Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

UniCast: A Universal Model for Multi-Lead-Time and Modality-Specific Predictions

Jiaqi Zheng
Sun Yat-sen University
Qing Ling
lingqing556@mail.sysu.edu.cn

Sun Yat-sen University
Yerong Feng
GBA Academy of Meteorological Research (Shenzhen Institute of Meteorological Innovation)

Article

Keywords:

Posted Date: September 15th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10894236/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

UniCast: A Universal Model for Multi-Lead-Time and Modality-Specific Predictions

Jiaqi Zheng[1], Qing Ling[1*], Yerong Feng[2*]

[1]School of Computer Science and Engineering/Guangdong Provincial Key Laboratory of Computational Science, Sun Yat-Sen University, Guangzhou, 510006, China.
[2]Shenzhen Institute of Meteorological Innovation, Shenzhen, 518038, China.

*Corresponding author(s). E-mail(s): lingqing556@mail.sysu.edu.cn; yerong_feng@gd121.cn;
Contributing authors: zhengjq23@mail2.sysu.edu.cn;

Abstract

Existing Deep Learning Weather Prediction (DLWP) models typically rely on fixed-lead-time (e.g., 6-hour) autoregression, creating a temporal conflict: precipitation demands rapid update to provide timely warning, while upper-air variables require longer lead times to reduce accumulation of autoregressive errors. Such a conflict leads to separate, resource-intensive models for the two modalities, ignoring the naturally coupled atmospheric state. To address this issue, we propose UniCast, a universal model that delivers multi-lead-time and modality-specific predictions. Based on a semi-autoregressive strategy, UniCast predicts weather variables autoregressively at different lead times (e.g., 1 hour or 6 hours), simultaneously satisfying the high-frequency requirement of precipitation predictions and the long-lead-time requirement of upper-air predictions. To guarantee high accuracies across the two modalities, UniCast tailors model training tasks to their specific challenges. For upper-air predictions, UniCast employs a large, foundational generalist that integrates advanced spatial and temporal operators, Window Multi-head Self-attention (W-MSA) and adaptive Layer Normalization (adaLN), to capture local atmospheric processes within the semi-autoregressive strategy; for precipitation predictions, UniCast introduces a small, pluggable specialist trained subject to a water-vapor-related equation to mitigate blurry predictions. Through jointly handling the two modalities, UniCast fully leverages atmospheric correlations while reducing computational overhead. Experimental results demonstrate that UniCast outperforms mainstream DLWP baselines on both precipitation and upper-air forecasting. The code is available online.*

1 Introduction

Modern weather prediction is now a fundamental and indispensable pillar of societal safety and economic planning [1, 2]. For decades, Numerical Weather Prediction (NWP) has been the gold standard, relying on high-performance supercomputing to solve the complex equations of fluid dynamics and thermodynamics [3, 4]. However, NWP demands immense computational resources [5, 6], which creates a severe bottleneck in the forecast speed, and restricts both update frequency and ensemble scalability. Such constraints are detrimental when predicting rapidly evolving, high-impact weather events where timeliness is critical [7]. In addition, precipitation remains exceptionally difficult for NWP to accurately forecast in terms of exact location, timing, and intensity [8]. The reason is that, precipitation is driven by complex microphysics and sub-grid scale phenomena; simulating these processes introduces significant errors into NWP models [9].

*https://github.com/Yumenomae/UniCast

In response to these challenges, Deep Learning Weather Prediction (DLWP) has emerged as a promising complement in recent years. By leveraging the vast archives of weather data, modern GPU hardware, and established deep learning architectures, DLWP introduces a data-driven paradigm that dramatically accelerates forecast speed while maintaining competitive accuracy [10–12]. Crucially, DLWP demonstrates potential in regimes where NWP usually struggles, such as precipitation nowcasting [13–15]. In those rapid, localized scenarios, where explicit physical equations and robust numerical methods are often inadequate or too slow, DLWP is able to capture complex atmospheric patterns directly from data [16–18]. Despite the potential, existing DLWP models still face several long-standing limitations. Among them, a critical one is the divide caused by fixed-lead-time autoregression, preventing existing DLWP models from fully leveraging correlations among multiple variables.

As weather variables influence each other, we naturally expect DLWP to predict all weather variables simultaneously if data are available. However, the trajectory of DLWP research actually bifurcates into two streams: predicting slowly-evolving variables only [10–12] and predicting rapid downstream events like precipitation only [17, 19, 20]. Such a divide stems from the reliance on fixed-lead-time autoregressive forecasting. For instance, many DLWP models operate on a 6-hour lead time [11, 12, 21]. While sufficient for gradual upper-air variables, this coarse resolution shall miss the temporal dynamics required by precipitation forecasting. Conversely, shrinking the lead time to 1 hour requires the model to perform six times as many rollout steps to reach the same forecast time point. Such a drastic increase in iterations causes autoregressive errors to accumulate rapidly, and hence degrades accuracy [10]. Facing this dilemma, a common remedy is to train separate models for coupled atmospheric variables, failing to leverage the statistical correlations between upper-air variables and precipitation [22].

Beyond resolving the temporal conflict to enable simultaneous predictions of all weather variables, another central challenge remains: achieving accurate predictions across these variables, particularly for the difficult case of precipitation. Driven by their data-centric nature, most models optimize for statistical metrics such as Mean Squared Error (MSE) and Cross-Entropy (CE). While minimizing the average error, they produce blurry forecasts to avoid large penalties [20, 23]. This "statistically safe" approach fails to capture the high variance and extreme intensities of severe weather, particularly strong precipitation [19]. Since capturing details of severe weather is important for operational forecasting, DLWP models must generate meteorologically coherent predictions [24]. To achieve this, the well-established physical laws of atmospheric science, such as conservation of mass and energy [17, 25], can serve as powerful constraints. These physical laws have already provided a fluid dynamical and thermodynamic lens for existing NWP models, making them physically grounded. In consequence, shifting from purely statistical to physics-assisted learning reins in unconstrained extrapolation of DLWP models, and eventually yields forecasts that meteorologists are able to trace and explain.

To address the above limitations, we propose UniCast, a universal model that delivers multi-lead-time and modality-specific predictions. At the core of its multi-lead-time capability is a semi-autoregressive forecasting strategy that reconciles the conflicting temporal demands of precipitation and upper-air variables [26]. To support this capability without sacrificing the accuracy, UniCast tailors training tasks to their specific challenges, training a generalist for upper-air-specific predictions alongside a specialist for precipitation-specific predictions. The generalist is a large, foundational deep learning model trained at multiple base lead times to capture the gradual evolution of upper-air variables as well as some surface variables, while the specialist is a small, pluggable deep learning adapter for the generalist to capture the rapid evolution of precipitation [27]. Because UniCast can seamlessly handle both modalities, we leverage this property to embed a physical constraint for the specialist: a water vapor conservation equation that dynamically links the generalist's upper-air predictions to the specialist's precipitation predictions. This equation explicitly couples the generalist's humidity, wind, and temperature forecasts with the specialist's precipitation forecast, thereby constraining the predictions in UniCast. Guided by a novel blur-penalizing loss function, this physics-assisted approach ensures that the model adheres to the atmospheric dynamics, delivering precipitation forecasts that are both structurally sharp and physically consistent.

2 Results

This section presents a comprehensive evaluation of UniCast to demonstrate its performance on multi-lead-time and modality-specific predictions. We first outline the experimental setup, briefly introducing the data, the model structure, and the training methods for generalist and specialist (detailed in Section

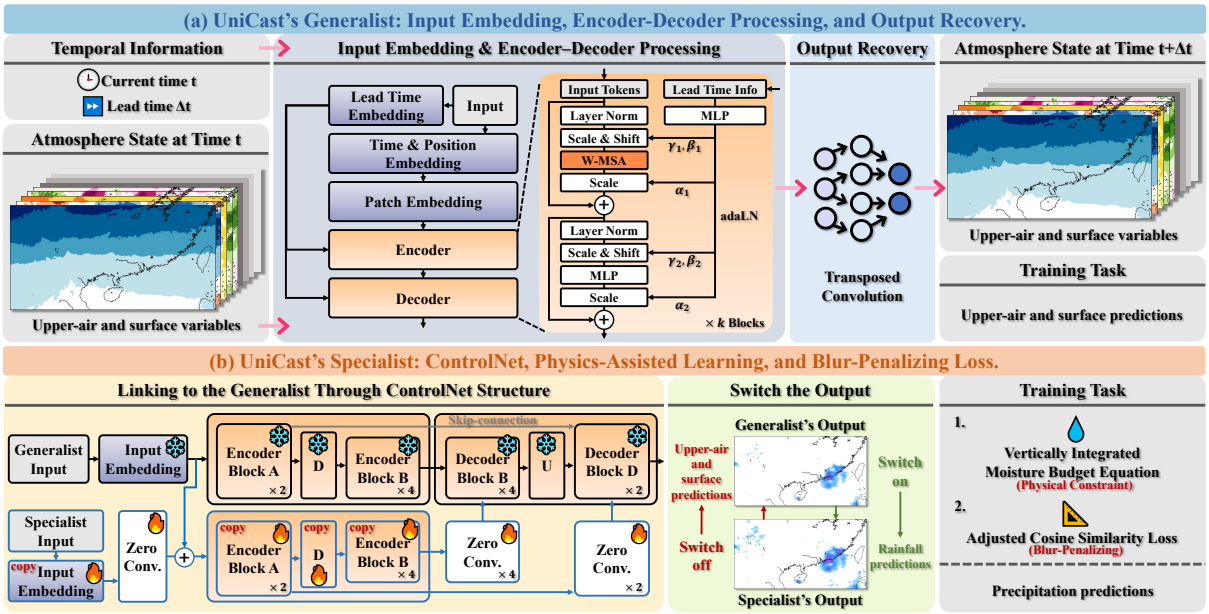


Fig. 1 Schematic of UniCast structure. (a) **The generalist.** The generalist predicts upper-air and surface variables at given lead times within the semi-autoregressive strategy. It consists of three modules: input embedding, encoder-decoder processing, and output recovery. The input embedding layer (steel gray) combines lead-time, temporal, positional, and patch embeddings to represent the atmospheric state. The encoder-decoder (light orange) uses Window Multi-head Self-Attention (W-MSA) and adaptive Layer Normalization (adaLN) to capture multi-scale spatiotemporal dependencies, which are mapped back to the weather field through transposed convolutions in output recovery. (b) **The specialist.** The specialist links to the generalist through the ControlNet structure to specialize in precipitation predictions. It introduces a trainable copy of the generalist's embedding and encoder layers, along with several zero-initialized convolutions. During training, only the specialist's parameters are updated while the generalist remains frozen. As a pluggable adapter, the specialist allows UniCast to switch modalities dynamically. It is also physically constrained: we incorporate the Vertically Integrated Moisture Budget (VIMB) equation and a blur-penalizing Adjusted Cosine Similarity Loss (ACSL) to enforce physical realism in the specialist's predictions.

4). Then, we conduct numerical experiments and performance analysis, aiming to evaluate the generalist on upper-air predictions and assess the specialist on precipitation predictions.

2.1 Data

We train and test UniCast within an operational setting, specifically, over the high-resolution reanalysis dataset from the China Meteorological Administration (CMA). This dataset integrates outputs from the CMA-3DVar assimilation system, the CMA-GD regional NWP model, and the CMPA surface precipitation analysis, from 2018 to 2023 [28]. Featuring a spatial resolution of 0.03° (a grid of 441 × 845 points) and a temporal resolution of 1 hour, this dataset provides the necessary precision for mesoscale modeling [29]. To evaluate the model's generalization capability, we use a chronological data partitioning strategy: the 2018–2021 period is utilized for training, while 2022 and 2023 are reserved for validation and testing, respectively.

2.2 Model Structure and Training Methods

UniCast is engineered as a DLWP system capable of predicting the evolution of 73 weather variables (see Table 1), including 1-hour accumulative precipitation, other surface variables, as well as several upper-air variables distributed across 13 vertical pressure levels ranging from 1000 hPa to 50 hPa.

To resolve the temporal conflict between upper-air variables and precipitation, UniCast adopts a semi-autoregressive forecasting strategy. This strategy balances computational efficiency and temporal precision: it enables UniCast to generate hourly precipitation predictions while predicting upper-air variables autoregressively at longer lead times.

Within UniCast, the generalist establishes a comprehensive understanding to the evolution of upper-air variables and other surface variables. As illustrated in Fig. 1(a), UniCast's generalist is a large, foun-

dational deep learning model, consisting of three main modules: input embedding, encoder-decoder core, and output recovery. In the encoder-decoder core, the integration of Window Multi-head Self-Attention (W-MSA) and adaptive Layer Normalization (adaLN) allows the model to capture complex, multi-scale interactions in both space and time [30, 31]. We use four base lead times (1, 3, 6, and 12 hours) for the generalist's semi-autoregressive strategy and train the generalist to predict at these lead times uisng MSE. The base lead times can help the generalist learn the evolution of both upper-air and surface variables at different temporal resolutions. Ultimately, the generalist serves as a strong backbone for UniCast and provides a stable basis for the specialist introduced next.

The specialist is a small, pluggable deep learning adapter to help the generalist generate precipitation-specific predictions. For this task, one important issue is to tackle the blurry effect that is very common in deep learning-based precipitation models. As depicted in Fig. 1(b), the link between the generalist and the specialist follows the ControlNet structure: the specialist injects precipitation-specific adjustments to the generalist through a symmetric encoder and several zero convolutions [32, 33]. This dynamic modality switching allows us to embed a physical constraint that shifts the specialist from purely statistical to physics-assisted learning: we introduce the Vertically Integrated Moisture Budget (VIMB) equation to couple the generalist's upper-air predictions to the specialist's precipitation predictions and minimize the deviation through a novel Adjusted Cosine Similarity Loss (ACSL) [34]. Such a physics-assisted approach effectively counteracts the blurry effect typical of models trained through MSE or CE, producing sharper, physically realistic forecasts of localized precipitation.

2.3 Numerical Experiments

We evaluate UniCast's generalist and specialist from two aspects. First, the generalist is assessed on its ability to predict upper-air and surface variables over 24 hours, utilizing the standard Root Mean Square Error (RMSE) metric [35]. The reason of limiting the evaluation to 24 hours is that training the baseline models on a 441×845 grid at high temporal (1-hour) and spatial (0.03°) resolutions results in heavy GPU memory burden, restricting the lead times they can learn. We therefore evaluate up to 24 hours, a lead time achievable by every model. Second, because the specialist is dedicated to precipitation, standard point-wise metrics like RMSE are inadequate, as they fail to account for spatial displacement and intensity shifts. Instead, we employ event-based metrics, Critical Success Index (CSI) and Fractional Skill Score (FSS), to evaluate the accuracy [36].

We benchmark UniCast against the regional NWP model (CMA-GD) and a number of DLWP models (FourCastNet [11], Pangu [10], GraphCast [12], GraphFM [37], YingLong [38], and Stormer [39]). Because these baselines are originally trained on coarser global datasets (typically 0.25° ERA5) [40], we retrain them from scratch on our high-resolution (0.03°) regional dataset. However, fully retraining such a large number of models turns to be computationally prohibitive. To enable comparisons with many baselines at an acceptable cost, we retrain them exclusively on a temporally subsampled subset in which initialization times are restricted to 00, 06, 12, and 18 UTC. For ensuring fair comparisons, we additionally introduce UniCast*, a variant of UniCast trained on the same subset. Detailed configurations of the baseline models are provided in Appendix.

We begin our quantitative evaluation by analyzing RMSE across six upper-air variables (z500, t500, q500, u500, t850, q850) and two surface fields (2t, 10u). Fig. 2 illustrates the evolution of forecast skill, focusing on hourly lead times up to 24 hours. The full UniCast model, benefiting from the comprehensive hourly training dataset, outperforms all baselines across most variables and lead times. Beyond the 6-hour lead time, semi-autoregressive architectures (specifically UniCast, UniCast*, and Stormer) demonstrate significantly lower RMSE than the others. While Stormer excels in predicting geopotential height (z500), UniCast* proves to be more robust across the remaining variables, particularly thermodynamic fields such as t850 and 2t. Standard autoregressive models perform well at short lead times (< 6 hours), especially YingLong. Meanwhile, graph-based baselines such as GraphCast and GraphFM suffer rapid performance degradation beyond the first 1 to 2 hours. This degradation stems from the memory-intensive nature of graph-based models, which confines them to training with shorter lead times than the others.

We evaluate the specialist's precipitation forecasts using two event-based skill scores: CSI and FSS, with FSS computed over a fixed 5×5 grid window (15km $\times$ 15km). Performance is assessed in terms of intensity and lead time. Intensity is measured across cumulative thresholds of 1, 2, 4, and 8mm, whereas lead time focuses on 1- and 3-hour forecasts, for which capturing rapid precipitation changes is essential.

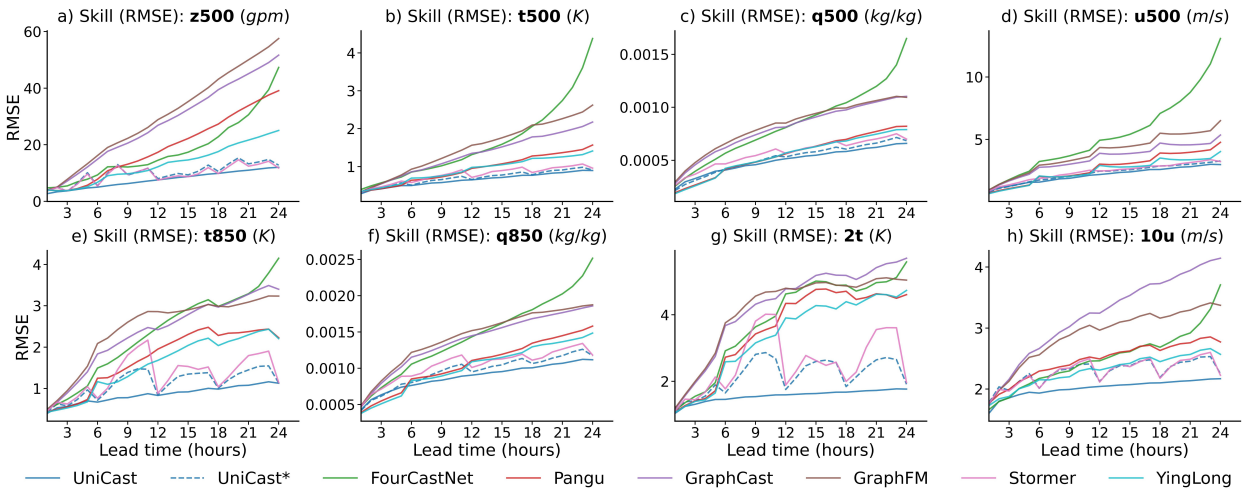


Fig. 2 Comparison between UniCast's generalist, UniCast*'s generalist and other DLWP models on RMSE skill, over the test dataset. Each subplot shows a single variable. The x-axis shows the lead time in 1-hour intervals over a 24-hour period, while the y-axis indicates the RMSE.

Tables 2 and 3 highlight the performance difference arising from the model structures. Graph-based models such as GraphCast and GraphFM are memory-intensive, which restricts them to shorter training lead times. Thus, their predictive skill peaks at the 1-hour lead time and drops sharply at the 3-hour lead time. Among the remaining baselines, Pangu and YingLong perform better, especially at the 3-hour lead time. Nevertheless, the specialist of UniCast* delivers the strongest overall performance, outperforming all baselines at every threshold for the 3-hour lead time. Compared with the generalists of UniCast* and UniCast, the corresponding specialists raise CSI by an average of 20.13% and 43.93%, and FSS by 18.02% and 40.78%, respectively. These gains grow with the difficulty of forecast: larger improvements appear at longer lead times and higher intensity thresholds. Such advances confirm that our physics-assisted learning recovers spatial details and successfully tunes the specialist to the rapid evolution of extreme precipitation.

While UniCast outperforms the baseline DLWP models, assessing its operational value requires direct comparisons with established NWP systems. Among our baseline DLWP models, Pangu and GraphCast have been reported to surpass the ECMWF's high-resolution Integrated Forecast System (IFS HRES) on the RMSE metric over the 0.25° ERA5 dataset. Their success is partly fueled by training on nearly four decades of global reanalysis data, while UniCast is trained on just four years of regional data. Moreover, regional NWP systems incorporate explicit boundary information (often from global models) to guide the forecast evolution, while our model must infer these complex dynamics purely from the limited training dataset. Therefore, we compare UniCast with numerical models not to claim absolute superiority, but to identify which atmospheric variables can be effectively learned in a data-scarce, high-resolution setting that lacks any additional boundary information.

We benchmark UniCast against CMA-GD, an operational regional NWP model run by the Guangdong Meteorological Service. It provides 96-hour forecasts at a 3km spatial resolution over a domain centered on Guangdong Province, China. This domain covers all of Guangdong, parts of the neighboring provinces (Guangxi, Hunan, Jiangxi, Fujian), and the northern part of South China Sea, with a grid of 813×558 points at 3km spacing. The model is nonhydrostatic and utilizes a terrain-following vertical coordinate, 67 vertical layers, and a model top at 30 km.

The comparisons are shown in Fig. 3. UniCast's generalist achieves competitive skill for thermodynamic variables, specifically temperature (t850, 2t) and specific humidity (q850, q500). This outcome is due to the local-attention layers within the generalist's encoder-decoder module, which is well suited to variables driven mainly by local processes. We observe that while UniCast outperforms CMA-GD at t850, it falls behind at t500. This divergence likely stems from our training loss design, which assigns greater weights to higher-pressure levels. Similarly, for variables governed by large-scale dynamical flows, such as wind (u500) and geopotential height (z500), UniCast cannot outperform CMA-GD. This fact underscores

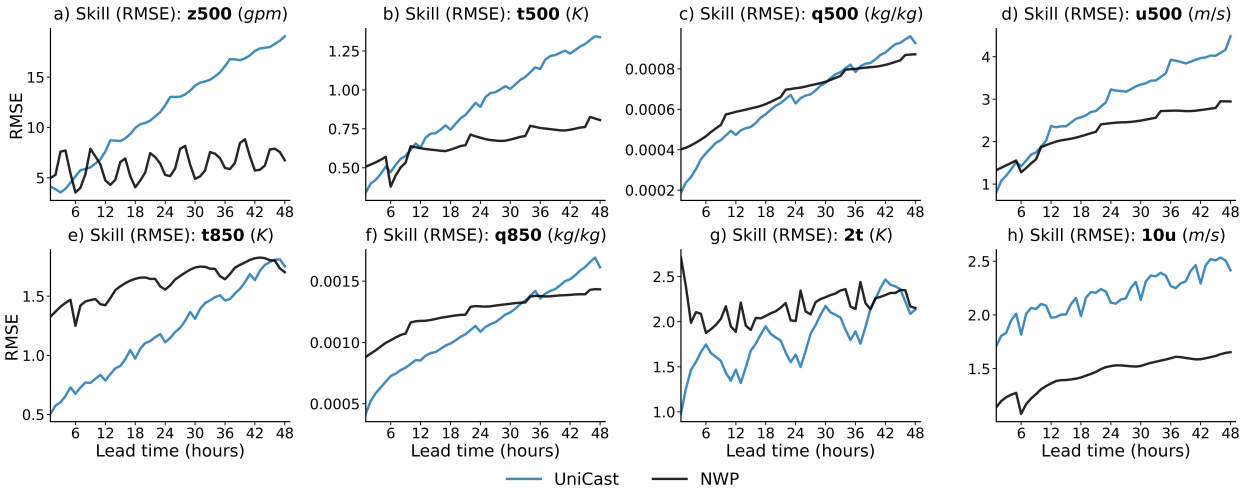


Fig. 3 Comparison between UniCast's generalist and CMA-GD on RMSE skill, over the test dataset. Each subplot shows a single variable. The x-axis shows the lead time in 1-hour intervals over a 48-hour period, while the y-axis indicates the RMSE.

the difficulty of capturing large-scale dynamics from a spatially and temporally limited regional dataset alone.

3 Discussions

After quantitatively evaluating UniCast's performance, we now examine the factors behind these results: the generalist (UniCast's foundational model), the specialist (UniCast's pluggable deep learning adapter), and the physics-assisted training objectives.

3.1 Generalist (Foundational Model)

We owe the performance of UniCast's generalist to the combination of Window Multi-head Self-Attention (W-MSA) and adaptive Layer Normalization (adaLN); the former acts as a spatial operator and the latter as a temporal operator. W-MSA sharpens the model's focus on local spatial details that high-resolution datasets are especially rich in. Meanwhile, adaLN injects the time-step information that UniCast's semi-autoregressive forecasting needs. Together, they helps resolve the tension between capturing the fine-scale spatial details and modeling the evolution over time. Note that these two components have already proven effective in earlier models such as Pangu and Stormer.

The empirical results demonstrated in Fig. 2 confirm the value of combining W-MSA with adaLN in UniCast's generalist. Compared with Pangu that integrates W-MSA but not adaLN, UniCast* initially shows a slight performance dip over the first 12 hours. But after that, adaLN allows UniCast* to maintain the accuracy. Ultimately, UniCast* delivers better performance in the second half of forecast lead times. In contrast, relative to Stormer that combines adaLN but not W-MSA, UniCast* benefits from the latter as it sharpens the processing of fine-scale spatial details, leading to better performance for most variables.

3.2 Specialist (Adapter)

Tables 2 and 3 already demonstrate the value of the specialist: it delivers precipitation-specific forecasts without interfering with the generalist, preserving UniCast's core capabilities while markedly improving precipitation predictions. In this section, we examine the advantages and trade-offs of implementing the specialist as a pluggable adapter rather than as a full deep learning model.

To better understand our design, we now compare three strategies. ControlNet Adapter is the adapter-based approach described in this paper, where a pluggable specialist attaches to the generalist through a ControlNet structure. Full Model is the alternative: a specialist is built as a stand-alone deep learning model independent of the generalist. Original Generalist serves as an additional baseline, within which

we continue training the generalist without introducing any new model. The key differences among these three setups are outlined below.

- ControlNet Adapter: The specialist is a pluggable adapter connected to the generalist via a ControlNet structure. UniCast then generates precipitation-specific predictions while the generalist remains frozen. In the physics-assisted task, upper-air predictions from the generalist are combined with precipitation predictions from the specialist to minimize the deviation of the water vapor conservation equation.
- Full Model: The specialist is a stand-alone model that replicates the generalist's architecture. UniCast generates precipitation predictions directly from the specialist, again with the generalist being frozen. In the physics-assisted task, upper-air predictions from the generalist are combined with precipitation predictions from the specialist to minimize the deviation of the water vapor conservation equation.
- Original Generalist: The generalist itself is further trained to serve as the specialist. UniCast generates precipitation predictions directly, but loses the ability to produce accurate upper-air predictions. Within the physics-assisted task, the model provides both upper-air and precipitation predictions based on the same parameters, and the deviation of the water vapor conservation equation is minimized using these self-provided fields.

To ensure fairness, all strategies are trained on the same CMA dataset, with the identical hyperparameters (learning rates, epochs), and with the same physics-constrained ACSL.

From a resource perspective, ControlNet Adapter is the best choice for operational use. As shown in Table 4, both Full Model and Original Generalist demand extra memory: Full Model increases the memory overhead by storing a complete copy of the generalist, while Original Generalist requires backpropagation through more autoregressive sequences, resulting in more gradient computation. In contrast, ControlNet Adapter cuts the number of trainable parameters by nearly 50% and avoids the extensive autoregressive gradient computation. This design enables UniCast to switch between the generalist and specialist with no need to maintain separate models, ultimately delivering far better operational efficiency.

Tables 5 and 6 highlight the necessity of freezing the generalist, and also reveal the trade-off between efficiency and accuracy. The resource-intensive approach of Original Generalist suffers significant performance degradation, in particular, at the critical 1-hour lead time. Such a failure arises from the lack of stable, accurate upper-air predictions. Since Original Generalist autoregressively generates all 73 variables but is optimized solely for precipitation, the remaining 72 variables drift into instability and imprecision. These degraded intermediates then propagate errors into subsequent forecast steps, severely destabilizing the model output. This phenomenon demonstrates the effectiveness of ControlNet Adapter: by keeping the generalist frozen and refining precipitation through an independent module, we can avoid corrupting the upper-air variables and the physics-assisted task. Ultimately, while ControlNet Adapter slightly lags Full Model in accuracy, its adapter structure drastically reduces memory overhead and model-switching latency, striking an balance between efficiency and accuracy.

3.3 Vertically Integrated Moisture Budget Constraint

To overcome the limitations of purely data-driven models, we introduce a training objective derived from the Vertically Integrated Moisture Budget (VIMB) equation (detailed in Section 4). This equation links precipitation to upper-air variables through the conservation of water vapor; specifically, it connects the Vertically Integrated Water Vapor (VIWV) to wind-driven advection and precipitating process. By enforcing this physical law, we supply the specialist with a physically grounded training signal.

We implement the VIMB equation utilizing a semi-discretized form combined with Euler's method to forecast VIWV at 2- and 4-hour lead times. These forecasts are produced via combining the frozen generalist's upper-air predictions at earlier steps (1 and 3 hours) with the precipitation predictions from the specialist. To ensure that the numerical discretization errors do not penalize the specialist, we compute the ground-truth VIWV targets by applying the exact same Euler scheme to the observational data. The deviation between the predicted and ground-truth VIWV is measured by our Adjusted Cosine Similarity Loss, and the specialist's parameters are optimized by minimizing this deviation.

In this section, we validate that the VIMB constraint works as intended, providing effective guidance for precipitation predictions. We compare our physics-assisted learning (Learning w/ Physics) against a control variant (Learning w/o Physics) that trains the specialist without the VIMB constraint. As shown in Tables 7 and 8, incorporating the VIMB constraint improves 1-hour forecast skill across both CSI and FSS metrics, while the performance at the 3-hour lead time remains stable. This localized improvement is

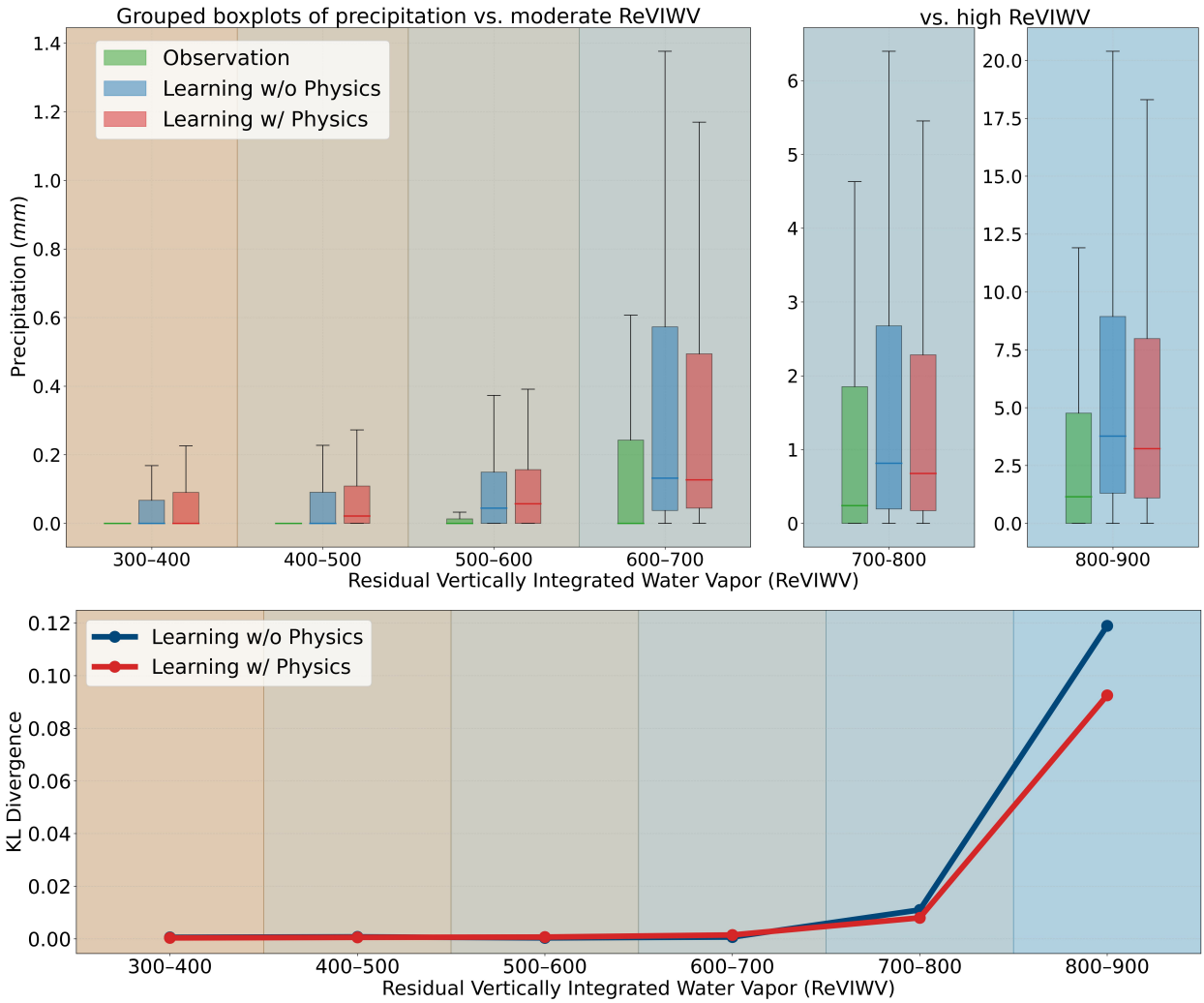


Fig. 4 Grouped boxplots of precipitation versus Residual Vertically Integrated Water Vapor (ReVIWV) over the test set. Top: Boxplots showing the distribution of observed and model-predicted precipitation for each ReVIWV range. The x-axis shows grouped ReVIWV ranges, while the y-axis presents the precipitation values. **Bottom:** Line plot of the KL divergence between observed and model-predicted precipitation distributions across the same ReVIWV ranges, highlighting differences in model performance.

consistent with our design, since only the specialist's 1-hour precipitation predictions are involved in the physics-assisted task. Although numerically solving the governing equations adds a modest computational overhead, the significant gains in 1-hour-lead-time accuracies, which are essential to high-impact weather nowcasting, fully justify the cost.

Beyond the standard performance metrics, our analysis also shows that the physics-assisted learning acts as a strong regularizer, reinforcing the physical consistency between moisture availability and rainfall intensity. To quantify this effect, we investigate the correlation between residual VIWV (ReVIWV), the portion of VIWV not explained by advection and compression, and precipitation predictions (see Fig. 4). Using spatially aggregated samples on a 5×5 grid (15km $\times$ 15km), we construct local joint distributions that compare model predictions directly with observational data. The results demonstrate a clear contrast: Learning w/o Physics drifts significantly in high-moisture regimes; however, for Learning w/ Physics, the joint distribution stays much closer to the observations, correctly pairing higher moisture with heavier precipitation. This stabilizing effect is quantitatively confirmed by a drop in the KL divergence. Together, our findings reveal that the VIMB constraint prevents the model from solely relying on statistics of the dataset, forcing it instead to learn the true physical dynamics that govern the precipitation evolution.

3.4 Adjusted Cosine Similarity Loss

Alongside the VIWV constraint, we introduce the Adjusted Cosine Similarity Loss (ACSL) to tackle the problem of forecast blur. This blur arises because precipitation follows a long-tailed distribution (extreme events are rare) so that the standard losses like MSE or CE, which enforce exact pixel-to-pixel matching, tend to yield a low-variance, blurry output. In contrast, ACSL focuses on aligning spatial structures and patterns rather than matching individual pixel values. This structure-aware design can preserve the high variance of real precipitation fields, rewarding sharp, distinct predictions over blurry ones.

Equally important, ACSL stabilizes our physics-assisted VIMB task. Numerical integration introduces unavoidable discretization errors; rigid losses like MSE and CE risk overfitting to these numerical artifacts. ACSL's structural flexibility avoids the rigid penalty, allowing spatial similarity, structural patterns, and the physical constraints to jointly guide the model toward meteorologically sound dynamics.

Our ablation study benchmarks ACSL against three alternative loss functions: standard MSE, Mean Absolute Error (MAE) and the spectral-domain Fourier Amplitude and Correlation Loss (FACL) [41]. The results (Tables 9 and 10) highlight distinct failure modes for each alternative. MSE yields only marginal gains, mostly confined to low-intensity thresholds (1 mm) at the 1-hour lead time, and its advantage does not carry over to longer forecast lead times. MAE performs even worse, degrading the overall skill since its robustness to outliers causes it to downweight the very extreme events that are critical for precipitation nowcasting. The spectral-based FACL offers some localized benefit, boosting accuracy for extreme events (4 and 8 mm at 3-hour) by penalizing amplitude mismatches in the Fourier domain, likely because aligning spectral amplitudes magnifies the influence of extreme events. Nevertheless, this advantage is confined to extreme intensities, and FACL underperforms in moderate-rain regimes. In contrast, ACSL is shown to be the most broadly effective objective. By aligning spatial structures and patterns rather than individual pixel values, ACSL delivers consistent improvements across all intensity thresholds, successfully resolving the structural details needed for accurate extreme-precipitation forecasts.

3.5 Case Study: Super Typhoon Saola

To ground our findings in a real-world operational context, Fig. 5 presents a case study of Super Typhoon Saola (September 1, 2023) [42]. A visual comparison shows that the specialist delivers a decisive improvement in the structural fidelity. Unlike the generalist, which tends to blurry extreme events, the specialist faithfully captures the storm's spiral rain bands and peak precipitation intensity. We can observe a minor reduction in light-rain accuracy, an expected trade-off when optimizing for heavy-tailed precipitation distributions, but this shift in sensitivity is clearly beneficial. For disaster mitigation, the ability to accurately forecast life-threatening, high-impact events far outweighs small errors in detecting light drizzle. Ultimately, this case study confirms that UniCast can successfully transform a general atmospheric model into a specialized tool for severe, high-impact weather.

4 Methods

This section describes the technical details of UniCast. We first introduce the regional CMA dataset used for training and evaluation. Next, we formalize the forecasting problem, examining the limitations of the standard autoregressive and non-autoregressive strategies to motivate our semi-autoregressive approach. Finally, we describe the core model structure and the training procedures for the generalist and specialist, which together enable UniCast's modality-specific forecasting capabilities.

4.1 Dataset

To ensure high-fidelity regional modeling, UniCast is trained and evaluated on a high-resolution ($0.03°$, $\approx$ 3km) regional dataset covering southern China. The atmospheric state is from regional reanalysis jointly built by the CMA-3DVar data assimilation system and the CMA-GD regional NWP model, while ground-truth precipitation data come from the China Merged Precipitation Analysis (CMPA). The temporal partitioning is designed to test the generalization ability: data from 2018–2021 are utilized for training, 2022 for validation, and 2023 for testing.

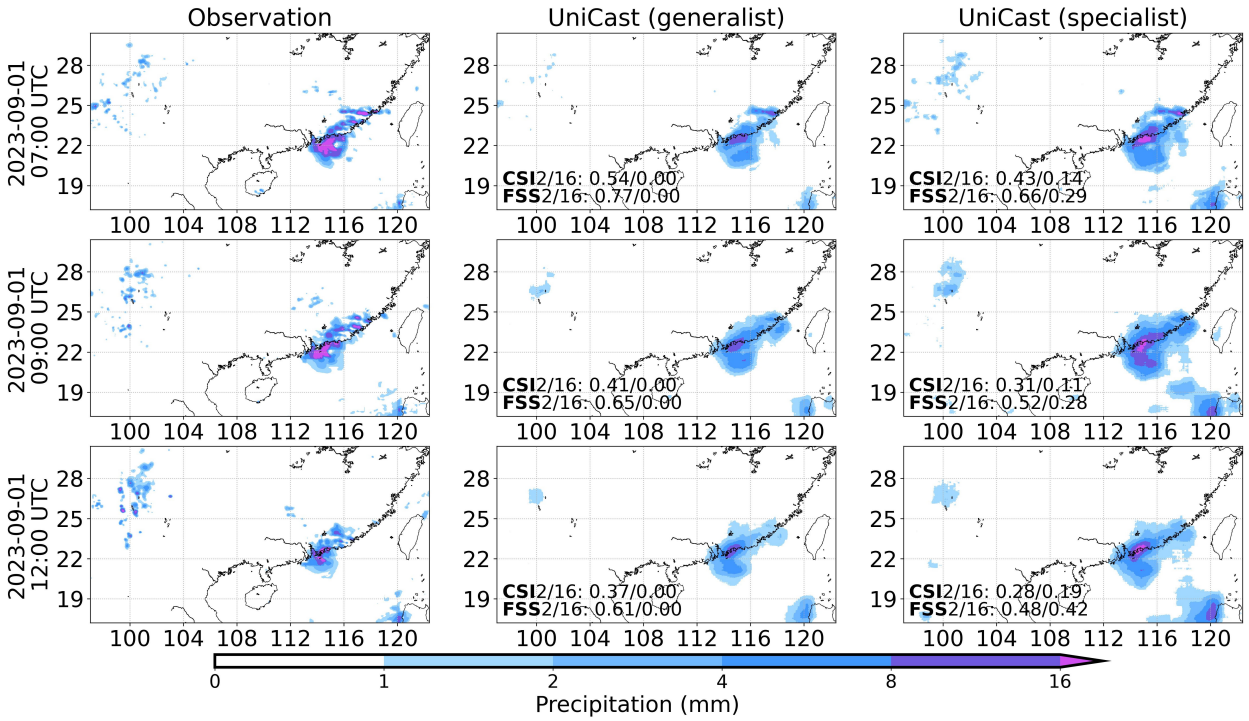


Fig. 5 Case study of model performance on a challenging precipitation event on 2023-09-01 at 06:00 UTC, featuring the spiral-structured intense rainfall associated with Super Typhoon Saola in the Guangdong-Hong Kong-Macao Greater Bay Area. The 06:00 UTC state serves as input for both UniCast (generalist) and UniCast (specialist). The three columns show observed and predicted precipitation from UniCast (generalist) and UniCast (specialist) at 07:00, 09:00, and 12:00 UTC. For each forecast, CSI and FSS scores at the 2 mm and 16 mm thresholds are reported in the bottom-left corner.

4.2 Semi-Autoregressive Approach

Weather forecasting can be cast as a large-scale spatiotemporal prediction problem. Let $X_t \in \mathbb{R}^{V\times H\times W}$ represent the weather state at time t, consisting of V weather variables over a spatial grid of height H and width W. The objective is to learn a mapping operator $\mathscr{T}$ that approximates the transition of the weather system from the current state X_t to a future state $X_{t+\Delta t}$.

The standard approach in both NWP and mainstream DLWP is autoregressive forecasting [10–12]. It relies on a recursive operator $\mathscr{T}$ to predict the weather state in a fixed lead time Δt (typically 6 hours for DLWP), as

$$X_{t+\Delta t} = \mathscr{T}(X_t), \quad X_{t+2\Delta t} = \mathscr{T}(X_{t+\Delta t}), \ \ldots$$

While this approach preserves the continuity of weather state, it creates a sharp conflict between temporal resolution and computational efficiency when applied to a universal model. Precipitation prediction demands high temporal resolution, that is, a short lead time (e.g., $\Delta t = 1$h), to support a precise forecasting, whereas efficient forecasting for upper-air and other surface variables typically operates at longer lead times. Enforcing a uniform 1-hour lead time across all variables imposes a prohibitive computational burden. For example, a model would require 24 iterations to produce a single 1-day forecast, but the same number of rollouts at a standard 6-hour lead time would yield a 6-day forecast. This increase in iterations not only raises computational cost but also amplifies error accumulation, making pure high-resolution autoregression impractical for extended lead times.

Conversely, non-autoregressive forecasting attempts to directly predict the weather state at an arbitrary lead time Δt in a single step [43, 44], as

$$X_{t+\Delta t} = \mathscr{T}(X_t, \Delta t).$$

Although this non-autoregressive approach offers flexibility, it struggles with the chaotic nature of atmospheric dynamics. As the lead time Δt increases, the system's predictability decays exponentially, making a direct mapping from X_t to a distant $X_{t+\Delta t}$ both statistically weak and ill-posed. Consequently, while

reliable for nowcasting, purely non-autoregressive methods typically fail to maintain physical consistency and accuracy in forecasts extending beyond 24 hours.

To bridge the aforementioned gap, UniCast utilizes a semi-autoregressive forecasting approach [39]. This hybrid approach employs non-autoregressive direct prediction for a predefined set of short lead times (e.g., 1- and 6-hour), defined as

$$\hat{X}_{t+\Delta t} = \mathscr{T}(X_t, \Delta t), \quad \Delta t \in \{\Delta t_1, \Delta t_2, \ldots, \Delta t_n\}. \tag{1}$$

For lead times extending beyond the predefined set, the model changes to an autoregressive approach, using the last predicted weather state as the new initial condition

$$\hat{X}_{t+\Delta t_n+\Delta t} = \mathscr{T}(\hat{X}_{t+\Delta t_n}, \Delta t), \quad \Delta t \in \{\Delta t_1, \Delta t_2, \ldots, \Delta t_n\}. \tag{2}$$

The semi-autoregressive approach reconciles the two conflicting requirements: it delivers hourly precipitation predictions through a direct, non-autoregressive approach, while maintaining efficient upper-air predictions through autoregressive approach at longer lead times. By periodically refreshing the weather state with direct, non-autoregressive predictions, UniCast reduces the number of autoregressive rollouts needed to reach extended lead times, hence balancing high temporal resolution, computational efficiency, and forecast accuracy.

4.3 Model Structure

This section describes the structure of UniCast. We first outline the model input and boundary handling, then show the design of the large, foundational generalist and the small, pluggable specialist. A schematic overview has been provided in Figure 1.

4.3.1 Model Input

The input is composed of three distinct feature groups, providing atmospheric and temporal context:

- Weather state variables (X_t): High-dimensional tensors ($\mathbb{R}^{73\times441\times845}$) representing the state of 73 key weather variables listed in Table 1.
- Spatiotemporal forcing ($\mathbf{f}_t$): An auxiliary tensor ($\mathbb{R}^{9\times441\times845}$) encoding deterministic cycles and geographic contexts. It includes sine and cosine embeddings for diurnal and annual periodicities to capture temporal dynamics, alongside static embeddings for topography, land-sea masks, and latitude/longitude coordinates to enforce spatial features [12, 37].
- Lead-time embedding (Δt): A continuous scalar input specifying the forecast lead time, enabling the semi-autoregressive flexibility described in Section 4.2.

4.3.2 Boundary Handling

Typically, a regional model relies on forecasts from a concurrent global model to supply real-time boundary conditions [45]. Because boundary-condition handling falls outside the scope of this paper, in UniCast we adopt a direct boundary forcing strategy [37], which treats grid points along the domain perimeter as prescribed external inputs. Such a strategy enables UniCast to evolve the internal weather state independently, without requiring a concurrent global model.

A limitation of this direct strategy is that the reliability of the boundary inputs degrades with the lead time, progressively shrinking the effective spatial forecast area. To ensure fair and meaningful comparisons, we rigorously maintain identical boundary settings across all numerical experiments. Therefore, this constraint does not compromise the validity of our comparative results. On the contrary, it suggests that integrating more sophisticated boundary-handling techniques could unlock further performance gains for UniCast.

4.3.3 Generalist (Foundational Model)

The generalist that serves as UniCast's backbone, is a 40-million-parameter foundational model explicitly designed to capture complicated spatiotemporal atmospheric dynamics. Built upon a U-Net-style encoder-decoder structure [46], it is composed of three modules: input embedding, encoder-decoder processing, and output recovery.

Input Embedding. The generalist initiates processing by fusing upper-air and surface data into an aligned latent space of dimension d via parallel pathways:

- Surface Pathway: Surface variables (listed in Table 1) are concatenated with spatiotemporal forcing terms and projected via a 5×5 patch embedding into a latent tensor ($d \times 89 \times 169$).
- Upper-Air Pathway: Upper-air variables (listed in Table 1) are processed through a $2 \times 5 \times 5$ volumetric patch embedding, yielding a latent tensor ($d \times 7 \times 89 \times 169$).

Empirically, we choose $d = 192$. The two data streams are then merged: the surface latent tensor is expanded to a tensor of shape $d \times 1 \times 89 \times 169$ and concatenated with the upper-air tensor, resulting in a combined input representation of size $d \times 8 \times 89 \times 169$. In parallel, the scalar lead time is encoded into a d-dimensional latent vector via a 4-bit binary mapping and passed through a Multi-Layer Perceptron (MLP) to generate the conditioning signal required for adaptive Layer Normalization (adaLN).

Encoder-Decoder Processing. The encoder-decoder processing module is the core of UniCast's generalist. Both the encoder and the decoder consist of two distinct layers, connected by DownSample and UpSample operations that adjust the spatial resolution of the latent features. These layers differ in depth: the higher-resolution layer contains a stack of two transformer blocks, while the lower-resolution layer uses four blocks. Within each block, processing alternates between Window Multi-head Self-Attention (W-MSA) and Shifted Window Multi-head Self-Attention (SW-MSA), each followed by an MLP [30]. W-MSA restricts attention to a local three-dimensional window of size $(2, 6, 12)$ to capture fine-scale patterns efficiently. To overcome the isolation of these local windows, SW-MSA shifts the window partition by $(1, 3, 6)$ units. This shifting strategy is critical: it enables cross-window information exchange and broadens the effective receptive field without the computational cost of global attention, ensuring that features near window boundaries are seamlessly integrated into the representation.

To support variable lead times, we replace standard Layer Normalization with adaptive Layer Normalization (adaLN) [31]. The latter treats the forecast lead time Δt as a dynamic conditioning signal: it transforms the lead-time embedding into shift, scale, and gating vectors that modulate the normalized latent features. Shift and scale adjust the statistics of the features to reflect the expected variability of the target lead time, while gating controls the contribution of residual connections, allowing the network to dynamically emphasize or suppress information based on how far ahead it is forecasting. In this way, adaLN seamlessly injects temporal context into every stage of feature processing.

Output Recovery. UniCast's output recovery module, augmented by the skip connections from the encoder's first layer [47], fuses multi-scale features from decoder's last layer into a dense latent tensor of shape $2d \times 8 \times 89 \times 169$. The skip connections directly transfer high-resolution spatial information across the network, preventing the loss of fine details. To handle the distinct dimensionalities of surface (2D) and upper-air (3D) fields, the tensor is then split along the level dimension, isolating the single-level surface component from the 7-level upper-air component. Each stream is independently restored to its original grid resolution using 2D and 3D deconvolution operations, respectively. Finally, the reconstructed fields are merged into a single output tensor of size $73 \times 441 \times 845$, representing the complete forecast for all 73 weather variables at the target lead time.

4.3.4 Specialist (Adapter)

To enable precipitation-specific predictions without sacrificing the computational cost or stability risks of fully retraining the generalist (e.g., catastrophic forgetting), UniCast uses a small, pluggable adapter as the specialist. We freeze the 40-million-parameter generalist and trains only the adapter's 20.7 million parameters, leading to faster convergence and better stability. Moreover, the adapter avoids model collapse that frequently occurs when large-scale networks are fully retrained [48].

The specialist adopts a copy-and-connect strategy following the ControlNet structure: it contains a trainable copy of the generalist's encoder, whose outputs are connected to the frozen generalist's decoder through zero-initialized convolution layers. As illustrated in Fig. 1, the output of each encoder block in the specialist is passed through a zero convolution and then symmetrically injected into the corresponding decoder block of the generalist. The zero convolutions guarantee that, at the start of training, the specialist adds strictly zero-valued residuals, making it functionally identical to the generalist alone. This noise-free starting point shields the specialist's parameters from destabilizing gradients during the early adaptation phase. The specialist can accept additional modality-specific inputs; but for a fair comparison, we feed it the same inputs as the generalist: the current weather state and the forcing terms $\mathbf{c}_t = \{X_t, \mathbf{f}_t\}$, together

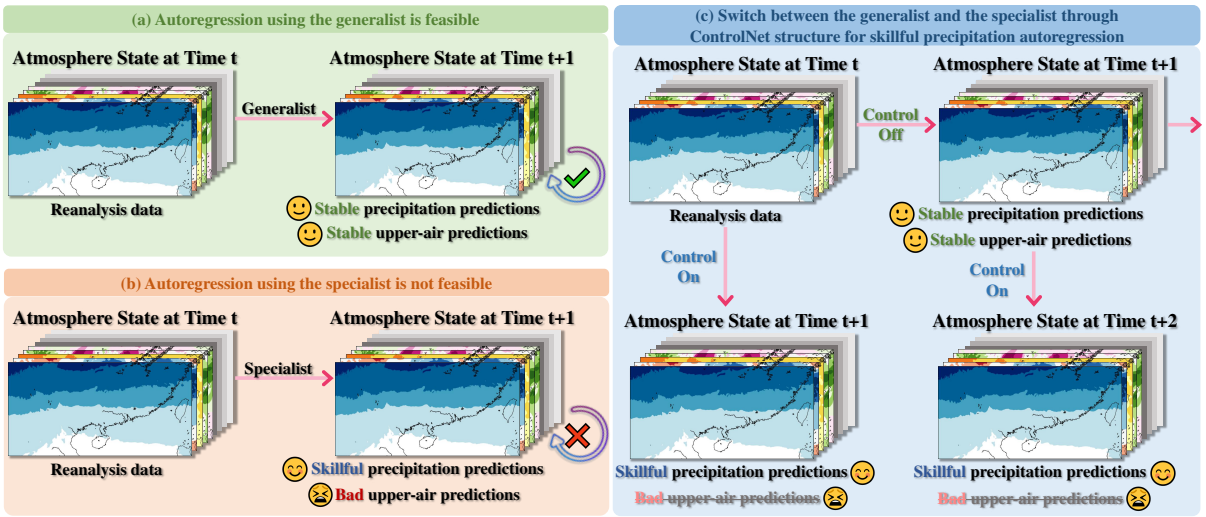


Fig. 6 UniCast's modality switching prevents autoregressive drift. (a) **Autoregression using the generalist.** Because the generalist is trained to predict all upper-air and surface variables, its predictions are stable enough to serve as initial weather state for subsequent autoregression. (b) **Autoregression using the specialist.** The specialist is an adapter that helps the generalist generate precipitation-specific predictions. This specialization degrades the accuracy of upper-air predictions, making its outputs unsuitable as initial weather state for further autoregression. In other words, the specialist suffers from autoregressive drift and cannot predict precipitation at longer lead times through autoregression alone. (c) **Autoregression by switching between the generalist and the specialist.** To resolve this conflict, we take advantage of UniCast's modality switching: we use the generalist to provide the initial weather state before target lead times, then switch to the specialist to predict precipitation for a single step, without running the specialist autoregressively.

with the lead time Δt. Over the course of training, the specialist gradually learns to inject precipitation-specific residuals, modulating the main processing stream to faithfully capture the dynamics of precipitation.

In an operational setting, the specialist acts as a modality router, allowing UniCast to switch between the generalist that propagates the broad weather state, and the specialist itself for precipitation predictions. As illustrated in Fig. 6, this decoupling can effectively prevent autoregressive drift, ensuring that optimizing for precipitation does not compromise the stability of the other weather variables needed for reliable autoregressive forecasting.

4.4 Training of Generalist

Let $\hat{X}$ denote the generalist's forecasts and X the corresponding observations. To capture diverse atmospheric dynamics, we train the generalist through minimizing a Variable-Weighted Mean Squared Error, $\mathcal{L}_{\text{generalist}}$, which sums the errors across all forecast initialization times, all spatial grid points, and a set of discrete lead times $\Delta T = \{1 \text{ hour}, 3 \text{ hours}, 6 \text{ hours}, 12 \text{ hours}\}$, as

$$\mathcal{L}_{\text{generalist}} = \frac{1}{|D_{\text{batch}}|} \sum_{d_0} \frac{1}{|\Delta T|} \sum_{\Delta t} \frac{1}{|\mathbf{S}|} \sum_{\mathbf{s}} \sum_{j} w_j (\hat{X}^{j,\mathbf{s}}_{d_0+\Delta t} - X^{j,\mathbf{s}}_{d_0+\Delta t})^2. \tag{3}$$

Here, we have:

- $d_0 \in D_{\text{batch}}$ denotes the forecast initialization times in a training batch,
- $\Delta t \in \Delta T = \{1 \text{ hour}, 3 \text{ hours}, 6 \text{ hours}, 12 \text{ hours}\}$ are the lead times,
- $\mathbf{s} \in \mathbf{S}$ indexes spatial grid points (latitude-longitude coordinates),
- $j \in J = \{\text{z1000, z850, ..., 10v, ps}\}$ indexes the weather variables listed on Table 1, and
- w_j is the weight assigned to variable j, specified in Fig. 7.

Fig. 7 introduces the alignment of variable-specific weights in $\{w_j\}$. To achieve stable convergence, we employ the AdamW optimizer paired with a cosine annealing scheduler. The training schedule begins with a one-epoch linear warm-up to mitigate early training instability, followed by a gradual decay of the learning rate from a peak of 5×10^{-4} down to 1×10^{-8} over a course of 100 epochs.

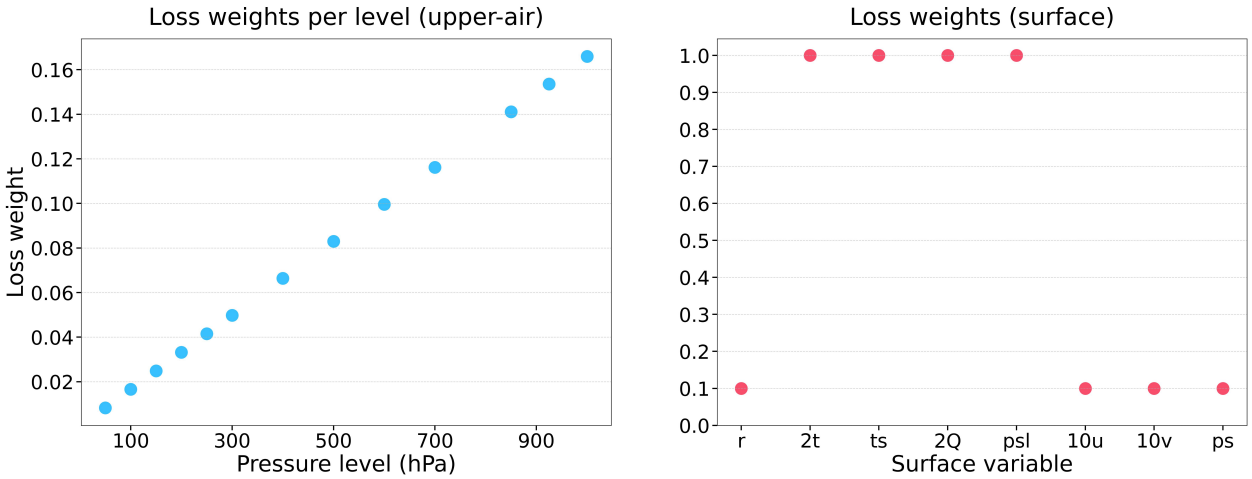


Fig. 7 Training loss weights. (a) Preset loss weights assigned to all pressure levels for upper-air variables. (b) Preset loss weights for surface variables. The final weight for each variable is computed via dividing its preset weight by the sum of all weights, ensuring that the normalized weights collectively sum to one.

4.5 Training of Specialist

To adapt the generalist for precipitation predictions, we freeze its parameters and train the specialist as an adapter guided by three principles. (1) Structural stability: by freezing the generalist, we preserve its ability to forecast the upper-air variables that the specialist's physics-assisted learning depends on. (2) Physical consistency: we incorporate the Vertically Integrated Moisture Budget (VIMB) equation so as to enforce water-vapor conservation. It ensures that precipitation forecasts are not merely statistically fit but physically grounded. (3) Targeted optimization: the Adjusted Cosine Similarity Loss (ACSL) focuses the specialist on the spatial structures and patterns of extreme precipitation, yielding a more stable adaptation than generic loss functions can provide.

4.5.1 Vertically Integrated Moisture Budget Constraint

We ground the specialist's training in atmospheric physics by enforcing the Vertically Integrated Moisture Budget (VIMB) equation. This constraint links precipitation predictions to Vertically Integrated Water Vapor (VIWV) and horizontal moisture flux divergence, guaranteeing that the model produces physically consistent outputs. It acts as a powerful regularizer to prevent unphysical forecasts (for instance, heavy rain appearing without moisture convergence) and improve accuracy by confining the solution to physically plausible states [49]. The governing equation is given by

$$\frac{\partial(\int_{p_t}^{p_s} qdp)}{\partial t} = -\alpha \int_{p_t}^{p_s} \frac{1}{\rho}(\frac{\partial \rho qu}{\partial x} + \frac{\partial \rho qv}{\partial y})dp - gR. \tag{4}$$

Here we have:

- q denotes the specific humidity, and $\int_{p_t}^{p_s} qdp$ represents the VIWV,
- $p \in \mathbb{R}$ denotes atmospheric pressure, where p_s is the surface pressure and $p_t = 50$hPa represents the top boundary of the atmospheric column,
- ρ is the air density, computed from the equation of state for moist air,
- u and v are the zonal and meridional winds, respectively,
- $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ denote the derivatives along the zonal and meridional directions,
- $\alpha \in \mathbb{R}$ is cloud percentage (cloud cover),
- $g = 9.8\text{m/s}^2$ is the gravitational acceleration, and
- R represents precipitation rate on the ground, which is considered as the result of condensation and evaporation along cloud column.

To maintain computational efficiency compatible with the model training, we employ the Euler's method with a 1-hour time step ($\Delta t = 1$h) for numerical integration, in which $\alpha = 0.79$ is chosen empirically to characterize the parameterized moisture transport from the environment into the cloud column.

As illustrated in Fig. 6(c), the VIMB equation is implemented as a sequential procedure that pairs the frozen generalist with the trainable specialist. Starting from the weather state at time t, the workflow enforces physical consistency in three steps.

1. **State generation by the generalist.** The generalist produces upper-air forecasts $\hat{X}$ at $t + 1$h and $t + 3$h. These provide the antecedent VIWV and moisture-advection terms required by the VIMB budget (see Eq. (5)). To anchor the physical constraint, all parameters of the generalist are frozen during this stage.

$$\mathrm{VIWV}^{\mathrm{pred}}(t + 3600\mathrm{s}) = \underbrace{\mathrm{VIWV}^{\mathrm{pred}}(t)}_{\text{From generalist}} \underbrace{- 3600 \times \alpha(\int_{p_t}^{p_s} (\frac{1}{\rho}\frac{\partial \rho \hat{q} u}{\partial x} + \frac{\partial \rho \hat{q} v}{\partial y}) dp)(t)}_{\text{From generalist}} - \underbrace{g\hat{R}(t \sim t + 3600\mathrm{s})}_{\text{From specialist}}. \quad (5)$$

2. **Coupled integration using the specialist and generalist.** From the upper-air forecast at $t+1$h and $t+3$h, the specialist predicts the cumulative 1-hour precipitation $\hat{R}$ for the intervals $[t+1\mathrm{h}, t+2\mathrm{h}]$ and $[t + 3\mathrm{h}, t + 4\mathrm{h}]$. We then apply Euler's method to combine the frozen upper-air fields $\hat{X}$ with the precipitation outputs $\hat{R}$ and compute the predicted VIWV, abbreviated as $\mathrm{VIWV}^{\mathrm{pred}}$, at $t + 2$h and $t + 4$h, exactly following the budget relation in Eq. (5).
3. **Supervised constraint.** A ground-truth $\mathrm{VIWV}^{\mathrm{obs}}$ is derived from the regional reanalysis and CMPA precipitation data using the identical Euler procedure. The specialist is optimized through minimizing the discrepancy between $\mathrm{VIWV}^{\mathrm{pred}}$ and $\mathrm{VIWV}^{\mathrm{obs}}$, which forces the specialist to produce precipitation fields that can physically explain the observed moisture evolution.

This constraint harmonizes UniCast's generalist and specialist by forming a physical feedback loop. When the generalist predicts strong moisture convergence but the specialist forecasts weak precipitation, the VIMB equation leads to a spurious accumulation of water vapor ($\mathrm{VIWV}^{\mathrm{pred}} > \mathrm{VIWV}^{\mathrm{obs}}$). Minimizing this discrepancy during training forces the specialist to increase its precipitation output, effectively "venting" the excess moisture and restoring budget balance. Conversely, a predicted moisture deficit compels the model to reduce the forecast rainfall. In this way, the constraint ensures that the two components produce physically consistent results, preventing a common failure mode in which individual variables appear accurate in isolation yet contradict each other when combined. However, pixel-wise loss functions such as MSE and CE amplify the discretization errors that arise from the numerical integration, potentially degrading precipitation accuracy. To overcome this issue, we instead optimize the specialist using the Adjusted Cosine Similarity Loss (ACSL), as described below.

4.5.2 Adjusted Cosine Similarity Loss

ACSL can shift the optimization priority from minimizing Euclidean distance to maximizing structural pattern alignment. Mathematically, ACSL acts as a proxy for the Pearson correlation coefficient. Through subtracting the shared mean (μ) from both the forecast ($\hat{\mathbf{v}}$) and observation ($\mathbf{v}$), the loss measures the cosine of the angle between the centered feature vectors, as

$$\mathcal{L}_{\mathrm{ACSL}}(\hat{\mathbf{v}}_t, \mathbf{v}_t) = 1 - \frac{\langle \hat{\mathbf{v}}_t - \mu, \mathbf{v}_t - \mu \rangle}{||\hat{\mathbf{v}}_t - \mu||_2 ||\mathbf{v}_t - \mu||_2} \in [0, 2]. \quad (6)$$

Here we have:

- $\hat{\mathbf{v}}_t$ and $\mathbf{v}_t$ denote the forecast and observed values of a given variable at time t,
- $\langle \cdot, \cdot \rangle$ denotes the inner product, and $|| \cdot ||_2$ the Euclidean norm, and
- μ is the shared mean of $\hat{\mathbf{v}}_t$ and $\mathbf{v}_t$, computed from training data.

We prefer ACSL over conventional loss functions for two reasons, each tied to the distinct requirements of our physics-assisted and primary tasks. (1) Structural tolerance for VIWV constraints: exact pixel-wise matching is harmful in the physics-assisted task because numerical discretization inevitably introduces

errors into the VIMB equation. Unlike MSE, which forces the model to replicate these noisy targets, ACSL prioritizes structural consistency. It attains the minimum whenever the forecast is an affine function of the observation, i.e., $\hat{\mathbf{v}}_t = k\mathbf{v}_t + (1-k)\mu,\ k > 0$. Thus, ACSL is invariant to linear scaling and shifts, allowing the model to learn the correct physical shape of the moisture field without overfitting the numerical noise from the solver. This flexibility makes it ideal for an auxiliary constraint: it guides the network toward a meteorologically plausible state rather than an exact but physically misleading match. (2) Counteracting blur in precipitation: the DLWP models trained with MSE tend to regress to the mean, producing overly blurry forecasts. ACSL provides an analytical corrective. As demonstrated in Eq. (7), the ACSL gradient is inversely proportional to the forecast's standard deviation $\hat{\sigma}$. It creates a built-in "gradient inflation": when the model outputs a low-variance, blurry field ($\hat{\sigma} \to 0$), the gradient magnitude increases sharply. The resulting penalty actively pushes the optimizer away from those blurry local minima, encouraging it to generate sharp, high-variance, and more realistic precipitation patterns.

$$\frac{\partial \mathcal{L}_{\text{ACS}}}{\partial \hat{\mathbf{v}}_t^{\mathbf{s}}} = \frac{1}{|\mathbf{S}|\hat{\sigma}} \left[\frac{(\hat{\mathbf{v}}_t^{\mathbf{s}} - \mu)\rho}{\hat{\sigma}} - \frac{\mathbf{v}_t^{\mathbf{s}} - \mu}{\sigma} \right]. \tag{7}$$

Here we have:

- $\hat{\mathbf{v}}_t^{\mathbf{s}}$ and $\mathbf{v}_t^{\mathbf{s}}$ are the forecast and observed values at time t and location $\mathbf{s}$,
- $\hat{\sigma}$ and σ are the spatial standard deviations of forecasts and observations, respectively, and
- ρ is the Pearson correlation coefficient between $\hat{\mathbf{v}}_t$ and $\mathbf{v}_t$.

From the analysis of the physics-assisted task and ACSL, we define the specialist's training objective as a combination of the primary precipitation loss and the physics-assisted VIWV constraint, as

$$\mathcal{L}_{\text{specialist}} = \gamma \mathcal{L}_{\text{precipitation}} + (1-\gamma)\beta \mathcal{L}_{\text{VIWV}}, \tag{8}$$

$$\mathcal{L}_{\text{precipitation}} = 1 - \frac{\sum_{d_0} \sum_{\Delta t} \sum_{\mathbf{s}} \hat{P}^{\mathbf{s}}_{d_0+\Delta t} * P^{\mathbf{s}}_{d_0+\Delta t}}{\sqrt{\sum_{d_0} \sum_{\Delta t} \sum_{\mathbf{s}} (\hat{P}^{\mathbf{s}}_{d_0+\Delta t})^2} \sqrt{\sum_{d_0} \sum_{\Delta t} \sum_{\mathbf{s}} (P^{\mathbf{s}}_{d_0+\Delta t})^2}}, \tag{9}$$

$$\mathcal{L}_{\text{VIWV}} = 1 - \frac{\sum_{d_0} \sum_{\Delta t} \sum_{\mathbf{s}} \hat{\text{VIWV}}^{\mathbf{s}}_{d_0+\Delta t} * \text{VIWV}^{\mathbf{s}}_{d_0+\Delta t}}{\sqrt{\sum_{d_0} \sum_{\Delta t} \sum_{\mathbf{s}} (\hat{\text{VIWV}}^{\mathbf{s}}_{d_0+\Delta t})^2} \sqrt{\sum_{d_0} \sum_{\Delta t} \sum_{\mathbf{s}} (\text{VIWV}^{\mathbf{s}}_{d_0+\Delta t})^2}}. \tag{10}$$

Here we have:

- $\hat{P}^{\mathbf{s}}_{d_0+\Delta t}$ and $\hat{\text{VIWV}}^{\mathbf{s}}_{d_0+\Delta t}$ denote the **normalized** precipitation and VIWV forecasts from the specialist at time $d_0 + \Delta t$ and spatial location $\mathbf{s}$,
- $P^{\mathbf{s}}_{d_0+\Delta t}$ and $\text{VIWV}^{\mathbf{s}}_{d_0+\Delta t}$ are the corresponding **normalized** observations,
- $d_0 \in D_{\text{batch}}$ denotes the forecast initialization times in a training batch,
- Δt indexes lead times (precipitation: $1, 3$ hours; VIWV: $2, 4$ hours),
- $\mathbf{s} \in \mathbf{S}$ indexes spatial grid points (latitude-longitude coordinates),
- $\beta = \mathbb{E}[\|\nabla \mathcal{L}_{\text{precipitation}}\|]/\mathbb{E}[\|\nabla \mathcal{L}_{\text{VIWV}}\|]$ is a scaling constant that equalizes the expected gradient norms of the two objectives, ensuring that the weighted VIWV loss $\beta\,\mathcal{L}_{\text{VIWV}}$ contributes comparably to the precipitation loss during optimization, and
- $0 \le \gamma \le 1$ is a convex combination coefficient that controls the trade-off between the precipitation loss and the physics-assisted VIWV loss.

Note that expressing the normalized variables in the form $\frac{v_t - \mu}{\sigma}$ recovers the ACSL loss (cf. Eq. (6)).

To keep the optimization balanced, we adopt a data-driven weighting scheme with two hyperparameters β and γ. Empirical monitoring shows a large-scale mismatch between the gradients of the precipitation and VIWV objectives. Therefore, we introduce and set $\beta = 15$ to align their magnitudes, preventing the primary task's stronger gradients from overwhelming the physical supervisory signal during backpropagation. While β equalizes the numerical influence of these two losses, the weighting parameter $\gamma = 0.95$ encodes our research priority: the precipitation loss receives a dominant weight of 0.95, and the auxiliary VIWV loss has a weight of 0.05. It ensures that the model is optimized primarily for precipitation forecast skill, with the moisture budget acting strictly as a regularizer to filter out the physically inconsistent solutions.

We employ the AdamW optimizer to update the 20.7 million trainable parameters within the specialist, while the generalist remains strictly frozen. To ensure stability during this high-precision adaptation, we implement a cosine annealing learning rate schedule. Over a focused duration of 10 epochs, the learning rate is smoothly decayed from 5×10^{-4} to 1×10^{-8}.

Table 1 Variables in the CMA dataset. Among them, upper-air variables are distributed across 13 vertical pressure levels (hPa): 1000, 925, 850, 700, 600, 500, 400, 300, 250, 200, 150, 100, 50.

Type	Full name	Abbreviation
Upper-air variables	Geopotential height	z
	Specific humidity	q
	Temperature	t
	U component of wind speed	u
	V component of wind speed	v
Surface variables	1-hour accumulative precipitation	r
	Temperature at 2m	2t
	Temperature at surface	ts
	Specific humidity at 2m	2q
	Pressure at sea level	psl
	U component of 10m wind speed	10u
	V component of 10m wind speed	10v
	Pressure at surface	ps

Table 2 Comparisons between UniCast, UniCast* and the other DLWP models in terms of CSI on the test set. Except for UniCast, the best model CSI at each lead time and threshold is in **bold** and the second best model CSI is underlined.

Model	CSI - 1-hour lead time				CSI - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast (generalist)	0.353	0.283	0.199	0.117	0.169	0.106	0.053	0.016
UniCast (specialist)	0.357	0.298	0.229	0.157	0.206	0.150	0.090	0.042
UniCast* (generalist)	0.343	0.274	0.198	0.131	0.161	0.110	0.066	0.031
UniCast* (specialist)	0.344	0.293	**0.223**	**0.153**	**0.190**	**0.137**	**0.086**	**0.047**
FourCastNet	0.341	0.282	0.197	0.120	0.169	0.112	0.055	0.017
Pangu	0.336	0.269	0.197	0.130	0.167	0.119	0.072	0.032
GraphCast	**0.372**	**0.307**	0.222	0.138	0.171	0.114	0.062	0.026
GraphFM	0.368	0.305	0.223	0.136	0.181	0.121	0.066	0.026
Stormer	0.336	0.270	0.188	0.117	0.179	0.119	0.064	0.026
YingLong	0.343	0.280	0.210	0.137	0.173	0.125	0.074	0.028

Data availability. The datasets generated and/or analysed during the current study are not publicly available due to commercial restrictions and the proprietary nature of the computationally generated data but are available from the corresponding authors on reasonable request.

Code availability. The source code for this study is available in Github and can be accessed via the link of https://github.com/Yumenomae/UniCast.

Table 3 Comparisons between UniCast, UniCast* and the other DLWP models in terms of FSS on the test set. Except for UniCast, the best model FSS at each lead time and threshold is in **bold** and the second best model FSS is underlined.

Model	FSS - 1-hour lead time				FSS - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast (generalist)	0.555	0.469	0.352	0.221	0.301	0.199	0.105	0.035
UniCast (specialist)	0.566	0.490	0.398	0.293	0.359	0.276	0.176	0.087
UniCast* (generalist)	0.544	0.459	0.353	0.249	0.289	0.208	0.133	0.067
UniCast* (specialist)	0.551	0.484	**0.390**	**0.287**	**0.336**	**0.254**	**0.169**	**0.098**
FourCastNet	0.541	0.466	0.346	0.221	0.297	0.205	0.107	0.033
Pangu	0.539	0.455	0.358	0.252	0.301	0.226	0.148	0.074
GraphCast	**0.586**	**0.505**	0.387	0.253	0.305	0.210	0.118	0.052
GraphFM	0.581	0.502	0.387	0.250	0.320	0.223	0.128	0.052
Stormer	0.553	0.464	0.342	0.227	0.329	0.229	0.128	0.055
YingLong	0.549	0.471	0.375	0.263	0.310	0.237	0.151	0.063

Table 4 Comparisons of different precipitation-specific strategies in terms of trainable parameters and memory usage.

Strategy	Trainable Parameters	Memory Usage
ControlNet Adapter	1.000×	1.000×
Full Model	1.932×	1.231×
Original Generalist	1.932×	1.230×

Table 5 Comparisons of different precipitation-specific strategies in terms of CSI on the test set. The best model CSI at each lead time and threshold is in **bold** and the second best model CSI is underlined.

Model	CSI - 1-hour lead time				CSI - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast* (generalist)	0.343	0.274	0.198	0.131	0.161	0.110	0.066	0.031
ControlNet Adapter	0.344	0.293	0.223	0.153	0.190	0.137	0.086	0.047
Full Model	**0.351**	**0.293**	**0.225**	**0.156**	**0.201**	**0.151**	**0.096**	**0.051**
Original Generalist	0.341	0.272	0.205	0.146	0.190	0.145	0.094	0.050

Table 6 Comparisons of different precipitation-specific strategies in terms of FSS on the test set. The best model FSS at each lead time and threshold is in **bold** and the second best model FSS is underlined.

Model	FSS - 1-hour lead time				FSS - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast* (generalist)	0.544	0.459	0.353	0.249	0.289	0.208	0.133	0.067
ControlNet Adapter	0.551	**0.484**	**0.390**	0.287	0.336	0.254	0.169	0.098
Full Model	**0.558**	0.483	0.390	**0.288**	**0.351**	**0.276**	**0.186**	**0.104**
Original Generalist	0.548	0.456	0.371	0.272	0.333	0.265	0.183	0.103

Table 7 Comparison of different training tasks in terms of CSI on the test set. The best model CSI at each lead time and threshold is in **bold** and the second best model CSI is underlined.

Model	CSI - 1-hour lead time				CSI - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast* (generalist)	0.343	0.274	0.198	0.131	0.161	0.110	0.066	0.031
Training w/ Physics	**0.344**	**0.293**	**0.223**	**0.153**	0.190	0.137	0.086	0.047
Training w/o Physics	0.336	0.286	0.217	0.149	**0.191**	**0.138**	**0.086**	**0.047**

Table 8 Comparison of different training tasks in terms of FSS on the test set. The best model FSS at each lead time and threshold is in **bold** and the second best model FSS is underlined.

Model	FSS - 1-hour lead time				FSS - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast* (generalist)	0.544	0.459	0.353	0.249	0.289	0.208	0.133	0.067
Training w/ Physics	**0.551**	**0.484**	**0.390**	**0.287**	**0.336**	0.254	0.169	0.098
Training w/o Physics	0.542	0.474	0.382	0.279	0.336	**0.255**	**0.170**	**0.100**

Table 9 Comparison of different loss functions for training in terms of CSI on the test set. The best model CSI at each lead time and threshold is in **bold** and the second best model CSI is underlined.

Model	CSI - 1-hour lead time				CSI - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast* (generalist)	0.343	0.274	0.198	0.131	0.161	0.110	0.066	0.031
Training w/ ACSL	0.344	**0.293**	**0.223**	**0.153**	**0.190**	**0.137**	0.086	0.047
Training w/ MSE	**0.351**	0.287	0.206	0.133	0.172	0.116	0.068	0.030
Training w/ MAE	0.313	0.253	0.190	0.128	0.134	0.098	0.062	0.029
Training w/ FACL	0.075	0.088	0.100	0.109	0.105	0.114	**0.088**	**0.050**

Table 10 Comparison of different loss functions for training in terms of FSS on the test set. The best model FSS at each lead time and threshold is in **bold** and the second best model FSS is underlined.

Model	FSS - 1-hour lead time				FSS - 3-hour lead time			
	1 mm	2 mm	4 mm	8 mm	1 mm	2 mm	4 mm	8 mm
UniCast* (generalist)	0.544	0.459	0.353	0.249	0.289	0.208	0.133	0.067
Training w/ ACSL	0.551	**0.484**	**0.390**	**0.287**	**0.336**	**0.254**	0.169	0.098
Training w/ MSE	**0.553**	0.477	0.365	0.251	0.307	0.219	0.136	0.066
Training w/ MAE	0.508	0.429	0.340	0.244	0.244	0.186	0.125	0.063
Training w/ FACL	0.232	0.261	0.274	0.261	0.269	0.252	**0.184**	**0.110**

References

[1] Ahmed, S. S., Bali, R., Khan, H., Mohamed, H. I. & Sharma, S. K. Improved water resource management framework for water sustainability and security. *Environ. Res.* **201**, 111527 (2021).

[2] Pielke, R. & Carbone, R. E. Weather impacts, forecasts, and policy: An integrated perspective. *Bull. Am. Meteorol. Soc.* **83**, 393–406 (2002).

[3] Bauer, P., Thorpe, A. & Brunet, G. The quiet revolution of numerical weather prediction. *Nature* **525**, 47–55 (2015).

[4] Coiffier, J. Fundamentals of numerical weather prediction (Cambridge University Press, 2011).

[5] Schultz, M. G. et al. Can deep learning beat numerical weather prediction? *Philos. T. R. Soc. A.* **379**, 20200097 (2021).

[6] Miller, C. A. & Edwards, P. N. Changing the atmosphere: Expert knowledge and environmental governance 31 (MIT Press, 2001).

[7] Majumdar, S. J. et al. Multiscale forecasting of high-impact weather: Current status and future challenges. *Bull. Am. Meteorol. Soc.* **102**, E635–E659 (2021).

[8] Sun, J. et al. Use of NWP for nowcasting convective precipitation: Recent progress and challenges. *Bull. Am. Meteorol. Soc.* **95**, 409–426 (2014).

[9] Khain, A., Ovtchinnikov, M., Pinsky, M., Pokrovsky, A. & Krugliak, H. Notes on the state-of-the-art numerical modeling of cloud microphysics. *Atmos. Res.* **55**, 159–224 (2000).

[10] Bi, K. et al. Accurate medium-range global weather forecasting with 3d neural networks. *Nature* **619**, 533–538 (2023).

[11] Kurth, T. et al. Fourcastnet: Accelerating global high-resolution weather forecasting using adaptive fourier neural operators. In *Proc. Platform for Advanced Scientific Computing Conference* (2023).

[12] Lam, R. et al. Learning skillful medium-range global weather forecasting. *Science* **382**, 1416–1421 (2023).

[13] Das, P. et al. Hybrid physics-ai outperforms numerical weather prediction for extreme precipitation nowcasting. *npj Clim. Atmos. Sci.* **7**, 282 (2024).

[14] Lavers, D. A., Simmons, A., Vamborg, F. & Rodwell, M. J. An evaluation of ERA5 precipitation for climate monitoring. *Q. J. R. Meteorolog. Soc.* **148**, 3152–3165 (2022).

[15] Gao, Y. et al. Oneforecast: A universal framework for global and regional weather forecasting. In *Proc. International Conference on Machine Learning* (2025).

[16] Ravuri, S. et al. Skilful precipitation nowcasting using deep generative models of radar. *Nature* **597**, 672–677 (2021).

[17] Zhang, Y. et al. Skilful nowcasting of extreme precipitation with nowcastnet. *Nature* **619**, 526–532 (2023).

[18] Tang, T. et al. Bias-targeted deep learning enhances short-range heavy rainfall forecasts. *npj Clim. Atmos. Sci.* **9**, 78 (2026).

[19] Shi, X. et al. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Proc. Annual Conference on Neural Information Processing Systems* (2015).

[20] Wang, Y. et al. Predrnn: A recurrent neural network for spatiotemporal predictive learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 2208–2225 (2023).

[21] Chen, L. et al. Fuxi: a cascade machine learning forecasting system for 15-day global weather forecast. *npj Clim. Atmos. Sci.* **6**, 190 (2023).

[22] Langousis, A. & Kaleris, V. Statistical framework to simulate daily rainfall series conditional on upper-air predictor variables. *Water Resour. Res.* **50**, 3907–3932 (2014).

[23] Baño-Medina, J. et al. A regional high resolution ai weather model for the prediction of atmospheric rivers and extreme precipitation. *npj Clim. Atmos. Sci.* **8**, 385 (2025).

[24] Kochkov, D. et al. Neural general circulation models for weather and climate. *Nature* **632**, 1060–1066 (2024).

[25] Hu, Y. et al. A hybrid framework for global weather forecasting via low-resolution dynamical core and multigrid neural operator. *npj Clim. Atmos. Sci.* (2026).

[26] Wang, C., Zhang, J. & Chen, H. Semi-autoregressive neural machine translation. In *Proc. Conference on Empirical Methods in Natural Language Processing* (2018).

[27] Ding, N. et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nat. Mach. Intell.* **5**, 220–235 (2023).

[28] Gao, Z. et al. Comprehensive comparisons of state-of-the-art gridded precipitation estimates for hydrological applications over southern china. *Remote Sens.* **12**, 3997 (2020).

[29] Dudhia, J. A history of mesoscale model development. *Asia-Pac. J. Atmos. Sci.* **50**, 121–131 (2014).

[30] Liu, Z. et al. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proc. International Conference on Computer Vision* (2021).

[31] Peebles, W. & Xie, S. Scalable diffusion models with transformers. In *Proc. International Conference on Computer Vision* (2023).

[32] Zhang, L., Rao, A. & Agrawala, M. Adding conditional control to text-to-image diffusion models. In *Proc. International Conference on Computer Vision* (2023).

[33] Li, X. et al. Analyzing and reducing catastrophic forgetting in parameter efficient tuning. In *Proc. International Conference on Acoustics, Speech, and Signal Processing* (2025).

[34] Trenberth, K. E. & Guillemot, C. J. Evaluation of the global atmospheric moisture budget as seen from analyses. *J. Clim.* **8**, 2255–2272 (1995).

[35] Rasp, S. et al. Weatherbench 2: A benchmark for the next generation of data-driven global weather models. *J. Adv. Model. Earth Syst.* **16**, e2023MS004019 (2024).

[36] Schaefer, J. T. The critical success index as an indicator of warning skill. *Weather Forecasting* **5**, 570–575 (1990).

[37] Oskarsson, J., Landelius, T., Deisenroth, M. P. & Lindsten, F. Probabilistic weather forecasting with hierarchical graph neural networks. In *Proc. Annual Conference on Neural Information Processing Systems* (2024).

[38] Xu, P. et al. An artificial intelligence-based limited area model for forecasting of surface meteorological variables. *Commun. Earth Environ.* **6**, 372 (2025).

[39] Nguyen, T. et al. Scaling transformer neural networks for skillful and reliable medium-range weather forecasting. In *Proc. Annual Conference on Neural Information Processing Systems* (2024).

[40] Hersbach, H. et al. The era5 global reanalysis. *Q. J. R. Meteorolog. Soc.* **146**, 1999–2049 (2020).

[41] Yan, C.-W. et al. Fourier amplitude and correlation loss: Beyond using l2 loss for skillful precipitation nowcasting. In *Proc. Annual Conference on Neural Information Processing Systems* (2024).

[42] Chan, P. W., Choy, C. W., Chiu, Y. Y. & He, J. Observational study of Super Typhoon Saola in 2023 when it was close to Hong Kong. *Weather* **80**, 258–265 (2025).

[43] Espeholt, L. et al. Deep learning for twelve hour precipitation forecasts. *Nat. Commun.* **13**, 5145 (2022).

[44] Nguyen, T., Brandstetter, J., Kapoor, A., Gupta, J. K. & Grover, A. Climax: A foundation model for weather and climate. In *Proc. International Conference on Machine Learning* (2023).

[45] Davies, T. Lateral boundary conditions for limited area models. *Q. J. R. Meteorolog. Soc.* **140**, 185–196 (2014).

[46] Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Proc. International Conference on Medical Image Computing and Computer Assisted Intervention* (2015).

[47] He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proc. Conference on Computer Vision and Pattern Recognition* (2016).

[48] Shumailov, I. et al. AI models collapse when trained on recursively generated data. *Nature* **631**, 755–759 (2024).

[49] Zhao, Y. et al. Enhancement of the summer extreme precipitation over North China by interactions between moisture convergence and topographic settings. *Clim. Dyn.* **54**, 2713–2730 (2020).

Acknowledgements. This work is supported by Guangdong Basic and Applied Basic Research Foundation (2024B1515520001), Shenzhen Science and Technology Program (KCXFZ20240903093759004), as well as GBA Meteorological S&T Collaborative Research Project (GHMA2024Y02).

Author contributions. J.Z. designed the project. Q.L. and Y.F. managed and oversaw the project. J.Z. performed the training and evaluation. All authors revised and approved the final manuscript.

Competing interests. All authors declare no financial or non-financial competing interests.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- appendix.zip