UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingeniería de Sistemas Informáticos

WindFormer: Pretraining a Spatio-Temporal Transformer for Wind Forecasting

MASTER THESIS

Submitted for the degree of Master by:

Samuel Reyes Sanz

Madrid, 2024

UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingeniería de Sistemas Informáticos

Master Degree in Aprendizaje Automático y Datos Masivos

WindFormer: Pretraining a Spatio-Temporal Transformer for Wind Forecasting

MASTER THESIS

Submitted for the degree of Master by:

Samuel Reyes Sanz

Under the supervision of:
Dr. Calos Camacho Gómez
Dr. Javier Huertas Tato

Madrid, 2024

Acknowledgement

I would like to express my sincere gratitude to my supervisors, Carlos and Javier, for their dedication and commitment to this project. They, along with the AIDA research group, have instilled in me a passion for research. Thank you for your guidance, support, and constant motivation.

Abstract

Efficient and accurate wind forecasting presents a substantial challenge in the field of meteorology, particularly when it involves high-resolution data. Accurate prediction of wind speed and direction is crucial for the safety of areas such as aviation and wind turbine operations, and for the efficiency of renewable energies. Traditional physical and numerical models often struggle with this complexity, resulting in diminished and slow performance. Similarly, recent advancements in deep learning have introduced encoder-decoder models aimed at improving forecasting accuracy. However, these models continue to face efficiency issues and struggle to balance the trade-off between accuracy and computational demand. This situation underscores the pressing need for innovative solutions capable of surpassing the constraints of current forecasting models by offering both high accuracy and efficiency.

In response to this challenge, WindFormer, a novel transformer-based model inspired by Video Vision Transformers and adapted into a decoder-only architecture, was developed. This design allows WindFormer to effectively capture temporal and spatial patterns in atmospheric data. WindFormer is trained on high-resolution ERA5 reanalysis data that specifically targets a 3D grid across the Iberian Peninsula with a spatial resolution of 0.25^{o} and a temporal granularity of one hour. The goal of WindFormer is twofold: to serve as a robust pre-trained model that can be fine-tuned with observational wind data and to offer efficient predictions over reanalysis data. Such an architecture proves advantageous in handling the vast datasets typical of weather data and allows for the intricate extraction of relevant features and patterns critical to forecasting. The results of this application demonstrate the effectiveness of WindFormer, achieving similar Root Mean Square Error (RMSE) and Anomaly Correlation Coefficient (ACC) values compared with state-of-the-art models, highlighting its capability to provide accurate and efficient wind speed forecasts using a decoder-only architecture.

This breakthrough not only achieves good results in wind forecasting but also provides a scalable framework that can be adapted for various predictive applications in meteorology and beyond.

The code is available at https://github.com/SamuReyes/WindFormer.

Resumen

La predicción eficiente y precisa del viento presenta un desafío sustancial en el ámbito de la meteorología, especialmente cuando involucra datos de alta resolución. Ser capaz de predecir con precisión la velocidad y dirección del viento es crucial para la seguridad en la aviación o de los aerogeneradores, así como para la eficiencia de la producción y consumo de las energías renovables. Los modelos físicos y numéricos tradicionales a menudo luchan con esta complejidad, lo que resulta en un lento rendimiento. De manera similar, los avances recientes en el aprendizaje profundo han introducido modelos de codificador-decodificador destinados a mejorar la precisión de las predicciones. Sin embargo, estos modelos continúan enfrentando problemas de eficiencia, luchando por equilibrar la compensación entre precisión y demanda computacional. Esta situación subraya la necesidad de soluciones innovadoras capaces de superar las limitaciones actuales ofreciendo tanto alta precisión como eficiencia.

En respuesta a este desafío, se desarrolló WindFormer, un innovador modelo basado en transformadores inspirado en los Video Vision Transformers y adaptado a una arquitectura solo de decodificador. Este diseño permite que WindFormer capture efectivamente los patrones temporales y espaciales en los datos atmosféricos. WindFormer está preentrenado en datos de reanálisis ERA5 de alta resolución sobre una cuadrícula 3D de la península ibérica con una resolución espacial de 0.25^{o} y una granularidad temporal de una hora. El objetivo de WindFormer es doble: servir como un modelo preentrenado robusto que puede ser afinado con datos de viento reales y ofrecer predicciones sobre datos de reanálisis de manera eficiente. Tal arquitectura resulta ventajosa para manejar los enormes conjuntos de datos meteorológicos y permite la extracción de características y patrones relevantes para la predicción. Los resultados de esta aplicación demuestran la efectividad de WindFormer, obteniendo valores de Error Cuadrático Medio (RMSE) y Coeficiente de Correlación de Anomalías (ACC) similares en comparación con los modelos estado del arte, destacando su capacidad para proporcionar pronósticos de viento precisos y eficientes utilizando una arquitectura de solo decodificador.

Este avance no solo obtiene buenos resultados en la predicción de viento, sino que también proporciona un marco escalable que podría adaptarse potencialmente para una amplia gama de aplicaciones predictivas en meteorología y otros ámbitos.

El código está disponible en https://github.com/SamuReyes/WindFormer.

Table of Contents

List of Figures

List of Tables

Abbreviations and acronyms

ACC Anomaly Correlation Coefficient

AI Artificial Intelligence

ARIMA Autoregressive Integrated Moving Average

CNN Convolutional Neural Network

CUDA Compute Unified Device Architecture

ERA5 Fifth Generation ECMWF Reanalysis for the Global Climate and Weather

GDDR6 Graphics Double Data Rate 6

GPU Graphics Processing Unit

HDF5 Hierarchical Data Format version 5

IFS Integrated Forecasting System

LLM Large Language Model

LSTM Long Short-Term Memory

MSE Mean Squared Error

NWP Numerical Weather Prediction

RAM Random Access Memory

RMSE Root Mean Square Error

SVM Support Vector Machine

ViT Vision Transformer

VRAM Video Random Access Memory

W-MAE Weather Masked AutoEncoder

3DEST 3D Earth-Specific Transformer

Chapter 1

Introduction

Precise wind prediction plays a key role in a wide range of applications, from wind energy forecasting to enhancing safety in numerous sectors. Accurate wind forecasts are essential to prevent flight diversions and delays [2, 3, 4], avoid the collapse of wind turbines during extreme wind events [5], and ensure reliable wind power generation [6]. These applications highlight the significance of accurate wind predictions, which can lead to economic benefits, increased safety, and improved efficiency in green energy practices.

However, predicting wind flows, especially in the short term, is challenging due to the inherently variable nature of atmospheric behavior. Wind dynamics are influenced by several factors including but not limited to topographical effects, atmospheric conditions, and temporal variability [7]. These factors contribute to the high randomness and non-linearity of wind data, making short-term predictions particularly challenging. Traditional forecasting methods often struggle with these complexities because their reliance on assumptions of stationary processes, which are rarely applicable to real-world wind patterns [8]. The unpredictable nature of wind requires robust analytical models capable of capturing both the spatial (1.1) and temporal correlations within meteorological data to improve prediction accuracy [9]. The five main methodologies for wind prediction are based on physical, statistical, machine learning, hybrid, and deep learning approaches.

Physical models predict wind speed by utilizing meteorological and geographic parameters, such as temperature, air density, topography, or pressure, which are provided by Numerical Weather Prediction (NWP) systems [10]. While often accurate, these methods are time-consuming and computation-intensive, requiring substantial computational resources. In addition, these models heavily rely on precise initial conditions, meaning that any inaccuracies or uncertainties in the initial observational data can lead to errors in the predictions.

Both **statistical methods** and **machine learning models** leverage historical wind speed data to enhance short-term forecasting accuracy. Statistical methods, such as Autoregressive Integrated Moving Average (ARIMA) models [11], are commonly used. Moreover, additional approaches like Particle Swarm Optimization and Kalman filters can be employed to refine predictions further [12]. However, statistical models may struggle with non-linear wind speed variations and lose accuracy over longer horizons due to their reliance on linear assumptions.

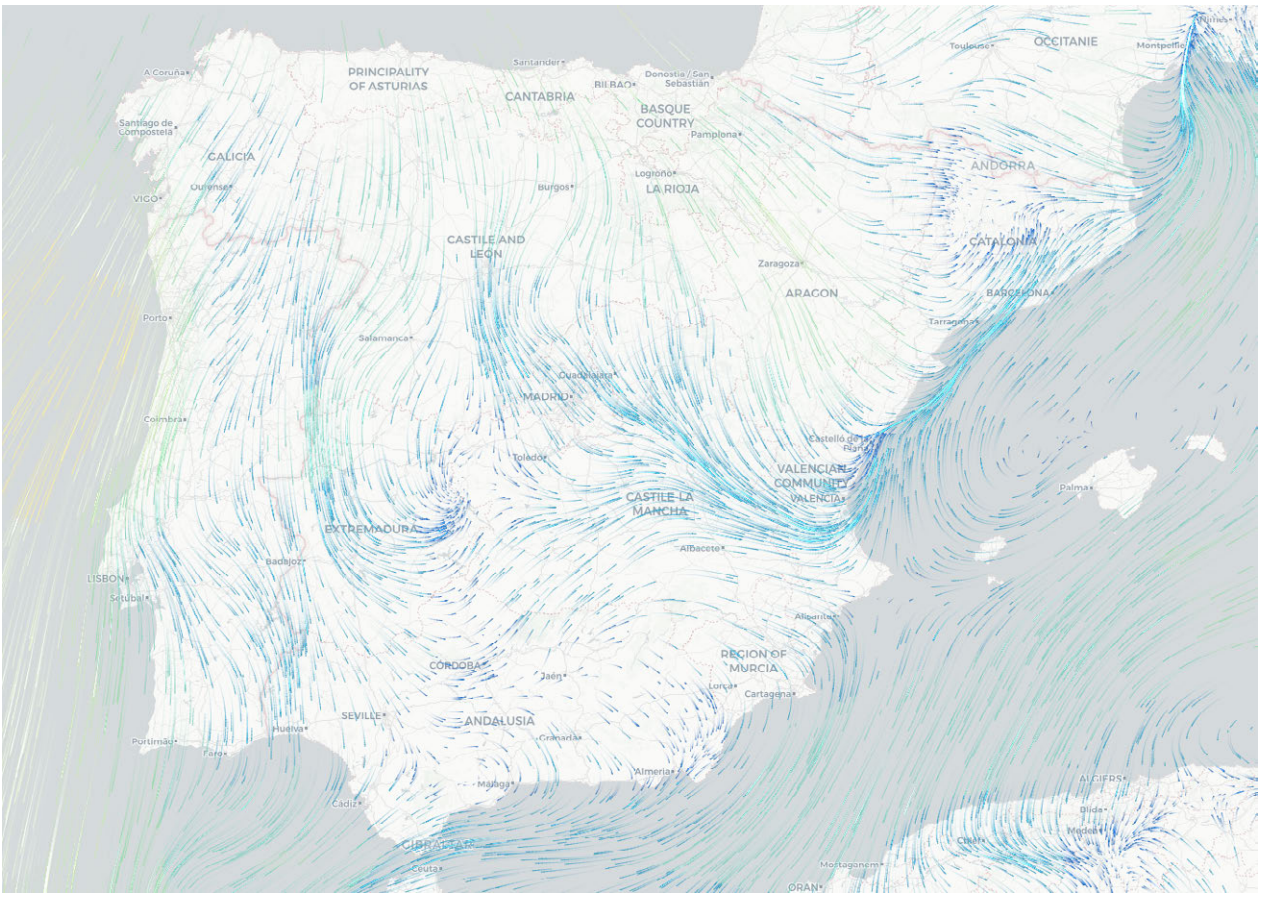

Figure 1.1: The image shows the wind speed and direction over the Iberian Peninsula.

In contrast, machine learning models, including Support Vector Machine (SVM) [13], offer a more robust solution by better fitting non-linear data.

The advent of deep learning technologies has reshaped expectations in wind prediction. **Deep learning models**, which excel in extracting complex features and harnessing large datasets, have surpassed statistical and machine learning models. Among these, Deep Convolutional Neural Networks (CNNs) [14] and Long Short-Term Memory networks (LSTMs) [15, 16] stand out for their precise wind speed forecasting.

Hybrid models combine the strengths of individual models to obtain high-precision predictions. Some hybrid techniques introduce signal preprocessing, such as Adaptive Decomposition Methods [17]. Others take advantage of several models to reduce their drawbacks and enhance their strengths [18, 19].

In recent years, the success of **transformers** [20] in language processing has led to their application in other domains such as image [21] and video analysis [22]. Inspired by these developments, models like PanguWeather [1] and ClimaX [23] have successfully adapted transformer technology for meteorological forecasting. These models excel in capturing spatial and temporal relationships among meteorological variables, improving accuracy and efficiency compared with traditional NWP models like the Integrated Forecasting System (IFS). Despite

these advancements, the encoder-decoder architectures used in these transformer models remain computationally expensive, posing challenges in operational settings where real-time forecasting is crucial or computational resources are limited. However, in recent years, the rise of decoder-only models, which have been shown to be equally effective in seq2seq tasks as encoder-decoder models [24, 25], has sparked interest due to their increased efficiency.

These transformer-based models are often initially trained as pre-trained models on reanalysis data, which consists of past observations reprocessed using modern forecasting techniques to create a consistent historical record. This pre-training allows the models to develop a deep understanding of weather patterns over time. Subsequently, they can be fine-tuned or adapted to specific forecasting tasks using real data.

Given these shortcomings, this study introduces a novel approach by developing a transformer-based model with a decoder-only architecture. This design enhances computational efficiency by focusing solely on the generative aspects of forecasting, thus optimizing speed and resource use. The main objectives of the model are to provide a scalable and efficient tool for meteorological forecasting, pre-trained on reanalysis data, that can later be fine-tuned with observational data for specific tasks. Additionally, this model can offer efficient regional predictions using reanalysis data. This model serves as a foundational work that could be adapted to other prediction problems leveraging spatial and temporal features or even video generation tasks.

The principal contributions of this article are as follows:

1. Development of WindFormer, a transformer-based model with a decoder-only architecture adapted for prediction tasks, which is more efficient than current encoder-decoder models.
2. Implementation of a model that operates at high resolutions (0.25^{o}) and various pressure levels on an hourly basis, capable of working with a large number of meteorological variables and offering very precise short-term predictions.
3. Demonstrating the versatility of the model to adapt to different regions and sets of variables.
4. Highlighting the potential of the model to serve as a pre-trained model on reanalysis data that can be fine-tuned with observational data.
5. Validating the model's performance through experiments, achieving similar Root Mean Square Error (RMSE) and Anomaly Correlation Coefficient (ACC) values compared to state-of-the-art models that use more inefficient architectures.

This paper is organized as follows: Chapter 2, "State of the Art" (2), reviews the foundational models influencing our development. Chapter 3, "Methodology" (3), introduces our model's architectural framework. Chapter 4, "Experimental Setup" (4), describes the dataset, preprocessing techniques, and experimental methodology. Chapter 5, "Experiments and Results" (5), assesses the model's performance, presents findings, addresses limitations, and suggests future work. Finally, Chapter 6, "Conclusions" (6), summarizes key outcomes and potential impacts in wind forecasting.

Chapter 2

State of the Art

In this section, a review of numerical and deep learning methods for weather forecasting will be conducted, along with a discussion of relevant techniques such as pretraining methods to enhance forecasting accuracy.

2.1 Numerical Weather Prediction

Numerical Weather Prediction (NWP) models are essential tools in meteorological forecasting, which use physical laws and mathematical equations to simulate atmospheric dynamics. These models integrate data such as temperature, pressure, and topography to accurately predict future weather states [26]. However, NWP models are computationally intensive, primarily due to their reliance on complex partial differential equations that simulate thermodynamics [27]. The need for high-resolution predictions often requires supercomputers because finer grid spacing substantially increases computational costs. Furthermore, the accuracy of these predictions critically hinges on the quality of the initial conditions, which are susceptible to errors due to uncertainties in observational data [26]. Efforts to optimize these models include adaptive grids and hybrid approaches [10], yet the computational demands remain a significant challenge, particularly affecting the timeliness of updates and limiting the scope for ensemble predictions essential for probabilistic forecasting [28].

2.2 AI-based Methods

Recent advancements in AI-based methods for weather forecasting have emphasized the adoption of transformer architectures. Notable examples include models like PanguWeather [1], ClimaX [23] or W-MAE [29], which use transformer technology, originally developed for natural language processing, to enhance numerical weather prediction. These architectures improve forecasting by capturing complex spatial and temporal patterns in meteorological data, integrating various data variables, and employing attention mechanisms.

These architectures were inspired by Vision Transformers (ViT) [30], a transformer-based model that works with image data by dividing it into fixed-size patches. Each patch is treated

as a token, similar to words in natural language processing. This allows ViT to apply the self-attention mechanism of transformers directly to patches of an image, enabling it to learn contextual relationships between different parts of the image.

Although transformer-based models are generally more efficient than traditional simulation-based NWP models, their application at very high resolutions requires significant computational resources. This high demand can compromise their effectiveness in tasks requiring real-time prediction. In this work, a decoder-only architecture is proposed to further enhance the efficiency of these models, in contrast to the predominant encoder-decoder architectures.

Pangu-Weather [1] is an AI-based system for global weather forecasting. It distinguishes itself by incorporating a 3D Earth-Specific Transformer (3DEST), which is an architecture based on the Vision Transformer. After performing patch embedding of the data, it aggregates them into a three-dimensional cube representing height, longitude, latitude, and embedding dimension. This cube is then processed by the 3DEST module using an encoder-decoder architecture to reconstruct the subsequent time step. Pangu-Weather has demonstrated superior performance over traditional NWP methods, achieving better accuracy in predicting various meteorological variables across different time ranges. This is achieved by predicting different lead times through the combination of models with varying lead times, effectively extending the forecast horizon. This model not only supports robust forecasts for general weather conditions but also excels in challenging tasks such as tracking tropical cyclones.

W-MAE [29] uses a Masked AutoEncoder (MAE) pre-training approach within a ViT framework to forecast weather by capturing intricate spatial correlations in meteorological data. This model stands out for its capability to efficiently process large datasets through a novel decoder design that optimizes computational resources. During its pre-training phase on ERA5 data, W-MAE employs a technique in which it randomly masks patches of meteorological images and reconstructs them, thereby learning to capture complex spatial correlations within the data. This method significantly reduces prediction errors and stabilizes forecast performance over varying time spans. Excelling in predicting key weather variables such as precipitation, W-MAE consistently outperforms traditional models like FourCastNet [31].

ClimaX [23] is an advanced implementation of transformer-based models for weather and climate forecasting. Designed to be both flexible and generalizable, ClimaX employs a mechanism to tokenize meteorological variables separately and then aggregate them to enhance the model's efficiency. This allows the model to be flexible with datasets containing different meteorological variables and scales. Once these tokens are processed using a modified ViT architecture, ClimaX can execute sophisticated predictive tasks, even with spatially incomplete or region-specific data. The model was pre-trained on extensive climate datasets using a self-supervised training approach.

2.3 Decoder-Only Architecture

The evolution of meteorological prediction has seen a shift from traditional NWP to deep learning models, particularly those employing transformer architectures like ViT [1] [23] [29] or utilizing attention mechanisms [32] [9]. These models employing encoder-decoder architectures

were not originally designed with forecasting as their primary function, leading to inefficiencies in computational resource usage. Recent advances have highlighted decoder-only models as state-of-the-art foundational models for time series forecasting [25], which are more suitable for predictive tasks.

In this work, a decoder-only model is proposed for spatio-temporal weather forecasting, drawing inspiration from models such as Video Vision Transformers (ViViT) [22], which effectively handle the relationships between patches across multiple video frames. This approach utilizes a decoder-only architecture, which streamlines processing by generating future states directly from past and present inputs without the intermediary encoder stage traditionally used to process data.

In the decoder-only transformer architecture, the prediction process for weather forecasting unfolds sequentially across time steps. Specifically, the model begins with the initial time step and uses it to predict the next one. For each subsequent time step, the model takes all previous predicted steps as input and builds upon them to predict the next one. This cumulative approach continues for a total of t time steps, with each prediction informed by the data from all preceding moments. In this way, the model receives as input t time steps and returns as output the same number of time steps shifted by one unit.

2.4 Pretraining Models

Large Language Models (LLMs) demonstrate the effectiveness of pretraining in deep learning. These models are initially trained with self-supervised learning on vast corpora of text, enabling them to gain a broad understanding of language patterns and structures. Once pre-trained, LLMs can be fine-tuned for specific tasks such as text classification, sentiment analysis, or content generation, showcasing their versatility and adaptability [33].

This pretraining approach is not exclusive to natural language processing and has been effectively applied to other fields, including meteorology. Using large reanalysis datasets like ERA5 [34, 35], this method has proven highly effective for developing models that capture climate relationships. These models can subsequently be adapted to specific tasks or observational data with minimal additional input. Examples include meteorological forecasting [23] and other applications like fire prediction [36].

Chapter 3

Methodology

This study introduces WindFormer, an advanced predictive model for regional wind forecasting, which is trained on an extensive 45-year ERA5 reanalysis dataset in a limited area. This rich dataset encompasses hourly records of numerous meteorological variables, providing a deep historical context.

WindFormer is designed as a versatile pre-trained model that can be swiftly adapted to various forecasting tasks and datasets. The architecture of WindFormer, which employs a decoder-only setup, draws inspiration from groundbreaking models like the Video Vision Transformers [22] and other transformer-based meteorological models [23, 1]. This decoder-only architecture allows greater focus on prediction tasks and efficiency.

3.1 Task Definition

The task addressed by WindFormer is to forecast future meteorological conditions based on historical data. This model processes time series data represented by 2-dimensional matrices of surface data and 3-dimensional matrices of altitude data over specified time steps. The architecture is designed to ensure that the input and output dimensions are identical, allowing the outputs to be recursively used as inputs for extended forecast horizons. Its core function is to predict the future sequence of these matrices by shifting them forward by one time step. Specifically, for t time steps, WindFormer predicts each subsequent time step using a cumulative approach: the first time step to predict the second, the first and second to predict the third, and so on up to t. This process is facilitated by its decoder-only transformer architecture, which is optimized for generative tasks.

Initially, WindFormer serves as a pre-trained model that captures complex meteorological relationships from extensive reanalysis datasets. The pre-training phase equips the model with a robust understanding of weather dynamics, enabling it to be fine-tuned with specific real-world data for targeted forecasting tasks.

3.2 Model Architecture

This section describes the design of the proposed architecture, which is referred to as WindFormer. The architecture is based on several models known for their effective management of spatio-temporal dynamics. It draws inspiration from the Factorized Encoder model used in Video Vision Transformers (ViViT) [22], as well as other encoder-decoder frameworks such as ClimaX [23] or Pangu Weather [1], which have proven highly effective in weather forecasting. The overall structure of WindFormer is depicted in Figure 3.1. Adapting to these influences, WindFormer employs a decoder-only approach, specifically tailored to enhance its efficiency and predictive capabilities.

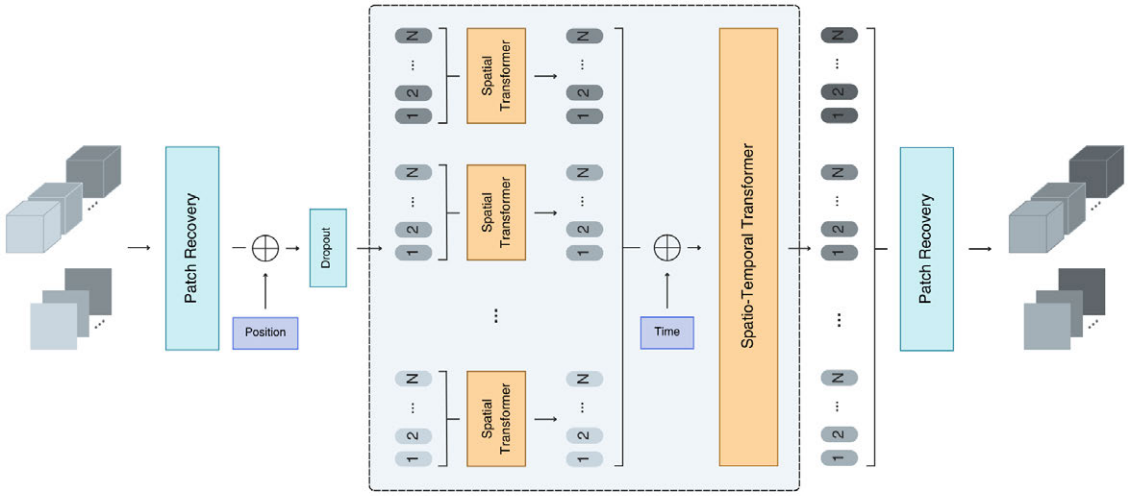

Figure 3.1: An overview of the WindFormer architecture.

The model processes two types of data inputs: surface data and upper-level data. The input dimensions are [B, T, Z, W, H, C] for upper-level data and [B, T, W, H, C] for surface data, where B denotes batch size, T represents the temporal dimension, Z indicates pressure levels, W is latitude, H is longitude, and C represents meteorological variables. This data is first split into patches using a common technique in computer vision called patch embedding, and these patches are then converted into embeddings employing convolutions and adding the positional encoding. These embeddings are separately processed through the same spatial transformer for each time step, after which the temporal encoding is added before being combined and processed in a spatio-temporal transformer. The resulting embeddings are transformed to match the original input dimensions. In line with the decoder-only architecture, the model takes T time steps as input, referred to as the context window, and outputs the same number of time steps, each shifted one step forward in time.

Patch Embedding. As shown in Figure 3.2, upper-level data is split into patches with dimensions [z, w, h]=[3,6,6] and surface-level data in [w, h]=[6,6], resulting in a total of 36 patches, assuming the parameters outlined in Table 4.2. Similar to the ViT architecture

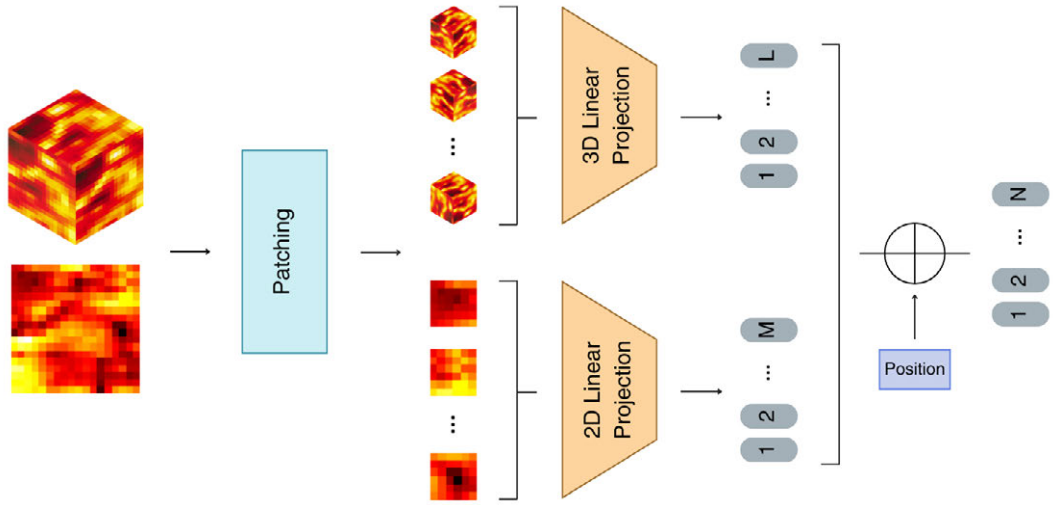

Figure 3.2: Illustration of the patch embedding module, showing how upper-level and surface-level data are divided into L and M patches respectively, and then processed through linear projections and embeddings into a total of N patches.

[21], a linear projection is applied to each patch, flattening them to the pre-established embedding dimension. Subsequently, positional embedding is added to the patches to preserve information related to their spatial order.

Spatial Transformer. The spatial transformer employed in the model uses the original architecture as proposed in [20]. It features a self-attention mechanism in which all patches within the same time step attend to each other, enhancing the model's ability to capture spatial interactions.

Spatio-temporal Transformer. The architecture for the spatio-temporal transformer is similar to that of the spatial transformer. However, it incorporates a custom attention mask to ensure that each patch can attend only to patches from the same or preceding time steps and not to any future time steps. This constraint is crucial for preserving the causal structure of the data and preventing information leakage from future states. The custom attention mask used in this transformer is illustrated in Figure 3.3. Before entering the spatio-temporal transformer, temporal embedding is added to the patches.

Patch Recovery. In the case of Patch Recovery, after passing the embeddings through the transformer layers, linear projection and convolution are applied. This projection restores the patch's original shape, after which the patches are concatenated to form an image similar to the input but with a higher channel resolution. A final convolutional layer then adjusts the image to match the original input dimensions.

Model Parameters. The model architecture is available in two versions: "Base" and "Small," each inspired by the parameter settings used in the BERT model [37]. The "Base" model features an embedding dimension of 768, a multilayer perceptron (MLP) size of 3072, 12 layers of depth, and 12 attention heads, designed for robust processing capabilities. It comprises approximately 183 million parameters, allowing for complex pattern recognition and learning. The "Small" model, optimized for faster processing, includes an embedding dimension of

Figure 3.3: Attention mask employed in the Spatio-temporal Transformer, assuming three time steps (t) and three patches (p) per time step.

Table 3.1: Parameters of the WindFormer Base and Small Models

Parameter	**Base**	**Small**
Embedding dimension	768	512
Depth	12	6
Heads	12	6
Dimension per Head	64	32
MLP Size	3072	1024
Transformer Dropout	0.1	0.1
Embedding Dropout	0.1	0.1
Total parameters	183M	26M

512, MLP size of 1024, 6 layers of depth, and 6 attention heads, totaling about 26 million parameters. This reduction in parameters enhances computational efficiency while maintaining effective performance. Both models maintain a consistent approach with dimensions per head of 64 and 32, respectively and employ a dropout rate of 0.1 for both transformer and embedding dropout to prevent overfitting and help generalization. Detailed specifications of each model are summarized in Table 3.1.

Design Choices. The architecture of WindFormer has been crafted to ensure maximum versatility and automation in data processing and model adjustments, implemented using the PyTorch framework [38]. Key design choices include the ability to adjust input sizes, patch dimensions, and specific meteorological variables, as well as the size of the context window. These flexible design parameters have facilitated extensive testing, although they are often limited by Graphics Processing Unit (GPU) memory and training time constraints. This adaptability is crucial in a domain where computational resources are a key factor, enabling tailored application of the model to specific problem contexts.

Chapter 4

Experimental Setup

4.1 Dataset Description

In this section, the dataset used as the foundation for training the WindFormer model and the preprocessing techniques employed are described. Additionally, the experimental setup, including the selection of hyperparameters and evaluation metrics, is detailed.

4.1.1 The ERA5 Reanalysis Dataset

Throughout this work, the ERA5 dataset [34, 35], the fifth generation reanalysis from the European Centre for Medium-Range Weather Forecasts (ECMWF), is utilized. This dataset provides a detailed hourly record of global climate and weather from 1940 to the present. ERA5 merges historical observations with the state-of-the-art meteorological forecasting model called the Integrated Forecasting System (IFS) [39] through a process known as data assimilation, thereby enhancing the accuracy of the reanalysis. It encompasses a broad spectrum of variables, including atmospheric temperatures, wind speeds and precipitation, across ocean-wave and land-surface parameters. The data are presented on a high-resolution 0.25° x 0.25° (approximately 28km x 28km) global latitude-longitude grid and include variables at 137 atmospheric pressure levels, ranging from the surface up to 1 hPa.

A subset of the full ERA5 reanalysis dataset, detailed in Table 4.1, was selected primarily for its relevance to wind prediction and the computational constraints related to its extensive size. The geographical scope of the dataset covers the entire Iberian Peninsula, spanning from 45°N to 35°N in latitude and from 10°W to 4°E in longitude. Data were extracted across multiple pressure levels, ranging from near-surface (950 hPa) to upper atmospheric levels (250 hPa), with measurements taken hourly. The selected period extends from 1979 to 2023, reflecting the higher quality of ERA5 data post-1979. The final dataset dimensions for surface data are $(n_{time}, n_{lat}, n_{lon}, n_{variables}) = (394440, 41, 57, 9)$ and for upper levels data are $(n_{time}, n_{levels}, n_{lat}, n_{lon}, n_{variables}) = (394440, 9, 41, 57, 8)$.

Table 4.1: Downloaded Data from the ERA5 Dataset

Variable Type	Abbreviation	Variables
Surface Variables	u100	100m u-component of wind
	v100	100m v-component of wind
	u10	10m u-component of wind
	v10	10m v-component of wind
	d2m	2m dewpoint temperature
	t2m	2m temperature
	z	geopotential
	msl	mean sea level pressure
	i10fg	instantaneous 10m wind gust
Upper-air Variables	u	u-component of wind
	v	v-component of wind
	q	specific humidity
	t	temperature
	d	divergence
	z	geopotential
	w	vertical velocity
	vo	vorticity
Area		[45N, 10W, 35N, 4E]
Spatial resolution		0.25º
Pressure Levels (hPa)		950, 925, 900, 850, 800, 700, 600, 500, 250
Time Coverage		1979 to 2023
Temporal resolution		Hourly

4.1.2 Atmospheric Variables Relevant to Wind Prediction

In this study, a selective focus was on a range of atmospheric variables from the ERA5 dataset that are particularly related to wind speed and direction. No specific filtering was conducted to identify the best predictors of wind, aiming instead for a generalized model that interprets meteorological relationships and predicts outcomes in subsequent stages. As a pre-training model, it can be adapted to additional variables or a different subset as needed. The details of these variables are as follows [34, 35]:

- **U-component of wind (m/s)**: Eastward horizontal wind component. This value is positive when the wind moves from west to east. This measure is available at multiple levels: 10m, 100m above the surface, and at different pressure levels.
- **V-component of wind (m/s)**: Northward horizontal wind component. This value is positive when the wind moves from the south to the north. This measure is available at multiple levels: 10m, 100m above the surface, and at different pressure levels.
- **Instantaneous 10m wind gust (m/s)**: Peak short-duration wind speed at 10m, key for analyzing extreme wind events. It provides information only about the speed of the

wind and not its direction or components.

- **Temperature (K)**: Measures temperature at different pressure levels and 2m above the earth's surface.
- **2m dewpoint temperature (K)**: The temperature at which air reaches saturation at 2m above the surface, combining factors of humidity, temperature, and atmospheric pressure.
- **Specific humidity (kg/kg)**: Mass of water vapor per kilogram of moist air, which is essential for understanding humidity distribution at higher altitudes.
- **Geopotential (m^2/s^2)**: Refers to the potential energy per unit mass at a specific elevation above sea level, reflecting the energy required to elevate a mass from sea level to that height.
- **Mean sea level pressure (Pa)**: This parameter represents atmospheric pressure at Earth's surface, adjusted to mean sea level. It calculates the weight of air above a point, assuming it is at sea level, and is used to identify high and low pressure areas. Closer contours on maps suggest stronger winds.
- **Divergence (s^-1)**: Rate of horizontal spreading or convergence of air, affecting cloud formation and weather systems at altitude.
- **Vertical velocity (Pa/s)**: Speed of air movement vertically, instrumental for identifying regions of atmospheric lifting and subsidence.
- **Vorticity (s^-1)**: Measure of rotation of air around a vertical axis, key for analyzing large-scale rotation in weather systems.

4.1.3 Data Preparation

For data preparation, the approach began with a series of monthly NetCDF files from 1979 to 2023. For each month, there are two distinct files: one containing surface variables and another containing altitude variables. Given the total volume of nearly 150 GB, merging all files directly into a single structure was impractical due to memory constraints. Therefore, data from each year were merged into separate numpy arrays, one for surface data and one for altitude data. These arrays were then stored in an HDF5 file, which allows efficient data organization and loading by year and type without overwhelming the RAM capacity.

To prepare the data for training the model, a series of transformations were applied to the raw meteorological data. Initially, anomalies were calculated for each variable at each geographic location. Anomalies are defined as deviations of observed values from their long-term averages during the reference period (1981-2010) [40], known as climatology. For each meteorological variable, climatology is calculated as the daily mean over a specified historical period. By subtracting this climatological mean from each observed value, the anomalies are obtained. This process helps highlight significant deviations that exceed typical seasonal variations, thereby enhancing the focus on unusual or extreme meteorological events. This step is crucial because it mitigates the effects of annual seasonality, allowing the model to better

generalize across different temporal segments [40]. After extracting the anomalies, the data were standardized by calculating the mean and standard deviation of each meteorological variable at every altitude level. Standardization, rather than normalization, was employed because of its effectiveness in managing outliers in the dataset. To minimize the computational cost of the model, the dataset's spatial dimensions are reduced by focusing on a targeted geographical area. The selected region is centrally located on the peninsula, bounded by latitudes from 38.5ºN to 42.5ºN and longitudes from 5.5ºW to 1.5ºW. This spatial constriction effectively narrows the dataset to a manageable size of 18x18 grid points. Figure 4.1 illustrates this spatial reduction, with the blue rectangle representing the available geographical data and the green rectangle highlighting the selected area for analysis.

Figure 4.1: The blue rectangle delineates the available geographical data. Within this, the green rectangle highlights the selected area for analysis.

The dataset was strategically partitioned into training, validation, and test sets. Data from the years 1979 through 2019 were used for training the model. The years 2020 and 2021 were designated for validation, and the years 2022 and 2023 were reserved for testing. Both the validation and test sets were designed as representative samples.

4.2 Experiment Setup

A series of experiments were designed to train the WindFormer model using the extensive ERA5 reanalysis dataset. The primary goal is to pre-train the model effectively, enabling it to serve as a foundational model for subsequent fine-tuning with real-world data. This approach leverages historical climate data to establish robust pattern recognition capabilities before application to current and future meteorological conditions.

For optimization, the Adam algorithm [41] is employed with a learning rate of 0.0001. The learning rate is managed by a linear scheduler that decreases the learning rate linearly from the initial set value, ensuring gradual reduction in learning rate adjustments as training progresses. The model is trained using the Mean Squared Error (MSE) loss function, which quantifies the difference between predicted and actual values, providing a clear measure of model accuracy.

To enhance training efficiency and manage computational resource use, mixed precision training is used. By executing operations in 16 bits precision, the model benefits from faster computations and reduced memory usage, which is crucial given the large size of the training dataset and the high number of model parameters.

Experiment tracking and monitoring were conducted using Weights & Biases [42], a platform for experiment tracking and model management in machine learning. Key metrics such as training and validation loss, root mean square error (RMSE), and computational resource usage are recorded. Additionally, heatmaps of the results are generated to visually assess the model performance. Validation loss is logged both mid-epoch and at the end of the epoch to provide timely feedback on the model's performance, given the substantial volume of training data.

The hardware employed for this task is an RTX 8000, with 48 GB of GDDR6 memory and 4608 CUDA cores. VRAM (Video Random Access Memory) is a crucial aspect in this problem, especially as larger transformer models, wider context windows, or larger areas to cover increase memory requirements. Despite the data being stored in memory by year, processing the entire dataset requires a good amount of RAM (Random Access Memory), at least 64GB of RAM to avoid memory saturation.

The complete code and implementation details can be found at the following GitHub repository: https://github.com/SamuReyes/WindFormer.

4.3 Hyperparameters

The project code and model architecture offer significant versatility, allowing for modifications in various aspects such as geographical area, meteorological variables, surface patch size, altitude patch size, transformer model size, and other adjustable parameters. In addition, typical parameters such as the learning rate and batch size can be adjusted. Table 4.2 presents the base parameters used as references for the experiments.

Table 4.2: Summary of Hyperparameters

Parameter	Value
Upper-air Variables	u, v, q, t, d, z, w, vo
Surface Variables	u100, v100, u10, v10, d2m, t2m, z, msl, i10fg
Pressure Levels (hPa)	250, 500, 600, 700, 800, 850, 900, 925, 950
Years Covered	1979 to 2023
Training Data Split	1979 to 2019
Validation Data Split	2020 to 2021
Testing Data Split	2022 to 2023
Model Image Size (3D)	9x18x18x8
Patch Size (3D)	3x6x6
Model Image Size (2D)	18x18x9
Patch Size (2D)	6x6
Sequence Length	12
Embedding Dimension	768
Depth	12
Heads	12
Dimension per Head	64
MLP Size	3072
Dropout Rate	0.1
Embedding Dropout	0.1
Reconstruction Dropout	0.1
Batch Size	64
Learning Rate	0.0001

4.4 Evaluation Metrics

Following prior spatio-temporal forecasting methods [23, 1, 29], to accurately assess the WindFormer model, the following latitude-weighted Root Mean Square Error (RMSE) and Anomaly Correlation Coefficient (ACC) are employed:

$$RMSE = \sqrt{\frac{\sum_{i,j}^{N_{\text{lat,lon}}} L(i) \left(\hat{A}_{i,j} - A_{i,j}\right)^2}{N_{\text{lat}} \times N_{\text{lon}}}}, \tag{4.1}$$

$$ACC = \frac{\sum_{i,j}^{N_{\text{lat,lon}}} L(i) \hat{A}_{i,j} A_{i,j}}{\sqrt{\sum_{i,j}^{N_{\text{lat,lon}}} L(i) \left(\hat{A}_{i,j}\right)^2 \times \sum_{i,j}^{N_{\text{lat,lon}}} L(i) \left(A_{i,j}\right)^2}}, \tag{4.2}$$

where:

- $\hat{A}_{i,j}$ is the predicted anomaly at the latitude index i and longitude index j.

- $A_{i,j}$ is the corresponding true anomaly.
- N_{lat} and N_{lon} are the total number of latitude and longitude points, respectively, covering the geographic area under study.
- $L(i)$ represents the latitude weighting factor, defined as:

$$L(i) = N_{\text{lat}} \times \frac{\cos \phi_i}{\sum_{i=1}^{N_{\text{lat}}} \cos \phi_i} \tag{4.3}$$

- ϕ_i is the latitude angle at index i.

These metrics are applied for each meteorological variable, pressure level, and lead time, ensuring a comprehensive assessment across the model's entire prediction spectrum.

The Root Mean Square Error (RMSE) measures the average magnitude of the prediction errors. It is the square root of the average of squared differences between predicted and observed values. RMSE provides a way to quantify the difference between values predicted by the model and those observed in reality, with lower values indicating better performance.

The Anomaly Correlation Coefficient (ACC) is a commonly used metric for verifying spatial fields. It represents the spatial correlation between forecast anomalies relative to climatology and the corresponding analysis anomalies relative to climatology. ACC measures how well the forecast anomalies represent the observed anomalies, indicating how well the predicted values from a forecast model "fit" with the real-life data. ACC values range from +1 to -1, with values near +1 indicating good agreement, values around 0 indicating poor agreement, and values near -1 indicating the forecast is in anti-phase with observations, meaning the predictions are very unreliable [43].

This refined approach adjusts the influence of each error term by the cosine of the latitude, thereby addressing the disparity in grid point spacing between the poles and the equator. At the poles, grid points are densely packed, leading to a higher degree of spatial correlation among points. In contrast, grid points are more widely spaced near the equator. This variation in spacing necessitates a correction to ensure that the metric accurately reflects the variable spatial interdependencies across different latitudes. The cosine adjustment ensures that the weighting reflects this increased interdependence, ensuring a balanced assessment of errors throughout the dataset.

Chapter 5

Experiments and Results

In this section, experiments are conducted using the ERA5 dataset to evaluate the performance of the proposed WindFormer model for spatio-temporal wind speed forecasting. Furthermore, the results and performance metrics are presented to illustrate the model's capabilities and effectiveness.

5.1 Experiments Methodology

The primary objective of the experiments with WindFormer is to verify whether a decoder-only architecture can achieve results comparable to those of encoder-decoder architectures while being more computationally efficient. In addition, the goal is to develop a model capable of understanding meteorological relationships, which can later be fine-tuned with real observational data to adapt to specific forecasting tasks.

As defined in Chapter 4, validation and test sets were used to evaluate the model. Initially, the model was trained for 50 epochs, a number chosen based on available time and resources to allow for testing various parameter combinations while always saving the model with the lowest validation error.

Resource limitations played a significant role in parameter selection. With a constraint of 50GB GPU memory, the total number of meteorological variables was maintained, and the region was cropped as described in Chapter 4. Given this setup and using the base model parameters, the context window could not be increased beyond 12 time steps, and the number of patches was limited to 36.

Comparing WindFormer with state-of-the-art models proved challenging for several reasons. Reproducing these models at their original scale is computationally prohibitive. Additionally, differences in spatial resolution, meteorological variables, selected areas, and data preprocessing make direct comparisons difficult. Nevertheless, the same evaluation metrics were used, and while a direct comparison is not possible, the results suggest that the decoder-only architecture is a feasible solution, achieving similar performance to more complex models.

Although additional experiments were conducted to assess the effectiveness of different

components of WindFormer, they are not highlighted here due to variations in experimental conditions.

5.2 Results Analysis

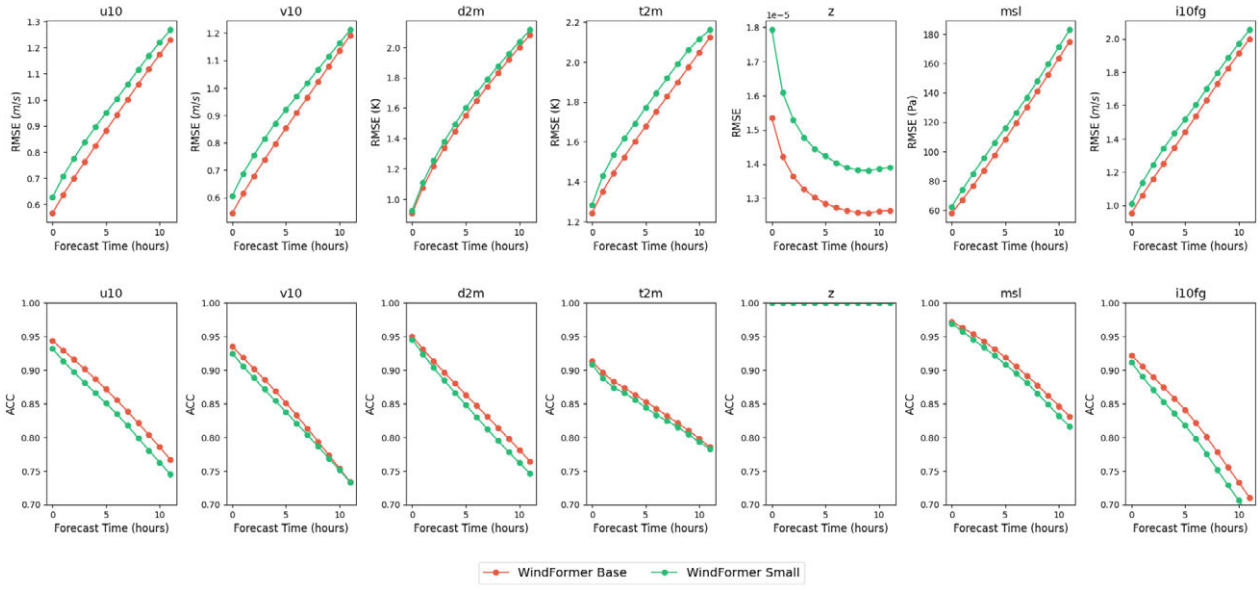

Figure 5.1: RMSE and ACC plots for surface meteorological variables across various lead times, comparing the base and small models.

		Surface		950 hPa		500 hPa		250 hPa	
		u	v	u	v	u	v	u	v
WindFormer Small	RMSE	0.626	0.606	2.167	2.414	0.834	0.791	0.659	0.623
	ACC	0.932	0.925	0.955	0.961	0.936	0.933	0.931	0.927
WindFormer Base	RMSE	0.566	0.543	2.060	2.292	0.734	0.694	0.584	0.548
	ACC	0.943	0.935	0.963	0.962	0.949	0.943	0.945	0.939

Table 5.1: Comparison of WindFormer Small and Base models on RMSE and ACC for u and v wind components at different pressure levels for a 1-hour lead time.

Figure 5.1 compares the performance of the base and small models (defined in Table 3.1) in terms of RMSE and ACC across various lead times for surface variables. The base model consistently achieves slightly better results for all variables and lead times for both metrics. The models exhibit high levels of precision for all variables when using a short prediction horizon. However, as lead time increases, RMSE rises and ACC decreases, reflecting the accumulation of errors due to the iterative prediction process. Notably, the variable z (geopotential) shows better performance, possibly due to its low variability over time, suggesting that it could be eliminated in future experiments (see Table 5.1 for detailed results).

Figure 5.2 presents a comparison of RMSE and ACC metrics for altitude variables at several pressure levels: 950 hPa, 800 hPa, 600 hPa, and 250 hPa, with 950 hPa being the closest

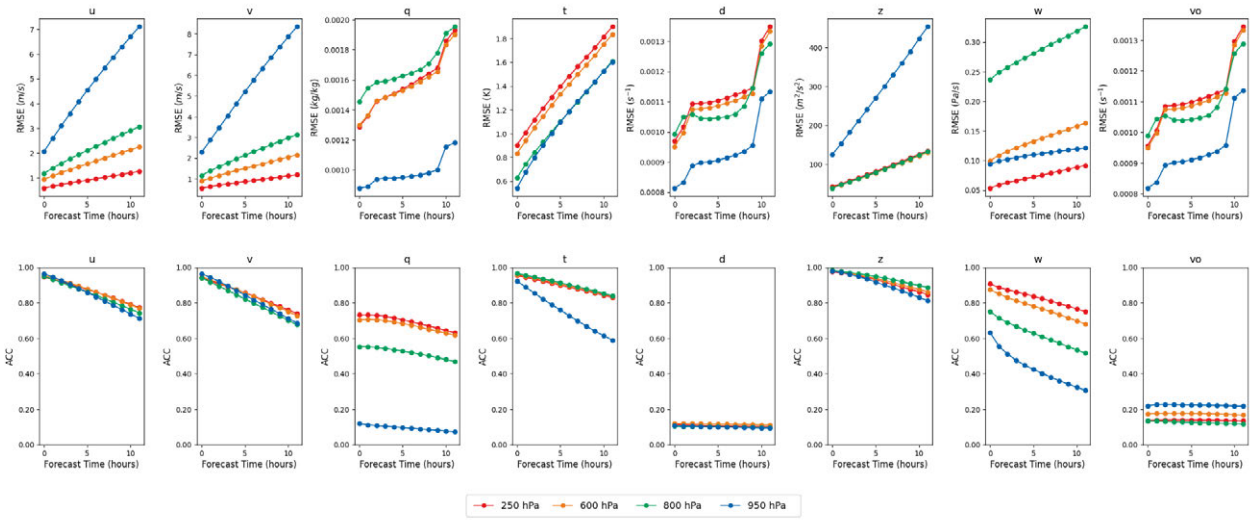

Figure 5.2: RMSE and ACC plots for upper meteorological variables across various lead times, comparing the performance of the base model across different altitude levels, where 950 hPa is the nearest to the surface and 250 hPa is the farthest.

to the surface and 250 hPa being the farthest. The results indicate that depending on the variable, better or worse results are obtained when comparing different pressure levels. This variation is due to the different natures of each type of variable, which can be more volatile or stable at different altitudes. It is important to note that RMSE is not a good metric for comparing a variable across different levels, as these meteorological variables have different magnitudes depending on the altitude at which they are found. Additionally, variables such as divergence (d) and vorticity (vo) exhibit poorer performance, likely due to their more dynamic and less predictable behavior.

In Figure 5.3, the prediction results with the base model for various meteorological variables at the surface and several altitude levels for the first lead time are presented. As observed, the predictions closely match the actual values, indicating good performance. Notably, variables d, vo, and q exhibit significant similarity. Divergence (d) and vorticity (vo) are inversely proportional, suggesting that one of them could be eliminated. At higher levels for t and z, some square patterns are visible, which may result from a lack of convergence in the final convolutional layers of the model. As previously shown in Figure 5.2, predictions at higher pressure levels tend to be of lower quality.

A new experiment was conducted in which the surface variable z and the altitude variables q and vo were eliminated. Figure 5.4 compares the performance of the small model with the reduced set of variables against the original small model for some surface variables and variables at 800 hPa. As can be seen in the results, reducing the number of variables improves the performance, as the problem is simplified and the accumulation of errors is reduced.

As previously mentioned, comparing WindFormer with state-of-the-art models [1, 23, 39] is challenging because WindFormer is a regional model, not a global one. Additionally, it does not use the same set of meteorological variables or the same resolution, and the lead

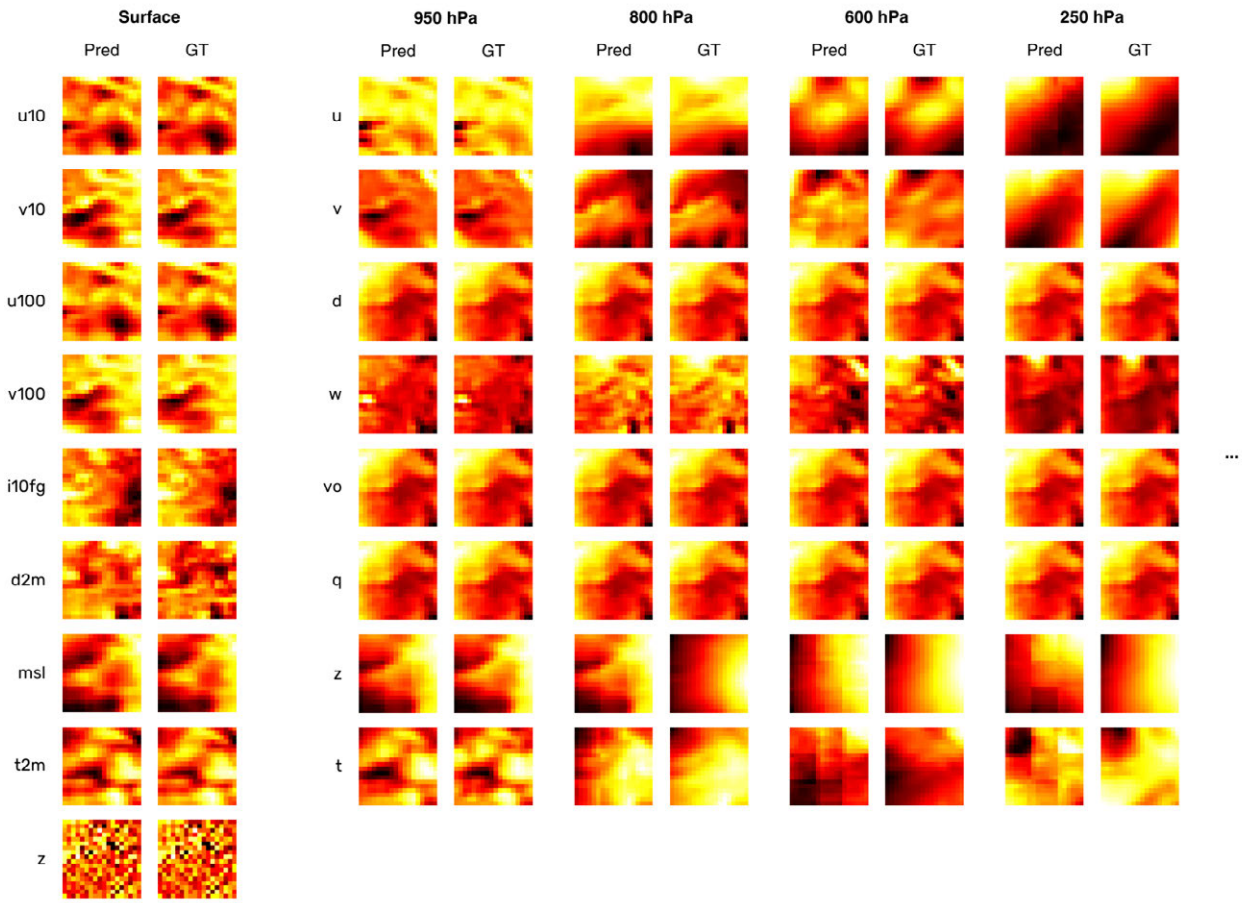

Figure 5.3: Example results from the base model for each surface and upper-level variable at various altitude levels. Predictions are shown for a lead time of 1. The variables are labeled according to Table 4.1.

times used by these models differ. Both RMSE and ACC metrics were employed to evaluate the models. While RMSE might not be the most reliable metric due to differences in how anomalies are calculated, it is notable that WindFormer achieves RMSE values of a similar magnitude, and even lower in some cases. Furthermore, WindFormer exhibits similar or even better ACC values compared to these models. This could be attributed to the smaller number of predictions that it makes. This analysis demonstrates that the model is capable of achieving the task effectively, although it cannot precisely be determined whether it performs better or worse than other models.

Figure 5.5 shows the RMSE and ACC metrics for wind and temperature variables at the surface for Pangu-Weather, IFS, and FourCastNet models. Although the conditions and setups differ, comparing the magnitude of these metrics with those from WindFormer indicates that WindFormer can resolve the task similarly, showcasing its effectiveness.

Furthermore, another objective of this work is to demonstrate the efficiency of the decoder-only architecture compared with the encoder-decoder architecture of these models. The encoder-decoder architecture inherently requires more layers and computational power, whereas the

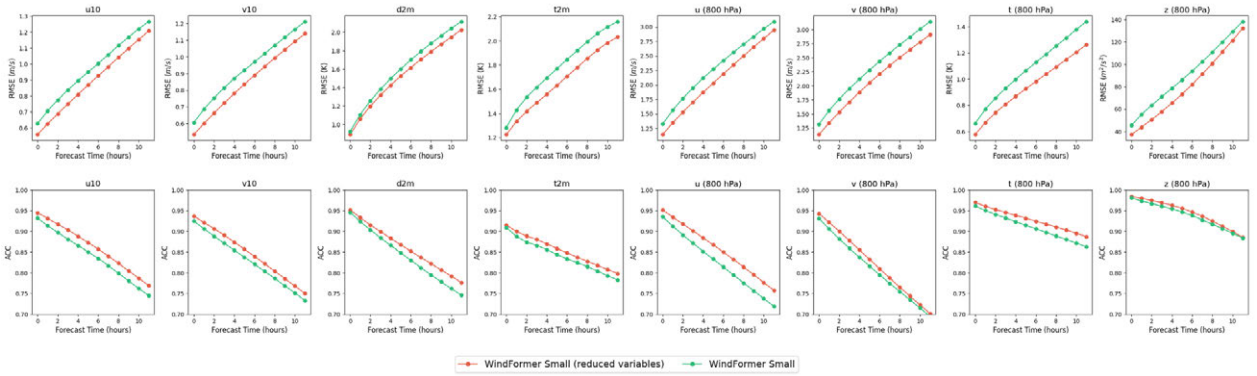

Figure 5.4: Comparison of the small model with the reduced set of variables against the original small model for some surface variables and variables at 800 hPa.

decoder-only architecture is more effective for tasks like prediction [25]. This suggests that the decoder-only approach used in WindFormer is not only capable but also more efficient for the given task.

5.3 Known Limitations and Future Work

While WindFormer demonstrates promising results, several limitations and areas for future improvement have been identified. First, more experiments are necessary to fully explore the potential of the model. Testing different patch sizes, incorporating topographical data, and experimenting with architectural modifications such as applying CoordConv [44] in patch embedding are some avenues to consider. Additionally, trying various combinations or subsets of meteorological variables could help reduce the error propagation that occurs over time, such as conducting more exhaustive studies on the variables d, vo, and q, which present high similarity, or on the surface variable z, which shows little change over time.

Another area for improvement would be to conduct a more precise comparison with state-of-the-art models. To achieve this, these models should be reproduced at the same scale as WindFormer and should use the same training data and preprocessing methods. In addition, it might be beneficial to transform WindFormer from a regional to a global model. However, this would require significant computational resources.

Another significant limitation is the accumulation of errors, which degrades the accuracy of predictions over longer forecast horizons. To address this, solutions like those implemented in PanguWeather [1] could be considered. This involves training multiple models with different lead times and combining their outputs to minimize the propagation of errors in long-term predictions. Additionally, Das et al. [25] propose a modification to the decoder architecture that predicts a longer forecast horizon than the input sequence. Instead of predicting the same input shifted by one time step, this approach forecasts a predetermined horizon, potentially improving long-term prediction accuracy.

Finally, the principal next step is to test the WindFormer architecture with real observational data. This aligns with one of the primary objectives of the model, which is to develop a

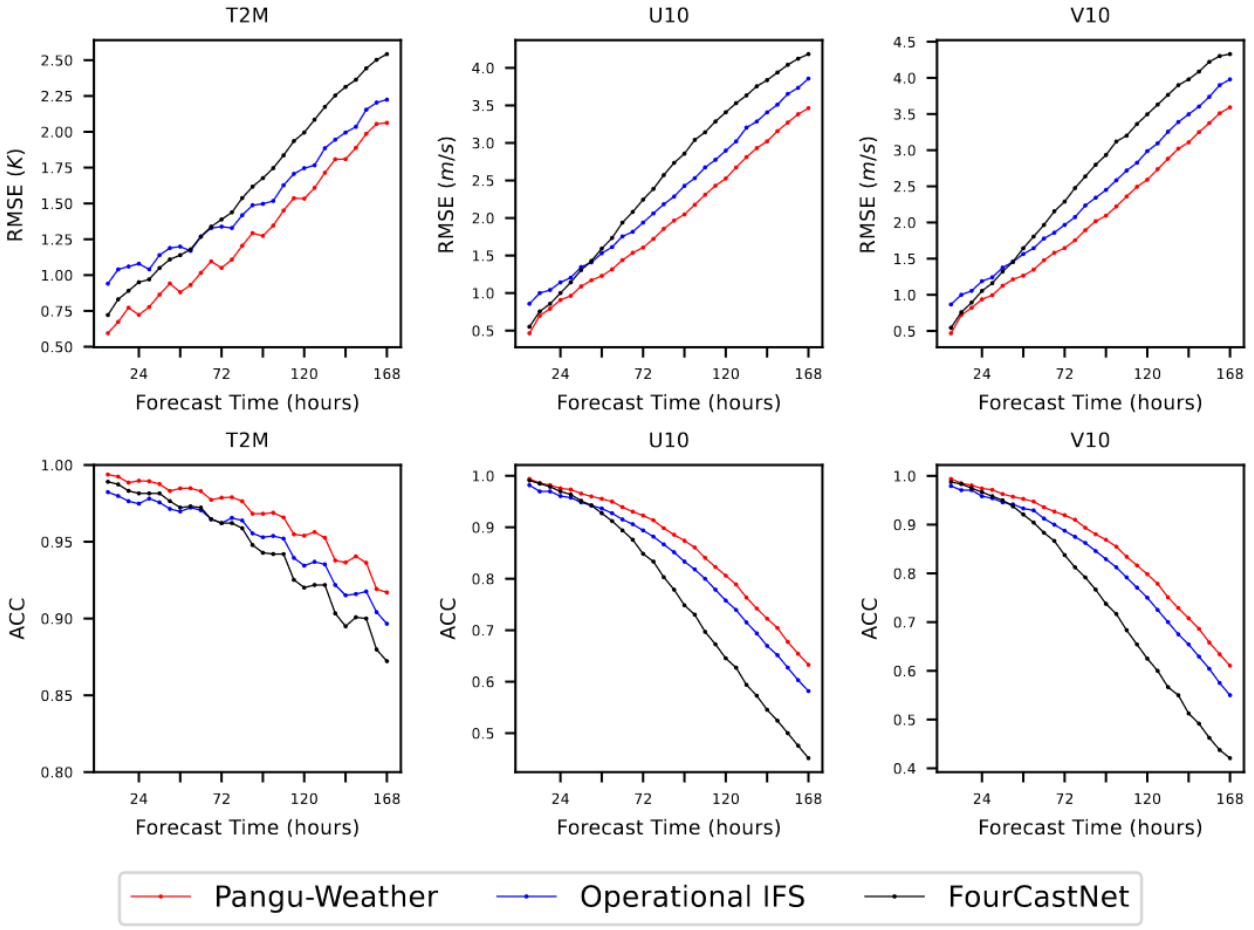

Figure 5.5: Comparison of RMSE and ACC plots for temperature, u and v wind components across various lead times for Pangu-Weather, IFS, and FourCastNet [1].

robust forecasting tool that can adapt to real-world data. This fine-tuning with real data can enable WindFormer to adapt to a wide range of tasks, such as wind prediction, forecasting wind extremes, predicting wind droughts, and other meteorological tasks related to different variables.

Chapter 6

Conclusions

This study introduces WindFormer, a transformer-based model with a decoder-only architecture for spatio-temporal regional wind speed forecasting. Trained on the ERA5 reanalysis dataset, WindFormer effectively provided efficient and accurate predictions, serving both as a robust pre-trained model for fine-tuning with observational wind data and offering efficient predictions over reanalysis data.

WindFormer showcases several key contributions that address the limitations of current forecasting models. The development of a decoder-only architecture specifically adapted for prediction tasks enhances computational efficiency compared with state-of-the-art encoder-decoder models. This architecture allows WindFormer to perform predictions with high precision and efficiency, validating its design through extensive experiments.

One of the standout features of WindFormer is its ability to operate at high resolutions (0.25^{o}) and across various pressure levels on an hourly basis, while handling a large number of meteorological variables, providing very precise short-term predictions. The model's versatility is further demonstrated by its adaptability to different regions and sets of variables, making it a flexible tool for various meteorological applications.

Validation of the model's performance through experiments shows that WindFormer achieves similar Root Mean Square Error (RMSE) and Anomaly Correlation Coefficient (ACC) values compared with state-of-the-art models. This validation underscores the model's efficiency and effectiveness, demonstrating that it can perform on par with more complex and computationally demanding architectures.

The study also highlighted several areas for future research, including architectural modifications, conducting more precise comparisons with state-of-the-art models, and employing it as a pre-trained model with observational data.

In conclusion, WindFormer not only achieves promising results in wind forecasting but also provides a scalable and adaptable framework for a wide range of predictive applications in meteorology and beyond. Its innovative architecture lays the foundation for future developments in the use of decoder-only models for global prediction, potentially serving as a pre-trained model for diverse meteorological tasks and applications in other fields.

Bibliography

[1] Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Pangu-Weather: A 3D High-Resolution Model for Fast and Accurate Global Weather Forecast, November 2022. arXiv:2211.02556 [physics].

[2] S. M. Tse, P. W. Chan, and W. K. Wong. A case study of missed approach of aircraft due to tailwind associated with thunderstorms. *Meteorological Applications*, 21(1):50–61, January 2014.

[3] Ramon Dalmau and Gilles Gawinowski. The effectiveness of supervised clustering for characterising flight diversions due to weather. *Expert Systems with Applications*, 237:121652, March 2024.

[4] P. W. Chan. A significant wind shear event leading to aircraft diversion at the Hong Kong international airport. *Meteorological Applications*, 19(1):10–16, March 2012.

[5] Jui-Sheng Chou, Yu-Chen Ou, and Kuan-Yu Lin. Collapse mechanism and risk management of wind turbine tower in strong wind. *Journal of Wind Engineering and Industrial Aerodynamics*, 193:103962, October 2019.

[6] Yongning Zhao, Lin Ye, Pierre Pinson, Yong Tang, and Peng Lu. Correlation-Constrained and Sparsity-Controlled Vector Autoregressive Model for Spatio-Temporal Wind Power Forecasting. *IEEE Transactions on Power Systems*, 33(5):5029–5040, September 2018.

[7] Saurabh S. Soman, Hamidreza Zareipour, Om Malik, and Paras Mandal. A review of wind power and wind speed forecasting methods with different time horizons. In *North American Power Symposium 2010*, pages 1–8, Arlington, TX, USA, September 2010. IEEE.

[8] Yao Zhang, Jianxue Wang, and Xifan Wang. Review on probabilistic forecasting of wind power generation. *Renewable and Sustainable Energy Reviews*, 32:255–270, April 2014.

[9] Chengqing Yu, Guangxi Yan, Chengming Yu, and Xiwei Mi. Attention mechanism is useful in spatio-temporal wind speed prediction: Evidence from China. *Applied Soft Computing*, 148:110864, November 2023.

[10] Sultan Al-Yahyai, Yassine Charabi, Abdullah Al-Badi, and Adel Gastli. Nested ensemble NWP approach for wind energy assessment. *Renewable Energy*, 37(1):150–160, 2012. ISBN: 0960-1481.

[11] Hui Liu, Hong-qi Tian, and Yan-fei Li. An EMD-recursive ARIMA method to predict

wind speed for railway strong wind warning system. *Journal of Wind Engineering and Industrial Aerodynamics*, 141:27–38, 2015.

[12] Lu Cao, Dong Qiao, and Xiaoqian Chen. Laplace l1 Huber based cubature Kalman filter for attitude estimation of small satellite. *Acta Astronautica*, 148:48–56, 2018.

[13] Ling-Ling Li, Xue Zhao, Ming-Lang Tseng, and Raymond R. Tan. Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm. *Journal of Cleaner Production*, 242:118447, 2020. ISBN: 0959-6526.

[14] Shubhi Harbola and Volker Coors. One dimensional convolutional neural network architectures for wind prediction. *Energy Conversion and Management*, 195:70–75, September 2019.

[15] Hui Liu, Xi-wei Mi, and Yan-fei Li. Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and Elman neural network. *Energy Conversion and Management*, 156:498–514, January 2018.

[16] Lionel P. Joseph, Ravinesh C. Deo, Ramendra Prasad, Sancho Salcedo-Sanz, Nawin Raj, and Jeffrey Soar. Near real-time wind speed forecast model with bidirectional LSTM networks. *Renewable Energy*, 204:39–58, March 2023.

[17] He Jiajun, Yu Chuanjin, Li Yongle, and Xiang Huoyue. Ultra-short term wind prediction with wavelet transform, deep belief network and ensemble learning. *Energy Conversion and Management*, 205:112418, February 2020.

[18] Lionel P. Joseph, Ravinesh C. Deo, David Casillas-Pérez, Ramendra Prasad, Nawin Raj, and Sancho Salcedo-Sanz. Short-term wind speed forecasting using an optimized three-phase convolutional neural network fused with bidirectional long short-term memory network model. *Applied Energy*, 359:122624, April 2024.

[19] Sujan Ghimire, Ravinesh C. Deo, David Casillas-Pérez, Sancho Salcedo-Sanz, Ekta Sharma, and Mumtaz Ali. Deep learning CNN-LSTM-MLP hybrid fusion model for feature optimizations and daily solar radiation prediction. *Measurement*, 202:111759, October 2022.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. *Advances in neural information processing systems*, 30, 2017.

[21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, June 2021. arXiv:2010.11929 [cs].

[22] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. ViViT: A Video Vision Transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6816–6826, Montreal, QC, Canada, October 2021. IEEE.

[23] Tung Nguyen, Johannes Brandstetter, Ashish Kapoor, Jayesh K Gupta, and Aditya Grover. ClimaX: A foundation model for weather and climate. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 25904–25938. PMLR, July 2023.

[24] Zihao Fu, Wai Lam, Qian Yu, Anthony Man-Cho So, Shengding Hu, Zhiyuan Liu, and Nigel Collier. Decoder-Only or Encoder-Decoder? Interpreting Language Model as a Regularized Encoder-Decoder. *arXiv*, 2024.

[25] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting, April 2024. arXiv:2310.10688 [cs].

[26] Peter Bauer, Alan Thorpe, and Gilbert Brunet. The quiet revolution of numerical weather prediction. *Nature*, 525(7567):47–55, September 2015. Publisher: Nature Publishing Group.

[27] Peter Lynch. The origins of computer weather prediction and climate modeling. *Journal of Computational Physics*, 227(7):3431–3444, 2008.

[28] Peter Bauer, Tiago Quintino, Nils Wedi, Antonio Bonanni, Marcin Chrust, Willem Deconinck, Michail Diamantakis, Peter Düben, Stephen English, Johannes Flemming, and others. *The ECMWF scalability programme: Progress and plans*. European Centre for Medium Range Weather Forecasts, 2020.

[29] Xin Man, Chenghong Zhang, Jin Feng, Changyu Li, and Jie Shao. W-MAE: Pre-trained weather model with masked autoencoder for multi-variable weather forecasting, December 2023. arXiv:2304.08754 [physics].

[30] Asher Trockman and J. Zico Kolter. Patches Are All You Need? *Transactions on Machine Learning Research*, 2023.

[31] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, Pedram Hassanzadeh, Karthik Kashinath, and Animashree Anandkumar. FourCastNet: A Global Data-driven High-resolution Weather Model using Adaptive Fourier Neural Operators, February 2022. arXiv:2202.11214 [physics].

[32] Xingbo Fu, Feng Gao, Jiang Wu, Xinyu Wei, and Fangwei Duan. Spatiotemporal Attention Networks for Wind Power Forecasting. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 149–154, Beijing, China, November 2019. IEEE.

[33] Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore, December 2023. Association for Computational Linguistics.

[34] Copernicus Climate Change Service. ERA5 hourly data on pressure levels from 1940 to present, 2018.

[35] C3S. ERA5 hourly data on single levels from 1940 to present, 2018.

[36] Helena Liz-López, Javier Huertas-Tato, Jorge Pérez-Aracil, Carlos Casanova-Mateo, Julia Sanz-Justo, and David Camacho. Spain on fire: A novel wildfire risk assessment model based on image satellite processing and atmospheric information. *Knowledge-Based Systems*, 283:111198, January 2024.

[37] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. arXiv:1810.04805 [cs].

[38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[39] Nils Wedi, Peter Bauer, Willem Deconinck, Michail Diamantakis, M. Hamrud, Christian Kuehnlein, S. Malardel, Kristian Mogensen, G. Mozdzynski, and Piotr Smolarkiewicz. The modelling infrastructure of the Integrated Forecasting System: Recent advances and future challenges. *European Centre for Medium-Range Weather Forecasts*, 2015. Publisher: [object Object].

[40] John Taylor and Ming Feng. A deep learning model for forecasting global monthly mean sea surface temperature anomalies. *Frontiers in Climate*, 4, 2022.

[41] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *https://arxiv.org/abs/1412.6980*, 2014.

[42] Lukas Biewald. Experiment Tracking with Weights and Biases, 2020.

[43] R. G. Owens and T. D. Hewson. *ECMWF Forecast User Guide*. ECMWF, Reading, 2018.

[44] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the CoordConv solution. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.