ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF STATISTICAL SCIENCES

Paolo Fortunati

Second Cycle Degree in Greening Energy Market and Finance

Curriculum: Renewable Technologies

GraphCast vs. IFS: A Comparative Analysis of Weather Forecasting Models for Optimized Renewable Energy Trading

Master's Thesis in Machine Learning and Artificial Intelligence

Supervisor:
Prof. Antonio Petruccelli

Co-Supervisor:
Prof. Matteo Amabili

Candidate:
Sveva Lombardini

Graduation Session: March 2025
Academic Year: 2023/2024

Contents

Abstract

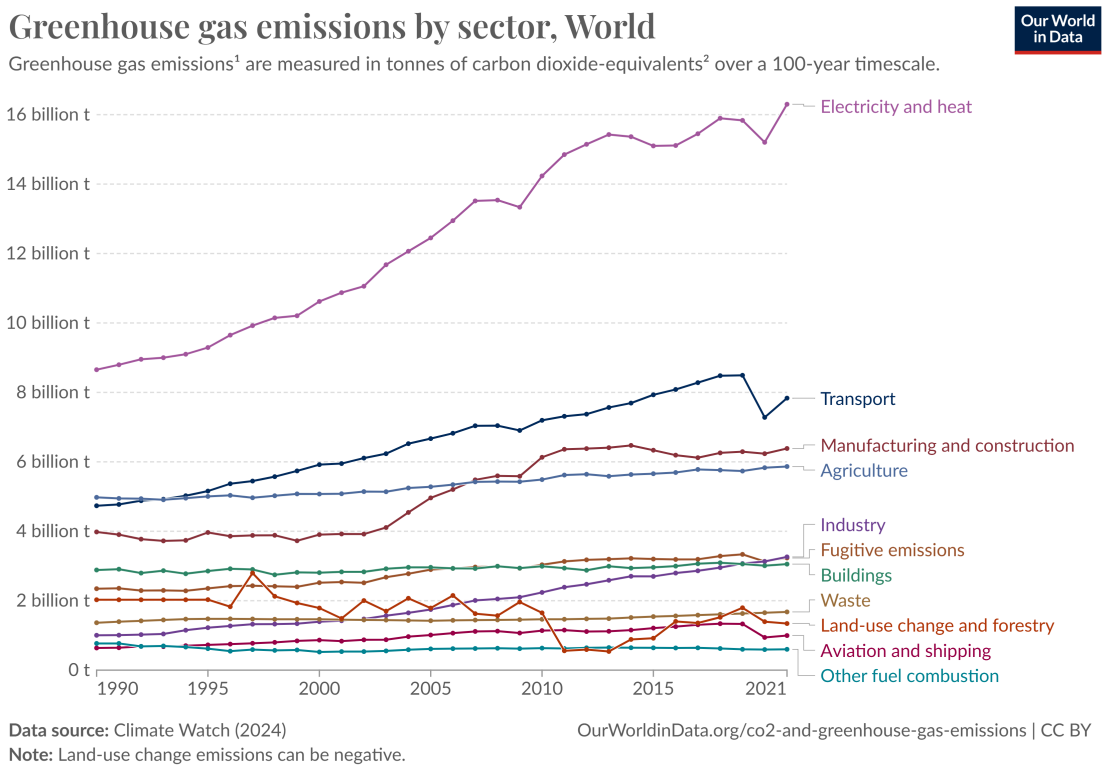
Renewable electricity generation is projected to grow significantly as the transition to sustainable energy sources accelerates in response to climate change. While the environmental benefits of this shift are clear, the increasing reliance on intermittent renewable resources poses new challenges to their efficient integration into the power grid. The energy output of solar panels and wind turbines, leading renewable technologies globally, is entirely dependent on weather conditions, which are inherently variable and difficult to predict. This uncertainty complicates the accurate forecasting of renewable energy profiles, making it challenging for energy traders to optimally place their bids on the market. As a result, grid stability becomes harder to maintain, and suppliers risk financial penalties for failing to meet supply commitments. Currently, Numerical Weather Prediction (NWP) models, particularly the Integrated Forecasting System (IFS) operated by the European Centre for Medium-Range Weather Forecasts (ECMWF), are considered the gold standard in weather simulation. However, in recent years, the field of Machine Learning Weather Prediction (MLWP) has emerged as a promising area of research due to its potential for improving prediction accuracy and efficiency. This thesis evaluates GraphCast, a machine learning-based forecasting tool developed by Google DeepMind, against ECMWF's IFS. The research specifically focuses on predicting wind speed and temperature—key factors influencing wind and solar energy outputs—by deploying one of GraphCast's publicly available pre-trained model versions. The performance of both models is assessed by comparing six months of forecasts with actual measurements from six SYNOP stations across Italy. Results show that GraphCast generally achieves accuracy comparable to IFS, demonstrating greater stability over time, with less degradation in predictive performance for extended forecasts. These findings highlight the potential of MLWP to complement or even enhance traditional numerical models in specific applications, particularly in long term forecasting. As weather prediction plays a pivotal role in optimizing renewable energy trading, the continued advancement of ML-based forecasting systems such as GraphCast is critical to shaping a cleaner, more resilient energy landscape aligned with global climate goals.

Chapter 1

Introduction

1.1 Renewable Energy Transition: A Strategic and Urgent Response to Climate Change

The transition from fossil fuels to sustainable energy sources is gaining momentum as part of global efforts to reduce greenhouse gas emissions and prevent severe climate change. As shown in Figure 1.1, electricity and heat generation represent the largest contributors to global greenhouse gas emissions, followed by transport, manufacturing, construction (mainly cement and similar materials), and agriculture [1]. In 2023, global energy-related CO_2 emissions reached a record 37.4 billion metric tons [2], underscoring the urgent need for decarbonization in this sector to achieve climate goals. According to the International Energy Agency's (IEA) report "CO_2 Emissions in 2023" [2], energy-related emissions increased by 1.1% compared to 2022. However, the growth of clean energy technologies has played a critical role in mitigating what could have been a more substantial rise [2].



1. Greenhouse gas emissions: A greenhouse gas (GHG) is a gas that causes the atmosphere to warm by absorbing and emitting radiant energy. Greenhouse gases absorb radiation that is radiated by Earth, preventing this heat from escaping to space. Carbon dioxide (CO_2) is the most well-known greenhouse gas, but there are others including methane, nitrous oxide, and in fact, water vapor. Human-made emissions of greenhouse gases from fossil fuels, industry, and agriculture are the leading cause of global climate change. Greenhouse gas emissions measure the total amount of all greenhouse gases that are emitted. These are often quantified in carbon dioxide equivalents (CO_2eq) which take account of the amount of warming that each molecule of different gases creates.

2. Carbon dioxide equivalents (CO_2eq): Carbon dioxide is the most important greenhouse gas, but not the only one. To capture all greenhouse gas emissions, researchers express them in "carbon dioxide equivalents" (CO_2eq). This takes all greenhouse gases into account, not just CO_2. To express all greenhouse gases in carbon dioxide equivalents (CO_2eq), each one is weighted by its global warming potential (GWP) value. GWP measures the amount of warming a gas creates compared to CO_2. CO_2 is given a GWP value of one. If a gas had a GWP of 10 then one kilogram of that gas would generate ten times the warming effect as one kilogram of CO_2. Carbon dioxide equivalents are calculated for each gas by multiplying the mass of emissions of a specific greenhouse gas by its GWP factor. This warming can be stated over different timescales. To calculate CO_2eq over 100 years, we'd multiply each gas by its GWP over a 100-year timescale (GWP100). Total greenhouse gas emissions – measured in CO_2eq – are then calculated by summing each gas' CO_2eq value.

Figure 1.1: Greenhouse gas emissions by sector, World [1]

The IEA outlines these challenges and solutions in its report "Net Zero Roadmap: A Global Pathway to Keep the 1.5 °C Goal in Reach" [3]. This report translates the Paris Agreement's critical goal of limiting global warming to 1.5 degrees Celsius (°C) into a concrete road map for the global energy sector. In [3], the IEA identifies the tripling of renewable energy capacity as the single largest driver of emissions reductions by 2030 in the Net Zero Emissions (NZE) by 2050 scenario. As shown in Figure 1.2, which illustrates CO_2 emissions reductions by mitigation measures in the NZE by 2050 scenario, the largest cumulative emissions savings come from wind and solar photovoltaic (PV), followed by advancements in energy efficiency and electrification.

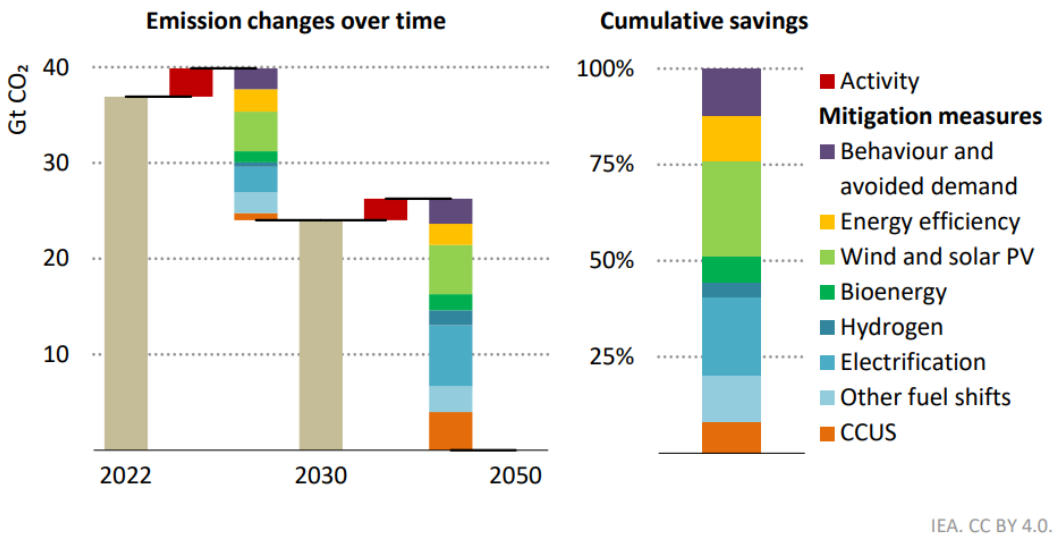


Figure 1.2: CO_2 emissions reductions by mitigation measure in the Net Zero Emissions by 2050 Scenario, 2022-2050 [3]. Activity = energy services demand changes from economic and population growth, CCUS = Carbon Capture, Utilization and Storage.

In addition to the urgent environmental drivers, the economic case for renewables is now stronger than ever. As renewable technologies become increasingly cost-effective, they accelerate the transition to cleaner energy sources. In its report "Renewable Power Generation Costs in 2023" [4], the International Renewable Energy Agency (IRENA) states that "renewable power generation has become the default source of least-cost new power generation". The global weighted average cost per unit of electricity from new solar PV and onshore and offshore wind power plants has fallen steadily year on year and is now significantly lower than that of fossil fuel-fired power plants. This cost evolution is shown in Figure 1.3 which illustrates the change in global weighted average Levelized Cost of Energy (LCOE)[1] for solar and wind compared to fossil fuels from 2010 to 2023. In 2023, the LCOE for new utility-scale solar PV installations was 56% lower than the weighted average of fossil fuel alternatives, having been 414% more expensive in 2010. For new onshore wind projects, the LCOE was even lower - 67% below the fossil fuel equivalent.

[1]Levelized cost of energy (LCOE) is a measure of the average net present cost of electricity generation for a power plant over its lifetime.

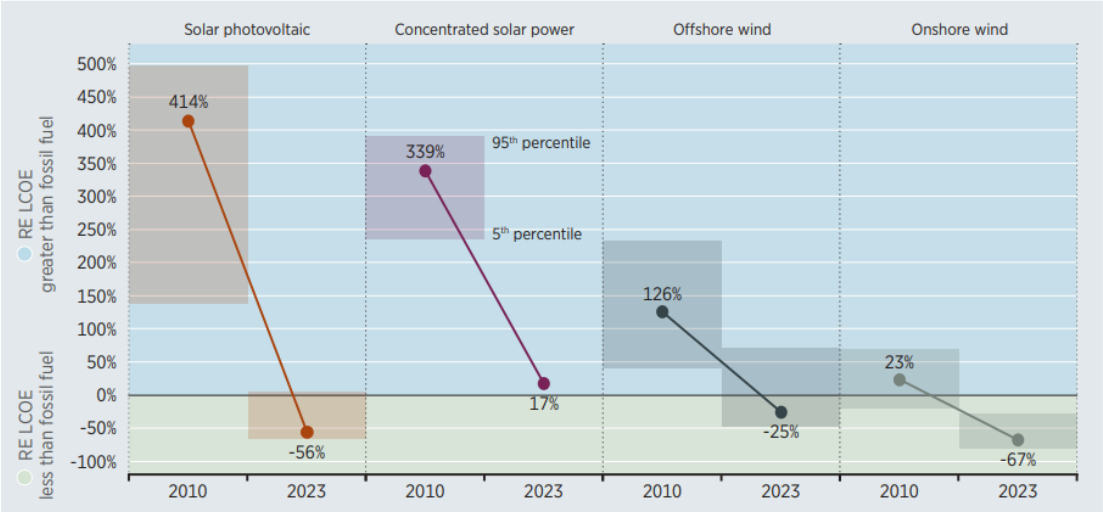


Figure 1.3: Change in global weighted average LCOE for solar and wind compared to fossil fuels, 2010-2023 [4]. RE = Renewable Energy, Concentrated solar power = mirrors or lenses to concentrate sunlight, generating heat for electricity production.

To fully understand the impact of these cost reductions, consider that "new renewable capacity added since 2000 is estimated to have reduced electricity sector fuel costs by at least $409 billion in 2023 alone" [4]. IRENA's 2023 report highlights the growing competitiveness of renewable energy technologies compared to carbon-based ones, despite fossil fuel prices returning to historical cost levels.

The urgent need to address climate change and the growing economic advantage of power generation from renewable energy sources are accelerating the transition to a renewable-based electricity sector. As a result, renewable power generation is poised for significant growth, paving the way for a cleaner, more resilient energy landscape that aligns with global climate objectives.

1.2 Forecasting Uncertainty in Renewable Energy Outputs: The Challenge of Weather Prediction

The efficient integration of solar PV and wind is critical to the NZE by 2050 scenario, as their share in total power generation in most regions reaches levels in 2030 seen only in a few countries today [3]. As shown in Figure 1.4, according to the NZE by 2050 scenario, solar PV and wind will lead the decarbonization of the electricity sector, becoming the largest sources of power by 2030.

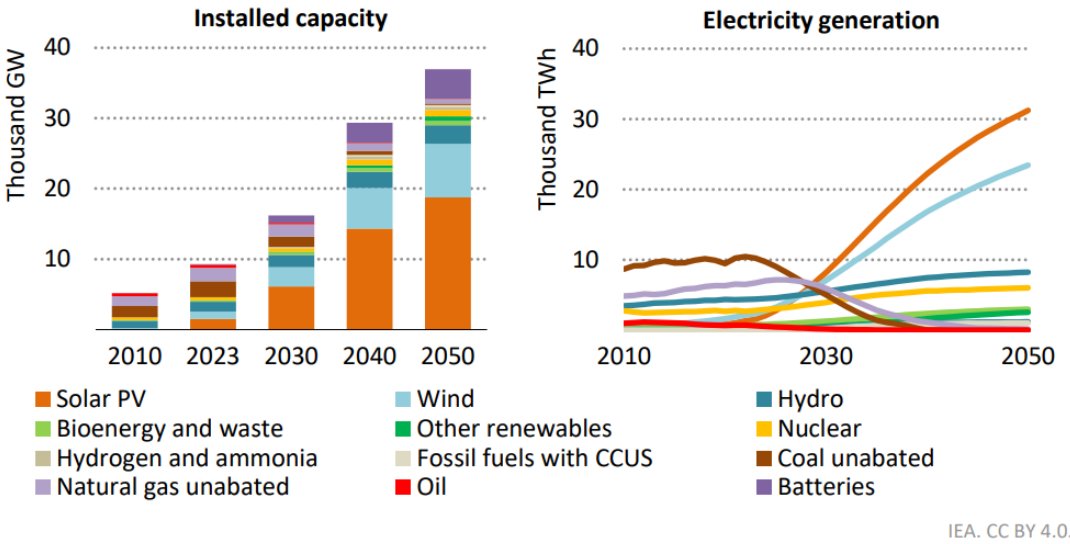


Figure 1.4: Total installed capacity and electricity generation by source in the Net Zero Emissions by 2050 scenario, 2010-2050 [3]. GW = GigaWatts, TWh = TeraWatts hours.

However, despite the clear environmental and economic benefits, integrating this increased installed capacity into existing electrical systems and markets presents significant challenges. These arise primarily due to the inherent uncertainty of predicted renewable energy outputs, as production from renewable energy systems can be difficult to forecast accurately. One of the key advantages of traditional coal and gas power plants is their reliability, as they provide a highly flexible supply of electricity around the clock. In contrast, the energy output of solar panels and wind turbines, leading renewable systems globally, entirely depends on weather conditions, which are inherently variable and challenging to predict [5].

The challenge of weather forecasting itself has deep historical roots, dating back to ancient civilizations, evolving gradually from simple observation of natural signs to a more scientific approach. The formalization of weather prediction as an initial value problem can be traced back to the early 20th century, when meteorologist Vilhelm Bjerknes [6] made significant strides in understanding the dynamics behind weather systems, laying the foundation for modern meteorology. The first step in any weather forecast is to gather a large amount of data from various sources to create the most accurate possible representation of the current state of Earth's atmosphere and surface (land and oceans) weather conditions. Real-time atmospheric, ocean and land-surface data are gathered from a coordinated network of individual surface- and space-based observing systems. This network includes radiosondes, satellites, buoys, radars, SYNOP[2] stations, as well as aircraft and ships, all designed to measure temperature, humidity, pressure, precipitation, wind, solar radiation, and other variables (see Figure 1.5). However, despite this advanced network

[2]SYNOP stands for SYNOPtic observations, a global network of ground-based meteorological stations that provide standardized weather reports at regular intervals.

of meteorological observation stations, the data collected is always limited—spatially and temporally—due to the vastness of the Earth's atmosphere and the unpredictability of weather patterns.

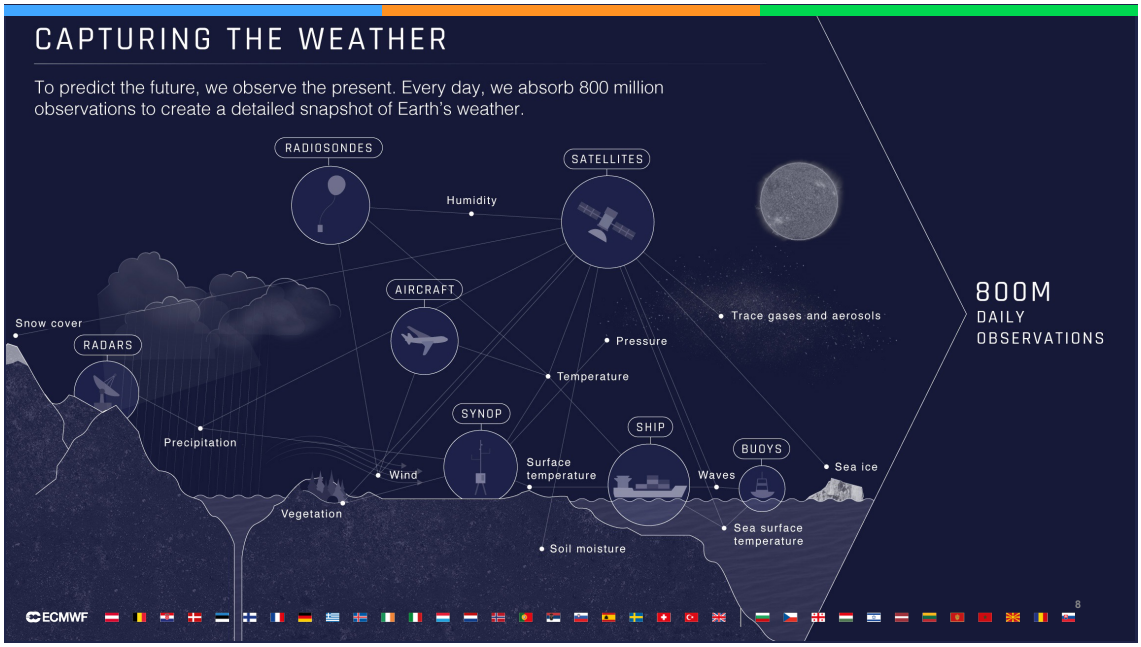


Figure 1.5: **ECMWF Data Collection Network.** Visualization of the extensive observational network used by the European Centre for Medium-Range Weather Forecasts (ECMWF), one of the leading organizations in global weather prediction. *Credit: Andrea Montani, ECMWF.*

According to Bjerknes [7], accurate weather prediction relies on two key elements:

1. Initial conditions must be characterized as accurately as possible.

2. The intrinsic laws governing atmospheric dynamics, which dictate how subsequent states evolve from previous ones, must be known.

Final forecasts of future atmospheric states are obtained by feeding the initial data into a model that approximates atmospheric dynamics [8]. This stands as the core challenge to accurate weather forecasting, as the atmosphere ranks among the most complex physical systems on Earth [9]. Building a model that can simulate the physical processes governing Earth's weather is a complex task that remains the focus of active research. As a result, weather forecasts are inherently uncertain, never reaching 100% accuracy due to the current inability to precisely translate the chaotic nature of the atmosphere into a model simulation, along with the limits of data availability.

One of the main obstacles to the efficient integration of renewables as a major source of electricity generation lies in this uncertainty, which directly affects the accuracy of predicted renewable energy profiles. Inaccurate forecasts of energy production hinder the ability of traders to participate effectively in the market, which in turn complicates

network management and increases the risk of financial penalties. The following section delves further into these dynamics and their impact on market operations.

1.3 The Role of Accurate Forecasting in Managing Grid Stability and Energy Trading Operations

In the wholesale electricity market, producers submit bids to sell electricity based on their anticipated output, and consumers submit bids to buy electricity based on their expected needs. These bids are matched, and transactions are settled at a market price (Figure 1.6).

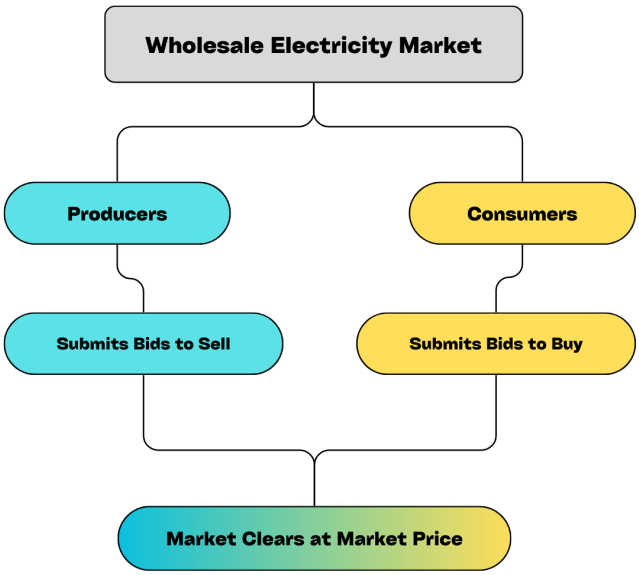


Figure 1.6: Flowchart of Wholesale Electricity Market Transaction Process

However, actual electricity production and consumption may differ from the forecasted amounts bought and sold on the market. When forecasts are inaccurate, grid management, a task performed by the Transmission System Operator (TSO), becomes increasingly complex. This added complexity can lead to penalties for producers who fail to meet their declared energy output, as imbalances—whether from under- or overproduction—affect system operational efficiency.

The electrical grid is an intricate network of power plants, transmission lines, substations, and distribution systems that deliver electricity from producers to consumers. For the grid to function reliably, it must remain in balance — meaning the amount of electricity generated must match the amount consumed at any given moment. Any imbalance, whether due to excess supply or demand, can lead to instability, potentially causing blackouts or equipment damage. Maintaining this balance is crucial, and it requires real-time adjustments to keep the frequency and voltage within safe limits, a task managed by

the TSO. If a producer generates less electricity than forecasted, they must purchase the shortfall from the TSO on the balancing market, where the imbalance price may be higher or lower than the market price for which they were originally paid.

When the system is subject to an energy surplus (i.e. long on energy), the imbalance price is typically lower than the market price, allowing the producer to buy the missing energy at a lower cost and make a profit. Conversely, if the system is experiencing an energy deficit (i.e. short on energy), the imbalance price will be higher than the market price, leading to a loss, as the producer must buy the shortfall at a higher price than what he was originally paid.

On the other hand, if the producer generates more electricity than forecasted, the TSO buys the surplus at the imbalance price, which also fluctuates based on the system's balance. When the system is long, the imbalance price is lower than the market price, causing the producer to sell their excess energy at a loss. If the system is short, the imbalance price is higher, leading to a profit for the producer, as they are paid more than they would have been on the market. These imbalance scenarios, along with the associated losses and profits for electricity producers, are summarized in Figure 1.7.

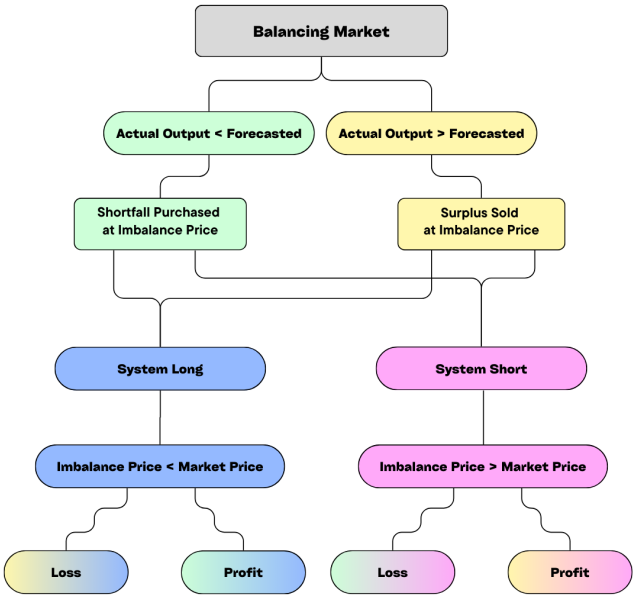


Figure 1.7: Imbalance Scenarios: Losses and Profits for Electricity Producers

Essentially, producers benefit financially when their output helps the TSO balance the grid — generating extra energy when the system is short, or producing less during surplus conditions. Conversely, they face penalties when their production complicates the TSO's balancing job, such as producing less in deficit conditions or generating excess energy in a surplus. Thus, improving the accuracy of forecasted energy profiles is crucial as electricity generation incrementally relies on renewable energy sources. Increased prediction reliability would allow suppliers to position themselves more effectively in the

market, assisting the TSO in maintaining grid stability and better managing the risks and opportunities in renewable energy trading [5].

1.4 Investigation Scope

This thesis evaluates the accuracy of weather forecasts generated using a pre-trained version of the GraphCast model, outsourced by DeepMind, and compares them against ECMWF's IFS forecasts. The focus is on predicting wind speed—a critical parameter for predicting wind energy production—and temperature, which directly influences the energy output of both wind turbines and solar panels. Model deployment was carried out during an internship at a renewable energy trading company, where accurate weather forecasts are crucial for predicting energy outputs on which market bids are based. By leveraging open-source data, pre-trained models, and publicly available code, this study aims to showcase a cost-effective and replicable implementation of GraphCast within an operational framework. Ultimately, this work seeks to assess GraphCast's performance and demonstrate its practical application in generating accurate weather predictions under real-world constraints for industries reliant on renewable energy.

Chapter 2 explores the two major approaches to medium-range weather forecasting, which involves predicting atmospheric variables up to 10 days in advance. This chapter starts with the foundational principles of Numerical Weather Prediction (NWP) and a general description of ECMWF's Integrated Forecasting System (IFS), which currently serves as the benchmark for traditional NWP. It then introduces Machine Learning Weather Prediction (MLWP) placing special emphasis on GraphCast, a global medium-range weather forecasting system developed by Google DeepMind in 2023 that leverages machine learning. This last section provides an overview of GraphCast's key features, including its autoregressive forecast generation process and innovative Graph Neural Network (GNN) architecture, and the detailed training process behind the system.

Chapter 3 provides an account of the methodology used to implement the selected pre-trained version of the GraphCast model, including the forecast generation process. It outlines the collection of initialization data and the key modifications made to the publicly available demonstration code to enable long-range operational forecasting.

Chapter 4 presents a comparative evaluation of six months of forecasts from both GraphCast and IFS, benchmarked against observational data from six SYNOP stations across Italy—covering various geographic regions. To assess the performance of both models, statistical metrics such as Pearson's correlation coefficient, Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) are employed.

Chapter 5 concludes with a discussion of the findings and insights drawn from this analysis.

Chapter 2

Evolution of Weather Forecasting: From Numerical Models to GraphCast

The following chapter begins by illustrating Numerical Weather Prediction (NWP), the traditional approach to weather forecasting, with a focus on its most advanced practical application: ECMWF's Integrated Forecasting System (IFS), currently considered the leading model in the field. It then explores Machine Learning Weather Prediction (MLWP), an area of growing research with significant potential. The last section delves into GraphCast, a machine learning-based weather forecasting system developed by Google DeepMind in 2023, which was used to obtain the forecasts center of this research.

2.1 Numerical Weather Prediction and the Integrated Forecasting System

Numerical Weather Prediction (NWP) models are based on systems of Partial Differential Equations (PDEs) that govern the evolution of the atmosphere [6]. These equations describe fluid dynamics [11], which are derived from the fundamental conservation laws of momentum, mass, and energy. Specifically, they illustrate how fluid substances, including air, respond to changes in factors such as pressure, temperature, and density. As a result, they serve as a fundamental tool for simulating wind patterns, precipitation, and other weather phenomena. The set of universal equations for NWP, commonly known as the primitive equations, includes [6]:

1. Newton's second law of motion or conservation of momentum, corresponding to fluid velocity in the x, y, and z directions (u, v, and w components)[1], collectively

[1] u, v, and w represent the velocity components along the x (east-west), y (north-south), and z (vertical) directions, respectively, with units of m s^{-1}.

known as the Navier-Stokes equations,

$$\frac{d\vec{V}}{dt} = -\alpha\vec{\nabla}p - \vec{\nabla}\Phi + \vec{F} - 2\vec{\Omega} \times \vec{V} \tag{2.1}$$

where $\vec{V} = (u, v, w)$ is the velocity vector (m s^{-1}), t is time (s), α is the specific volume (m^3 kg^{-1}), p is the pressure (Pa), Φ is the geopotential height (m^2 s^{-2}), $\vec{F}$ is the friction force per unit mass (m s^{-2}), and $\vec{\Omega}$ is the Earth's angular velocity vector (s^{-1}).

2. The continuity equation or conservation of mass,

$$\frac{\partial\rho}{\partial t} = -\vec{\nabla} \cdot (\rho\vec{V}) \tag{2.2}$$

where ρ is the air density (kg m^{-3}).

3. The equation of state for ideal gases,

$$p\alpha = RT \tag{2.3}$$

where R is the gas constant (for dry air $R_d \approx 287.05$ J kg^{-1} K^{-1}) and T is the temperature (K).

4. The first law of thermodynamics or conservation of energy,

$$Q = C_p\frac{dT}{dt} - \alpha\frac{dp}{dt} \tag{2.4}$$

where Q is heating per unit mass (J kg^{-1}), and C_p is the specific heat capacity at constant pressure[2] (≈ 1004 J kg^{-1} K^{-1} for dry air).

5. A conservation equation for water mass,

$$\frac{\partial\rho q}{\partial t} = -\nabla \cdot (\rho\vec{V}q) + \rho(E - C) \tag{2.5}$$

where q is the water vapor mixing ratio (kg kg^{-1}), and E and C represent evaporation and condensation rates respectively (kg m^{-3} s^{-1}).

While these primitive equations form the universal foundation for all NWP models, each model applies a unique combination or adaptation of these equations. These customizations account for the specific requirements of the model, such as spatial resolution,

[2] C_p represents the amount of heat required to raise the temperature of a unit mass of a substance by one degree while keeping the pressure constant.

geographic focus, or the inclusion of particular physical processes. This results in variations in how the equations are implemented and combined, making each model uniquely suited to its intended application.

To predict future atmospheric states, these PDEs must be solved using initial conditions derived from previous states [6], a task performed by supercomputers. Since these equations describe the dynamics of fluids, they involve derivatives representing infinitesimally small changes in space and time. While computers excel at arithmetic (addition, subtraction, multiplication, etc.), they cannot directly perform calculus, which is needed to solve continuous equations [11]. To overcome this, the PDEs are discretized [6], and so translated into discrete form that computers can process. Discretization involves dividing the atmosphere, or more generally any space-time domain, into a grid, where both space and time are divided into intervals. This process approximates the continuous derivatives by calculating the differences between adjacent grid points. After discretization, numerical methods are employed to solve these equations. These methods enable the computer to handle the problem by performing arithmetic operations on the discrete points of the grid. Enhancing NWP models is a complex, resource-intensive process led by highly trained experts developing increasingly accurate physical equations and leveraging increased computational power [9]. Developing these models requires expertise in atmospheric physics and a deep understanding of fluid dynamics and thermodynamics. Additionally, as models become more precise and detailed, they require increasingly sophisticated calculations, hence the need to invest in greater High-Performance Computing (HPC)[3] hardware.

The leading NWP-based system for medium-range weather forecasting is the Integrated Forecasting System (IFS), operated by the European Centre for Medium-range Weather Forecasts (ECMWF). The IFS combines both deterministic and probabilistic approaches to deliver some of the most accurate weather forecasts available today. The system generates a set of 50 possible future weather scenarios (ensemble members) by introducing small perturbations to both initial conditions and the original model configuration, to account for uncertainties in observations and model physics [12]. Together, the ensemble members form a probabilistic distribution of possible weather outcomes, as illustrated in Figure 2.1.

[3]High-Performance Computing (HPC) refers to the use of supercomputers and parallel processing techniques for solving complex computational problems.

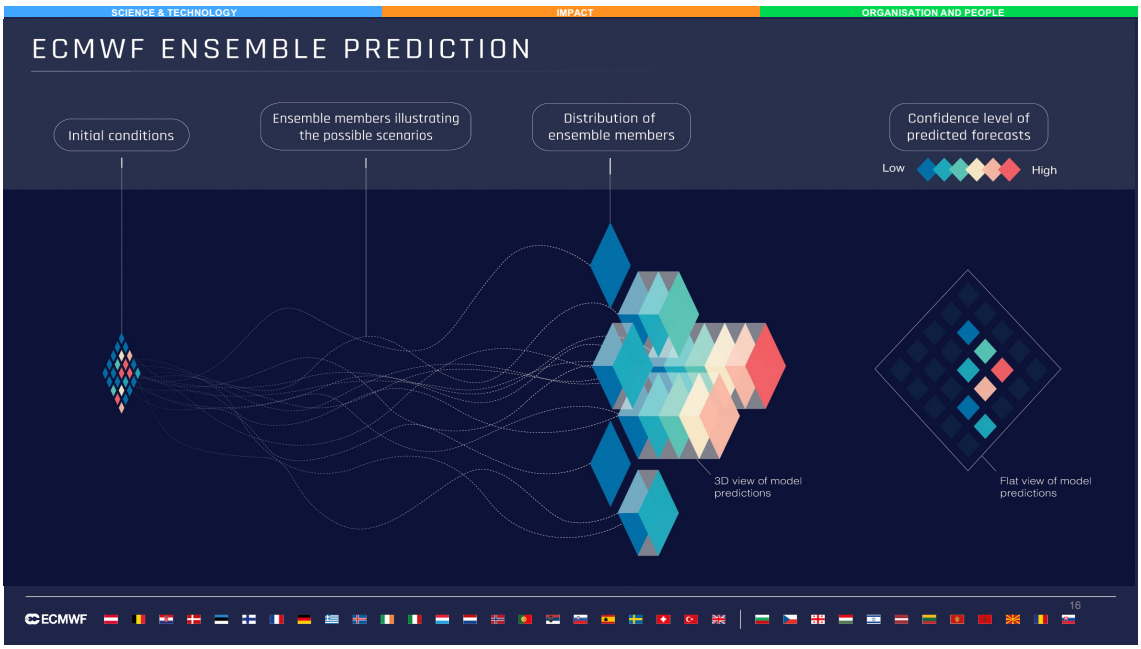


Figure 2.1: **ECMWF Ensemble Prediction Process.** The IFS generates 50 ensemble members by perturbing initial conditions and model configurations, providing a probabilistic distribution of forecasts with associated confidence levels. *Credit: Andrea Montani, ECMWF.*

Alongside these perturbed members, the system includes an unaltered model run, known as the ”ensemble control”, which serves as a deterministic forecast [4] [12]. This combination within the IFS allows users to access single-value forecasts as well as probabilistic ranges with associated confidence levels. Ensemble forecasts are typically visualized in time series plots (meteograms), which display key percentiles (e.g., 10th, 25th, median, 75th, and 90th). Figure 2.2 shows an ensemble meteogram of ECMWF's IFS forecasts for Bologna (44.46°N, 11.32°E – 54m altitude). The meteogram displays the control member forecast (High Resolution Forecast)[5] and the distribution of predictions from the 50 ensemble members (ENS Distribution) from Thursday, May 23, 2024, to Sunday, June 2, 2024, based on the 12 UTC [6] run times.

[4]The term ”deterministic” refers to a forecast generated from a single set of initial conditions and model configuration, as opposed to ensemble forecasts which are derived from multiple sets of initial conditions and modifications to the model configuration, with small variations introduced to account for uncertainty.

[5]The graph shows forecasts from an ECMWF IFS version prior to Cy49r1 (forecast cycle 49 revision 1), implemented after autumn 2024. In that version, the control member (High Resolution Forecast) had a 9 km resolution, while ensemble members had 18 km. As of Cy49r1, both now feature a 9 km resolution, thus the control member is no longer referred to as High Resolution Forecast.

[6]Coordinated Universal Time (UTC) is the primary time standard by which the world regulates clocks and time. It is consistent worldwide and does not observe daylight saving time.

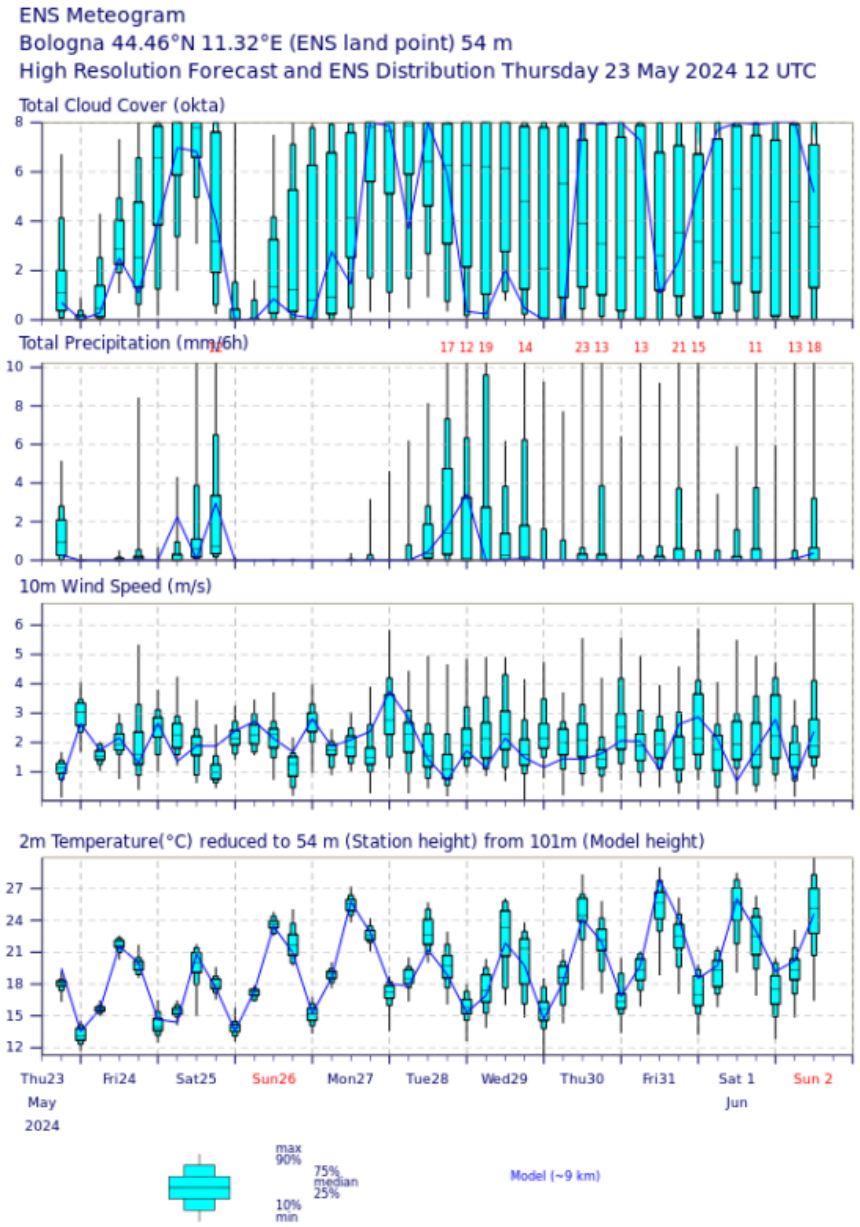


Figure 2.2: **Ensemble meteogram for Bologna (44.46°N, 11.32°E – 54m altitude) from Thursday, May 23, 2024, to Sunday, June 2, 2024, based on 12 UTC run times.** Predictions include Total Cloud Cover (okats: eighths of the sky covered by clouds), Total Precipitation (mm/6h), 10m Wind Speed (m/s), 2m Temperature (C°), with temperature values adjusted to Bologna's grid point elevation (54m) which is different from the nearest ENS model land grid point (101m). The blue line follows the control member single-value predictions while the vertical bars display the percentiles of the forecast distribution from the 50 ensemble members. *Credit: Andrea Montani, ECMWF.*

These visualizations enable users to interpret both deterministic forecasts and probabilistic ranges, providing insights into the likelihood of specific weather events. The ensemble control typically demonstrates higher average skill compared to individual perturbed members when performance is evaluated over many forecasts. However, in certain cases, a perturbed member can outperform the control, highlighting the value of the en-

semble in capturing a wide spectrum of possibilities [12].

The IFS produces forecasts up to 15 days ahead, with initialization times at 00 UTC and 12 UTC for both the ensemble control and the ensemble members. Additionally, shorter-term forecasts are produced at 06 UTC and 18 UTC, with the ensemble reaching up to 6 days and the control providing predictions for up to 3.5 days [12]. Both the ensemble and control forecasts of IFS share a horizontal resolution of approximately 9 km [12]. This means each weather state is represented by a grid cell with a spatial resolution of 0.1° latitude by 0.1° longitude. Additionally, both systems incorporate a vertical resolution of 37 levels [12], enabling the prediction of atmospheric variables across 37 distinct pressure levels, each corresponding to a specific height in the atmosphere (see Section 2.3.1 for a more detailed explanation of horizontal and vertical resolution).

2.2 Machine Learning Weather Prediction: Paving a New Way in Forecasting

Machine Learning-based Weather Prediction (MLWP) offers a different approach compared to traditional NWP: using data instead of physical equations to develop forecasting models. MLWP models are trained on decades of historical weather data to detect and learn the relationships that drive the evolution of weather systems from present to future states [9]. This data-driven approach has the potential to improve forecast accuracy by capturing complex patterns and scales in the data that may be difficult to represent in explicit equations [9].

Tom Mitchell, in his book "Machine Learning" [13], defined ML as follows: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E" [13]. In the context of machine learning for weather prediction, computer programs learn from past atmospheric observations (E), to predict future weather states (T), and their accuracy (P) continuously improves as they are exposed to higher-quality information. Unlike traditional programming, where input data is processed by a program explicitly designed based on knowledge of a specific phenomenon to produce results, ML-based systems are not programmed with predefined rules for a particular task. Instead, they learn automatically from data and have the ability to evolve and improve as more comprehensive records become available—an especially valuable feature in an era of changing climate, where weather patterns are continuously shifting. Additionally, this enables faster and more cost-effective model improvements, as adjustments are based on data rather than physical laws and increased computational power. Past weather observations have been collected for decades in extensive archives such as ECMWF's Meteorological Archive System (MARS) and combined with NWP models to

'fill in the blanks' where data is incomplete. This process enables the accurate reconstruction of a rich record of global historical weather data, known as reanalysis data such as ECMWF's ERA5 dataset. ERA5 is a state-of-the-art global reanalysis product offering atmospheric, surface, and oceanic variables spanning decades [14]. The dataset is freely available through the Copernicus Climate Data Store, making it an invaluable resource for scientists and researchers seeking to analyze historical weather patterns or validate models. The goal of machine learning is to identify relationships between input features and target outputs, which are then approximated into a model. In the context of MLWP, during training, models learn patterns and dependencies within reanalysis data, such as the relationships between variables defining initial states (e.g., temperature gradients) and target outputs (e.g., wind movements), in order to approximate the underlying physical processes. A machine learning model is "trained" by feeding historical data into an optimization algorithm. This algorithm aims to minimize a loss function, which quantifies the error between the model's predictions and the actual values, thereby measuring how well the model fits the data. The objective is to build a model that reduces this error as much as possible. Once trained, the model can be used for inference to generate predictions on new, unseen data, leveraging the learned relationships to approximate future outcomes.

ML-based methods also offer opportunities for increased efficiency, as they operate on modern deep learning hardware rather than traditional supercomputers. This hardware includes specialized processors like Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs), designed to handle the complex computations required for training and executing machine learning models more effectively. Generating a ML-based 10-day forecast takes less than a minute on a single GPU or TPU machine and consumes only a fraction of the energy needed for a conventional approach, such as IFS, which typically requires an hour of computation on a supercomputer that involves hundreds of machines simultaneously [10].

2.3 GraphCast Model Overview

In a recent paper [9], Google DeepMind introduced GraphCast, a new global medium-range deterministic weather forecasting system based on machine learning. The paper presents a comprehensive performance evaluation of GraphCast against ECMWF's High-RESolution (HRES) system, widely regarded as the most accurate operational deterministic weather simulation tool.

For clarity, it is important to note that with effect from Cy49r1, implemented after autumn 2024, IFS ensemble control member is no longer referred to as High-RESolution (HRES). Previously, IFS control member had a 9 km resolution (0.1 degree latitude-longitude), while ensemble members had a coarser resolution of 18 km (approximately 0.162 degrees latitude-longitude near the equator). Thus HRES is now known as the

ensemble control member of IFS whose output is equivalent to that of ex-HRES [12]. However, to align with the terminology used in the research paper by DeepMind [9], this chapter will continue to refer to the IFS control member as HRES.

According to [9], 1380 combinations of forecasts—spanning various meteorological variables, at different atmospheric pressure levels, and prediction lead times—were used to systematically evaluate the accuracy of HRES versus GraphCast, with GraphCast outperforming HRES on 90% of these verification targets. Two different baseline datasets were used to evaluate HRES and GraphCast forecasts: HRES-fc0 served as the ground truth for evaluating HRES forecasts, while ERA5 was used when assessing GraphCast. HRES-fc0 (”HRES forecast at step 0”) is a ground truth dataset constructed by Google DeepMind researchers to evaluate the skill of HRES in the research paper [9]. This dataset comprises the initial time step of each HRES forecast, at initialization times 00z, 06z, 12z, and 18z [7] (see Figure 3.1). For HRES-fc0 data, each time step corresponds to the HRES forecast at lead-time 0, essentially providing an ”initialization” from HRES.

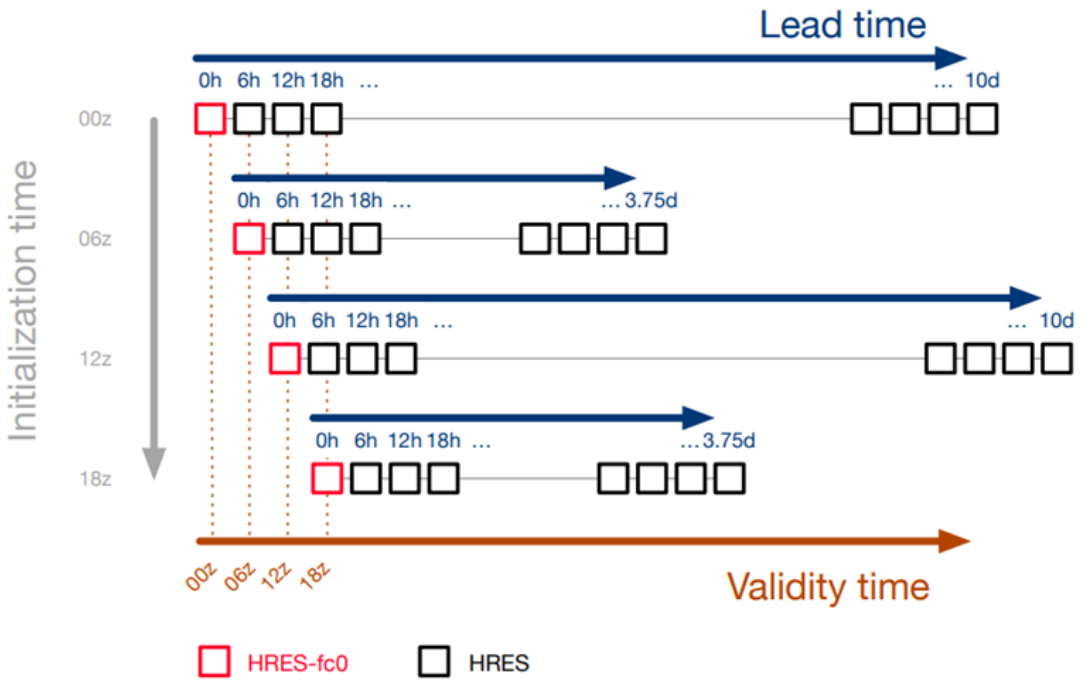


Figure 2.3: Schematic of HRES-fc0 [9]. Each horizontal line represents a forecast made by HRES, initialized at a different time (grey axis). In versions previous to Cy49r1 HRES forecasts initialized from 00z and 12z made predictions up to 10 days lead-time (blue axis), while HRES forecasts initialized from 06z and 18z made predictions up to 3.75 days ahead. Each square represents a state predicted by HRES, in 6 hour increments (states in the middle of a forecast trajectory are omitted from the schematic). Red squares represent the forecast at time 0 for each HRES forecast and define the data points included in HRES-fc0. The brown axis represents the validity time and allows visualizing the alignment of predictions from different initialization times.

[7] Zulu time (Z) is used to refer to UTC time. The notation 00z/06z/12z/18z indicates forecasts initialized at 00:00/06:00/12:00/18:00 UTC respectively.

Figure 2.4 shows GraphCast's RMSE skill and skill score versus HRES for the year 2018, as presented in the DeepMind research paper [9]. The absolute RMSE represents the raw error magnitude in predicting outcomes, where lower values indicate higher accuracy. In contrast, the RMSE skill score normalizes the RMSE difference between the two models, offering a relative measure of performance. Together, these metrics provide a comprehensive view of GraphCast's forecasting accuracy relative to HRES.

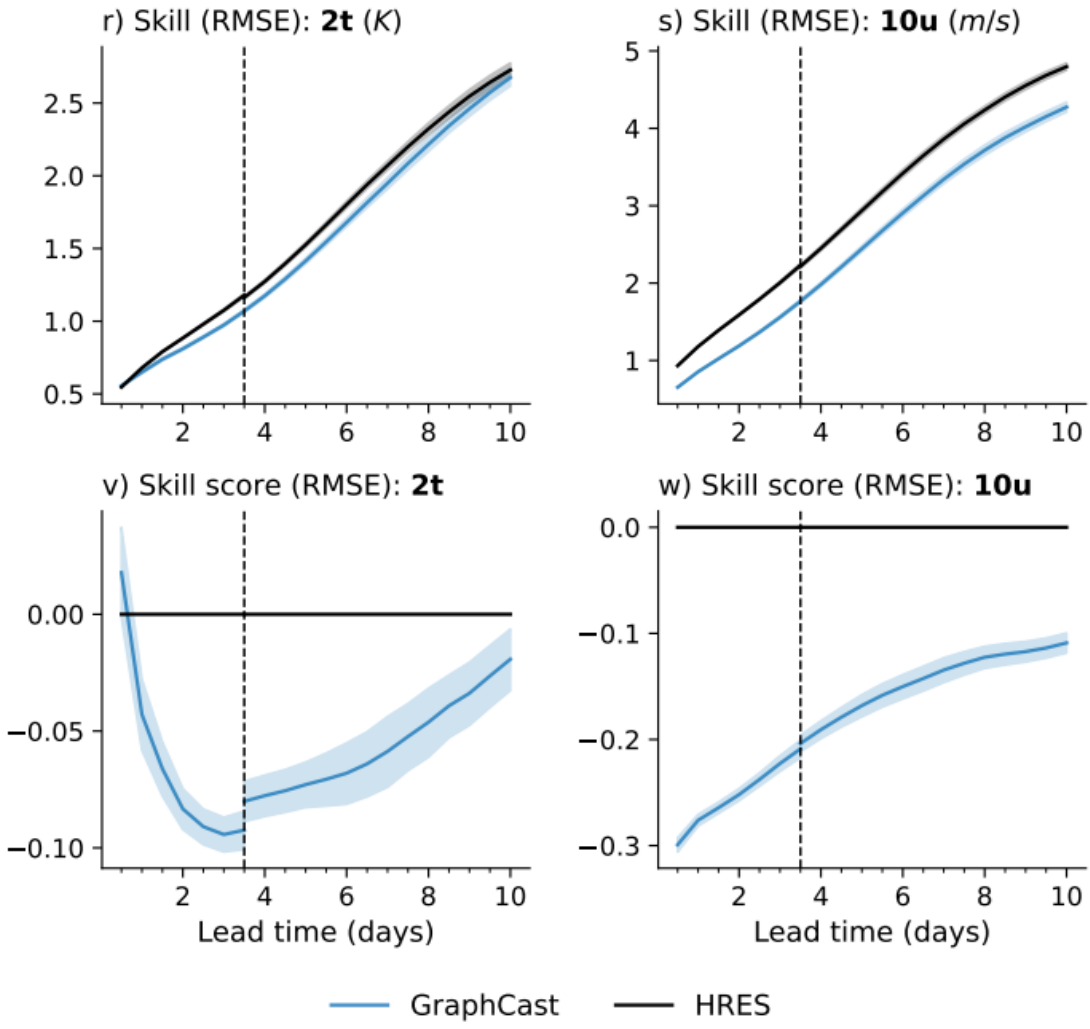


Figure 2.4: Root Mean Square Error (RMSE) skill and skill score (y-axis) for GraphCast (blue) and HRES (black) as a function of lead time for 2-meter temperature (2t) and 10-meter u wind component (10u). ERA5 and HRES-fc0 data from 2018 were used as ground truth for GraphCast and HRES, respectively. Lower values indicate better performance. [9].

The first row of Figure 2.4 displays the absolute RMSE values (y-axis) for GraphCast (blue line) and HRES (black line) calculated using Equation 2.6. The plots also include 95% confidence interval error bars, which indicate the range within which the true RMSE or RMSE skill score is expected to lie with 95% probability. The second row displays the RMSE skill score (y-axis), derived as the normalized RMSE difference between GraphCast and HRES using Equation 2.7). These graphs also include 95% confidence interval

error bars for clarity. The x-axis in all graphs corresponds to forecast lead times, measured in 12-hour intervals over a 10-day period. Across all four graphs, the data indicates that GraphCast consistently outperforms HRES. The absolute RMSE for GraphCast (blue line) is lower than that of HRES across all lead times. Furthermore, the negative skill score confirms that GraphCast demonstrates superior performance compared to HRES.

$$\text{RMSE}_{model} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i,model} - y_{i,baseline})^2} \tag{2.6}$$

Where:

- n: Total number of grid cells in the latitude-longitude grid.
- $y_{i,baseline}$: Actual observed weather value from the baseline dataset: ERA5 for GraphCast or HRES-fc0 for HRES, at the i-th grid cell.
- $x_{i,model}$: Predicted weather value from the forecasting model (GC or HRES) for the same i-th grid cell.

$$\text{Skill Score} = \frac{\text{RMSE}_{GC} - \text{RMSE}_{HRES}}{\text{RMSE}_{HRES}} \tag{2.7}$$

A vertical dashed line at 3.5 days marks the transition from HRES forecasts initialized at 06z/18z to those initialized at 00z/12z, which explains the discontinuity observed in GraphCast's skill score curves. Skill scores up to 3.5 days are computed between GraphCast (initialized at 06z/18z) and HRES's 06z/18z initialization, while after 3.5 days skill scores are computed with respect to HRES's 00z/12z initialization. Each set of initial conditions (06z/18z vs. 00z/12z) comes from a slightly different observational dataset and assimilation process. This can cause subtle differences in forecast accuracy. Since the RMSE skill score is a normalized difference between GraphCast's RMSE and HRES's RMSE, any significant change in HRES's RMSE after the transition can cause a noticeable discontinuity in the skill score curve.

The following paragraphs provide an in-depth description of the GraphCast model, detailing its architecture and forecasting process.

2.3.1 Autoregressive Forecast Generation, Grid State Representation and Modeled Weather Variables

GraphCast predicts global weather conditions up to 10 days in advance in under one minute employing a single Google Cloud TPU v4 device [8] [9]. To generate forecasts, GraphCast requires input from the two most recent states of Earth's weather—the current

[8] A Google Cloud TPU v4 is a specialized type of Tensor Processing Unit (TPU) developed by Google.

state and the state from six hours prior. Based on these consecutive inputs, GraphCast predicts the next weather state, six hours ahead [9]. Like traditional NWP-based systems, GraphCast employs an autoregressive forecast generation process: its prediction cycle can be repeated iteratively, using each newly predicted state as the ”current” weather state and the previous prediction as the state from six hours earlier. By feeding its own forecasts back in as input, a process known as “autoregressive roll-out” (Figure 2.5c), GraphCast can produce a 10-day sequence of weather states, with updates provided in 6-hour increments [9]. A single weather state, both as input (Figure 2.5a) and as forecasted output (Figure 2.5b), is represented by a cell with a spatial resolution of 0.25° latitude by 0.25° longitude. This resolution creates a global grid comprising 721 cells along the latitude axis and 1440 cells along the longitude axis, resulting in a total of $721 \times 1440 = 1,038,240$ grid points that span the entire Earth's surface [9]. At the equator, each grid cell covers an area of approximately 28 x 28 square kilometers. However, due to the Earth's curvature, the distance covered by each degree of longitude decreases as latitude increases. As a result, each 0.25° x 0.25° grid cell represents a progressively smaller area (in square kilometers) as you move from the equator toward the poles. Each cell on the grid represents a vertical slice of the atmosphere, 0.25 degrees wide in latitude and longitude, to which a specific set of surface and atmospheric variables (listed in Table 2.1) is associated [9].

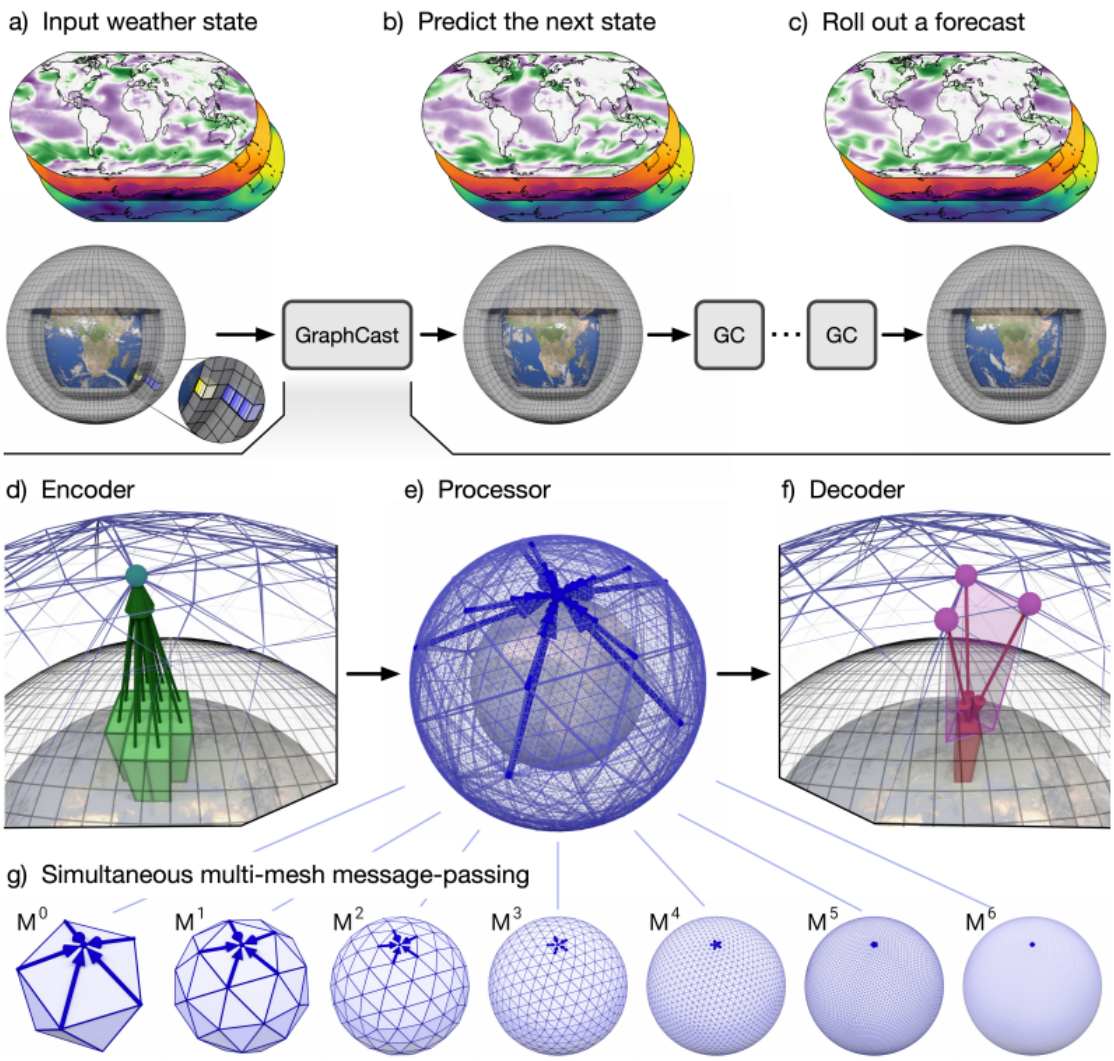


Figure 2.5: GraphCast Model Schematic [9]. (a) Input grid constituted of single input weather states on 0.25° latitude-longitude cells comprising a total of $721 \times 1440 = 1,038,240$ grid points. Yellow layers in the close-up window represent the 5 surface variables. Blue layers represent the 6 atmospheric variables, repeated at 37 pressure levels ($5+6\times37 = 227$ variables per point in total), resulting in a global weather state representation of $235,680,480$ values used as starting point to generate the forecasts. (b) GraphCast predicts the next state of the weather on the grid 6 hours into the future. (c) Autoregressive generation of a forecast (roll-out): GraphCast is iteratively applied to the previous forecast and the state before to predict the new state. (d) The Encoder maps two consecutive regions from the latitude-longitude grid to a node representation on the multi-mesh. (e) The Processor performs message-passing on the multi-mesh, enabling simultaneous local and large-scale information propagation. (f) The Decoder maps the learned features back to the grid, providing a prediction for the next time step as a residual update. (g) The multi-mesh is derived from a regular icosahedron dividing each triangular face recursively into 4 smaller triangles. Starting with the base mesh (M^0), with 12 nodes, the process ends with the final mesh (M^6), which consists of 40,962 nodes. This creates a flat hierarchy with varying edge lengths across resolution levels, allowing simultaneous message-passing across multiple resolutions while maintaining a consistent structure.

The yellow square in the close-up window in Figure 2.5 a) represents the 5 surface variables modeled by GraphCast, while the blue squares denote the six atmospheric vari-

ables, repeated across 37 different pressure levels. These values provide a comprehensive view of weather states, both at the Earth's surface and throughout the atmosphere at various altitudes. With a total of $5 + 6 \times 37 = 227$ values per cell, this yields a global representation of Earth's weather state comprising $227 \times 1,038,240 = 235,680,480$ values [9]. Table 2.1 synthesizes all the weather variables and pressure levels modeled by GraphCast, which are also listed and briefly detailed below.

Surface Variables:

1. **2-meter temperature (2t)** – Air temperature measured at a height of 2 meters above the ground (K).
2. **10-meter u wind component (10u)** – The horizontal wind component in the east-west direction at 10 meters above the ground (m s^{-1}), positive u for winds from the east, negative u for winds from the west.
3. **10-meter v wind component (10v)** – The horizontal wind component in the north-south direction at 10 meters above the ground (m s^{-1}).
4. **Mean sea-level pressure (msl)** – The atmospheric pressure at sea level (hPa).
5. **Total precipitation (tp)** – The accumulated precipitation (rainfall or snowfall) over 6 hours (m).

Atmospheric Variables:

1. **Temperature (T)** – Air temperature at each atmospheric pressure level (K).
2. **U component of wind (U)** – The east-west component of wind at different pressure levels (m s^{-1}).
3. **V component of wind (V)** – The north-south component of wind at different pressure levels (m s^{-1}).
4. **Geopotential (z)** – Gravitational potential energy per unit mass at a given pressure level (m^2 s^{-2}).
5. **Specific humidity (q)** – The amount of water vapor per unit mass of air at a given pressure level (kg kg^{-1}).
6. **Vertical wind speed (w)** – The speed of vertical air movement at each pressure level (m s^{-1}). Vertical wind speed indicates whether air is rising (positive w) or sinking (negative w).

Pressure Levels:

Each pressure level represents a specific value of atmospheric pressure in hPa (hectoPascals), which corresponds to a certain height. These heights are not fixed, as atmospheric pressure varies depending on factors such as temperature and density. For instance, the 1000 hPa level represents the altitude at which the atmospheric pressure is 1000 hPa in that particular forecast.

- **1000 hPa to 850 hPa** – Near-surface levels
- **800 hPa to 500 hPa** – Mid-levels
- **450 hPa to 200 hPa** – Upper levels
- **150 hPa to 1 hPa** – Very high levels

Surface Variables (5)	**Atmospheric Variables (6)**	**Pressure Levels (37)**
2-meter temperature (2t)	Temperature (T)	1, 2, 3, 5, 7, 10, 20, 30, 50, 70,
10-meter u wind component (10u)	U component of wind (U)	100, 125, 150, 175, 200, 225,
10-meter v wind component (10v)	V component of wind (V)	250, 300, 350, 400, 450, 500,
Mean sea-level pressure (msl)	Geopotential (z)	550, 600, 650, 700, 750, 775,
Total precipitation (tp)	Specific humidity (q)	800, 825, 850, 875, 900, 925,
	Vertical wind speed (w)	950, 975, 1000

Table 2.1: Weather variables and pressure levels modeled by GraphCast [9]. Pressure levels are expressed in hPa (hectoPascal).

2.3.2 Graph Neural Network Architecture

GraphCast model architecture is implemented as a Graph Neural Network (GNN) which follows an "encode-process-decode" sequence to generate forecasts [9]. GNNs are particularly effective in modeling data with spatial dependencies, as they can capture arbitrary sparse patterns of spatial interactions. This makes them especially suitable for modeling weather dynamics, which are characterized by intricate interdependencies across different regions of the Earth [9]. In GNNs, data is structured as a graph, where each region of interest (point on the grid) is represented as a node. The interactions between these regions—such as how one region's weather affects another—are represented as edges connecting these nodes [9]. The internal "multi-mesh" graph representation (Figure 2.5g) allows capturing dependencies between nodes (regions) that span large distances within a few message-passing steps [9]. This enables the model to learn the underlying weather state without being computationally expensive. A critical distinction of the multi-mesh representation is its homogeneous spatial resolution, meaning it provides a consistent level of detail across the entire globe [9]. This stands in contrast to the latitude-longitude

grid, which suffers from a non-uniform distribution of grid points. In a latitude-longitude grid, as one moves toward the poles, the grid cells shrink, resulting in higher resolution and more computationally expensive processing in polar regions. GraphCast's multi-mesh representation, by contrast, avoids this issue, as it maintains a uniform resolution across the globe and does not suffer from the disproportionate computational load at higher latitudes. In addition, the traditional latitude-longitude grid tends to constrain interactions with adjacent regions, in contrast, GraphCast can model arbitrary sparse interactions, meaning that weather dependencies between regions that are not directly adjacent can still be modeled.

The multi-mesh graph used in GraphCast is derived from a regular icosahedron, a geometric shape with 12 nodes, 20 faces, and 30 edges. To create the final multi-mesh structure, each triangular face of the icosahedron is recursively subdivided into 4 smaller triangles [9]. Each refinement step is repeated six times and increases the number of faces and edges by four while maintaining the overall structure. Starting with the base mesh (denoted as M^0), which contains 12 nodes, the process culminates with the final mesh (M^6), which consists of 40,962 nodes (Figure 2.5g) [9]. The refinement process results in a series of meshes with increasingly higher resolution, preserving all the edges of each intermediate mesh from M^0 to M^6 [9]. This creates a flat hierarchy of edges, where the edges at each level have varying lengths depending on the mesh resolution, but the overall structure remains consistent across all levels. The hierarchical nature of the multi-mesh structure enables message-passing over edges that span multiple levels of resolution [9]. During the message-passing process, information flows from node to node through the edges, and all edges are treated as bi-directional, meaning each edge is counted twice, once for each direction. The learned message-passing occurs simultaneously across all mesh levels (Figure 2.5g) [9]. As a result, each node is updated based on information from all of its incoming edges, regardless of which mesh level those edges belong to. This allows the model to learn from high-resolution and low-resolution interactions at the same time, leveraging both fine-grained local details and coarse global patterns. The table below provides the number of nodes, faces, and edges (including multilevel edges) for the multi-mesh structure at each refinement level from level 0 (M^0) to level 6 (M^6).

Refinement Level	Num Nodes	Num Faces	Num Edges	Num Multilevel Edges
0	12	20	60	60
1	42	80	240	300
2	162	320	960	1,260
3	642	1,280	3,840	5,100
4	2,562	5,120	15,360	20,460
5	10,242	20,480	61,440	81,900
6	40,962	81,920	245,760	327,660

Table 2.2: Multi-mesh refinement statistics [9]. Number of nodes, faces, edges, and multilevel edges for each refinement level, from the base mesh (level 0) to the final mesh (level 6) used in GraphCast. The multi-mesh structure undergoes recursive refinement, resulting in progressively higher resolution and more complex connections between nodes at each level.

GraphCast architecture is characterized by an "encode-process-decode" configuration:

- **The Encoder** (Figure 2.5, point d, and Figure 2.6, point a) component maps two consecutive input regions on the latitude-longitude grid (green boxes), to a node representation on the multi-mesh internal graph (green, upward arrows which terminate in the green-blue node).

  This component uses a single GNN layer[9], which transforms the input data (surface and atmospheric variables) in node attributes on the multi-mesh.

- **The Processor** (Figure 2.5, point e, and Figure 2.6, point b) performs learned message-passing on the multi-mesh (heavy blue arrows that terminate at a node) using 16 unshared GNN layers.

  Unlike hierarchical approaches, the processor does not require explicit hierarchy between higher and lower resolution edges, thus enabling simultaneous local and large-scale information propagation with few message-passing steps.

- **The Decoder** (Figure 2.5, point f, and Figure 2.6, point c) component maps the learned features on the multi-mesh (purple nodes) back onto the grid representation (red, downward arrows which terminate at a red box).

  The output is presented as a prediction for the next time step, expressed as a residual update to the most recent input state.

  This component uses a single GNN layer, similar to the encoder, to translate the processed features into a forecasted state.

[9] A layer in a Graph Neural Network (GNN) applies graph-specific operations resulting in feature transformations, such as message passing or mapping features from the latitude-longitude grid to the multi-mesh representation and back

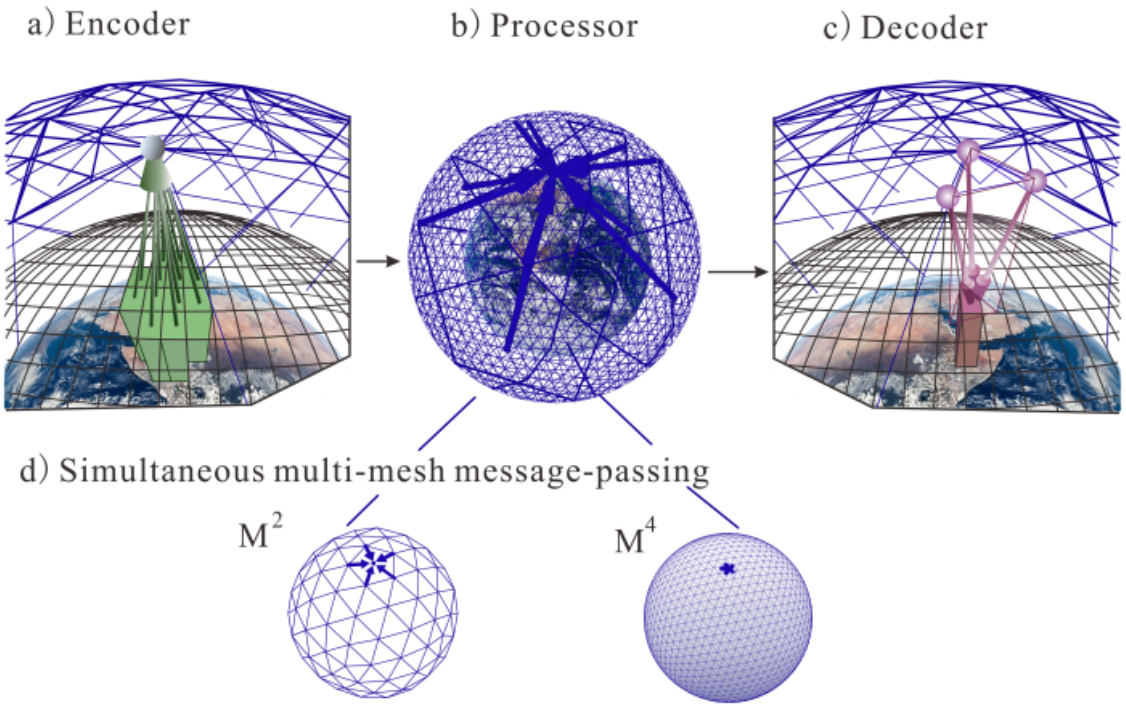


Figure 2.6: Close-up on Encoder, Processor, Decoder, and simultaneous multi-mesh message-passing [15]. (a) The Encoder maps pairs of adjacent latitude-longitude regions to node representations on the multi-mesh. (b) The Processor enables simultaneous local and large-scale information propagation via message-passing. (c) The Decoder transforms learned features back to the grid, predicting the next time step as a residual update. (d) The multi-mesh, derived from an icosahedron, recursively subdivides each triangular face into four, progressing from a 12-node base mesh (M^0) to a 40,962-node final mesh (M^6). This structure enables efficient multi-resolution message-passing while maintaining consistency.

2.3.3 Training Process and Details

GraphCast was trained to minimize the Mean Squared Error (MSE) between its predicted weather state and the corresponding weather state from ECMWF's ERA5 reanalysis dataset which served as ground truth (see Equation 2.8). DeepMind researchers split ERA5 data from 1979 to 2021 into two sets: a "development set" used for training, which included 39 years of historical weather observations from 1979 to 2017, and a "test set" covering the years 2018–2021, used to evaluate the model's performance [9]. The only difference between the two datasets is that the development set comprises only dates earlier than those in the test set. To prevent bias, neither the researchers nor the training software were allowed to access data from the test set until the development phase was completed [9]. This ensured that all decisions made regarding the model's architecture and training process could not use any future information, preserving the integrity of the evaluation.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \tag{2.8}$$

Where:

- n: Total number of grid cells in the latitude-longitude grid.
- y_i: Actual observed weather value from the ERA5 reanalysis for the i-th grid cell.
- x_i: Predicted weather value from the GraphCast model for the same i-th grid cell.

To account for the varying importance of different pressure levels, the MSE, also known as the objective or loss function, was averaged and weighted by vertical level [9]. For each atmospheric variable measured at multiple pressure levels, the MSE was calculated by taking into account the weight of each pressure level, using a weighted average. Heavier weights were assigned to levels closer to the surface, while lighter weights were given to higher levels (as shown in Figure 2.7). This approach ensured that levels closer to the ground had a greater influence on the final error calculation. The lower atmosphere is denser, concentrating more mass and energy, and most weather events, such as storms and rain, occur near the surface. Therefore, variables predicted at lower (in terms of height) pressure levels (like 850 to 1000 hPa) are more relevant for day-to-day weather forecasting.

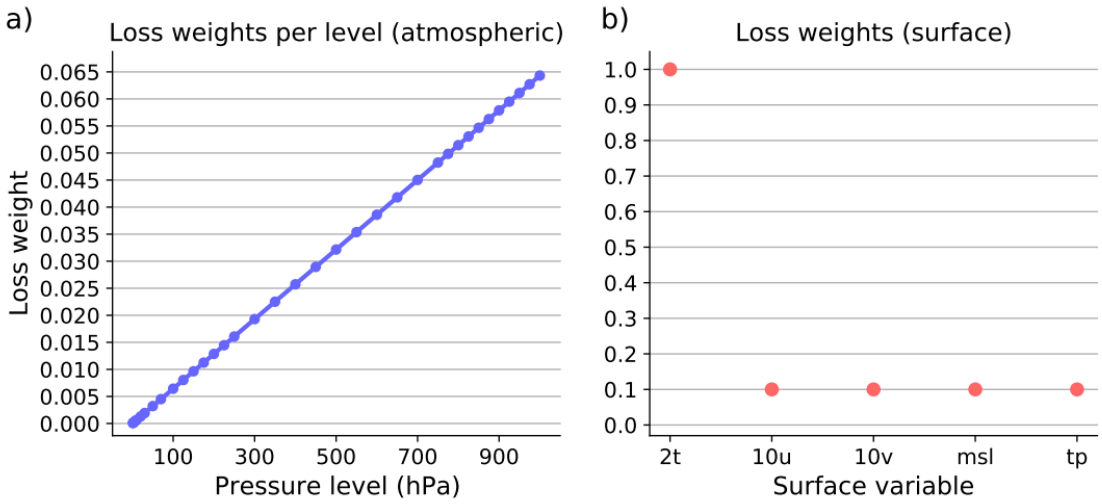


Figure 2.7: Loss weights used during model training [9]. (a) Vertical loss weights for atmospheric variables, prioritizing lower pressure levels near the surface. (b) Loss weights for surface variables, with 2-meter temperature (2t) assigned the highest weight (1.0) due to its importance, while other variables (10u, 10v, msl, tp) were weighted at 0.1 for balanced training.

Additionally, specific weights were assigned to each surface variable to reflect their relative importance in the error calculation process. During model training, the loss weights for surface variables were adjusted to ensure that no single variable disproportionately influenced the model's overall performance [9]. 2-meter temperature (2t), being directly measurable and highly influential on weather conditions that affect daily life, was assigned a higher weight (1.0). In contrast, smaller weights (0.1) were assigned to the

other atmospheric variables: 10-meter u wind component (10u), 10-meter v wind component (10v), mean sea-level pressure (msl), and total precipitation (tp) (Figure 2.7). This weight adjustment ensured a balanced and fair training process, allowing the model to give appropriate attention to all variables [9].

As GraphCast's final model was designed to predict weather variables over multiple forecasting steps, an autoregressive training regime was employed [9]. In this regime, the model's predicted state for a given time step was used as input to predict the subsequent time step, creating a feedback loop. The number of autoregressive steps was increased incrementally from 1 to 12 (i.e., six hours to three days) throughout training [9]. The error between GraphCast's predicted state and the corresponding ERA5 state was computed for each of the 12 autoregressive steps. The gradients of these errors with respect to the model parameters were then backpropagated through the entire sequence of model iterations using a technique known as Backpropagation Through Time (BPTT) [9].

Backpropagation is a fundamental technique for training artificial neural networks [16]. It involves the backward adjustment of model parameters, such as the weights assigned to edges connecting neurons, to minimize the cost function (see Figure 2.8) [16]. In the context of GraphCast's GNN architecture, the weights influencing the output of each node were iteratively updated through the network for each of the 12 steps in the forecasting sequence [9].

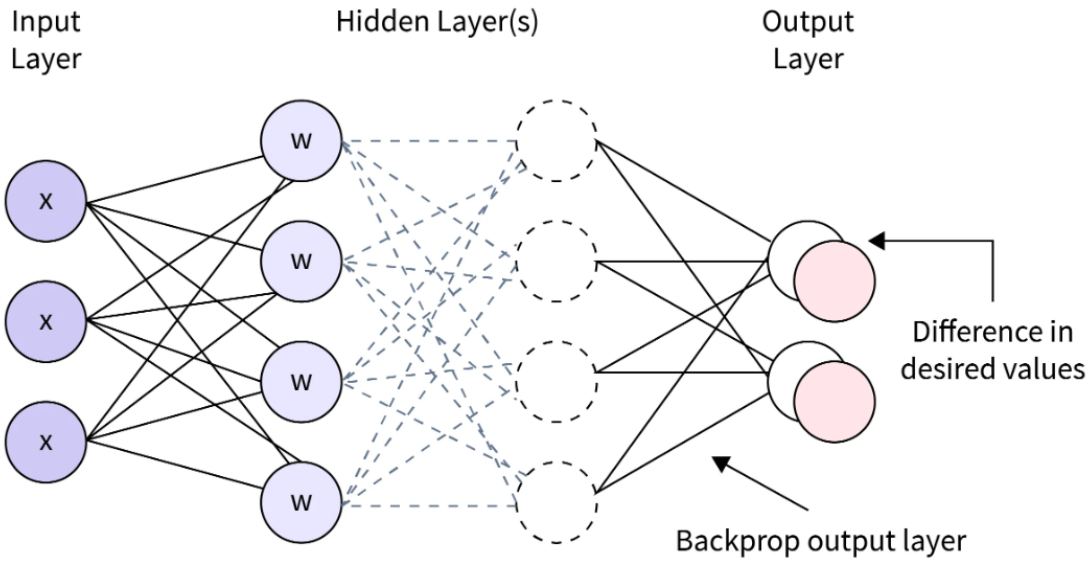


Figure 2.8: Illustration of the backpropagation process in a neural network [17]. The error, calculated as the difference between the desired and predicted values, is propagated backward to update the weights (W) in the previous layers.

Backward error propagation works hand-in-hand with gradient descent, a standard optimization algorithm in machine learning that directs the search for parameter values

to minimize the cost function. It uses the gradient, a mathematical measure of the slope or steepness of the cost function, to guide updates to the model's parameters [16]. The gradient indicates the direction of the steepest ascent of the function, while the negative gradient points in the direction of the steepest descent [16].

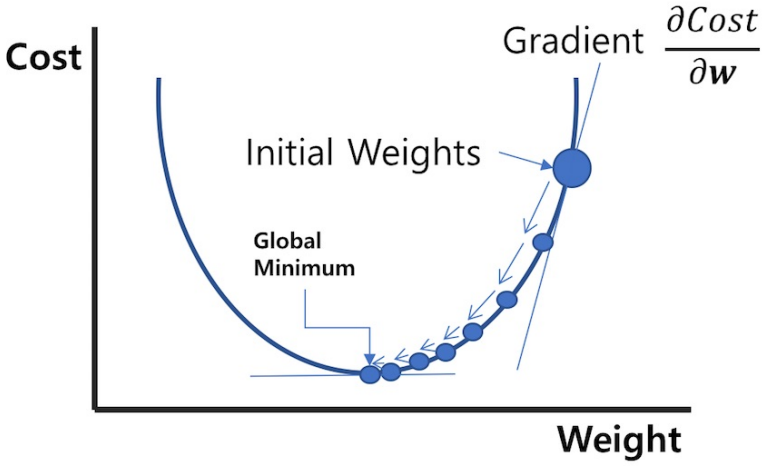


Figure 2.9: Visualization of gradient descent optimization [18]. Starting from the initial weights, the algorithm uses the gradient ($\frac{\partial \text{Cost}}{\partial w}$) to iteratively adjust the weights, moving step by step towards the global minimum of the cost function (loss function). Each step minimizes the cost, guiding the model parameters closer to the optimal solution.

During each training iteration, the partial derivative (gradient) of the loss function with respect to each weight is computed. Subsequently, the weights are adjusted by subtracting a multiple of the gradient. This multiple is determined by a learning rate, a hyperparameter that controls the magnitude of the updates. The updates ensure that each parameter is modified in proportion to its contribution to the total error, gradually moving the model towards a local or global minimum of the loss [16]. In GraphCast's GNN architecture, weight adjustments occur across all 18 layers starting at the last layer (output layer), the one employed by the decoder to map input variables to node attributes on the "multi-mesh". It then moves through all the 16 processor layers, which are responsible for message-passing within the multi-mesh representation. Then, it reaches the encoder's first layer (input layer), which is closest to the input lat-lon grid [9]. Through this iterative refinement, the model progressively enhances its predictive accuracy, minimizing the discrepancy between its outputs and the target values.

To facilitate research and practical applications, DeepMind has open-sourced three pre-trained versions of the GraphCast deterministic model, along with demonstration code for their implementation, available in their GitHub repository. These pre-trained models—GraphCast, GraphCas_small, and GraphCast_operational—vary in resolution,

pressure levels, and training data, catering to different computational and operational needs (Table 2.3). In this study, the provided code was adapted and used to implement a selected pre-trained model and generate forecasts. The predictions obtained were subsequently compared to the IFS ensemble control forecasts to assess performance. A detailed evaluation of these forecasts is presented in Chapter 4, while Chapter 3 (Methodology) outlines the modifications made to the original implementation to optimize it for operational forecasting. Below is an overview of the available pre-trained GraphCast models, highlighting their key differences and intended use cases.

Graphcast pre-trained model versions are available for loading from the Google Cloud Bucket and include GraphCast, GraphCast_small, and GraphCast_operational, which are summarized in Table 2.3. GraphCast is the full-scale version of the model, featuring a high resolution of 0.25 degrees and 37 pressure levels. It utilizes a six-times refined icosahedral mesh and was trained on ERA5 data from 1979 to 2017. This is the version introduced in the DeepMind paper on GraphCast [9], where it has been evaluated on data from 2018 onward. GraphCast_small is a more compact variant designed for environments with limited computational resources, such as free Colab notebooks. It operates at a lower resolution of 1 degree with 13 pressure levels and a five-times refined mesh. Trained on ERA5 data from 1979 to 2015, it has a smaller memory footprint but lower performance compared to the full-scale version. GraphCast_operational is tailored for real-world forecasting applications. It maintains the high resolution of 0.25 degrees and operates with 13 pressure levels, utilizing the same six-times refined icosahedral mesh as the full-scale version. The model was trained on ERA5 data from 1979 to 2017 and further fine-tuned on HRES-fc0 data from 2016 to 2021. Unlike the other versions, which require ERA5 data as input, this model must be initialized directly from HRES-fc0 data and can be evaluated on data from 2022 onward.

Model	**Resolution**	**Pressure Levels**	**Mesh Refinement Level**	**Training Data**
GraphCast	0.25°	37	6	ERA5 (1979–2017)
GraphCast_small	1°	13	5	ERA5 (1979–2015)
GraphCast_operational	0.25°	13	6	ERA5 (1979–2017) - HRES-fc0 (2016–2021)

Table 2.3: Summary of Pre-trained GraphCast Model Versions. Differences in resolution, pressure levels, mesh refinement, and training data. GraphCast is the standard high-resolution model, GraphCast_small is optimized for lower computational resources, and GraphCast_operational is fine-tuned for real-world forecasting, maintaining high resolution while incorporating additional fine-tuning on HRES-fc0 data for improved performance in operational settings.

Chapter 3

Methodology

The primary objective of this thesis was to evaluate the capabilities of GraphCast in its publicly available form for operational forecasting in the context of energy trading. In particular, forecasts of wind speed and temperature were generated up to 48 hours into the future, a critical time window for energy production estimates used in trading strategies. By relying solely on open-source data, pre-trained models, and publicly available code, this study aims to provide a cost-effective and replicable demonstration of GraphCast implementation subject to real-world operational constraints.

Among the open-sourced pre-trained versions described in Chapter 2 (Table 2.3), GraphCast_operational was selected as it is optimized for real-world applications. Unlike other GraphCast versions, which rely on historical data (ERA5) for training and initialization, GraphCast_operational initializes forecasts using HRES-fc0 (see Figure 3.1 in Chapter 2), a dataset derived from ECMWF IFS initialization data (forecast outputs at lead-time 0). This data is made available in near real-time through the ECMWF Production Data Store (ECPDS). Additionally, GraphCast_operational retains the high resolution of the full-scale version (0.25° spatial resolution, 13 pressure levels), aligning with the resolution of ECMWF's publicly available IFS operational deterministic forecasts, which enables a meaningful and direct evaluation.

This chapter provides an overview of the methodology adopted to implement GraphCast_operational and generate the forecasts analyzed in Chapter 4. The modifications to the publicly available demonstration code primarily focused on two key aspects: (1) integrating custom input datasets, and (2) producing forecasts up to 10 days ahead, starting with an input dataset containing only two time steps—a capability not supported by the demo.

3.1 Starting Point: GraphCast Repository

The starting point for this work was the GraphCast GitHub repository provided by Google DeepMind. This repository includes a demonstration codebase that can be executed in Colaboratory to run and train GraphCast.

The demo illustrates examples of:

- Loading example input datasets.
- Loading three different pre-trained model versions (Table 2.3) and associated model weights.
- Loading normalization data.
- Generating predictions based on these configurations.
- Training the model.

Several example datasets are available for loading through the demo, collected from two different sources: ERA5 and HRES[1] operational forecasts archive stored in the ECMWF Production Data Store (ECPDS). These datasets come with different resolutions (0.25° and 1°) and pressure levels (13 and 37), and cover a varying number of time steps. The resolution and number of pressure levels of the model version must match those of the input data. Therefore, not all combinations of models and input datasets are available. GraphCast and GraphCast_small, models 1 and 2 of Table 2.3, are designed to be initialized with ERA5 data, and thus require precipitation as input. All models predict precipitation, however, the ERA5 dataset includes the total_precipitation_6hr variable (cumulative precipitation over a 6-hour period), while the HRES data does not. GraphCast_operational (model 3) does not depend on precipitation, as it is specifically trained to use HRES-fc0 data as input, which does not include this variable.

In GraphCast, the input variables on the latitude-longitude grid are normalized to have zero mean and unit variance. The learned features on the multi-mesh are also normalized when mapped back to the grid as a residual update to the previous state, ensuring unit variance on the residuals. Zero mean normalization shifts the data's average to zero by subtracting the mean from each value. This prevents a skewed central tendency, which could introduce bias in the model's predictions. By centering the data, the model can focus more on the relationships between variables, rather than any specific offset or baseline. Unit variance scaling standardizes the data by dividing each value by its standard

[1]To maintain consistency with the original terminology, we will continue to refer to IFS's control member as HRES. However, as previously mentioned, starting from Cycle 49r1, IFS's control member is no longer referred to as "High-RESolution". It is now known simply as the ensemble control member of IFS, with output equivalent to that of the former HRES.

deviation, ensuring all features have the same scale. This promotes uniformity in the magnitude of features, making the optimization process more stable during training. Without consistent variance, some features could dominate the learning process, making it harder for the model to learn the relationships between all variables. By scaling the data to unit variance, the model can learn from all features equally, improving the convergence of the gradient descent algorithm.

GraphCast normalization statistics include specific datasets that can be loaded along with the pre-trained model weights and example inputs from the Google Cloud Bucket:

- diffs_stddev_by_level: standard deviations of the differences at each pressure level.
- mean_by_level: mean values at each level.
- stddev_by_level: standard deviations at each level.

3.2 Code Modifications

The key modifications made to the publicly available demonstration code for deploying the GraphCast_operational model were implemented to extend the forecast capabilities of the demo. The original example implementation is limited in its ability to generate forecasts for multiple time steps in the future, as it requires input datasets containing all the required time steps for a given forecasting horizon. To obtain a prediction for just one time step ahead (6 hours) the demo needs an input dataset containing at least three consecutive time steps (e.g., 00z, 06z, 12z). The first two time steps are used as model inputs, while the third time step is included only to maintain the correct structure to store the final forecasts. Additionally, higher-resolution example input datasets are only available for fewer time steps due to the memory requirements of loading them, making it practically impossible to obtain high-resolution forecasts for multiple time steps in the future using the Colab-based implementation. When run in Colab, the demo code allows predictions using the small GraphCast model on 1-degree resolution data for up to 4 steps ahead. Forecasts using the other higher-resolution models are available for shorter time steps, a limitation inherent to the Colab environment's constraints. In an operational setting, this approach would result in significant data redundancy and inefficiencies. For instance, to generate forecasts up to 48 hours ahead (8 time steps), the would demo require an input dataset covering at least 10 time steps, which is impractical in real-world forecasting environments.

To overcome these constraints, custom logic was developed to enable GraphCast to generate long-range forecasts (up to 10 days) while requiring only two time steps of input data. Instead of relying on pre-existing datasets with multiple time steps, the code developed in this research dynamically expands the input data, maintaining GraphCast's

expected initialization structure while reducing unnecessary storage and computational overhead. The solution consists of creating future time steps dynamically, using null placeholders. The last available time step from the input dataset is identified, and future timestamps are generated in 6-hour increments to match the forecast horizon. A new dataset is initialized with the same variable structure as the original input, but with missing values for the additional time steps. All relevant meteorological variables, such as wind speed and temperature, are replicated in this extended data structure, which is then concatenated with the original input along the time dimension, preserving the expected initialization structure while minimizing the required input data. The resulting dataset is then passed to GraphCast's autoregressive inference process, where the model iteratively replaces the null placeholders with predicted values during the forecast rollout. This implementation brings several advantages. First, it reduces input data requirements, making it possible to generate forecasts up to 10 days ahead with only a two time steps initialization dataset. Second, it minimizes storage and computational requirements, significantly improving GraphCast's usability for real-world applications.

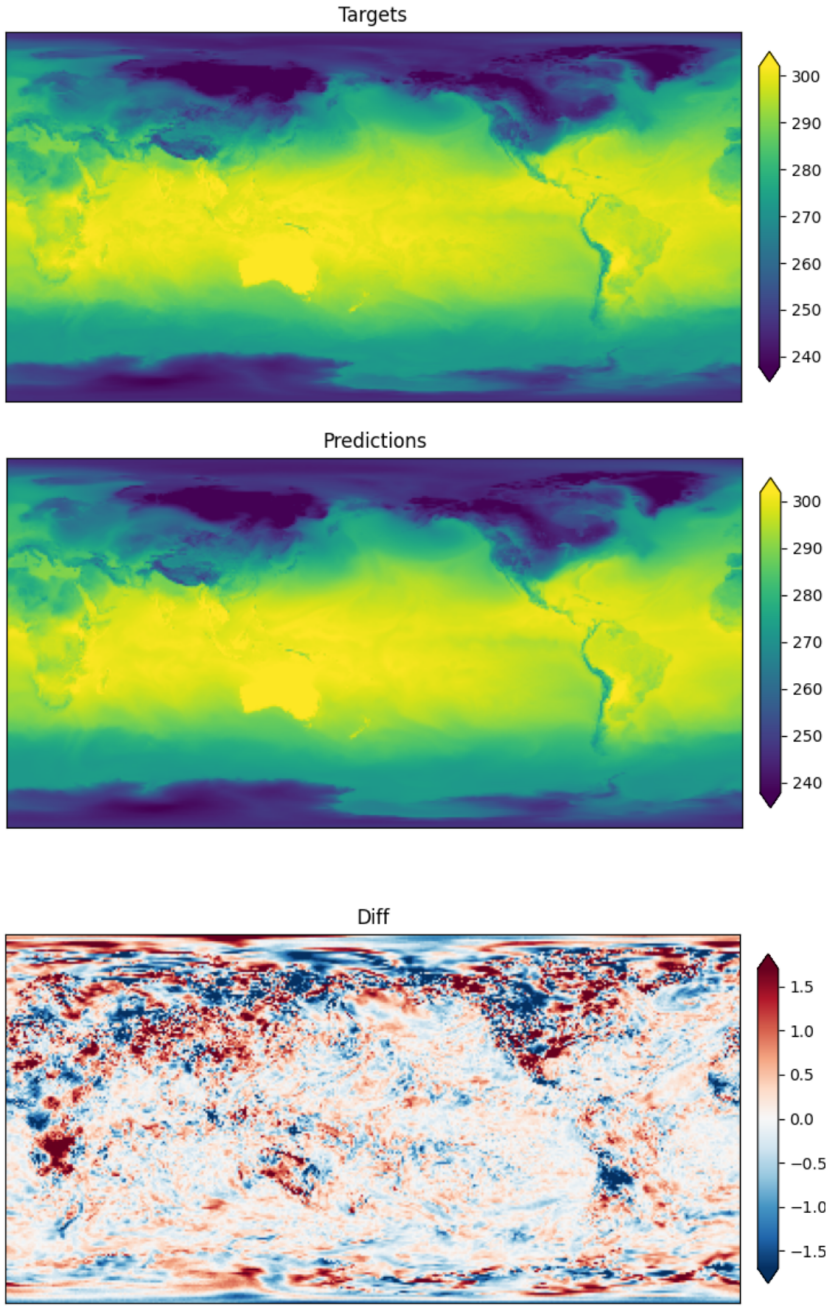


Figure 3.1: Global 2-meter temperature prediction obtained by running the GraphCast_small model on Colab. The image shows: (up) target 2-meter temperature values in Kelvins (K) from the ERA5 dataset at 1-degree resolution for 2022-01-01 at 06z, (center) GraphCast forecasts for the same time, day, and resolution, and (down) the difference between the two.

The HRES-fc0 dataset needed for initialization was replicated by retrieving HRES operational forecasts at 0-hour lead-time for each initialization time 00z, 06z, 12z, and 18z from the ECPDS for the period from the 8th of March, 2024, at 00z to the 29th of August, 2024, at 00z. The final implementation of the GraphCast_operational pre-trained model was executed using custom HRES-fc0 datasets as initialization data. Each forecast sequence extended 48 hours into the future, covering 8 time steps (6 hours per step × 8

= 48h = 2 days). For each initialization time (00z, 06z, 12z, and 18z), a corresponding HRES-fc0 dataset was used, containing initial conditions from both the current run-time and 6 hours prior. Forecasts were generated for the entire globe, covering all variables and pressure levels modeled by GraphCast. These forecasts extended 48 hours ahead, with outputs provided at 6-hour intervals (00z, 06z, 12z, and 18z). The resulting dataset spans from March 8, 2024 (06z), to August 29, 2024 (18z), covering a period of approximately six months (175 days).

Chapter 4

Comparative Evaluation of GraphCast and IFS Forecasts

4.1 Data Collection and Preprocessing

To enable a fair comparison between the outputs of GraphCast and ECMWF's IFS models against observational data, a structured approach was followed for data collection, preprocessing, and standardization. Forecasts and observations were aligned in terms of period (8 March 2024, 00z to 29 August 2024, 00z), temporal resolution (6-hour intervals at 00z, 06z, 12z, and 18z), geographical coordinates, and key evaluation variables: 2-meter temperature (2t) and wind speed expressed through 10-meter wind components (u and v).

Data Sources

Three primary datasets were used for evaluation:

- **GraphCast forecasts**: Predictions generated using the GraphCast_operational model.
- **IFS forecasts**: Operational forecasts retrieved from ECMWF's ECPDS archive.
- **Observational data**: Ground truth measurements from six SYNOP stations in Italy.

Observational Data

The observational dataset, serving as the baseline for evaluating GraphCast against IFS forecasts, consisted of daily 2-meter temperature (2t) and wind speed measurements recorded at 6-hour intervals from six SYNOP stations across Italy (summarized in Table 4.1). These observations, spanning from 8 March 2024, 00z to 29 August 2024, 00z, were obtained from Meteomanz, an online platform providing access to historical meteorological data.

Region	Station	WMO Index	Latitude	Longitude
Piemonte	Torino Caselle	16059	45.191847	7.650664
Emilia-Romagna	Cervia	16148	44.223256	12.305817
Puglia	Bari/Palese	16270	41.133014	16.750081
Sicilia	Palermo Punta Raisi	16405	38.180594	13.096353
Sardegna	Cagliari Elmas	16560	39.243372	9.060217
Calabria	Lamezia Terme	16362	38.908303	16.253581

Table 4.1: **SYNOP meteorological stations in Italy.** The table lists the SYNOP stations from which ground truth measurements of 2-meter temperature (2t) and wind speed were retrieved. It includes their geographical coordinates and corresponding WMO (World Meteorological Organization) Index, a unique identifier used for standardized global weather data reporting.

The selected SYNOP stations are geographically distributed across different regions of Italy, ensuring coverage of various climatic conditions. Torino Caselle (Piemonte) represents northern Italy, while Cervia (Emilia-Romagna) is located along the northern-central Adriatic coast, experiencing a transitional climate influenced by both continental and maritime conditions. The remaining four stations—Bari/Palese (Puglia), Palermo Punta Raisi (Sicilia), Cagliari Elmas (Sardegna), and Lamezia Terme (Calabria)—are situated in southern Italy and along the major Mediterranean islands, where most of the country's wind farms are concentrated. According to the wind energy report by GSE (Gestore dei Servizi Energetici) [19], 97% of Italy's wind capacity is located in the southern regions, with Puglia alone accounting for a quarter of the national total, followed by Sicilia (18%), Campania (14%), Basilicata (13%), Calabria and Sardegna each contributing 10%. This geographic distribution makes these areas particularly relevant for studying and optimizing meteorological forecasts for renewable energy production.

To ensure comparability with the forecast datasets, the observational data underwent preprocessing to align its structure and temporal resolution with the model outputs. First, temperature was converted from Celsius to Kelvin and wind speed from km/h to m/s, then since the original dataset contained hourly measurements, it was filtered to include only records corresponding to the forecasting time steps (00z, 06z, 12z, and 18z), ensuring direct comparability. Additionally, latitude and longitude coordinates were appended to each record based on the respective station's location, preserving the geospatial consistency required for the analysis.

IFS Forecasts

IFS deterministic forecasts were retrieved from the IFS operational forecasts archive on the ECPDS. To ensure consistency with GraphCast predictions, only the first 8 forecast time steps were selected for each initialization time (00z, 06z, 12z, and 18z), covering the

period from 8 March 2024, 00z to 29 August 2024, 00z. The variables of interest 2t (K), 10u (m/s), and 10v (m/s) were extracted and the u and v components used to compute wind speed (m/s) following equation 4.1. The dataset was then filtered to match the locations of selected SYNOP stations, ensuring alignment with ground truth measurements for evaluation.

GraphCast Forecasts

A similar procedure was applied to GraphCast forecasts, initially obtained at a global scale for all modeled variables and pressure levels. The required fields—2t (K), 10u (m/s), and 10v (m/s)—were extracted, and wind speed (m/s) was computed from the u and v components used using Equation 4.1. The data was then refined to align with the locations of the SYNOP stations for consistency in evaluation.

$$\text{Wind Speed}_{model} = \sqrt{u_{i,model}^2 + v_{i,model}^2} \quad \text{[m/s]} \tag{4.1}$$

Where:

- $u_{i,model}$ [m/s]: 10-meter u wind component (10u) value forecasted by the forecasting model (GC or IFS).
- $v_{i,model}$ [m/s]: 10-meter v wind component (10v) value forecasted by the forecasting model (GC or IFS).

Forecast Filtering to Avoid Overlap

To ensure a non-overlapping and structured evaluation of forecast performance, only forecasts initialized at midnight (00z) were selected for both GraphCast and IFS. The forecasts were then filtered into two distinct 24-hour periods:

- **Day 1 Forecasts (0–24h):** The first four prediction time steps were retained: 06z, 12z, 18z, and 1 day 00z.
- **Day 2 Forecasts (24–48h):** The following four time steps were selected: 1 day 06z, 1 day 12z, 1 day 18z, and 2 days 00z.

This filtering approach ensures that forecasts do not overlap while allowing for a structured assessment of model performance at two key forecasting horizons: within the first 24 hours and from 24 to 48 hours.

Calculation of Metrics

Various statistical indicators were computed to assess the accuracy of the predictions provided by GraphCast and IFS. The Pearson correlation coefficient (r) was used to measure

the linear relationship between predicted and observed values. This coefficient ranges from -1 to 1, where values close to 1 indicate a strong positive correlation, while values near 0 suggest little to no linear relationship. The correlation was computed using the formula:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2}\sqrt{\sum(y_i - \bar{y})^2}} \tag{4.2}$$

where x_i and y_i represent the forecasted and observed values, respectively, and $\bar{x}$, $\bar{y}$ denote their respective means. To directly compare GraphCast and IFS, the correlation between their predictions was also computed for both wind speed and temperature. This analysis provided insights into the level of agreement between the two forecasting methodologies, highlighting any systematic differences in their predictive patterns.

In addition to correlation, two key error metrics were calculated to quantify discrepancies between predictions and actual observations: the Root Mean Square Error (RMSE) and the Mean Absolute Percentage Error (MAPE). The RMSE provides a measure of the average deviation between forecasted and observed values, expressed in the same units as the analyzed variable:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum(x_i - y_i)^2} \tag{4.3}$$

Where n is the total number of observations.

On the other hand, MAPE expresses the forecast error as a percentage relative to the actual values, enabling a normalized evaluation of prediction accuracy:

$$\text{MAPE} = \frac{100}{n}\sum\left|\frac{x_i - y_i}{y_i}\right| \tag{4.4}$$

These metrics were computed separately for wind speed and 2-meter temperature, allowing for an assessment of each model's performance across both meteorological variables.

4.2 Results of the Comparative Evaluation

This section presents a comparative analysis of the performance of GraphCast and IFS forecasts, leveraging statistical correlation measures and error metrics to assess predictive accuracy. The evaluation covers a six-month period (March 8, 2024 – August 29, 2024), encompassing a seasonally diverse timeframe that includes both spring and summer. This period is particularly relevant for assessing wind speed forecasts, as it spans some of the windiest months of the year (March to May) in several regions of Italy.

The following subsections detail the correlation and error analysis for 2-meter temperature and wind speed forecasts, highlighting differences in model performance, spatial

variations and the relative strengths of each forecasting system over short-term (0-24h) and extended (24-48h) horizons.

4.2.1 Correlation Analysis

The accuracy of GraphCast and IFS forecasts was evaluated by computing the Pearson correlation coefficient (r), which quantifies the linear relationship between predicted values and ground truth observations. Correlation values close to 1 indicate a strong agreement between the forecast and the actual measurements, while lower values suggest weaker predictive performance.

Two key aspects were analyzed:

- **Model-to-observations correlation** – Evaluating how closely GraphCast and IFS forecasts align with ground truth data.
- **Model-to-model correlation** – Direct comparison between GraphCast and IFS, assessing how similar their predictions are to each other.

The following tables present the Pearson correlation coefficients for 2-meter temperature and wind speed forecasts, comparing GraphCast and IFS against ground truth and highlighting their differences. They also show how the relative performance of the two models evolves over time, with positive values indicating GraphCast's increasing advantage at longer lead times.

2-Meter Temperature Forecasts

Station	Forecasting Horizon	Ground Truth & GraphCast	Ground Truth & IFS	GraphCast & IFS
Torino	0-24h	0.983	0.973	0.988
	24h-48h	0.982	0.971	0.989
Lamezia	0-24h	0.959	0.961	0.981
	24h-48h	0.956	0.960	0.978
Cervia	0-24h	0.974	0.980	0.979
	24h-48h	0.974	0.978	0.978
Bari	0-24h	0.942	0.917	0.990
	24h-48h	0.946	0.914	0.986
Cagliari	0-24h	0.970	0.981	0.989
	24h-48h	0.969	0.979	0.987
Palermo	0-24h	0.955	0.945	0.994
	24h-48h	0.958	0.945	0.991

Table 4.2: **Pearson correlation coefficients for 2-meter temperature forecasts.** Comparing GraphCast and IFS against ground truth observations, and against each other, across different forecasting horizons (0–24h and 24–48h). Higher values indicate a stronger relationship between the predictions and actual measurements, reflecting better forecast accuracy.

Station	Difference (GC - IFS, 24h)	Difference (GC - IFS, 48h)	Change in Difference (48h - 24h)
Torino	+0.010	+0.011	+0.001
Lamezia	-0.002	-0.004	-0.002
Cervia	-0.006	-0.004	+0.002
Bari	+0.025	+0.032	+0.007
Cagliari	-0.011	-0.010	+0.001
Palermo	+0.010	+0.013	+0.003

Table 4.3: **Differences in Pearson correlation coefficients for 2-meter temperature forecasts.**. The first column shows the difference between GraphCast and IFS correlation values with ground truth at the 24-hour forecast horizon, while the second shows the same difference at the 48-hour. The third column indicates how this difference evolves over time (48h - 24h). Positive values suggest a better performance by GraphCast, whereas negative values favor IFS.

At first glance, the differences in correlation values between GraphCast (GC) and IFS at the 24-hour and 48-hour forecast horizons suggest a balanced performance between the two models. GraphCast shows higher correlation than IFS in Torino, Bari, and Palermo, while IFS outperforms GC in Lamezia, Cervia, and Cagliari. This might indicate that

both models perform similarly overall, each excelling in certain regions. However, the true advantage of GraphCast becomes evident when examining how the relative difference between GC and IFS evolves over time. The "Change in Difference (48h - 24h)" column provides valuable insight into how the predictive accuracy of each model deteriorates as the forecasting horizon increases. GraphCast demonstrates a more stable predictive accuracy across all stations except Lamezia, as evidenced by the trends in correlation differences. When GraphCast initially outperforms IFS (i.e., positive values in the 24h column), its advantage grows further at 48h. This pattern is seen in Torino (+0.010 $\rightarrow$ +0.011), Bari (+0.025 $\rightarrow$ +0.032), and Palermo (+0.010 $\rightarrow$ +0.013), indicating that GraphCast either loses less correlation over time (Torino) or even improves while IFS declines or stays the same (Bari and Palermo). Conversely, when IFS initially has a higher correlation (i.e., negative values in the 24h column), the gap between the two models narrows at 48h. In Cervia (-0.006 $\rightarrow$ -0.004) and Cagliari (-0.011 $\rightarrow$ -0.010), GraphCast's correlation remains stable (Cervia) or declines at a slower rate than IFS (Cagliari). This means that even in cases where IFS starts with a higher correlation, its forecasts deteriorate more over time relative to GraphCast, favoring GraphCast over longer periods. Lamezia is the only exception, where GraphCast's correlation declines slightly more than IFS's over time (-0.002 $\rightarrow$ -0.004). Despite this, the broader trend remains consistent: GraphCast demonstrates greater resilience in preserving predictive accuracy.

However, the extremely high correlation between GraphCast and IFS predictions indicates that the two models follow similar predictive patterns, capturing comparable meteorological trends.

Wind Speed Forecasts

Station	Forecasting Horizon	Ground Truth & GraphCast	Ground Truth & IFS	GraphCast & IFS
Torino	0-24h	0.437	0.360	0.276
	24h-48h	0.419	0.313	0.269
Lamezia	0-24h	0.813	0.821	0.894
	24h-48h	0.818	0.800	0.884
Cervia	0-24h	0.585	0.609	0.720
	24h-48h	0.563	0.613	0.668
Bari	0-24h	0.550	0.624	0.801
	24h-48h	0.538	0.600	0.749
Cagliari	0-24h	0.696	0.773	0.797
	24h-48h	0.679	0.751	0.754
Palermo	0-24h	0.657	0.673	0.875
	24h-48h	0.658	0.638	0.817

Table 4.4: **Pearson correlation coefficients for wind speed forecasts.** Correlation coefficients for wind speed forecasts between ground truth, GraphCast, and IFS across different stations and forecasting horizons.

Station	Difference (GC - IFS, 24h)	Difference (GC - IFS, 48h)	Change in Difference (48h - 24h)
Torino	+0.077	+0.106	+0.029
Lamezia	-0.008	+0.018	+0.026
Cervia	-0.024	-0.050	-0.026
Bari	-0.074	-0.062	+0.012
Cagliari	-0.077	-0.072	+0.005
Palermo	-0.016	+0.020	+0.036

Table 4.5: **Differences in Pearson correlation coefficients for wind speed forecasts.** Variation in correlation between GraphCast and IFS relative to observed values across different forecasting horizons.

As observed in temperature forecasts, the relative difference between GraphCast and IFS wind speed correlation values shifts in favor of GC at longer forecasting horizons. While IFS generally provides better predictions up to 24 hours ahead, for forecasts extending from 24 to 48 hours, the performance becomes more balanced: GraphCast surpasses IFS in Torino, Lamezia, and Palermo, while IFS maintains an advantage in Cervia, Bari, and Cagliari. The Change in Difference (48h - 24h) column once again highlights a key insight: while IFS may be more accurate in short-term wind speed predictions, GraphCast is better at preserving its accuracy over longer timeframes. In Lamezia, for example,

GraphCast starts slightly behind IFS at 24h (-0.008), however, over the next 24 hours, GraphCast's correlation increases by 0.005, while IFS declines by 0.021. As a result, by 48h, the difference reverses to +0.018 in favor of GraphCast. In Cervia, Bari, and Cagliari, IFS maintains a stronger correlation at both 24h and 48h, however, even in these locations (except for Cervia), IFS loses slightly more accuracy over time compared to GraphCast, reducing its initial advantage.

The correlation values between GraphCast (GC) and IFS forecasts remain consistently high but are notably lower for wind speed compared to temperature, indicating that while both models generally capture similar predictive patterns for wind, greater discrepancies emerge relative to temperature forecasts. This distinction is evident in Figure 4.1, where temperature correlations (red lines) remain stable and high across all stations, reflecting the smoother spatial and temporal variability of temperature. In contrast, wind speed correlations (blue lines) exhibit pronounced spatial variability, with overall lower values. Among the analyzed stations, Torino stands out with significantly lower wind speed correlations. This can be attributed to topographical influences, local meteorological variability, and the challenges models face in accurately resolving wind dynamics in mountainous terrain.

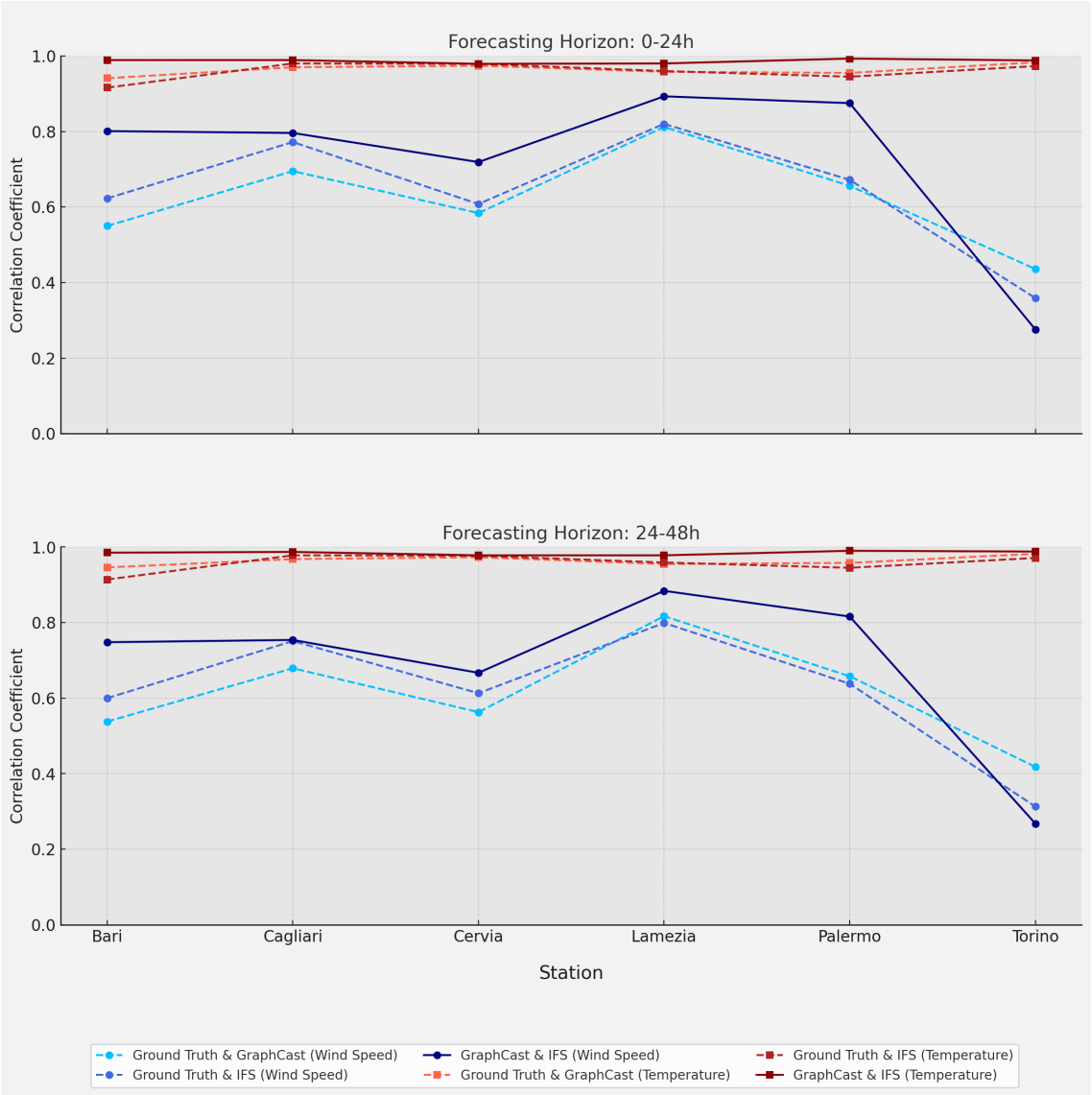


Figure 4.1: **Comparison of correlation coefficients for wind speed and 2m temperature forecasts across different stations and forecasting horizons.** The figure shows Pearson correlation values for GraphCast and IFS forecasts over 0–24h (top) and 24–48h (bottom). Temperature correlations (red) remain high and stable, while wind speed correlations (blue) vary more due to local turbulence and topographical effects. Solid lines represent model-to-model correlations, while dashed lines indicate model-to-observation correlations.

This difference likely arises from the greater complexity involved in predicting wind speed, which is inherently more variable due to local turbulence, rapid fluctuations, and finer-scale meteorological processes that are not fully resolved at the model's spatial resolution. One key factor contributing to this discrepancy is the difference in resolution between ground truth observations and model forecasts. SYNOP stations provide high-resolution point measurements at specific locations, while GraphCast and IFS predictions analyzed in this study are generated at a spatial resolution of 0.25° (around 28 km grid cells). This means that when retrieving a forecast for a specific station, such as Torino

Caselle (WMO 16059, 45.191847, 7.650664), the 10-meter u and v wind components, from which wind speed is derived, represent an approximation over a 28 km area, rather than an exact match to the point where the SYNOP station is located. As wind speed is highly dependent on local topography and small-scale atmospheric disturbances, the discrepancy between point-based ground truth measurements and grid-based forecasts is likely larger than for temperature, which tends to vary more smoothly over larger areas. Including temperature in the analysis serves as a valuable control variable, providing a clearer understanding of the forecasting models' performance and helping to contextualize the lower correlation values observed for wind speed.

To further illustrate the correlation patterns discussed in this section, Figures 4.2 to 4.3 present a selection of scatter plots comparing forecasted and observed values between GraphCast and IFS for Cagliari and Lamezia. While Figures 4.4 to 4.5 show inter-model correlations for Cervia, Bari and Cagliari. To maintain clarity and conciseness in this section, only a subset of correlation plots has been included in the main text. A more comprehensive set of scatter plots, covering the other analyzed SYNOP stations, is provided in the appendix for reference.

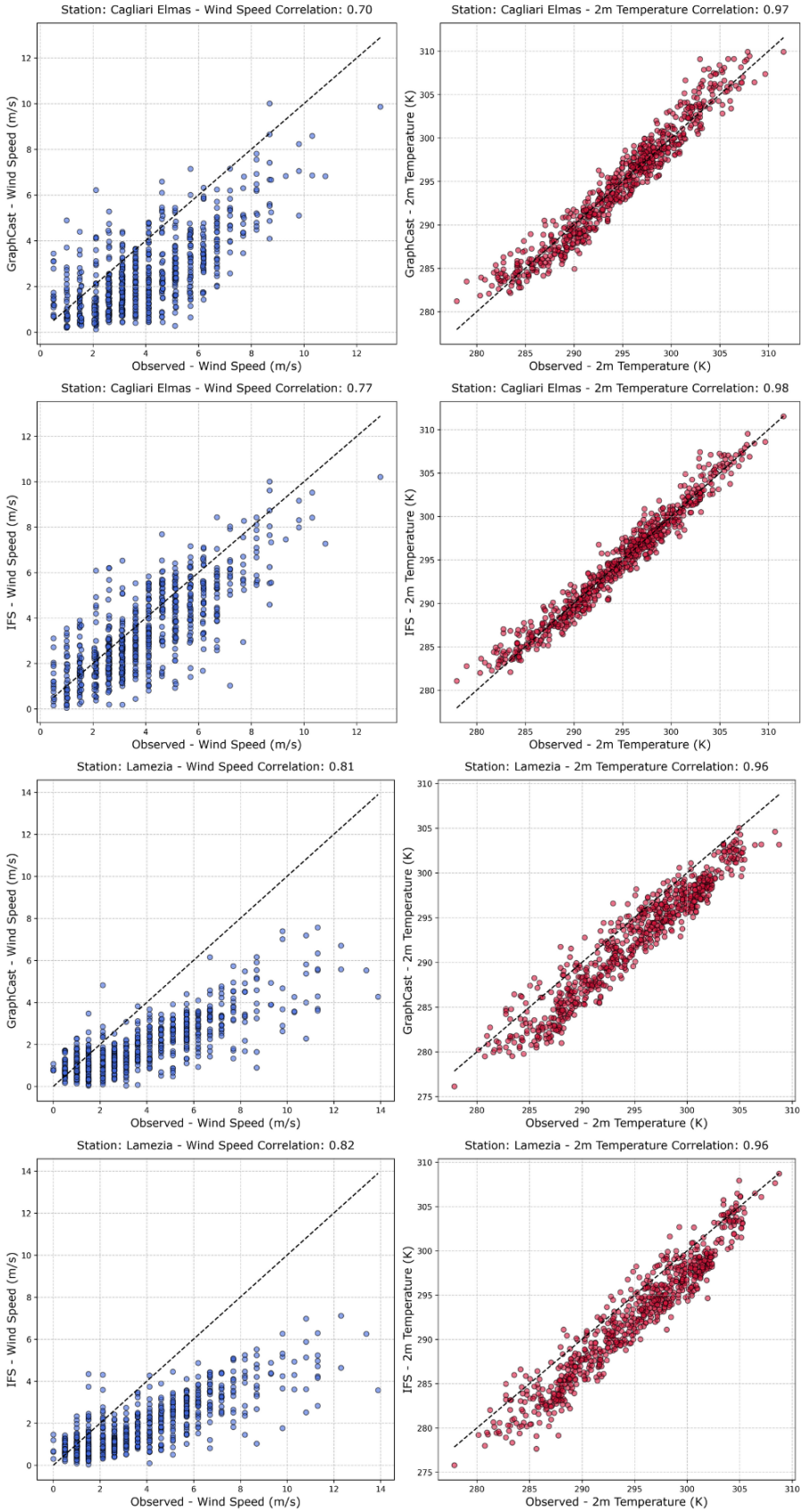


Figure 4.2: **Scatter plots of forecasted vs. observed wind speed and 2m temperature values for Cagliari and Lamezia (0-24h).** GraphCast (first and third row) and IFS (second and fourth) predictions vs. observed ground truth values for Cagliari (first two rows) and Lamezia (bottom rows). Wind speed (left, blue) and 2m temperature (right, red) are forecasted for 0-24h.

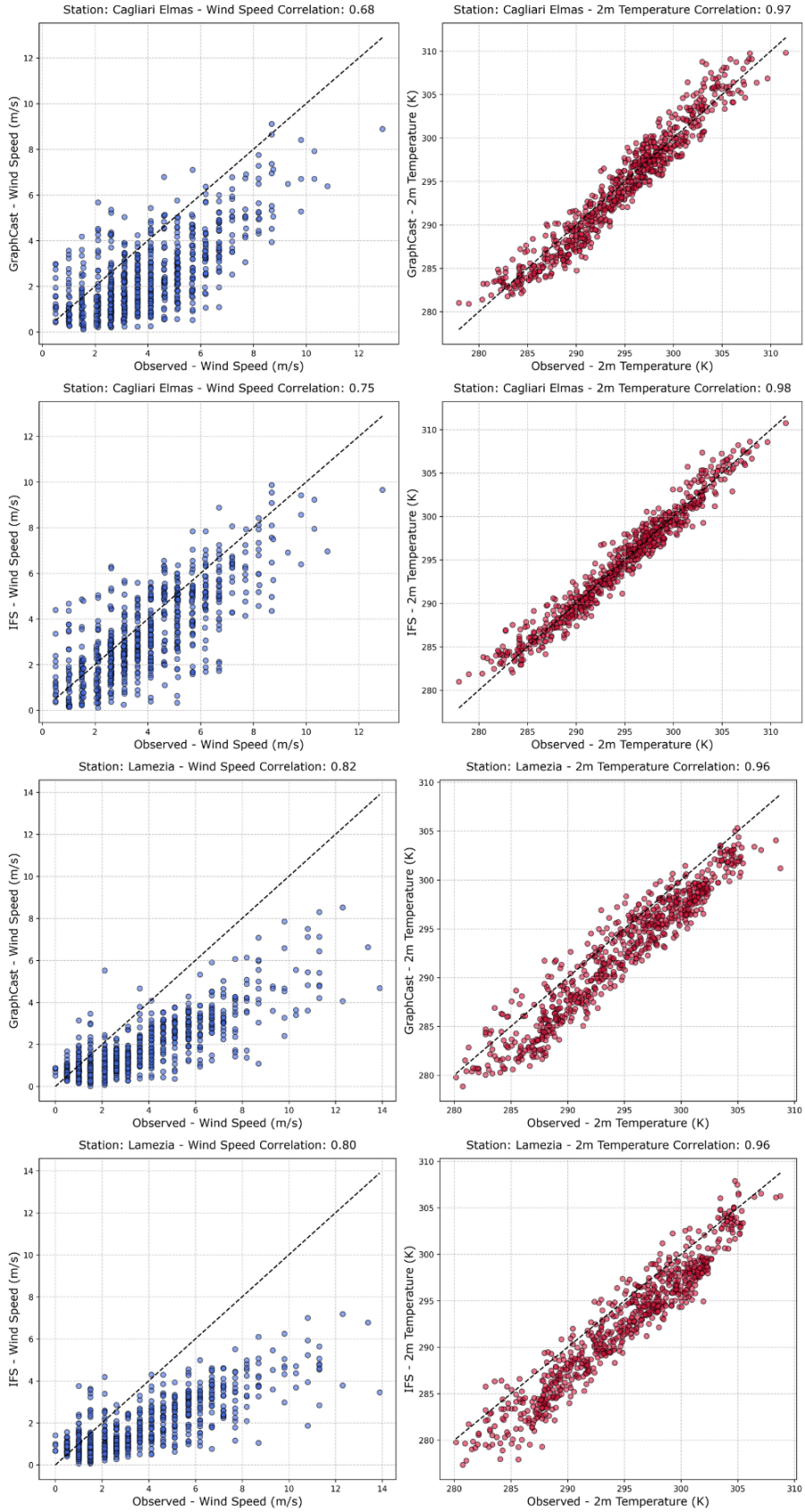


Figure 4.3: **Scatter plots of forecasted vs. observed wind speed and 2m temperature values for Cagliari and Lamezia (24-48h).** GraphCast (first and third row) and IFS (second and fourth) predictions vs. observed ground truth values for Cagliari (first two rows) and Lamezia (bottom rows). Wind speed (left, blue) and 2m temperature (right, red) are forecasted for 24-48h.

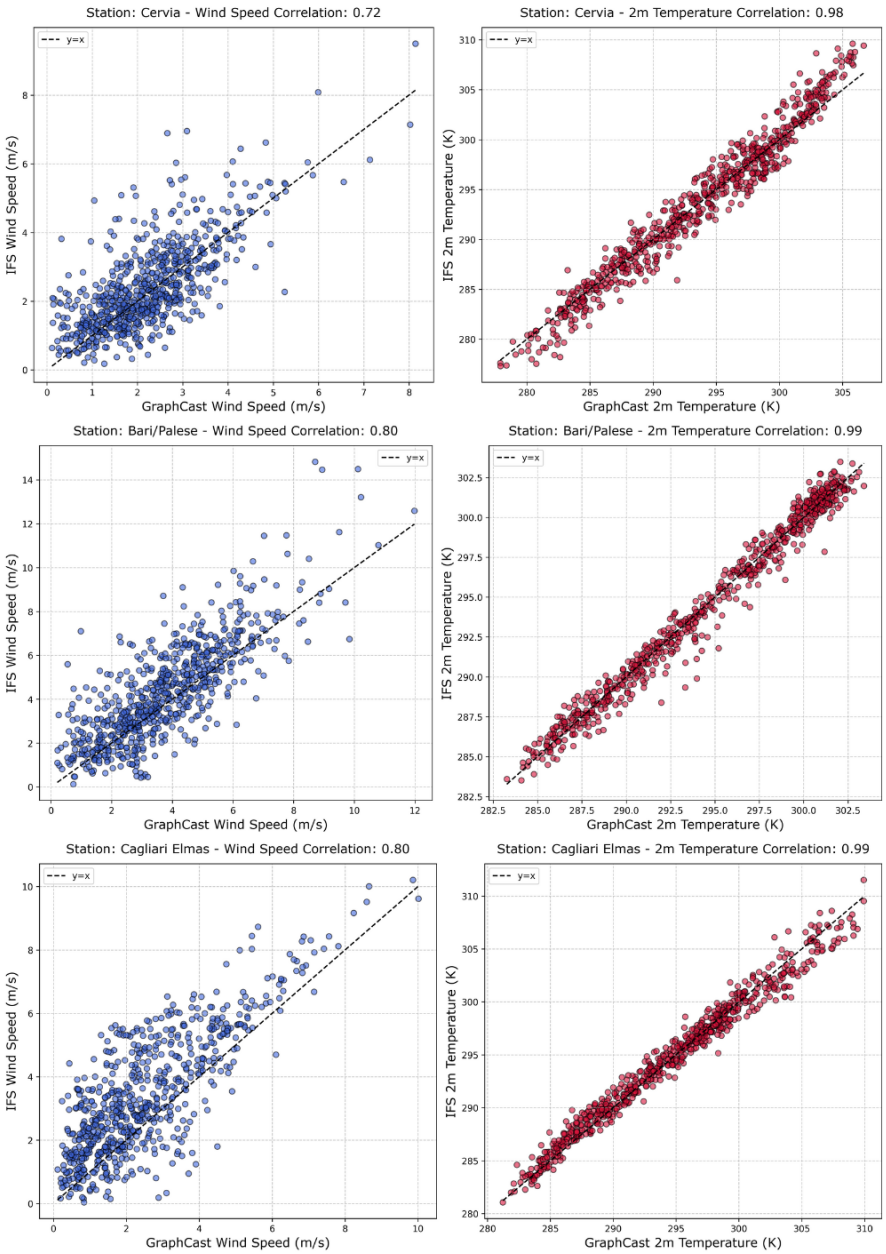


Figure 4.4: **Scatter plots of GraphCast vs. IFS forecasted wind speed and 2m temperature values for Cervia, Bari and Cagliari (0-24h).** Each subplot compares GraphCast and IFS predictions against each other for Torino, Lamezia and Palermo, evaluating the agreement between the two models. Wind speed is displayed in the left panels (blue) and 2m temperature in the right panels (red), covering a forecasting horizon of 0-24h.

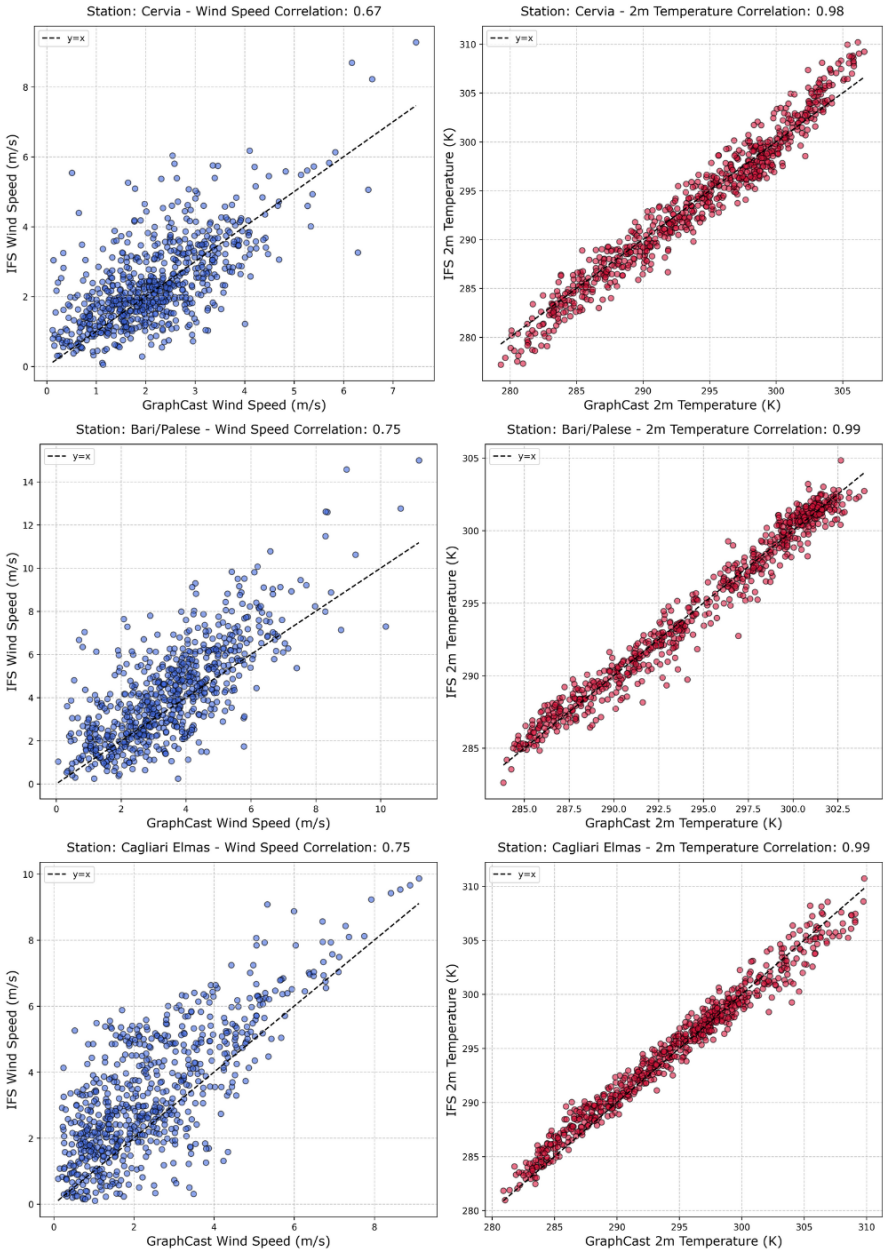


Figure 4.5: **Scatter plots of GraphCast vs. IFS forecasted wind speed and 2m temperature values for Cervia, Bari and Cagliari (24-48h).** Each subplot compares GraphCast and IFS predictions against each other for Torino, Lamezia and Palermo, evaluating the agreement between the two models. Wind speed is displayed in the left panels (blue) and 2m temperature in the right panels (red), covering a forecasting horizon of 24-48h.

4.2.2 RMSE and MAPE Analysis

The evaluation of GraphCast and IFS forecasts using Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) provides a detailed assessment of their predictive performance for 2-meter temperature and wind speed. RMSE measures absolute deviations from observed values, while MAPE represents the relative percentage error. Lower values for both metrics indicate better forecasting accuracy.

2-Meter Temperature Forecasts

Table 4.6 and Figure 4.6 summarize the RMSE and MAPE values for 2m temperature predictions across all stations and forecasting horizons.

Station	Forecasting	Model	MAPE (%)	RMSE (K)
Torino	0-24h	GraphCast	0.46	1.63
		IFS	0.52	1.97
	24-48h	GraphCast	0.46	1.63
		IFS	0.55	2.04
Lamezia	0-24h	GraphCast	0.90	3.02
		IFS	0.90	3.04
	24-48h	GraphCast	0.85	2.86
		IFS	0.91	3.07
Cervia	0-24h	GraphCast	0.44	1.60
		IFS	0.38	1.49
	24-48h	GraphCast	0.45	1.65
		IFS	0.41	1.57
Bari	0-24h	GraphCast	0.65	2.41
		IFS	0.75	2.76
	24-48h	GraphCast	0.63	2.31
		IFS	0.75	2.78
Cagliari	0-24h	GraphCast	0.43	1.59
		IFS	0.32	1.22
	24-48h	GraphCast	0.47	1.70
		IFS	0.33	1.28
Palermo	0-24h	GraphCast	0.41	1.69
		IFS	0.46	1.84
	24-48h	GraphCast	0.43	1.70
		IFS	0.46	1.84

Table 4.6: **RMSE and MAPE for 2m temperature forecasts.** RMSE and MAPE values for GraphCast and IFS 2t predictions across different stations and forecasting horizons (0-24h and 24-48h).

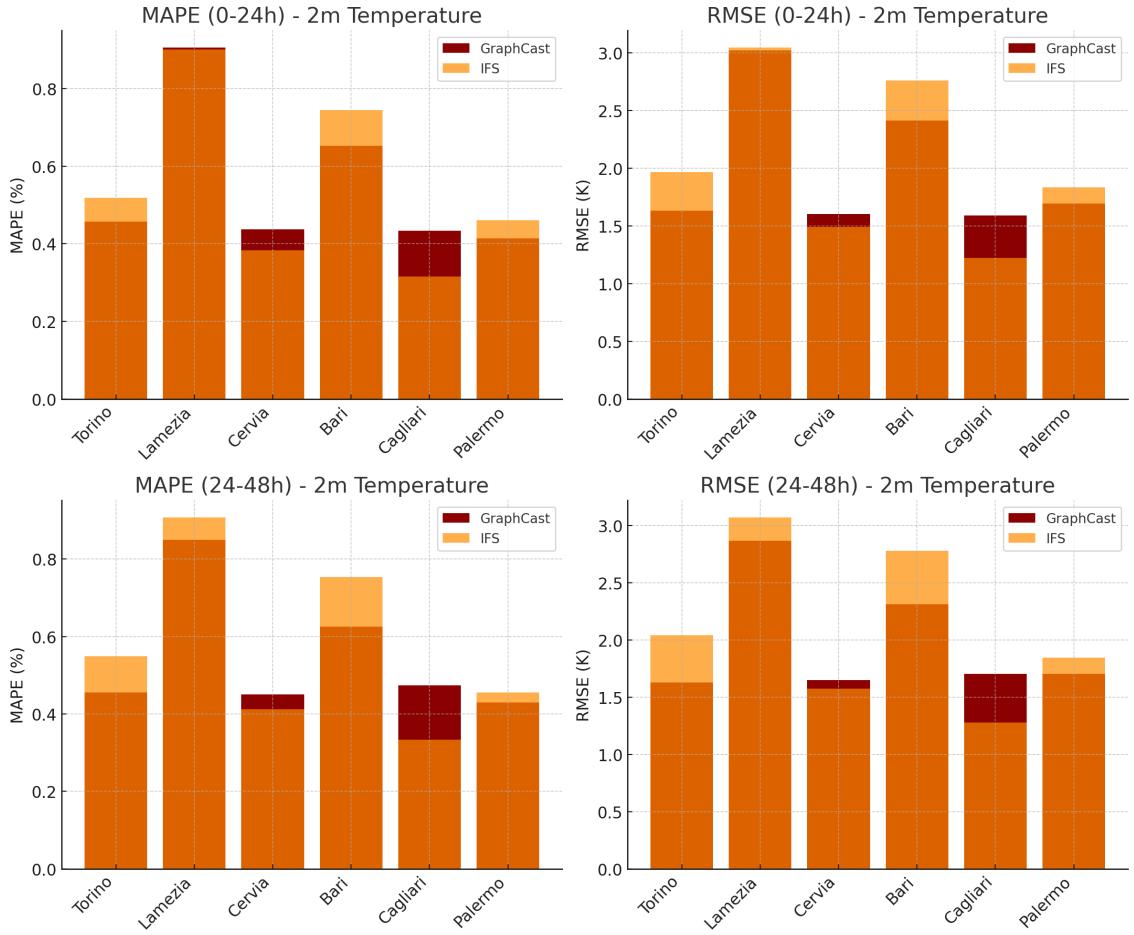


Figure 4.6: **Comparison of RMSE and MAPE for 2-meter temperature forecasts.** The figure illustrates the RMSE (left) and MAPE (right) for GraphCast (red) and IFS (dark red) across multiple stations, highlighting differences in error trends between short-term (0-24h) and extended (24-48h) forecasts.

Looking at short-term forecasts (0-24h), GraphCast generally exhibits slightly lower RMSE and MAPE values than IFS, indicating a smaller deviation from actual observations. The most significant differences appear in Bari, where GraphCast records an RMSE of 2.41 K and a MAPE of 0.65%, compared to IFS with 2.76 K and 0.75%, and Cagliari where GraphCast records an RMSE of 2.41 K and a MAPE of 0.65%, compared to IFS with 2.76 K and 0.75%. In Cagliari and Cervia, IFS outperforms GraphCast with lower error values, making it the more accurate model in these regions. In Lamezia, for 0-24 both models perform similarly with nearly identical RMSE and MAPE values; however, it is notable that this station consistently exhibits higher error values than other stations. This could be attributed to local meteorological conditions or geographical factors that increase forecast uncertainty. Over longer forecast horizons (24–48 hours), the trend shifts slightly. In several locations—Bari, Cervia, Lamezia, and Torino—the relative differences between MAPE and RMSE values between GraphCast and IFS shift in favor of GC over the longer horizon. Specifically, in Bari, Lamezia, and Torino, where GC already

outperforms IFS in the short term, the advantage increases at 24–48h. This improvement stems from either stable (Torino) or decreasing (Bari, Lamezia) GC errors, while IFS errors increase. In Cervia, although IFS maintains a lower MAPE and RMSE, the gap narrows (from +0.06 to +0.04) as GC's error grows more slowly than IFS's. Conversely, in Cagliari and Palermo, IFS's advantage widens over time due to GC's more pronounced error growth. Overall, these results confirm that while IFS may have a slight edge in certain locations, GraphCast demonstrates greater stability over time, with less degradation in predictive accuracy for extended forecasts.

Wind Speed Forecasts

Table 4.7 and Figure 4.7 display the RMSE and MAPE values for wind speed predictions across all stations and forecasting horizons. Unlike temperature forecasts, wind speed predictions exhibit higher errors overall, with notable differences between forecasted and observed values for both GraphCast and IFS.

Station	Forecasting	Model	MAPE (%)	RMSE (m/s)
Torino	0-24h	GraphCast	56.1	1.34
		IFS	47.4	1.18
	24-48h	GraphCast	55.8	1.33
		IFS	49.0	1.21
Lamezia	0-24h	GraphCast	52.7	2.51
		IFS	53.9	2.55
	24-48h	GraphCast	52.5	2.47
		IFS	54.7	2.56
Cervia	0-24h	GraphCast	40.2	1.60
		IFS	37.2	1.50
	24-48h	GraphCast	41.1	1.64
		IFS	37.4	1.50
Bari	0-24h	GraphCast	53.3	1.91
		IFS	63.9	2.34
	24-48h	GraphCast	47.5	1.73
		IFS	62.7	2.30
Cagliari	0-24h	GraphCast	44.0	2.08
		IFS	28.2	1.44
	24-48h	GraphCast	44.9	2.13
		IFS	29.6	1.52
Palermo	0-24h	GraphCast	42.8	2.13
		IFS	42.2	2.17
	24-48h	GraphCast	43.4	2.15
		IFS	43.8	2.26

Table 4.7: **RMSE and MAPE for wind speed forecasts.** Comparison of GraphCast and IFS across different stations and forecasting horizons (0-24h and 24-48h).

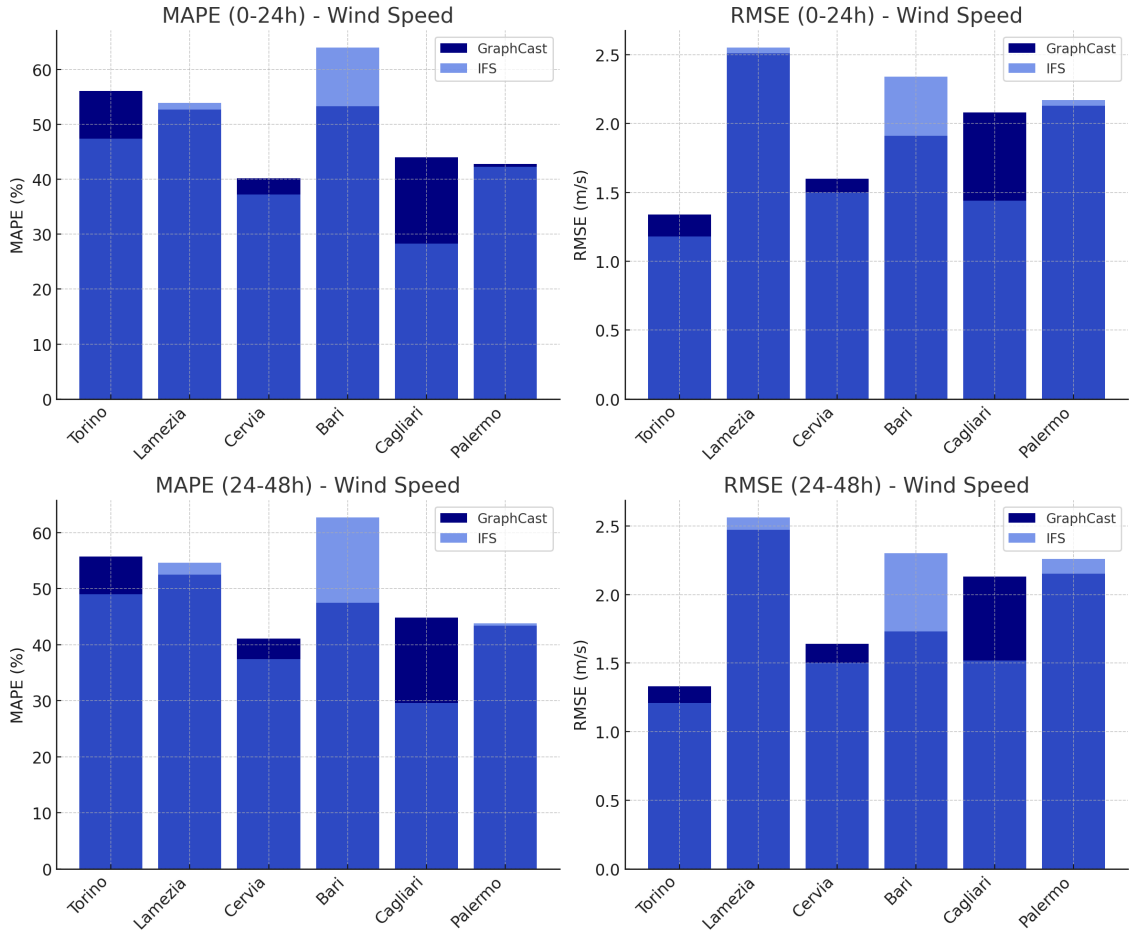


Figure 4.7: **Comparison of RMSE and MAPE for wind speed forecasts.** The figure presents the RMSE (left) and MAPE (right) for GraphCast (blue) and IFS (dark blue) across different SYNOP stations and forecasting horizons (0-24h and 24h-48h). Lower RMSE values indicate better predictive accuracy, while lower MAPE percentages reflect reduced relative error.

In the short-term forecasts (0–24h), IFS generally outperforms GraphCast in most stations with lower RMSE and MAPE values, reflecting better absolute and relative predictive accuracy. For instance, in Cagliari, IFS significantly surpasses GraphCast, with a notably lower RMSE (1.44 m/s vs. 2.08 m/s) and MAPE (28.2% vs. 44.0%). Bari is a key exception, where GraphCast performs better, recording an RMSE of 1.91 m/s and a MAPE of 53.3%, whereas IFS shows higher errors (2.34 m/s and 63.9%).

Over the extended forecast horizon (24–48h), the trends balance out, with IFS maintaining lower RMSE and MAPE values in Torino, Cervia, and Cagliari, while GraphCast performs better in Lamezia, Bari, and Palermo. GraphCast generally exhibits less degradation in predictive accuracy across all stations except for Cervia. In Torino and Lamezia, GraphCast's errors decrease while IFS's increase, allowing GC to extend its advantage over time. In Bari, both models show improved accuracy, but GC's reduction in errors is more pronounced, further widening the gap in its favor. Conversely, in Cagliari and Palermo, errors increase for both models, yet IFS's error growth is more substantial. For

instance, in Palermo, GraphCast's RMSE rises only slightly from 2.13 m/s to 2.15 m/s, while IFS's increases more noticeably from 2.17 m/s to 2.26 m/s, resulting in GC overtaking IFS over the longer horizon. These patterns highlight GraphCast's superior stability in predictive accuracy over extended forecasts, despite IFS retaining an edge in certain locations.

A key observation is the larger discrepancy between MAPE values in wind speed forecasts compared to temperature. High MAPE percentages—often exceeding 50%—highlight significant relative errors, particularly when observed wind speeds are low, making percentage-based metrics more sensitive. In contrast, RMSE offers a more stable measure of absolute error, with differences between the two models generally narrower. However, it's important to note that despite the percentage-based sensitivity, RMSE values for wind speed remain comparatively high relative to temperature. While temperature RMSE values typically range around 1–3 K—where a 2 K error represents a small fraction of the temperature magnitude—wind speed RMSE values often reach around 2.5 m/s, which constitutes a substantial portion of typical wind speeds (often ranging between 0–10 m/s). This underscores that, beyond the influence of small observed magnitudes inflating MAPE, wind speed forecasts inherently carry larger absolute errors. This difference can be partly attributed to the greater complexity and variability of wind dynamics, which are more influenced by local topography, turbulence, and small-scale atmospheric disturbances. As with the correlation analysis, the discrepancy between point-based ground truth measurements and grid-based forecasts is more pronounced for wind speed than temperature. Temperature tends to vary more smoothly over larger areas, making it easier to predict with higher consistency, whereas wind speed's local variability and rapid fluctuations challenge the models' ability to capture precise values at the station scale.

Chapter 5

Conclusions

The global energy landscape is undergoing a profound transformation, driven by the urgent need to address climate change, enhance energy security, and leverage the economic advantages of renewable energy sources. However, the large-scale integration of these intermittent power sources presents significant challenges due to their dependence on meteorological conditions. Accurate weather forecasts are crucial for reliable renewable energy production estimates, as errors can disrupt market bids, complicate grid management, and expose suppliers to financial penalties.

Weather prediction is inherently complex, requiring the modeling of chaotic atmospheric dynamics with limited data availability. Despite advancements, uncertainty persists, with perhaps its most significant impact on renewable energy forecasting. This underscores the need for continuous research to improve the accuracy of forecasting methodologies, thus supporting the integration of renewable energy into power systems.

Traditionally, weather forecasts have been generated using physics-based models, known as Numerical Weather Prediction (NWP) systems. Among them, the Integrated Forecasting System (IFS), developed by the European Centre for Medium-Range Weather Forecasts (ECMWF), represents the benchmark in operational meteorology. However, producing these forecasts comes at a high computational cost, as NWP models rely on solving complex physical equations through high-performance computing, which demands significant computational resources. Moreover, improving these models is an increasingly resource-intensive process, requiring highly trained experts with extensive knowledge of atmospheric physics, developing increasingly accurate equations and leveraging increased computing power.

In recent years, an alternative approach has emerged with the development of Machine Learning Weather Prediction (MLWP) models, employing data-driven techniques to generate forecasts without relying explicitly on physical equations. This methodology enables MLWP models to capture non-linear meteorological relationships in the data that may be difficult to represent in explicit equations, and to improve performance more efficiently by retraining on increasingly comprehensive historical records. This capability

is particularly valuable in an era of changing climate, where shifting weather patterns challenge traditional forecasting methods. Additionally, MLWP models offer significant efficiency gains, operating on modern deep learning hardware rather than traditional supercomputers. The potential of such systems led to growing interest in their accuracy evaluation against numerical-based methods.

This thesis aims to contribute to this area of research by assessing the capabilities of a freely available pre-trained version of GraphCast, a global weather forecasting model developed by Google DeepMind in 2023. GraphCast is one of the most promising MLWP systems introduced to date, demonstrating high accuracy and efficiency in medium-range forecasting. Among the available pre-trained model versions outsourced by DeepMind GraphCast_operational was implemented for operational forecasting and used to generate forecasts at 0.25° resolution for wind speed (m/s) and 2-meter temperature (K), two key variables in estimating wind and solar energy production. These predictions were then evaluated against IFS operational forecasts at the same resolution and for the same variables to assess their relative accuracy over a six-month period, using observational data from six SYNOP meteorological stations across Italy as baseline. The assessment, conducted through statistical metrics—including Pearson's correlation coefficient, Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE)—provides valuable insights into how GraphCast compares to IFS at two key forecasting horizons crucial for energy production estimates in trading strategies: 1 day forecasting (0–24 hours) and 2 day forecasting (24–48 hours).

Key Findings

The comparison between GraphCast and IFS forecasts against ground truth data for both 2-meter temperature and wind speed highlights key differences in their predictive performance, particularly in how their accuracy degrades over longer forecasting horizons.

For temperature forecasts, both models exhibit high correlation values, consistently exceeding 0.9, indicating their strong ability to capture temperature variations. However, their predictive accuracy evolves differently over time. While IFS correlation values remain stable at best, GraphCast demonstrates a notable advantage in preserving, and in some cases even improving, its correlation with observations over the extended 24-48h forecast horizon. A similar trend is observed in RMSE and MAPE: although GraphCast already shows slightly lower errors in the short term (0-24h), its advantage becomes even more pronounced over 24-48h. While IFS errors tend to increase consistently, GraphCast exhibits a slower degradation in predictive accuracy, and in some cases, RMSE and MAPE even decrease, suggesting an improvement in relative accuracy over time. Additionally, the correlation between GraphCast and IFS temperature forecasts remains exceptionally high across all stations and lead times, reinforcing the fact that both models capture similar large-scale temperature patterns with only minor deviations.

For wind speed forecasts, IFS generally holds an advantage in correlation, RMSE, and MAPE across most locations in the short-term (0-24h), indicating better initial predictive accuracy. However, as with temperature, GraphCast demonstrates superior long-term stability. Over the 24-48h period, IFS correlation values decrease more significantly, whereas GraphCast's correlation either declines less sharply or even improves relative to observations. Similarly, GraphCast's RMSE and MAPE degrade at a slower rate than IFS's, reinforcing its ability to sustain predictive accuracy over longer periods.

A key distinction between temperature and wind speed forecasts lies in the overall accuracy levels for both models. Temperature forecasts maintain high correlation values, low errors, and relatively stable performance across all stations. In contrast, wind speed predictions exhibit significantly lower correlations, higher RMSE and MAPE, and much greater spatial variability. This is due to the inherent complexity of wind dynamics—wind speed is highly variable due to local turbulence, rapid fluctuations, and finer-scale meteorological processes that models struggle to capture at their spatial resolution. As a result, when compared to point-based SYNOP observations, the coarser grid-based forecasts show larger discrepancies, leading to lower correlations and higher errors and a greater range of variations in accuracy across different geographical areas in wind speed predictions.

Overall, these findings indicate that GraphCast holds significant promise, particularly for longer-range forecasts where its predictive skill appears to degrade more slowly than IFS. While further analysis at higher spatial resolutions would be necessary to fully evaluate wind speed performance, the results for temperature—a more stable and well-represented variable—strongly suggest that GraphCast demonstrates performance that is equal to or superior to IFS in many aspects, particularly for longer forecast horizons.

Final Remarks

This research provides valuable insights into the performance of GraphCast for short-term operational weather forecasting. However, several limitations must be acknowledged. First, the study focused on a relatively short forecast horizon (0–48h), whereas the full potential of MLWP models, particularly in medium- to long-range forecasting (up to 10 days), remains an area for further investigation. Second, the evaluation was limited to a specific set of stations in Italy, meaning the results may not fully generalize to other geographical regions with different climatic conditions. Expanding the study to a broader range of locations and meteorological settings could provide a more comprehensive understanding of the model's capabilities.

By demonstrating the feasibility of implementing a freely available version of GraphCast for operational forecasting, this study highlights the potential of MLWP models in practical meteorology applied to energy markets. The results are highly promising, showing that GraphCast can perform at a level comparable to, and in some cases even superior

to, IFS, particularly in maintaining predictive accuracy over longer time horizons. If ML-based models like GraphCast can consistently match the performance of traditional NWP models—such as IFS, which is considered among the most precise forecasting systems—then their efficiency advantages make them an extremely valuable resource, especially in operational applications. However, beyond efficiency, MLWP models also introduce a key advantage: their flexibility and scalability. ML models provide the opportunity for customization by allowing companies to train them on specialized datasets that reflect their operational requirements, market conditions, or geographical areas of interest. This makes them not only highly adaptable but also easily scalable—a crucial feature for operational activities reliant on weather forecasting, such as energy trading, where precise and continuously improving forecasts enhance decision-making and overall performance.

As computational methods evolve and more high-quality observational data become available, MLWP models are likely to play an increasingly significant role in weather forecasting. Their ability to provide fast, accurate, and cost-effective predictions will have direct benefits for renewable energy trading activities, grid stability, and ultimately renewable energy integration. However, this does not imply that ML models can replace traditional NWP systems. Numerical weather prediction remains fundamental, as the historical datasets used to train ML models rely on NWP-generated reanalysis data. Instead of serving as a direct substitute, GraphCast can complement NWP models like IFS, improving forecasting efficiency and potentially enhancing accuracy in specific scenarios.

Future research should focus on optimizing the integration of GraphCast with NWP systems to leverage the strengths of both approaches, with a particular emphasis on improving spatial resolution. While GraphCast has demonstrated strong performance for 2m temperature at its current resolution (0.25°), achieving higher-resolution forecasts is crucial for its practical application in operational forecasting, especially for weather predictions related to renewable energy, such as wind speed forecasts. Moreover, IFS currently operates at a higher resolution (0.1°) than the 0.25° (approximately 28 x 28 square kilometers) used in this study, which corresponds to the publicly available operational forecasts. In reality, IFS's internal resolution is even finer (9km x 9km), providing it with a significant advantage in capturing small-scale atmospheric phenomena. To become truly competitive with IFS, GraphCast must improve its spatial resolution, which would enhance the precision of its forecasts and, consequently, the accuracy of energy production estimates.

Appendix A

Supplementary Correlation Plots

This appendix provides an extended collection of scatter plots comparing forecasted vs. observed values as well as forecasted vs. forecasted values for wind speed and 2m temperature across the SYNOP stations not analyzed in Chapter 4. These visualizations illustrate the correlation strength between:

- Model predictions and actual measurements, evaluating how well each forecasting system (GraphCast and IFS) aligns with observed ground truth data.
- GraphCast and IFS forecasts, assessing the consistency and agreement between the two models.

The analysis covers two key forecasting horizons:

- 0-24h: Forecasts initialized at 00:00 UTC with lead times covering the first 24 hours, i.e., predictions for 06:00, 12:00, 18:00, and 00:00 of the following day.
- 24-48h: Forecasts initialized at 00:00 UTC with lead times covering the subsequent 24-hour period, i.e., predictions for 06:00, 12:00, 18:00, and 00:00 of the second forecast day.

Each subplot consists of:

- Left panels (blue): Wind speed forecasts vs. ground truth.
- Right panels (red): 2m temperature forecasts vs. ground truth.
- The dashed diagonal line represents the ideal 1:1 relationship (perfect accuracy).
- Correlation coefficients in the titles quantify how closely forecasts align with observations, with values closer to 1 indicating stronger agreement. These additional

While Chapter 4 includes a select set of comparisons focusing on specific stations to highlight key trends, this appendix offers an extended view, allowing for a station-by-station evaluation of model performance. The figures below present results for both GraphCast and IFS, facilitating a detailed comparison of each model's accuracy, their respective deviations from ground truth, and the alignment between the two forecasting systems across the different meteorological conditions analyzed in this study.

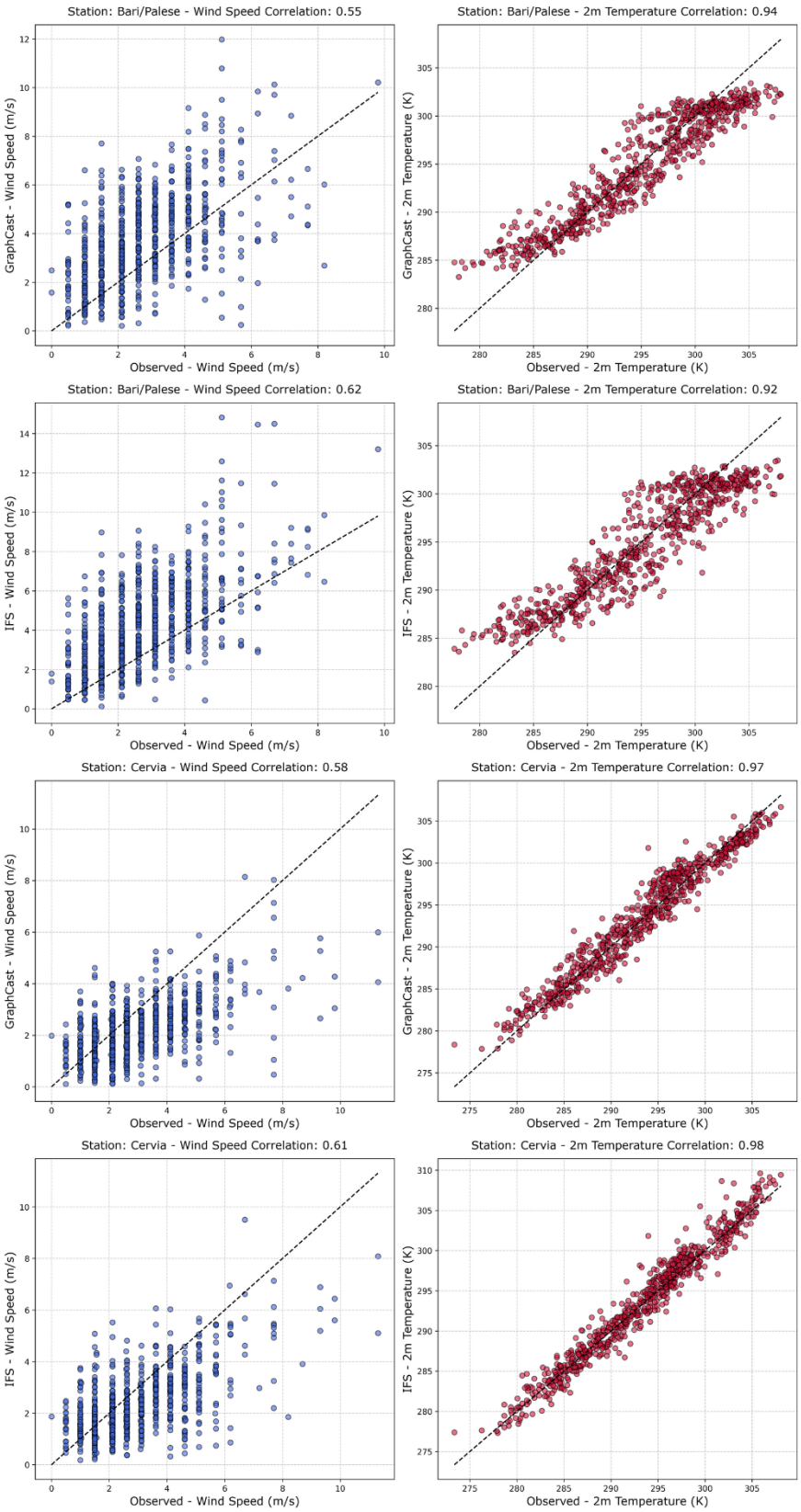


Figure A.1: **Scatter plots of forecasted vs. observed wind speed and 2m temperature values for Bari and Cervia (0-24h).** GraphCast (first and third row) and IFS (second and fourth) predictions vs. observed ground truth values for Bari (first two rows) and Cervia (bottom rows). Wind speed (left, blue) and 2m temperature (right, red) are forecasted for 0-24h.

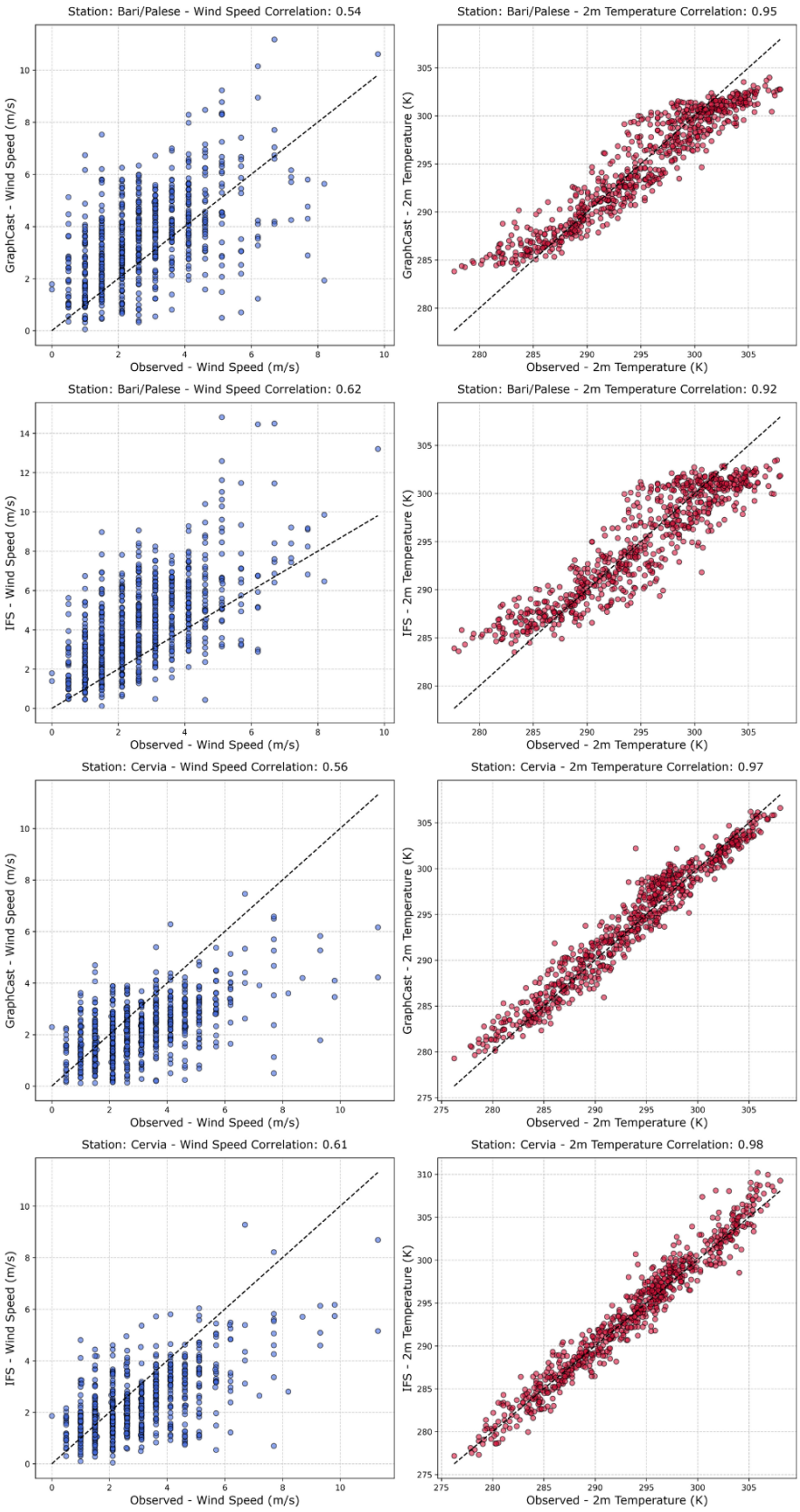


Figure A.2: **Scatter plots of forecasted vs. observed wind speed and 2m temperature values for Bari and Cervia (24-48h).** GraphCast (first and third row) and IFS (second and fourth) predictions vs. observed ground truth values for Bari (first two rows) and Cervia (bottom rows). Wind speed (left, blue) and 2m temperature (right, red) are forecasted for 24-48h.

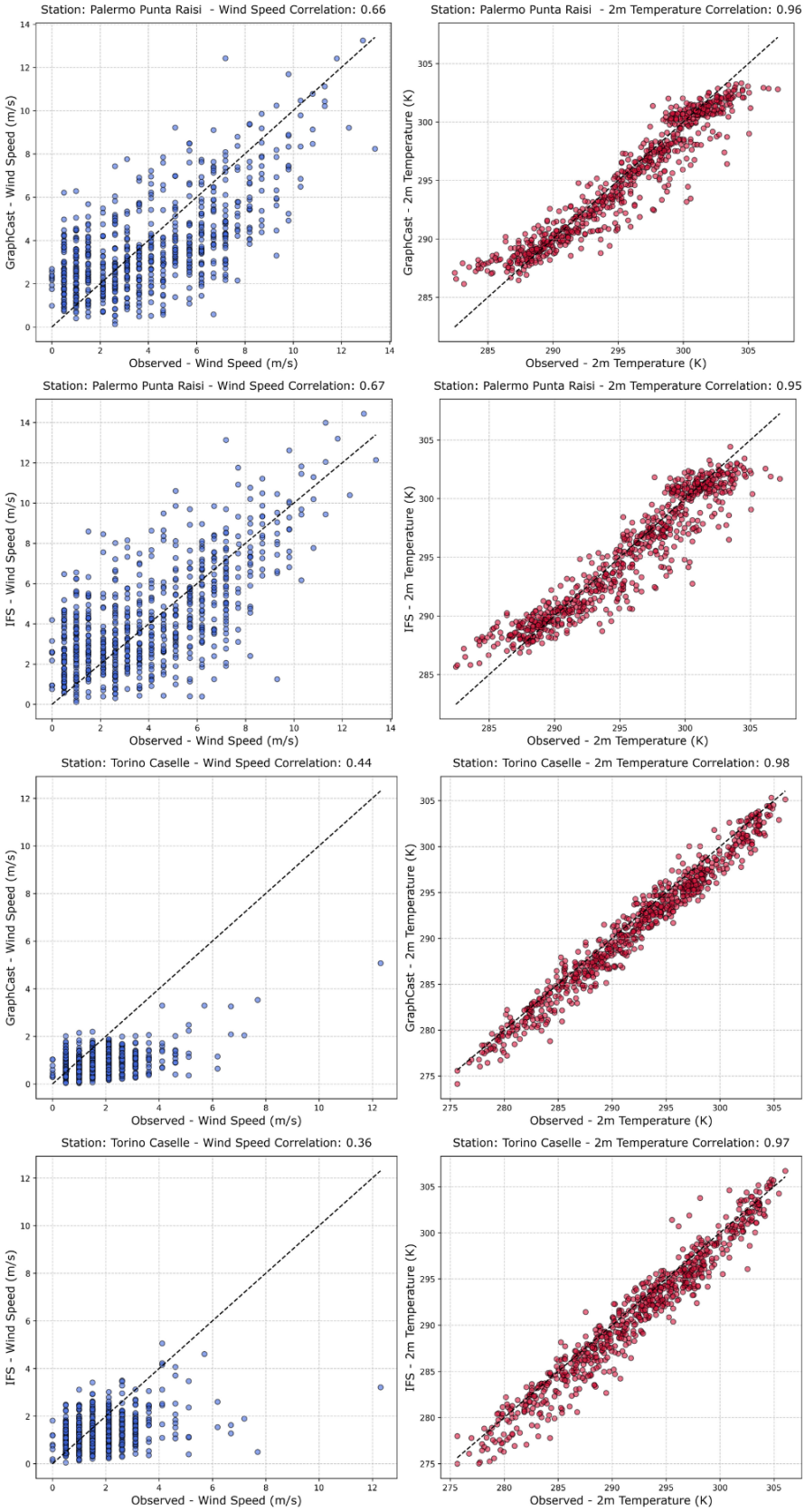


Figure A.3: **Scatter plots of forecasted vs. observed wind speed and 2m temperature values for Palermo and Torino (0-24h).** GraphCast (first and third row) and IFS (second and fourth) predictions vs. observed ground truth values for Palermo (first two rows) and Torino (bottom rows). Wind speed (left, blue) and 2m temperature (right, red) are forecasted for 0-24h.

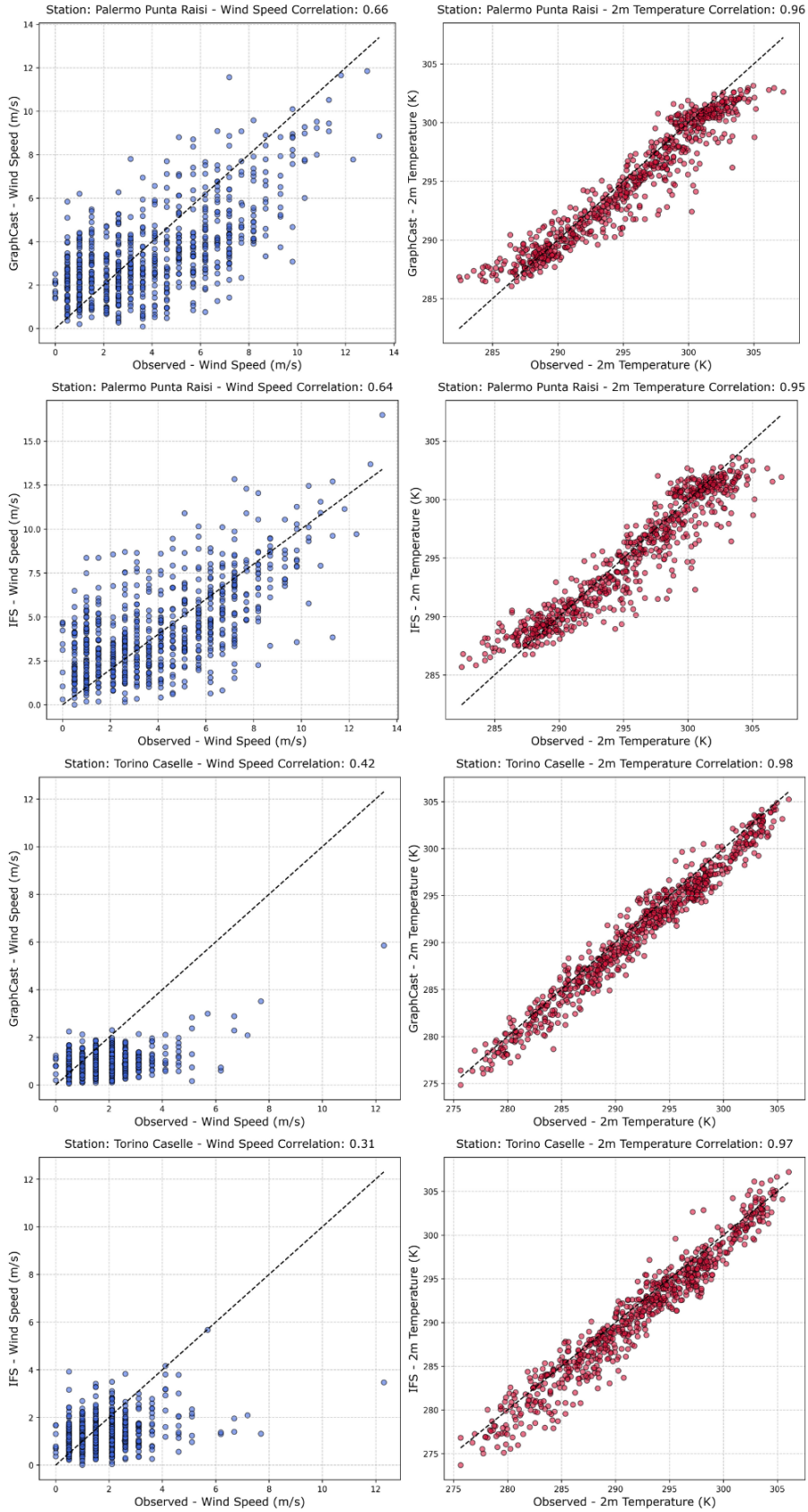


Figure A.4: **Scatter plots of forecasted vs. observed wind speed and 2m temperature values for Palermo and Torino (24-48h).** GraphCast (first and third row) and IFS (second and fourth) predictions vs. observed ground truth values for Palermo (first two rows) and Torino (bottom rows). Wind speed (left, blue) and 2m temperature (right, red) are forecasted for 24-48h.

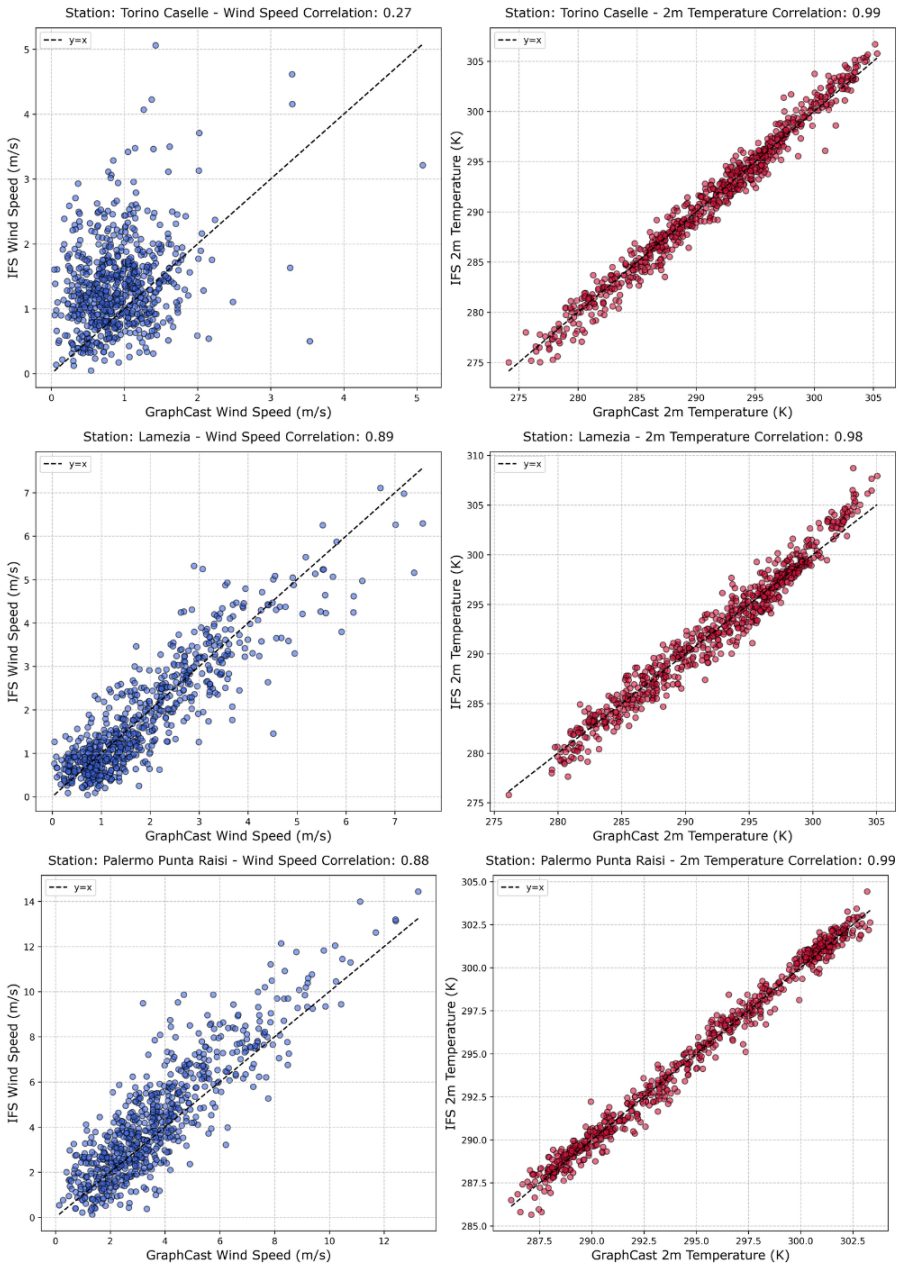


Figure A.5: **Scatter plots of GraphCast vs. IFS forecasted wind speed and 2m temperature values for Torino, Lamezia and Palermo (0-24h).** Each subplot compares GraphCast and IFS predictions against each other for Torino, Lamezia and Palermo, evaluating the agreement between the two models. Wind speed is displayed in the left panels (blue) and 2m temperature in the right panels (red), covering a forecasting horizon of 0-24h.

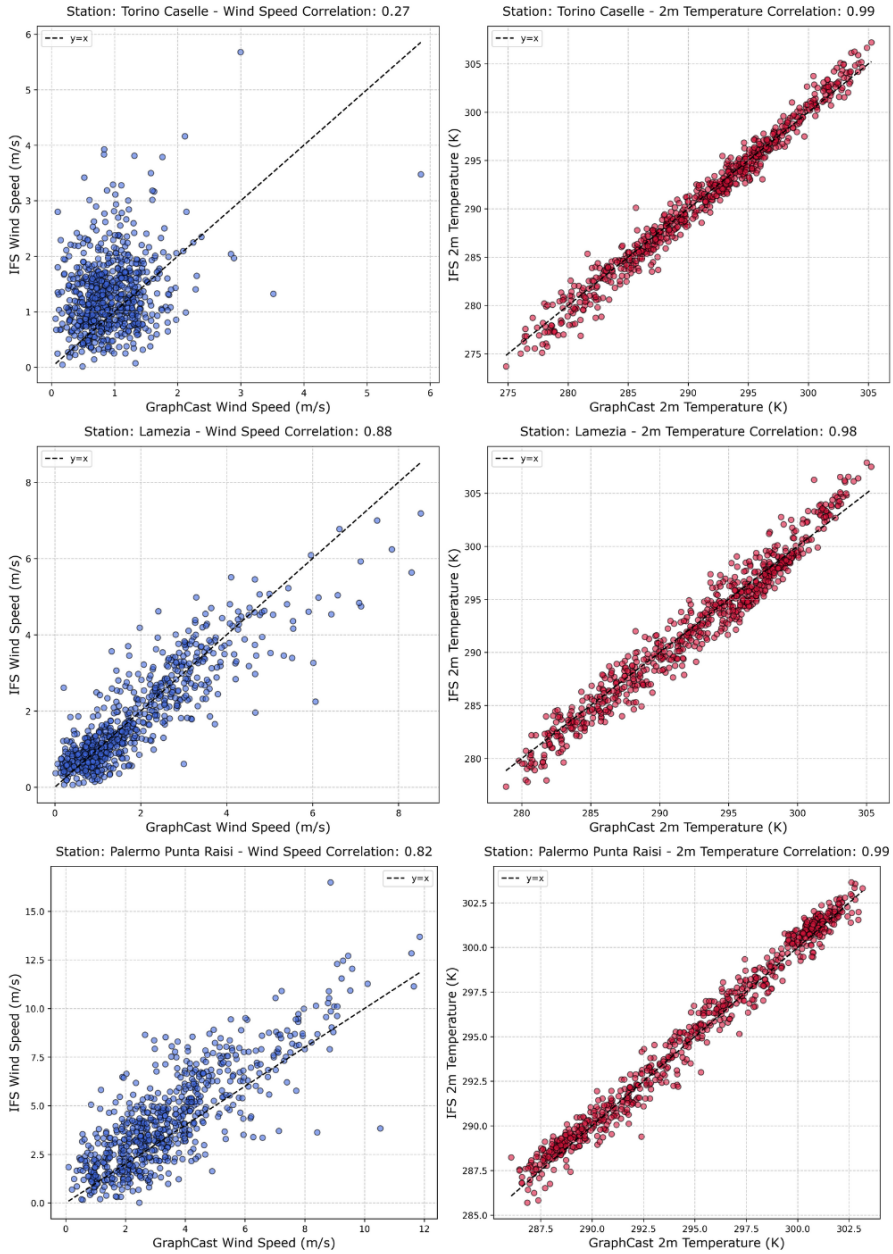


Figure A.6: **Scatter plots of GraphCast vs. IFS forecasted wind speed and 2m temperature values for Torino, Lamezia and Palermo (24-48h).** Each subplot compares GraphCast and IFS predictions against each other for Torino, Lamezia and Palermo, evaluating the agreement between the two models. Wind speed is displayed in the left panels (blue) and 2m temperature in the right panels (red), covering a forecasting horizon of 24-48h.

Bibliography

[1] Ritchie, H., Rosado, P., Roser, M. (2020). Breakdown of carbon dioxide, methane, and nitrous oxide emissions by sector. Our World in Data. Accessed 24th October 2024 at https://ourworldindata.org/emissions-by-sector

[2] International Energy Agency (IEA), Paris. (2024). CO_2 Emissions in 2023. Accessed 24th October 2024 at https://www.iea.org/reports/co2-emissions-in-2023, Licence: CC BY 4.0

[3] International Energy Agency (IEA), Paris. (2023). Net Zero Roadmap: A Global Pathway to Keep the 1.5 °C Goal in Reach. Accessed 20th October 2024 at https://www.iea.org/reports/net-zero-roadmap-a-global-pathway-to-keep-the-15-0c-goal-in-reach, Licence: CC BY 4.0

[4] International Renewable Energy Agency (IRENA), Abu Dhabi. (2024). Renewable Power Generation Costs in 2023. Accessed 15th October 2024 at https://www.irena.org/-/media/Files/IRENA/Agency/Publication/2024/Sep/IRENA_Renewable_power_generation_costs_in_2023.pdf

[5] McIntosh, B. (2024). How to address risk from the intermittency of renewable energy in power markets. Accessed 29th September 2024 at https://www.woodmac.com/news/opinion/how-to-address-risk-from-the-intermittency-of-renewable-energy-in-power-markets/

[6] Pu, Z., Kalnay, E. (2018). Numerical Weather Prediction Basics: Models, Numerical Methods, and Data Assimilation. In: Duan, Q., Pappenberger, F., Thielen, J., Wood, A., Cloke, H., Schaake, J. (eds) Handbook of Hydrometeorological Ensemble Forecasting. Springer, Berlin, Heidelberg. Accessed 29th September 2024 at https://www.inscc.utah.edu/~pu/6500_sp12/Pu-Kalnay2018_NWP_basics.pdf

[7] Bjerknes, V. (1904). Das Problem der Wettervorhersage betrachtet vomStandpunkt der Mechanik und Physik. Meteorol. Z. 21, 1–7. Accessed 10th December 2024 at https://www.semanticscholar.org/paper/Das-Problem-der-Wettervorhersage%2C-betrachtet-vom-Bjerknes/8aeb9ffd5bb9fa37c8c7baefb0a9da68d9869a81

[8] Bloomer, M. The Challenges and Complexities of Weather Forecasting. National Weather Service - National Oceanic and Atmospheric Administration (NOAA). Accessed 29th September 2024 at https://www.weather.gov/car/weatherforecasting

[9] Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., et al. (2023). GraphCast: Learning skillful medium-range global weather forecasting. Google DeepMind, Google Research. Accessed 8th September 2024 at https://www.science.org/doi/10.1126/science.adi2336

[10] Chantry, M., Bouallegue, Z. B., Magnusson, L., Maier-Gerber, M., Dramsch, J. (2023). The Rise of Machine Learning in Weather Forecasting. Accessed 10th October 2024 at https://www.ecmwf.int/en/about/media-centre/science-blog/2023/rise-machine-learning-weather-forecasting

[11] Numerical Weather Prediction (Weather Models). National Weather Service - National Oceanic and Atmospheric Administration (NOAA). Accessed 28th November 2024 at https://www.weather.gov/media/ajk/brochures/NumericalWeatherPrediction.pdf

[12] European Centre for Medium-range Weather Forecasts (ECMWF), (2023). Forecast User Guide, Section 2.1.2.4 Ensemble Control ex-HRES, Section 2.1.2.1 Medium Range Ensemble forecasts. Accessed 20th October 2024 at https://confluence.ecmwf.int/display/FUG/Section+2.1.2.4+HRES+-+High+Resolution+Forecasts

[13] Mitchell, T. M. (1997). Machine Learning. McGraw-Hill. Accessed 20th December 2024 at https://www.cs.cmu.edu/~tom/mlbook.html

[14] Hersbach, H., Bell, B., Berrisford, P., et al. (2020). The ERA5 global reanalysis. QJR Meteorol Soc, 146: 1999–2049. Accessed 10th December 2024 at https://doi.org/10.1002/qj.3803

[15] Chen, L., Han, B., Wang, X., Zhao, J., Yang, W., Yang, Z. Machine Learning Methods in Weather and Climate Applications: A Survey. Appl. Sci. 2023, 13, 12019. Accessed 20th December 2024 at https://doi.org/10.3390/app132112019

[16] Alake, R. (2022). A Data Scientist's Guide to Gradient Descent and Backpropagation Algorithms. NVIDIA Developer. Accessed 20th December 2024 at https://developer.nvidia.com/blog/a-data-scientists-guide-to-gradient-descent-and-backpropagation-algorithms/

[17] Sivakumar, D. (2023). Introduction to Feed Forward Neural Network. Accessed 4th January 2025 at https://www.scaler.com/topics/deep-learning/introduction-to-feed-forward-neural-network/

[18] Dutt, A. (2018). Intro to Machine Learning: Gradient Descent. Accessed 4th January 2025 at https://anujdutt9.github.io/linear-regression-gradient-descent

[19] Giannì, M., Benedetti, L. (2019), Rome. Wind energy in Italy: recent trends. GSE Studies and System Monitoring Direction. Presented at IEA WIND TASK 11 TOPICAL Expert Meeting 96 – WIND PLANT DECOMMISSIONING, REPOWERING, RECYCLING. Accessed 4th Frebruary 2025 at https://www.gse.it/documenti_site/Documenti%20GSE/Studi%20e%20scenari/Wind%20energy%20in%20Italy_recent_trends_v5.pdf