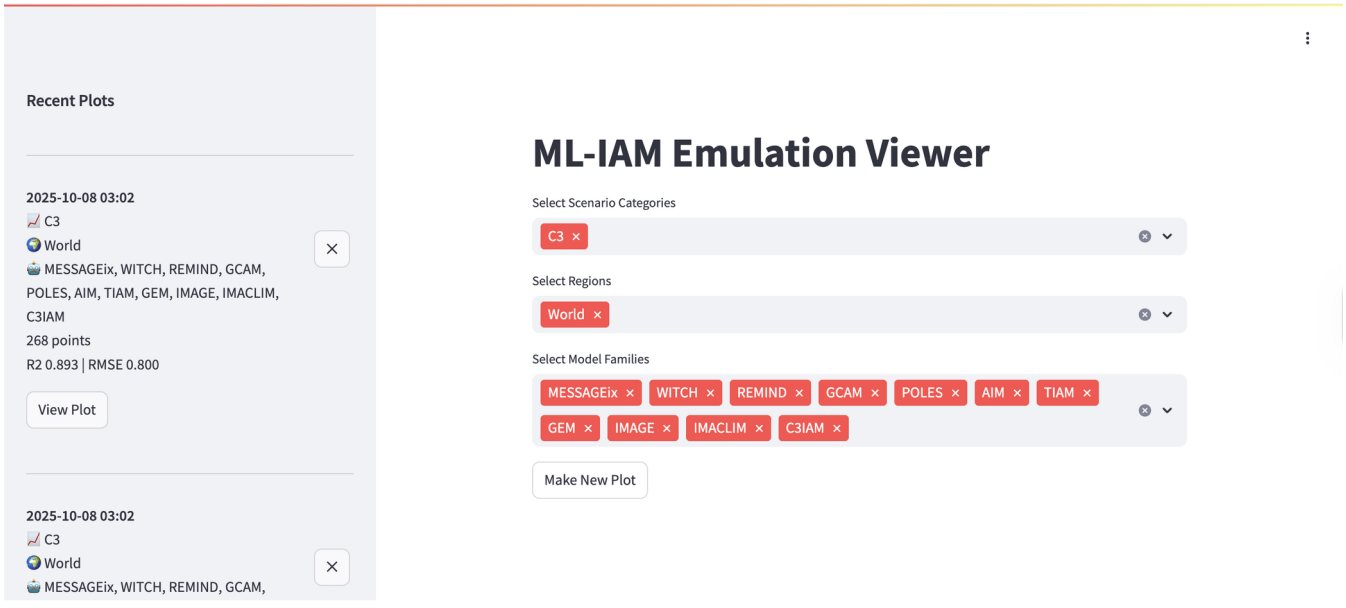
Table D1 summarizes the R^2 values for all output variables across the three ML models. Extended visualizations including scatter plots (Figures S1–S3), temporal trajectories (Figures S4–S6), and SHAP analysis (Figures S7–S9) are provided in the Supplementary Information.

Table D1. R^2 values for ML model predictions versus original IAM outputs across all target variables.

Output Variable	XGBoost	LSTM	TFT
Primary Energy|Coal	0.966	0.911	−0.169
Primary Energy|Gas	0.977	0.958	0.351
Primary Energy|Oil	0.985	0.963	0.380
Primary Energy|Solar	0.978	0.960	0.190
Primary Energy|Wind	0.976	0.960	0.285
Primary Energy|Nuclear	0.891	0.885	0.449
Emissions|CO2	0.981	0.944	0.301
Emissions|CH4	0.989	0.964	0.422
Emissions|N2O	0.992	0.970	0.373
Average	**0.970**	**0.946**	**0.287**

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Appendix E: ML-IAM Emulation Viewer



(a) Select Categories, Regions, and Model Families

(b) Make New Plot

Figure E1. Screenshots of the interactive Emulation Viewer, which generates graphs based on user-selected scenarios. Available at https://mliam.dev/.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Author contributions. YS, CL, MC, BK, JW, AO, and HM designed the study. YS and KP performed the literature review. JW, MC, BK, JH, and HM contributed to the data curation and processing. CL, EK, and JM contribute to the development of the ML models. EK and HM contributed to the visualizations. YS developed the model code, conducted the analysis, and wrote the original manuscript draft with contributions from all co-authors.

Competing interests. No competing interests.

Acknowledgements. This research was supported by the Korea Environmental Industry & Technology Institute(KEITI) through Project for developing an observation-based GHG emissions geospatial information map, funded by Korea Ministry of Environment(MOE) (RS-2023-00232066).

The authors used generative AI tools to assist with code development and to improve the readability and clarity of the manuscript. All AI-generated content was reviewed and edited by the authors, who take full responsibility for the final publication.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

References

Alonso, N. I., Antolin Camarena, J., et al.: Physics-informed neural networks (pinns) in finance, Julian, Physics-Informed Neural Networks (PINNs) in Finance (October 10, 2023), 2023.

Anesti, N., Kalamara, E., and Kapetanios, G.: Forecasting with Machine Learning methods and multiple large datasets, Econometrics and Statistics, 2024.

Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Arango, S. P., Kapoor, S., et al.: Chronos: Learning the language of time series, arXiv preprint arXiv:2403.07815, 2024.

Bates, D., Mächler, M., Bolker, B., and Walker, S.: Fitting linear mixed-effects models using lme4, Journal of statistical software, 67, 1–48, 2015.

Bodnar, C., Bruinsma, W. P., Lucic, A., Stanley, M., Vaughan, A., Brandstetter, J., Garvan, P., Riechert, M., Weyn, J., Dong, H., Gupta, J. K., Thambiratnam, K., Archibald, A., Wu, C.-C., Heider, E., Welling, M., Turner, R. E., and Perdikaris, P.: A foundation model for the Earth system, Nature, 629, 900–908, https://doi.org/10.1038/s41586-025-09005-y, 2025.

Brian C. O'Neill, Kriegler, E., Edmonds, J., Hallegatte, S., Ebi, K. L., Kram, T., Riahi, K., Winkler, H., and van Vuuren, D. P.: A new scenario framework for climate change research: the concept of shared climate policy assumptions, Climatic Change, 122, 401–414, https://doi.org/10.1007/s10584-013-0971-5, 2014.

Byers, E., Krey, V., Kriegler, E., Riahi, K., Schaeffer, R., Kikstra, J., Lamboll, R., Nicholls, Z., Sandstad, M., Smith, C., van der Wijst, K., Al Khourdajie, A., Lecocq, F., Portugal-Pereira, J., Saheb, Y., Stromann, A., Winkler, H., Auer, C., Brutschin, E., Gidden, M., Hackstock, P., Harmsen, M., Huppmann, D., Kolp, P., Lepault, C., Lewis, J., Marangoni, G., Mller-Casseres, E., Skeie, R., Werning, M., Calvin, K., Forster, P., Guivarch, C., Hasegawa, T., Meinshausen, M., Peters, G., Rogelj, J., Samset, B., Steinberger, J., Tavoni, M., and van Vuuren, D.: AR6 Scenarios Database, https://doi.org/10.5281/zenodo.5886911, 2022.

Calvin, K. and Bond-Lamberty, B.: Integrated human-earth system modeling—state of the science and future directions, Environmental Research Letters, 13, 063 006, 2018.

Chen, T. and Guestrin, C.: Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785–794, 2016.

Chen, X.-S., Kim, M. G., Lin, C.-H., and Na, H. J.: Development of per capita GDP forecasting model using deep learning: including consumer goods index and unemployment rate, Sustainability, 17, 843, 2025.

Crane-Droesch, A.: Machine learning methods for crop yield prediction and climate change impact assessment in agriculture, Environmental Research Letters, 13, 114 003, https://doi.org/10.1088/1748-9326/aae159, 2018.

Cuomo, S., Di Cola, V. S., Giampaolo, F., Rozza, G., Raissi, M., and Piccialli, F.: Scientific machine learning through physics–informed neural networks: Where we are and what's next, Journal of Scientific Computing, 92, 88, 2022.

Das, A., Kong, W., Sen, R., and Zhou, Y.: A decoder-only foundation model for time-series forecasting, in: Forty-first International Conference on Machine Learning, 2024.

Dauphin, M. J.-F., Dybczak, M. K., Maneely, M., Sanjani, M. T., Suphaphiphat, M. N., Wang, Y., and Zhang, H.: Nowcasting gdp-a scalable approach using dfm, machine learning and novel data, applied to european economies, International Monetary Fund, 2022.

Dekker, M. M., Daioglou, V., Pietzcker, R., Rodrigues, R., de Boer, H.-S., Dalla Longa, F., Drouet, L., Emmerling, J., Fattahi, A., Fotiou, T., Fragkos, P., Fricko, O., Gusheva, E., Harmsen, M., Huppmann, D., Kannavou, M., Krey, V., Lombardi, F., Luderer, G., Pfenninger,

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

S., Tsiropoulos, I., Zakeri, B., van der Zwaan, B., Usher, W., and van Vuuren, D.: Identifying energy model fingerprints in mitigation scenarios, Nature Energy, 8, 1395–1404, https://doi.org/10.1038/s41560-023-01399-1, 2023.

Dong Thi Ngoc, L., Hoan, N. D., and Nguyen, H.-N.: Gross Domestic Product Forecasting Using Deep Learning Models with a Phase-Adaptive Attention Mechanism, Electronics, 14, 2132, 2025.

Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., and Taylor, K. E.: Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6) experimental design and organization, Geoscientific Model Development, 9, 1937–1958, https://doi.org/10.5194/gmd-9-1937-2016, 2016.

Gasser, T., Guivarch, C., Tachiiri, K., Malko, A., and Ciais, P.: The compact Earth system model OSCAR v2.2: description and first results, Geoscientific Model Development, 10, 271–305, https://doi.org/10.5194/gmd-10-271-2017, 2017.

Goulet Coulombe, P.: The macroeconomy as a random forest, Journal of Applied Econometrics, 39, 401–421, 2024.

Hajjem, A., Bellavance, F., and Larocque, D.: Mixed-effects random forest for clustered data, Journal of Statistical Computation and Simulation, 84, 1313–1328, 2014.

Han, Y., Tian, Y., Yu, L., and Gao, Y.: Economic system forecasting based on temporal fusion transformers: Multi-dimensional evaluation and cross-model comparative analysis, Neurocomputing, 552, 126 500, 2023.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., De Chiara, G., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R. J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., de Rosnay, P., Rozum, I., Vamborg, F., Villaume, S., and Thépaut, J.-N.: The ERA5 global reanalysis, Quarterly Journal of the Royal Meteorological Society, 146, 1999–2049, https://doi.org/https://doi.org/10.1002/qj.3803, 2020.

Hochreiter, S. and Schmidhuber, J.: Long short-term memory, Neural computation, 9, 1735–1780, 1997.

Holmes, A., Jensen, M., Coffland, S., Shen, H. M., Sizemore, L., Bassetti, S., Nieva, B., Tebaldi, C., Snyder, A., and Hutchinson, B.: Emulating the Global Change Analysis Model with Deep Learning, https://arxiv.org/abs/2412.08850, 2024.

IIASA Energy, Climate, and Environment (ECE) Program: MESSAGEix-Transport (model.transport) — *message-ix-models* documentation, https://docs.messageix.org/projects/models/en/stable/transport/index.html, read the Docs "stable" build; revision cf687da5. Accessed 2025-08-21, 2024.

Kipf, T.: Semi-supervised classification with graph convolutional networks, arXiv preprint arXiv:1609.02907, 2016.

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stott, J., Pritzel, A., Mohamed, S., and Battaglia, P.: Learning skillful medium-range global weather forecasting, Science, 382, 1416–1421, https://doi.org/10.1126/science.adi2336, 2023.

Li, P., Zhu, R., McJeon, H., Byers, E., Zhou, P., and Ou, Y.: Using deep learning to generate key variables in global mitigation scenarios, Nature Climate Change, pp. 1–9, 2025.

Lim, B., Arık, S. Ö., Loeff, N., and Pfister, T.: Temporal fusion transformers for interpretable multi-horizon time series forecasting, International journal of forecasting, 37, 1748–1764, 2021.

Liu, Y., Pan, R., and Xu, R.: Mending the crystal ball: Enhanced inflation forecasts with machine learning, Imf working paper, International Monetary Fund, 2024.

Liu, Y., Qin, G., Shi, Z., Chen, Z., Yang, C., Huang, X., Wang, J., and Long, M.: Sundial: A family of highly capable time series foundation models, arXiv preprint arXiv:2502.00816, 2025.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Lundberg, S. M. and Lee, S.-I.: A unified approach to interpreting model predictions, Advances in neural information processing systems, 30, 2017.

Maccarrone, G., Morelli, G., and Spadaccini, S.: GDP forecasting: machine learning, linear or autoregression?, Frontiers in Artificial Intelligence, 4, 757 864, 2021.

Meinshausen, M., Raper, S. C. B., and Wigley, T. M. L.: Emulating coupled atmosphere–ocean and carbon cycle models with a simpler model, MAGICC6—Part 1: Model description and calibration, Atmospheric Chemistry and Physics, 11, 1417–1456, https://doi.org/10.5194/acp-11-1417-2011, 2011.

Natel, C., Belda, D. M., Anthoni, P., Haß, N., Rabin, S., and Arneth, A.: Emulating grid-based forest carbon dynamics using machine learning: an LPJ-GUESS v4.1.1 application, Geoscientific Model Development, 18, 4317–4333, https://doi.org/10.5194/gmd-18-4317-2025, 2025.

Nguyen, T., Brandstetter, J., Kapoor, A., Gupta, J. K., and Grover, A.: ClimaX: A foundation model for weather and climate, arXiv preprint arXiv:2301.10343, https://arxiv.org/abs/2301.10343, 2023.

Nie, Y., Nguyen, N. H., Sinthong, P., and Kalagnanam, J.: A time series is worth 64 words: Long-term forecasting with transformers, arXiv preprint arXiv:2211.14730, 2022.

Oancea, B. and Simionescu, M.: Gross Domestic Product Forecasting: Harnessing Machine Learning for Accurate Economic Predictions in a Univariate Setting, Electronics, 13, 4918, 2024.

Peters, G. P., Andrew, R. M., Canadell, J. G., Fuss, S., Jackson, R. B., Korsbakken, J. I., Le Quéré, C., and Nakicenovic, N.: Key indicators to track current progress and future ambition of the Paris Agreement, Nature Climate Change, 7, 118–122, 2017.

Peters, G. P., Al Khourdajie, A., Sognnaes, I., and Sanderson, B. M.: AR6 scenarios database: an assessment of current practices and future recommendations, npj Climate Action, 2, 31, 2023.

Raissi, M., Perdikaris, P., and Karniadakis, G. E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Computational physics, 378, 686–707, 2019.

Rani, I. and Verma, C. K.: G-PINNs: A Bayesian-Optimized GRU-Enhanced Physics-Informed Neural Network for Advancing Short Rate Model Predictions, Engineering Analysis with Boundary Elements, 179, 106 396, 2025.

Riahi, K., van Vuuren, D. P., Kriegler, E., Edmonds, J., O'Neill, B. C., Fujimori, S., Bauer, N., Calvin, K., Dellink, R., Fricko, O., Lutz, W., Popp, A., Cuaresma, J. C., KC, S., Leimbach, M., Jiang, L., Kram, T., Rao, S., Emmerling, J., Ebi, K., Hasegawa, T., Havlik, P., Humpenöder, F., Da Silva, L. A., Smith, S., Stehfest, E., Bosetti, V., Eom, J., Gernaat, D., Masui, T., Rogelj, J., Strefler, J., Drouet, L., Krey, V., Luderer, G., Harmsen, M., Takahashi, K., Baumstark, L., Doelman, J. C., Kainuma, M., Klimont, Z., Marangoni, G., Lotze-Campen, H., Obersteiner, M., Tabeau, A., and Tavoni, M.: The Shared Socioeconomic Pathways and their energy, land use, and greenhouse gas emissions implications: An overview, Global Environmental Change, 42, 153–168, https://doi.org/https://doi.org/10.1016/j.gloenvcha.2016.05.009, 2017.

Riahi, K., Schaeffer, R., Arango, J., Calvin, K., Guivarch, C., Hasegawa, T., Jiang, K., Kriegler, E., Matthews, R., Peters, G. P., Rao, A., Robertson, S., Sebbit, A. M., Steinberger, J., Tavoni, M., and van Vuuren, D. P.: Chapter 3: Mitigation pathways compatible with long-term goals, in: Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change, edited by Shukla, P. R., Skea, J., Slade, R., Al Khourdajie, A., van Diemen, R., McCollum, D., Pathak, M., Some, S., Vyas, P., Fradera, R., Belkacemi, M., Hasija, A., Lisboa, G., Luz, S., and Malley, J., Cambridge University Press, Cambridge, UK and New York, NY, USA, https://doi.org/10.1017/9781009157926.005, accessed 2025-08-21, 2022.

Richardson, A., Mulder, T. v. F., and Vehbi, T.: Nowcasting New Zealand GDP using machine learning algorithms, CAMA Working Paper 47/2018, Centre for Applied Macroeconomic Analysis, Australian National University, 2018.

https://doi.org/10.5194/egusphere-2025-5305
Preprint. Discussion started: 9 January 2026

Saleem, H., Salim, F., and Purcell, C.: PACER: Physics Informed Uncertainty Aware Climate Emulator, arXiv preprint arXiv:2410.21657, https://arxiv.org/abs/2410.21657, 2024.

Shin, Y., Lee, C., Kim, E., Myung, J., Park, K., Ha, J., Choi, M.-Y., Kim, B., Ka, H. W., Woo, J.-H., Oh, A., and McJeon, H.: ML-IAM: Supporting Data for Machine Learning Emulator of Integrated Assessment Models, https://doi.org/10.5281/zenodo.17390113, 2025a.

Shin, Y., Lee, C., Kim, E., Myung, J., Park, K., Ha, J., Choi, M.-Y., Kim, B., Ka, H. W., Woo, J.-H., Oh, A., and McJeon, H.: ML-IAM: Machine Learning Emulator for Integrated Assessment Models - Source Code, https://doi.org/10.5281/zenodo.17390678, 2025b.

Shwartz-Ziv, R. and Armon, A.: Tabular data: Deep learning is not all you need, Information Fusion, 81, 84–90, 2022.

Sigrist, F.: Gaussian process boosting, Journal of Machine Learning Research, 23, 1–46, 2022.

Smith, C. J., Forster, P. M., Allen, M., Leach, N., Millar, R. J., Passerello, G. A., and Regayre, L. A.: FAIR v1.3: A simple emissions-based impulse response and carbon cycle model, Geoscientific Model Development, 11, 3131–3154, https://doi.org/10.5194/gmd-11-3131-2018, 2018.

Takakura, J., Fujimori, S., Takahashi, K., Hanasaki, N., Hasegawa, T., Hirabayashi, Y., Honda, Y., Iizumi, T., Park, C., Tamura, M., et al.: Reproducing complex simulations of economic impacts of climate change with lower-cost emulators, Geoscientific Model Development Discussions, 2020, 1–29, 2020.

Wang, Z., Li, K., Xia, S. Q., and Liu, H.: Economic recession prediction using deep neural network, arXiv preprint arXiv:2107.10980, 2021.

Watson-Parris, D., Rao, Y., Olivié, D., Seland, , Nowack, P., Camps-Valls, G., Stier, P., Bouabid, S., Dewey, M., Fons, E., Gonzalez, J., Harder, P., Jeggle, K., Lenhardt, J., Manshausen, P., Novitasari, M., Ricard, L., and Roesch, C.: ClimateBench v1.0: A Benchmark for Data-Driven Climate Projections, Journal of Advances in Modeling Earth Systems, 14, e2021MS002 954, https://doi.org/https://doi.org/10.1029/2021MS002954, 2022.

Williams, R. J. and Zipser, D.: A learning algorithm for continually running fully recurrent neural networks, Neural computation, 1, 270–280, 1989.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., and Sahoo, D.: Unified training of universal time series forecasting transformers, in: Proceedings of the 41st International Conference on Machine Learning, PMLR, 2024.

Wu, H., Xu, J., Wang, J., and Long, M.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, Advances in neural information processing systems, 34, 22 419–22 430, 2021.

Yang, Y., Xu, X., Ge, J., and Xu, Y.: Machine Learning for Economic Forecasting: An Application to China's GDP Growth, arXiv preprint arXiv:2407.03595, 2024.

Yoon, J.: Forecasting of real GDP growth using machine learning models: Gradient boosting and random forest approach, Computational Economics, 57, 247–265, 2021.

Zhang, J., Wen, J., and Yang, Z.: China's gdp forecasting using long short term memory recurrent neural network and hidden markov model, Plos one, 17, e0269 529, 2022.

Zhang, Y. and Yan, J.: Cross former: Transformer utilizing cross-dimension dependency for multivariate time series forecasting, in: The eleventh international conference on learning representations, 2023.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., and Jin, R.: Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting, in: International conference on machine learning, pp. 27 268–27 286, PMLR, 2022.