Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Of Computer Information Systems and Software Engineering
(Full name of faculty)
Computer science Department
(Full name of department)

QUALIFYING PAPER

For the degree of

master
(Degree name)

topic: Machine learning–driven weather forecasting integrating advanced data analysis methods

Submitted by: sixth year student, group ICTm-62
Specialty: 126 Information Systems and Technologies
(Code and name of specialty)

Valerie Kasongo Bwanga
(signature) (Surname and initials)

Supervisor Holotenko O.S.
(signature) (Surname and initials)

Standards verified by Nykytiuk V.V.
(signature) (Surname and initials)

Head of Department Bodnarchuk I.O.
(signature) (Surname and initials)

Reviewer Yasnii O.P.
(signature) (Surname and initials)

Ternopil
2026

Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Of Computer Information Systems and Software Engineering
(Full name of faculty)
Department Computer science Department
(Full name of department)

APPROVED BY
Head of Department
Bodnarchuk I.O.
(signature) (Surname and initials)
« » 2025

ASSIGNMENT

for QUALIFYING PAPER

for the degree of *master*
(Degree name)
specialty *126 Information Systems and Technologies*
(Code and name of the specialty)
student Valerie Kasongo Bwanga

1. Paper topic: Machine learning–driven weather forecasting integrating advanced data analysis methods

Paper supervisor: Holotenko Oleksandr Serhiyovych,, PhD, Assoc. Prof.
(Surname, name, patronymic, scientific degree, academic rank)

Approved by university order as of «___» ________ 2025 №_______.

2. Student's paper submission deadline: _________________

3. Initial data for the paper *information protection in local area networks, methods and tools for implementing the weather forecasting task, implementation of the forecasting system, methodology for user interaction*

4. Paper contents (list of issues to be developed)/
Analysis the methods of information protection in local area networks. Methods and tools for implementing the weather forecasting task based on machine learning and data analysis. Description and implementation of the forecasting system. Methodology for user interaction with the system. Development. of the startup project. Occupational safety and health

5. List of graphic material (with exact number of required drawings, slides).

6. Advisors of paper chapters.

Chapter	Advisor's surname, initials and position	Signature, date	
		assignment was given by	assignment was received by
Occupational Safety			
Safety in Emergency Situations			

7. Date of receiving the assignment.

TIME SCHEDULE

LN	Paper stages	Paper stages deadlines	Notes
1	Selection and approval of the Master’s Thesis topic		
2	Internship		
3	Analysis of existing solutions and approaches to the		
4	Selection of technologies for the implementation of artificial		
5	Development of the Software Product Structure		
6	Implementation, Testing, and Debugging of the Program		
7	Protection of the Software product		
8	Preparing of the Master’s Thesis Manuscript		
9	Pre-Defense of the Master’s Thesis		
10	Normative Review and Revision of the Master’s Thesis		
11	Submission of the Master’s Thesis		
12	Defense of the Master’s Thesis		

Student ____________________ Valerie Kasongo Bwanga

(signature) (surname and initials)

Paper supervisor ____________________ Holotenko O.S.

(signature) (surname and initials)

ABSTRACT

Machine learning–driven weather forecasting integrating advanced data analysis methods // The educational level "Master" qualification work // Valerie Kasongo Bwanga // Ternopil Ivan Pulyuy National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science, ISTm-62 group // Ternopil, 2026 // P. 97, fig. – 18, tables – 25, annexes – 1, ref. – 31.

First chapter analyzes the importance of weather forecasting, reviews existing solutions and methods, and formulates the main problems addressed by machine learning-based approaches.

Second chapter describes the machine learning models, statistical methods, and software tools used to develop an effective weather forecasting system.

Third chapter presents the system architecture, data integration process, model training, deployment, and security considerations of the developed system.

Fourth chapter explains how users interact with the system, including data input, model selection, parameter configuration, visualization, and results.

Fifth chapter outlines the startup concept, target audience, market opportunities, and strategy for commercializing the weather forecasting system.

Sixth chapter analyzes workplace conditions and identifies potential occupational risks to ensure safe and healthy working environments.

Conclusions summarizes the research results, confirms the effectiveness of machine learning methods for weather forecasting, and outlines directions for future development.

Object of the research – the process of weather forecasting based on historical and real-time meteorological data.

Subject of the research – machine learning methods and data analysis models (ARIMA, SARIMA, SARIMAX) used for modeling, forecasting, and improving the accuracy of weather predictions.

Keywords: weather forecasting, machine learning, ARIMA, SARIMA, SARIMAX, time series analysis, interactive interface, visualization, accuracy.

CONTENTS

INTRODUCTION

In the modern era, often referred to as the information age, people are surrounded by an unprecedented amount of data. The ability to process and effectively utilize this data is crucial for the functioning of industry, government, and society. Among various applications of data analytics, weather forecasting is particularly significant as it directly impacts sectors such as agriculture, transportation, disaster prevention, and urban planning. Weather is often unpredictable and serves as a key factor influencing daily human activities and long-term strategic decisions. Accurate weather forecasts have become essential tools for mitigating the effects of natural disasters, optimizing resource allocation, and improving operational efficiency.

However, traditional forecasting methods, primarily based on atmospheric physics and mathematical models, often face significant limitations. These models require high computational power, struggle to process large volumes of data, and are frequently unable to capture nonlinear relationships between weather variables. The emergence of machine learning and artificial intelligence has revolutionized the way complex problems are approached. Leveraging large datasets and advanced algorithms, these technologies can decipher hidden patterns and correlations that traditional methods overlook. In particular, time series forecasting models such as ARIMA (Autoregressive Integrated Moving Average), SARIMA (Seasonal Autoregressive Integrated Moving Average), and SARIMAX (Seasonal Autoregressive Integrated Moving Average with Exogenous Variables) have demonstrated excellent capabilities in predicting weather phenomena. These models incorporate seasonal trends and exogenous variables, providing greater accuracy, making them ideal for weather forecasting applications.

This work aims to address the challenges of existing forecasting systems and develop innovative solutions using machine learning technologies. The proposed system is designed to overcome the limitations of traditional approaches by automating the data preprocessing, model selection, and parameter optimization processes, providing a scalable and user-friendly platform for weather forecasting.

The importance of weather forecasting extends beyond the scientific realm, affecting various aspects of contemporary life. Agriculture relies on weather forecasts to optimize planting schedules, irrigation, and pest control. Transportation systems use forecasts to ensure safety and efficiency, while energy companies depend on accurate predictions to balance supply and demand. In disaster management, timely weather updates can save lives and reduce economic losses. Despite its significance, current forecasting systems often fail to meet the growing demands for accuracy, speed, and accessibility. This research addresses these shortcomings by integrating advanced machine learning methods and delivering more precise and reliable weather forecasts. This approach not only enhances the forecasting process but also ensures broader accessibility through a convenient interface.

The primary goal of this research is to develop and implement a computational system for weather forecasting using advanced machine learning algorithms. To achieve this, it is necessary to conduct a comprehensive analysis of current weather forecasting systems and their limitations, apply ARIMA, SARIMA, and SARIMAX models for time series data analysis, implement an interactive platform for visualizing weather forecasts and evaluating model performance, test the system using real datasets and compare its performance with traditional methods, and explore the practical application of the system across various industries. This research focuses on the methodology and system for weather forecasting, emphasizing the application of ARIMA, SARIMA, and SARIMAX models to meteorological time series data. To achieve these goals, the research employs data preprocessing methods for cleaning and normalizing datasets, the application of the ARIMA family of models, statistical evaluation metrics such as RMSE and MAE, and a Python-based system for interactive visualization.

The practical significance of the developed system is evident in its applications in agriculture, disaster management, and urban planning. Advanced forecasting can optimize resource distribution, improve emergency response, and support infrastructure development. The research also aims to bridge the gap between accessibility for a broad audience and advanced machine learning technology through user-friendly design.

The work consists of six comprehensive chapters, including an overview of subject areas, mathematical methods, system applications, user interaction, and potential applications. The work consists of an introduction, five chapters, and conclusions. The total volume of the dissertation is 100 pages, including 25 tables, 18 figures, and 3 pages of references containing 31 items.

This structured and detailed approach ensures that the proposed system is not only a theoretical contribution but also a practical tool for enhancing weather forecasting methodologies.

Approbation of the results. The main provisions of this work were discussed at: III International Scientific and Practical Conference "Technology development: shaping modern thinking and scientific approaches", Krakow, Poland, January 19–21, 2026.

Publications. The main results of the qualification work were published at the conference (See Appendix A).

1 OVERIEW OF THE SUBJECT AREA AND PROBLEM STATEMENT

Weather forecasting has become an essential component of modern life, influencing many sectors such as agriculture, energy management, transportation, urban planning, and disaster prevention. Its significance continues to grow in the context of climate change and increasing reliance on renewable energy sources. However, traditional forecasting methods, primarily based on Numerical Weather Prediction (NWP) models, face several limitations. These include high computational demands, difficulties in processing large datasets, and limited adaptability to rapidly changing climatic conditions.

Weather forecasting is one of the most critical tasks in modern science, affecting numerous areas of human activity such as agriculture, transportation, energy, and disaster management. As climate change continues to intensify, the frequency and severity of extreme weather events are increasing, inevitably creating a demand for accurate real-time weather forecasts. Traditional weather forecasting methods, based primarily on physical and mathematical models, often encounter limitations due to computational complexity, inefficiency in handling large data arrays, and difficulties in adapting to sudden weather changes. This section explores the need for innovative forecasting methods by combining machine learning techniques with data analysis. This section will utilize advanced computational methods to describe the objectives and tasks of weather forecasting, followed by a detailed analysis of existing solutions and technologies. This comprehensive overview provides the foundation for selecting and justifying the methods used in this research and highlights the potential to overcome existing limitations and enhance forecast accuracy and accessibility.

1.1 Objectives of Weather Forecasting Using Machine Learning and Data Analysis

Weather forecasting is an extremely important task, spanning scientific, economic, and social spheres. Advanced forecasting systems utilizing machine learning are designed to transform the ways forecasts are produced and utilized. Below, the broader context of weather forecasting, its applications, and the role of machine learning in addressing complex problems are considered.

Accurate weather forecasts are essential for several reasons:

- Disaster Preparedness: Reliable forecasting enables timely evacuation and disaster mitigation strategies, reducing human and economic losses during extreme weather events such as hurricanes, floods, and heatwaves.
- Agricultural Planning: Farmers rely on forecasts to determine optimal planting and harvesting times, as well as irrigation and harvesting schedules, which can help increase yields and reduce resource waste.
- Transportation Safety: Forecasts provide crucial information about hazardous conditions, ensuring the safe operation of aviation, maritime, and road transport.
- Energy Management: Renewable energy sources such as solar and wind are highly dependent on weather conditions. Accurate forecasts help balance energy supply and demand.

Potential Application Areas for machine learning-based weather forecasting systems are easily adaptable and can be utilized in various applications:

- General Weather Services: Improving daily and long-term weather forecasts for public use.
- Agriculture: Supporting precision farming technologies by predicting microclimatic conditions.
- Urban Planning: Assisting in designing climate-resilient infrastructure and smart cities.
- Renewable Energy Optimization: Forecasting solar radiation and wind speed to enhance energy production.

The machine learning-based forecasting system is designed for:

- Government Agencies: For timely weather alerts and disaster response management.
- Farmers and Agronomists: To optimize agricultural productivity and sustainability.
- Energy Companies: To integrate forecasts into energy production models.
- Researchers: To develop advanced models and conduct climate impact studies.

Machine learning models offer several advantages over traditional approaches:

- High Accuracy: These models improve forecast accuracy by detecting patterns in large datasets.
- Automation: Automatic parameter tuning and model selections reduce manual intervention.
- Scalability: These systems can process and analyze data across various geographical scales, from local to global.
- Real-time Updates: API integration ensures continuous updating and dynamic forecasting.

1.2 Analysis of Existing Solutions

The evolution of weather forecasting is characterized by the development of various methods and systems, each tailored to address specific challenges. Traditional approaches largely rely on Numerical Weather Prediction (NWP) models, which use hydrodynamic equations to simulate atmospheric processes. Despite their effectiveness, these models require high computational power and lack high accuracy for short-term forecasts. In recent years, the integration of machine learning technologies has revolutionized weather forecasting. Machine learning provides the ability to analyze complex interdependencies in data, adapt to changing conditions, and deliver accurate forecasts with lower computational costs [13]. This subsection provides a detailed overview of existing solutions for weather forecasting, their methodologies, and a comparison of their advantages and limitations.

1.2.1 Existing Ready-Made Solutions

Numerous platforms and tools are used for weather forecasting. Each has its own history, target audience, and specific features. Below is a detailed overview of key solutions currently relevant worldwide.

Google Weather API

The Google Weather API was created as part of Google's global strategy to provide access to real-time data. Initial API versions were launched in 2010 and have been gradually improved.

Principle of Operation: Google uses its own data collection network combined with forecasts from other sources, analyzed using machine learning algorithms.

Relevance: The API is one of the most popular solutions due to its ease of integration and reliability.

Advantages:

1. Ease of use.
2. High accuracy of real-time data.
3. Integration with other Google services.

Disadvantages:

1. Limited functionality in the free version.
2. Lack of deep customization for complex models.

OpenWeatherMap

Founded in 2014 as an open platform for developers, its popularity grew due to free access to data in the basic version.

Principle of Operation: Combines data from ground weather stations, satellites, and mathematical models.

Advantages:

1. Ability to obtain historical data.
2. Flexible API for integration.

Disadvantages:

1. Requires additional data processing for specific tasks.
2. Dependence on paid access for advanced features.

MeteoBlue

Founded in 2006, it utilizes high-resolution mathematical models.

Principle of Operation: Based on high-resolution mathematical models.

Advantages:

1. Accuracy of forecasts.
2. GIS integration.

Disadvantages:

1. High subscription cost.

Relevance: Medium, especially for researchers.

Table 1.1 provides a comparison of existing platforms.

Table 1.1 – Comparison of Existing Platforms

Platform	Year Founded	Data Source	Main Advantages	Platform	Year Founded
Google Weather API	2010	Ground stations, satellites	Simplicity, integration with Google services	Limited customization	High
OpenWeather Map	2014	Weather stations, satellites	Free access, historical data	Need for additional processing	High
MeteoBlue	2006	High-resolution mathematical models	Forecast accuracy, GIS integration	High subscription cost	Medium for researchers

1.2.2 Methods for Solving the Weather Forecasting Problem

Numerous methods exist for data analysis and weather forecasting, each with its own advantages and disadvantages. Among them, ARIMA, SARIMA, SARIMAX and their analogues stand out, as well as neural networks and hybrid models.

ARIMA (Autoregressive Integrated Moving Average)

History: The ARIMA method was developed in the 1970s by George Box and David Jenkins for time series analysis.

Principle of Operation: ARIMA is based on identifying autocorrelation and moving averages over time.

Advantages:

1. Simplicity of implementation.
2. Suitable for stationary time series.

Disadvantages:

1. Does not account for seasonality.
2. Limited accuracy for long-term forecasts.

SARIMA (Seasonal ARIMA)

Development: An enhancement of ARIMA to account for seasonal variations.

Principle of Operation: Adds parameters for modeling seasonality in time series.

Advantages:

1. High accuracy for seasonal series.
2. Flexibility in parameters.

Disadvantages:

1. Complexity in parameter tuning.
2. High dependence on data quality.

SARIMAX (Seasonal ARIMA with Exogenous Variables)

Features: Includes exogenous variables to improve forecasts.

Application: Ideally suited for complex tasks where external factors such as humidity or atmospheric pressure are considered.

Table 1.2 provides a practical comparison of all ARIMA-family models based on their advantages, disadvantages, features, and usage scenarios.

Table 1.2 – Comparison of ARIMA Methods

Method	Features	Advantages	Disadvantages	Usage Scenarios
ARIMA	Basic time series analysis	Simplicity, speed	Does not account for seasonality	Short-term forecasts
SARIMA	Accounts for seasonality	Accuracy for seasonal data	Complexity in tuning	Forecasts with seasonal trends
SARIMAX	Adds exogenous variables	Increased accuracy	High computational complexity	Integrated forecasts with additional data

Analogues of the ARIMA Family

Neural Networks (LSTM, GRU): Utilize neural architectures for processing time series. Advantages: Can learn from large datasets, capturing complex dependencies. Disadvantages: High computational resource requirements. Exponential Smoothing (ETS): Exponential smoothing method that works well for simple series. Disadvantages: Limited flexibility.

Hybrid Models (ARIMA + Neural Networks): Combines ARIMA's predictability with the adaptability of neural networks. Advantages: Combines ARIMA predictability with neural network adaptability. Disadvantages: Complexity in tuning.

Why Preference is Given to ARIMA, SARIMA, SARIMAX? Accuracy: High effectiveness for stationary and seasonal series. Flexibility: Ability to account for exogenous variables (SARIMAX). Ease of Implementation: Lower computational resource requirements compared to neural networks.

Table 1.3 shows the main reasons for preferring the ARIMA family.

Table 1.3 – Comparison of Analogues

Method	Main Characteristics	Advantages	Disadvantages	Application
LSTM (Long Short-Term Memory)	Neural network architecture for time series, with memory for long-term dependencies	Learning complex dependencies, processing large datasets	Requires significant computational resources, long training time	Forecasting complex time series, including climatic changes
GRU (Gated Recurrent Unit)	Simplified version of LSTM with fewer parameters	Requires fewer training resources, faster than LSTM	Less accurate than LSTM for complex time dependencies	Short-term forecasting of weather conditions
ETS (Exponential Smoothing)	Uses exponential smoothing for forecasts	Simplicity of implementatio n, effectiveness for simple series	Not suitable for complex or multivariate data	Simple forecasting without pronounced seasonality
Hybrid Models (ARIMA + LSTM)	Combines ARIMA accuracy with LSTM's ability to capture nonlinearities	High accuracy, ability to account for complex dependencies	Complex architecture, long development time	Forecasts with nonlinear and seasonal dependencies
Random Forest Regression	Uses ensemble method for forecasts	Robustness to overfitting, ability to work with many predictors	Not always effective for time series	Random influences without long-term dependencies
SVM (Support Vector Machines)	Classification and regression models for time series	Good generalization ability	Requires parameter tuning, does not account for forecasting	Solving specific tasks

1.3 Conclusions for Chapter 1

This chapter presented a comprehensive analysis of the subject area of weather forecasting using machine learning methods and data analysis, allowing for a detailed study of problems, existing solutions, and current methods. These results form the foundation for further development and application of modern forecasting systems. Specifically, weather forecasting has been identified as a crucial tool impacting many sectors such as agriculture, transportation, energy, and emergency management. It was found that modern forecasting systems, such as Google Weather API, OpenWeatherMap, and IBM Weather Company Data API, provide broad capabilities for obtaining data and forecasts. However, there are also disadvantages related to limited customization, the need for data processing, and the high cost of accessing advanced features. The analysis of existing methods, such as ARIMA, SARIMA, SARIMAX, neural networks, and hybrid models, shows their advantages and limitations. The ARIMA family method provides convenience and accuracy when applied to stationary and seasonal time series, but it is less flexible and can explain complex nonlinear patterns compared to modern neural networks like LSTM and GRU. The main results of this chapter include:

- Weather Forecasting Tasks: Key application areas for forecasting were identified, such as agriculture, transportation, disaster management, and energy, confirming the high relevance of creating modern forecasting systems.
- Review of Existing Solutions: Existing platforms have limitations that reduce their effectiveness in private applications but provide comprehensive access to weather data.
- Comparison of Forecasting Methods: Methods including ARIMA, SARIMA, SARIMAX, neural networks, etc., were considered. Neural networks are promising for complex scenarios, but the ARIMA family is in high demand due to its effectiveness and ease of implementation.
- Need for Innovation: The analysis results indicate the necessity of combining the advantages of traditional methods with modern machine learning algorithms to create an integrated system ensuring accuracy, scalability, and ease of

use.

As a result, this overview serves as the basis for the mathematical justification of the system and software application. The analysis results demonstrate that using machine learning methods in weather forecasting significantly enhances forecast quality and speed, making it accessible to a wide range of users.

2 METHODS AND TOOLS FOR IMPLEMENTING THE WEATHER FORECASTING TASK BASED ON MACHINE LEARNING AND DATA ANALYSIS

This chapter discusses the methods and tools used for developing a weather forecasting system based on machine learning and data analysis. Forecasting weather conditions is an extremely complex task that requires processing large volumes of heterogeneous data and accounting for numerous factors such as seasonality, time dependencies, and geographical specificities. Therefore, the application of modern machine learning methods is one of the most effective ways to build accurate forecasting models. Modern machine learning technologies allow not only for the analysis of large data volumes but also for uncovering hidden dependencies that traditional methods might miss. This chapter describes the main methods used for forecasting, the software tools applied for system implementation, as well as the key stages of developing this system.

Considering the complexity of weather systems, it is important to understand how machine learning methods can provide better accuracy by accounting for all possible influencing factors. The accuracy of weather forecasting is critical for many sectors such as agriculture, energy, transportation, and natural resource management. The use of machine learning enables the creation of more reliable forecasts, which aids in making informed decisions in these sectors.

2.1 Analysis of Weather Forecasting Methods

Weather forecasting is a complex task requiring the analysis of a large quantity of heterogeneous data, including historical meteorological data, current indicators, and geographical features of a region. To solve this task, machine learning methods were applied, specifically ARIMA, SARIMA, SARIMAX, and linear regression, which work effectively with time series.

Weather forecasting has many important aspects such as accuracy, computation speed, and scalability. Traditional approaches often face difficulties due to the complexity of modeling nonlinear relationships between variables, leading to inaccurate forecasts. In this context, modern machine learning methods demonstrate significant advantages, allowing for more accurate forecasts by considering various data features such as seasonality and dependencies between different meteorological variables. The use of models like ARIMA, SARIMA, and SARIMAX enables effective work with data that have temporal dependencies and seasonal fluctuations.

2.1.1 ARIMA Method

AutoRegressive Integrated Moving Average is used for analyzing and forecasting time series. It allows for modeling and predicting weather behavior based on previous observations. The ARIMA method consists of three components: autoregression (AR), integration (I), and moving averages (MA). This method is well-suited for data without pronounced seasonality. ARIMA can be used for predicting variables such as temperature or atmospheric pressure based on historical data. ARIMA provides reliable forecasts in cases where weather data are stationary or have a certain trend. However, for more complex weather patterns, ARIMA may be insufficient, requiring additional methods.

2.1.2 SARIMA Method

Seasonal ARIMA is an extension of ARIMA for cases where data exhibit seasonality. Seasonal components allow for accurately forecasting weather behavior depending on the time of year or other regular factors. SARIMA enables accounting for cyclical changes observed in weather data, significantly increasing forecast accuracy in cases with pronounced seasonal dependence, such as summer and winter temperatures. Using SARIMA is effective when there is a need to account for periodic changes and trends that regularly repeat. This makes SARIMA one of the best options for forecasting seasonal changes, such as annual temperature or precipitation levels.

Chart 2.1.1. Seasonal components in the SARIMA model for forecasting weather conditions over a year, showing how the model accounts for seasonal fluctuations for more accurate forecasts.

2.1.3 SARIMAX Method

Seasonal ARIMA with eXogenous factors is also an extension of SARIMA, allowing for the inclusion of exogenous variables or additional factors that may influence the forecast. Using SARIMAX allows for more accurate forecasts by adding the ability to model the influence of such external parameters as climate changes or other socioeconomic factors. SARIMAX enables modeling of more complex weather processes by accounting for the influence of exogenous factors such as humidity, pressure, solar radiation level, and other important indicators. This significantly improves forecast accuracy, especially for regions where weather conditions heavily depend on several factors.

2.1.4 Linear Regression

One of the fundamental machine learning methods, it allows for modeling the relationship between meteorological variables such as temperature, humidity, atmospheric pressure, etc. Linear regression is simple and fast to use, making it particularly useful for basic forecasting tasks when quick results with acceptable accuracy are needed. However, its accuracy may be insufficient for more complex tasks where nonlinear relationships between variables need to be considered. Linear regression is often used as a baseline approach for comparison with more complex models to assess improvements achieved through more sophisticated approaches.

In **Figure 2.1.** a comparison of accuracy metrics (RMSE, MAE) for different forecasting methods:

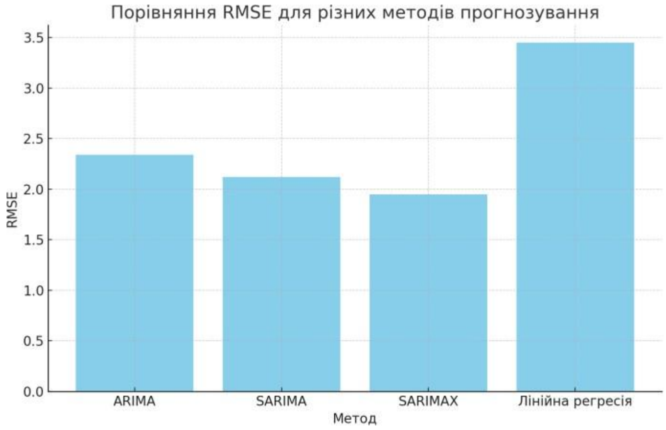


Figure 2.1 – Graph comparing accuracy metrics (RMSE, MAE)

These data demonstrate that the SARIMAX method has the lowest RMSE and MAE values, indicating its high accuracy compared to other methods.

2.1.5 Method for Automatic Parameter Tuning

Using the pmdarima library allows for quickly finding optimal parameters for ARIMA, SARIMA, and SARIMAX models. This significantly reduces model preparation time and improves their effectiveness. Automatic parameter tuning is an important stage as it reduces the likelihood of errors during modeling, particularly selecting incorrect parameters that can affect forecast quality. Using automated tools for parameter tuning significantly simplifies analysts' work and saves resources while ensuring high forecast quality.

2.1.6 Additional Statistical Tests

The Dickey-Fuller test (adfuller), used for checking data stationarity, is a key stage when working with ARIMA, SARIMA, and SARIMAX. Stationarity means that the main statistical characteristics of a series, such as mean and variance, remain constant over time. Non-stationary data can lead to inaccurate forecasts, so it is important before starting modeling to check if the data are stationary and, if necessary, apply transformations to achieve stationarity. Using statistical tests helps ensure data quality and increase forecast reliability.

2.2 Software Tools for Implementing the Forecasting System

In modern development of weather forecasting systems, the application of appropriate software tools is critically important for ensuring efficiency, reliability, and forecast accuracy. This section discusses the tools used for creating such a system and their contribution to the overall solution architecture. Software tools allow for working with large data volumes, creating and training machine learning models, as well as providing a convenient and intuitive interface for user interaction. By integrating these tools into the system, it becomes possible to fully automate the processes of collecting, analyzing, and forecasting weather conditions.

2.2.1 Python as the Primary Programming Language

Python is a high-level, interpreted programming language with a wide selection of libraries and tools for data processing, visualization, and developing machine learning models. Python is an ideal choice for developing a forecasting system due to its flexibility, ease of use, and broad support in the developer community. An important advantage of Python [15] is its ability to work with large data volumes and easily integrate various libraries and frameworks. Additionally, Python supports numerous machine learning libraries such as TensorFlow, scikit-learn, Keras, making it an indispensable tool for implementing forecasting models.

2.2.2 Streamlit

Streamlit is a library for creating interactive web applications based on Python. Streamlit allows for quickly creating interfaces to demonstrate the weather forecasting model. Using Streamlit, developers can create interactive web pages where users can interact with the model: upload data, set parameters for forecasting, and view results in real-time. The main advantage of Streamlit is its ease of use, as web programming knowledge is not required; creating a web interface is sufficient by writing Python code. This makes prototype development fast and efficient, and Streamlit integration makes the system accessible to end-users without requiring deep technical knowledge.

2.2.3 Pandas and NumPy

Pandas and NumPy are key tools for data processing and analysis. Pandas is a library for manipulating data in tabular form (DataFrame), allowing for easy loading, cleaning, and analyzing of data. Pandas provides a rich set of functions for data work such as merging tables, data grouping, filtering, and aggregation. It enables convenient data preparation for further modeling and ensures integration with other Python libraries. NumPy is used for working with multidimensional arrays and performing numerical computations. NumPy allows for efficient mathematical operations on large datasets, which is an important stage in data preparation before applying machine learning models.

Pandas and NumPy constitute the foundational computational and data manipulation stack for modern data science in Python, and they are indispensable in the pipeline of the developed weather forecasting system. Their complementary roles ensure efficient, accurate, and scalable data handling from raw input to model-ready arrays.

Pandas serves as the primary library for structured, tabular, and time-series data operations. Its core data structure, the DataFrame, provides an intuitive, spreadsheet-like interface with labeled axes, enabling complex data manipulations with concise syntax. Within the project's context, Pandas is utilized for:

- Data Ingestion & Inspection: Seamlessly loading historical weather data from diverse sources (e.g., CSV files, APIs) into DataFrames.
- Data Cleaning & Preprocessing: Handling missing values via interpolation or deletion, filtering outliers’ specific to meteorological parameters, and correcting data types.
- Time-Series Specific Operations: Leveraging its robust time-series functionality for

tasks such as resampling (e.g., converting hourly data to daily averages), calculating rolling statistics (moving averages for temperature trends), and shifting data for lag feature creation, a critical step for autoregressive models like ARIMA.

- Data Transformation & Feature Engineering: Merging datasets from different sources, grouping data by temporal periods (season, year), and performing aggregations to create summary features that enhance model performance.

NumPy, the fundamental package for scientific computing in Python, provides the underlying architecture for numerical operations. It introduces the powerful N-dimensional array object, which enables efficient storage and manipulation of large, homogeneous numerical datasets. Its role in this system is critical for:

- Numerical Backend for Pandas: Pandas DataFrames and Series are built atop NumPy arrays, meaning all high-level Pandas operations ultimately compile to optimized NumPy computations.
- Mathematical & Statistical Foundations: Performing vectorized operations, which apply functions to entire arrays without explicit loops, leading to execution speeds comparable to C/C++. This is essential for computing loss functions, gradients, and statistical metrics during model evaluation.
- Linear Algebra & Multi-dimensional Data Handling: Providing the essential routines for matrix decompositions, linear algebra calculations, and handling multi-dimensional data structures required by advanced modeling techniques.
- Memory Efficiency: Enabling the handling of large-scale meteorological datasets (e.g., multi-year, multi-station records) in memory with minimal overhead.

In synthesis, the workflow within the forecasting system sees Pandas as the data manager and wrangler, responsible for organizing, cleaning, and structuring the raw, heterogeneous weather data into a coherent timeline. This refined data is then leveraged as a NumPy array, where NumPy acts as the high-performance computational engine, executing the complex mathematical transformations and preparations required for statistical and machine learning modeling. Their integrated use ensures a seamless, efficient, and robust data preparation pipeline, forming the reliable data foundation upon which accurate forecasting models are built and evaluated.

2.2.4 Matplotlib and Seaborn

Matplotlib and Seaborn are used for data visualization. Matplotlib is a basic library for creating graphs and charts, providing a wide range of capabilities for visualization, including line graphs, histograms, pie charts, etc. An important part of system development is creating visual tools for presenting results, as this helps users understand the nature of the data and draw conclusions about forecast quality. Seaborn is built on top of Matplotlib, allowing for creating more attractive and informative graphs, including heat maps, pair plots, and other complex visualizations. Visualization helps analyze dependencies between variables and evaluate forecast results, as well as illustrate important trends and data features.

2.2.5 Plotly

Plotly is a library for creating interactive graphs. Using Plotly allows users to interact more deeply with data, scale charts, explore individual data points, and analyze them. Thanks to interactive visualization, the behavior of forecasted parameters can be understood in more detail and more accurate conclusions can be made. For example, using Plotly, dynamic temperature forecast graphs can be created where users can interact with the graph and get additional information about specific days or periods. This makes the forecasting system not only an analysis tool but also a powerful means for presenting data to users.

2.2.6 Scikit-learn

Scikit-learn is a machine learning library that provides a wide set of algorithms for classification, regression, clustering, as well as tools for evaluating model quality. Scikit-learn is used for implementing machine learning algorithms and for tuning models such as ARIMA, SARIMA, SARIMAX. The library also provides tools for data processing before model training, such as standardization or normalization of data, which allows for improving model accuracy and making their work more stable.

2.2.7 Statsmodels

Statsmodels is another important library used for statistical modeling and data analysis. Statsmodels allows for implementing ARIMA, SARIMA, and SARIMAX methods, providing extensive functionality for building, evaluating, and visualizing time series. Thanks to Statsmodels, developers can easily conduct statistical analysis, evaluate model parameters, and check their significance. This is important for ensuring forecast reliability and understanding which variables most influence modeling results.

2.2.8 Pmdarima

Pmdarima is a library used for automating the process of selecting parameters for ARIMA, SARIMA, and SARIMAX models. Automatic parameter selection is a critically important stage in developing time series models, as incorrectly chosen parameters can lead to reduced forecast accuracy. Pmdarima significantly simplifies this process by performing an automatic search for optimal parameters such as autoregression order, integration order, and moving averages order. This makes the model development process more efficient and less dependent on manual tuning.

Thus, the selected software tools provide a comprehensive approach to implementing the weather forecasting system. They allow for easy processing and analyzing of data, building models, creating interactive interfaces, and visualizing results, making the forecasting system effective and user-friendly. Each of these tools has its specific role and ensures the execution of important tasks, which together ensure a high level of reliability and forecast accuracy.

2.3 Main Stages of System Development

The process of developing a weather forecasting system consists of several key stages, each playing an important role in ensuring the quality and accuracy of results. Each of these stages has its specific tasks and challenges that must be solved to create a reliable system. For example, the data collection stage requires careful selection and filtering of information to avoid using incorrect or outdated data that could affect the final result. Next, model analysis and selection are critically important for determining the most effective forecasting approaches, considering the specifics of weather conditions and the type of data used. Each model has its advantages and disadvantages, so this stage requires an experimental approach for choosing the optimal solution.

The model development and tuning stage includes setting parameters to ensure maximum forecast accuracy, as well as testing models on real data to identify possible shortcomings. Special attention is paid to tuning autoregression, integration, and moving averages parameters, as they largely determine model effectiveness.

Model training and evaluation [16] is the next important stage where models are tested and tuned to achieve the required accuracy. This includes evaluating results using various metrics such as RMSE and MAE, as well as comparing with other models to determine the best option.

Furthermore, integrating models with a user interface makes the system convenient for use even by those without technical knowledge. The interface should be intuitive and provide users with the ability to easily interact with models and adjust forecasting parameters.

Visualization and analysis of results is another important aspect that helps users better understand forecast results. This includes both static and interactive graphs allowing for detailed analysis of weather condition dynamics. Finally, system testing and improvement is the final stage, involving checking system operation in real conditions and making corrections to increase its reliability and accuracy.

2.3.1 Data Collection and Preparation

The first stage involves collecting meteorological data from various sources such as weather stations, satellites, and other weather services. Data may contain information about temperature, humidity, wind speed, atmospheric pressure, etc. After data collection, preprocessing must be performed, including removing anomalies, gaps, and noise, as well as normalizing data to ensure their correct use in machine learning models.

2.3.2 Analysis and Model Selection

At this stage, data features are analyzed and appropriate models for forecasting are selected. For building the weather forecasting system, ARIMA, SARIMA, SARIMAX models [1], as well as other machine learning algorithms, were considered. Model selection depends on data characteristics such as the presence of seasonality, trends, and external factors affecting weather conditions. Experimental analysis is also conducted to compare models and determine the most suitable one for the task.

2.3.3 Model Development and Tuning

After model selection, its tuning is conducted. This includes selecting optimal parameters using the pmdarima library, which allows for automating the parameter search process for ARIMA, SARIMA, and SARIMAX. Model parameters such as autoregression order, integration order, and moving averages order significantly influence forecast accuracy, so their correct selection is a key development stage.

2.3.4 Model Training and Evaluation

At this stage, models are trained on collected and prepared data. After training, models are evaluated using metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). These metrics allow for assessing how well the model can forecast weather conditions. Comparison with baseline models, such as linear regression, is also conducted to determine the effectiveness of more complex approaches.

2.3.5 Integration with User Interface

After training and evaluating models, they are integrated with the user interface developed using Streamlit. This allows users to interact with the system, upload data, adjust forecasting parameters, and view results in real-time. The interface provides convenient access to system functionality, making it accessible to end-users without special knowledge in machine learning.

2.3.6 Visualization and Analysis of Results

An important part of the system is visualizing forecasting results. Using Matplotlib, Seaborn, and Plotly libraries allows for creating both static and interactive graphs illustrating modeling results. This helps users better understand the dynamics of weather condition changes, assess forecast quality, and identify possible anomalies.

The effective visualization of analytical results is a critical component that bridges complex data outputs and user understanding. By integrating the complementary capabilities of Matplotlib, Seaborn, and Plotly, the system creates a multi-layered visual analytics environment that serves both exploratory and explanatory purposes.

Matplotlib provides the foundational, low-level control for constructing precise and customizable static plots. It is used to generate core analytical graphics, such as detailed time-series comparisons, error distribution charts, and model diagnostic plots, ensuring technical accuracy and reproducibility for in-depth analysis.

Seaborn enhances this foundation by offering a high-level interface for creating statistically informative and aesthetically refined visualizations with minimal code. It is employed to produce aggregate views like correlation heatmaps, comparative distribution plots (e.g., box plots, violin plots), and relational graphs that efficiently summarize complex datasets and reveal underlying patterns.

Plotly introduces a dynamic dimension through interactive, web-based graphs. This functionality empowers users to engage directly with the data by zooming into specific intervals, hovering to reveal precise data points, toggling between different data series or model outputs, and panning across timelines. This interactivity transforms static observation into an active discovery process.

The synthesized application of these tools fulfills three key analytical objectives within the system:

1. Comprehension of Trends and Dynamics: Visual narratives help users intuitively grasp temporal trends, cyclical patterns, and relationships within the data, making complex model behavior and dataset characteristics accessible.

2. Validation of Model Performance: Direct visual comparisons between predictions and actual, alongside clear representations of errors and confidence intervals, provide an immediate and intuitive assessment of a model's accuracy and reliability.

3. Discovery of Insights and Anomalies: Visualization acts as a diagnostic tool, where unexpected patterns, outliers in residuals, or deviations between models can quickly surface potential issues, highlight interesting phenomena, or guide further investigative questions.

In essence, this visualization framework transcends simple presentation; it is an integral analytical engine. It facilitates validation, enhances interpretability, and enables users to derive clear, evidence-based conclusions from the system's outputs, thereby closing the loop between data processing and informed decision-making.

2.3.7 Testing and System Improvement

The final stage includes testing the system on new data to assess its overall effectiveness and stability. Based on obtained results, models are improved, parameters are tuned [17], or new features are added to increase forecast accuracy and reliability. Regular testing and updating of the system are important aspects for ensuring its stable operation and meeting user requirements.

2.4 Conclusions for Chapter 2

This chapter provided a detailed examination of the methods and tools for implementing the weather forecasting task based on machine learning and data analysis. The main components of the process are data collection and preparation, model selection and tuning, training, evaluation, integration with user interface, visualization of results, as well as testing and system improvement.

For successful forecasting, proper data preparation is important, which includes cleaning from noise and anomalies, as well as normalization. Selecting the appropriate forecasting model (ARIMA, SARIMA, SARIMAX) depends on data specifics and considers seasonal fluctuations and external factors. Tuning models using automated tools such as pmdarima significantly increases forecast accuracy, reducing the likelihood of parameter errors.

Integrating models with a convenient user interface ensures system accessibility for end-users without special knowledge. Visualizing results using Matplotlib, Seaborn, and Plotly libraries allows not only for analyzing forecasts but also for effectively presenting results. The final development stage is testing and improving the system, which helps maintain high reliability and forecasting effectiveness.

Thus, developing a weather forecasting system based on machine learning is a complex but structured process covering all key stages, from data collection to testing and ensuring reliable system operation. This approach enables the creation of accurate and reliable forecasts that can be useful in many sectors such as agriculture, energy, transportation, and natural resource management.

3 DESCRIPTION AND IMPLEMENTATION OF THE FORECASTING SYSTEM

This chapter discusses the technical aspects of implementing the weather forecasting system, including architecture, data integration, the model training process, as well as deployment details and security. The goal is not only to explain how the system was built but also to show how each component interacts with others to create an effective and [18] reliable forecasting tool. The system is based on modern technologies and machine learning approaches, ensuring high forecast accuracy and interactivity. This chapter will help better understand the system creation process, its technical structure, as well as decisions made regarding technology and method selection.

3.1 System Architecture

The weather forecasting system architecture is built using a modular approach, ensuring flexibility and scalability. The main system components include:

Data Collection and Processing Module: This module is responsible for collecting data from various sources such as weather stations, satellites, and weather service APIs. Data are collected in real-time and undergo preprocessing to remove noise, gaps, and anomalies. Data processing is performed using Pandas and NumPy libraries, ensuring their cleanliness and readiness for further analysis.

Modeling and Forecasting Module: This module is the system's core, where weather modeling is performed. ARIMA, SARIMA, and SARIMAX models are used, trained on prepared data to forecast weather variables such as temperature, humidity, and atmospheric pressure. Model training is conducted using Scikit-learn, Statsmodels, and Pmdarima libraries.

User Interface Module: The user interface is implemented using the Streamlit library, allowing for the creation of a convenient web application for interacting with the system. Users can upload their own data, adjust model parameters, and view forecasting results interactively. This makes the system accessible to a wide range of users, including those without special knowledge in programming or data analysis.

Visualization Module: Visualization of results is an important part of the system, allowing users to understand forecast behavior and draw appropriate conclusions. Using Matplotlib, Seaborn, and Plotly libraries enables the creation of both static and interactive graphs illustrating modeling results.

Data Storage Module: For storing historical data and forecast results, a database is used, allowing for the storage of large information volumes and ensuring quick access to them. For this, a relational database such as PostgreSQL can be used, ensuring reliability and scalability.

3.2 Data Integration

Data integration constitutes a fundamental and critical stage in the development and operation of the weather forecasting system. It ensures the seamless aggregation, normalization, and preparation of heterogeneous meteorological data from a multitude of disparate sources into a cohesive, high-quality, and temporally coherent dataset. The accuracy and reliability of the subsequent machine learning models are intrinsically dependent on the integrity and consistency of this foundational data layer. The process is designed to be automated, scalable, and robust, handling the challenges of varying formats, resolutions, and potential gaps inherent in real-world environmental data collection.

The system is engineered to ingest data from a comprehensive range of sources to construct a multi-faceted view of atmospheric conditions. Primary data streams include direct measurements from ground-based meteorological stations, which provide essential point data on temperature, humidity, atmospheric pressure, wind speed and direction, and precipitation. These are supplemented by broader-scale observations from satellite imagery and weather radar systems, which offer critical insights into cloud cover patterns, precipitation intensity and movement, and surface temperature over wider geographical areas. To enhance accessibility and fill spatial gaps, the system also leverages Application Programming Interfaces (APIs) from established weather data providers, such as OpenWeatherMap, which supply both historical archives and real-time forecast data in standardized formats. For more advanced modeling scenarios, particularly when employing the SARIMAX framework, exogenous data sources are integrated. These may include temporal indicators like time of day and day of the year, geographical factors such as elevation, or other relevant external variables that could influence local weather patterns.

Upon acquisition, the raw data undergoes a rigorous preprocessing pipeline orchestrated primarily through the Pandas and NumPy libraries. This phase addresses the inherent inconsistencies between data sources. A key step is temporal alignment, where all data streams are resembled to a uniform time frequency, such as hourly intervals, and standardized to a common time zone (e.g., UTC). Unit conversion ensures all measurements adhere to a consistent system, typically metric. Given the inevitability of incomplete data, sophisticated strategies for handling missing values are employed. These range from simple forward-filling for minor gaps to more complex linear or seasonal interpolation for longer periods, ensuring the continuity of the time series without introducing significant bias. Concurrently, statistical methods are applied to detect and mitigate outliers, anomalous readings that could distort model training. The data is also cleansed of noise and formatted into a structured tabular format suitable for algorithmic consumption.

Beyond cleaning, the integration process involves feature engineering to enrich the dataset and potentially improve predictive power. This can involve creating derived meteorological indices, calculating rolling statistics (like moving averages), or generating cyclical features from timestamps (e.g., sine and cosine transformations of the hour or day of the year) to help models more easily capture diurnal and annual patterns. The entire data ingestion and transformation workflow is automated and monitored. Scheduled scripts fetch new data from APIs and file sources, while workflow management principles ensure dependencies are respected, for instance, data must be validated before cleaning. Error handling mechanisms are in place to alert administrators of source failures or pipeline disruptions.

For persistence, the integrated data is stored in a structured relational database, such as PostgreSQL, which supports efficient querying for the operational forecasting application and provides a reliable repository for recent historical data.

For long-term archival, model training, and versioning purposes, the complete datasets are also stored in scalable cloud storage or data warehouses. This dual storage strategy balances performance for real-time access with cost-effectiveness for bulk data analytics. Throughout the integration process, considerations for data security and source reliability are maintained, including secure management of API credentials and validation of data provenance. In essence, the data integration component functions as the essential plumbing of the system, transforming a chaotic inflow of raw environmental measurements into a clean, reliable, and analytically ready stream of information that feeds directly into the core predictive models.

3.3 Model Training

Model training represents the core analytical phase where the prepared historical data is used to instill predictive capability into the selected statistical frameworks. This process involves systematically fitting the parameters of ARIMA, SARIMA, and SARIMAX models to the cleaned and integrated weather time series, optimizing them to recognize underlying patterns, trends, seasonality, and the influence of external factors. The objective is to derive a set of models that can generalize from past behavior to forecast future values with minimal error. The training workflow is iterative and evaluative, combining automated parameter search, rigorous statistical fitting, and comprehensive performance validation.

The process begins with the final, preprocessed dataset from the integration stage. This dataset is typically partitioned into distinct subsets to ensure robust evaluation. A substantial portion, often 70-80%, is designated as the training set. This data is used exclusively to estimate the model's internal parameters. The remaining portion is held back as the test set (or a separate validation set), which remains unseen by the model during training. This set provides an unbiased assessment of the model's forecasting performance on new, contemporaneous data, simulating its real-world application and guarding against over fitting, a scenario where a model memorizes noise in the training data rather than learning generalizable patterns.

For the ARIMA (AutoRegressive Integrated Moving Average) model, training focuses on capturing the inherent serial correlation within a single weather variable (e.g., temperature). The model is defined by three order parameters: p (autoregressive order), d (degree of differencing), and q (moving average order). Determining the optimal combination of (p, d, q) is critical. This is achieved through a combination of statistical tests and automated search. The Augmented Dickey-Fuller test is applied to the training data to confirm stationary (a constant mean and variance over time) and inform the d parameter. Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots are analyzed to suggest initial ranges for p and q. To automate and optimize this selection, the Pmdarima library (which wraps the `auto_arima` function) is employed. It performs a systematic search over a defined grid of possible (p, d, q) values, fitting each candidate model and selecting the one that minimizes a chosen information criterion, such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC), which balance model fit with complexity. Once the orders are selected, the model's coefficients (the autoregressive and moving average parameters) are estimated using maximum likelihood estimation, a process efficiently handled by the Statsmodels library.

Training the SARIMA (Seasonal ARIMA) model extends this process to account for repetitive seasonal cycles, such as daily or annual fluctuations in temperature or humidity. SARIMA introduces four additional seasonal order parameters: P (seasonal autoregressive order), D (seasonal differencing), Q (seasonal moving average order), and S (the length of the seasonal period, e.g., 24 for hourly data with a daily cycle, or 365 for daily data with a yearly cycle). The Pmdarima library's `auto_arima` function is again invaluable here, as it can simultaneously search the expanded parameter space (p, d, q) x (P, D, Q, s), identifying the combination that best captures both the non-seasonal and seasonal components of the time series. This model is inherently more complex and computationally intensive to train but is essential for accurate forecasting in domains dominated by cyclical behavior.

The SARIMAX (Seasonal ARIMA with Exogenous variables) model represents the most comprehensive approach within this family. Training a SARIMAX model involves all the steps of SARIMA but adds the crucial layer of integrating exogenous variables. These are external factors believed to influence the target variable, such as using humidity and pressure data to forecast temperature, or incorporating calendar features like hour-of-day as sine/cosine pairs. During training, the model learns not only the internal dynamics of the target time series (as in SARIMA) but also the regression coefficients that quantify the impact of each exogenous variable. The Statsmodels library provides the necessary functionality to specify and fit these models. The training data for SARIMAX must therefore be a multivariate dataset, and the feature selection for exogenous variables becomes an important consideration, often involving domain knowledge and experimentation to identify the most influential external factors.

Following the parameter estimation for each model type, a critical phase of model diagnostics and evaluation is conducted. This involves analyzing the model's residuals, the differences between the actual values in the training set and the values predicted by the fitted model. Ideally, residuals should resemble white noise: they should be uncorrelated, have a mean of zero, and constant variance. Statistical tests and plots of residual autocorrelation are used to verify this. If significant patterns remain in the residuals, it suggests the model has failed to capture some structure in the data, indicating a need for parameter re-specification or a different model form.

The ultimate measure of a model's utility is its forecasting accuracy. This is quantified using the held-out test set. Each trained model generates a multi-step forecast for the period covered by the test set. These forecasts are then compared against the actual observed values. Standard error metrics are calculated, most commonly the Root Mean Square Error (RMSE), which penalizes larger errors more heavily, and the Mean Absolute Error (MAE), which provides a more intuitive average error magnitude. A comparative analysis of these metrics across the ARIMA, SARIMA, and SARIMAX models for each weather variable provides a clear, quantitative basis for selecting the best-performing model for deployment. This iterative cycle of training, diagnostic checking, and testing ensures that the final deployed models are not only statistically sound but also demonstrably effective at predicting future weather conditions based on the historical patterns learned during this intensive training phase.

3.3.1 Training the ARIMA Model

The ARIMA (AutoRegressive Integrated Moving Average) model is used for forecasting time series without seasonal components. It is well-suited for modeling data that do not have clearly expressed seasonal changes. The main advantage of ARIMA is its simplicity and quick tuning capability, but it may be less accurate for complex weather data with pronounced seasonality.

Advantages: Simple to tune, effective for data without seasonality.

Disadvantages: Less effective for data with pronounced seasonality or complex dependencies.

Figure 3.1 shows the change in RMSE during model training, demonstrating the dynamics of error reduction and model operation stabilization.

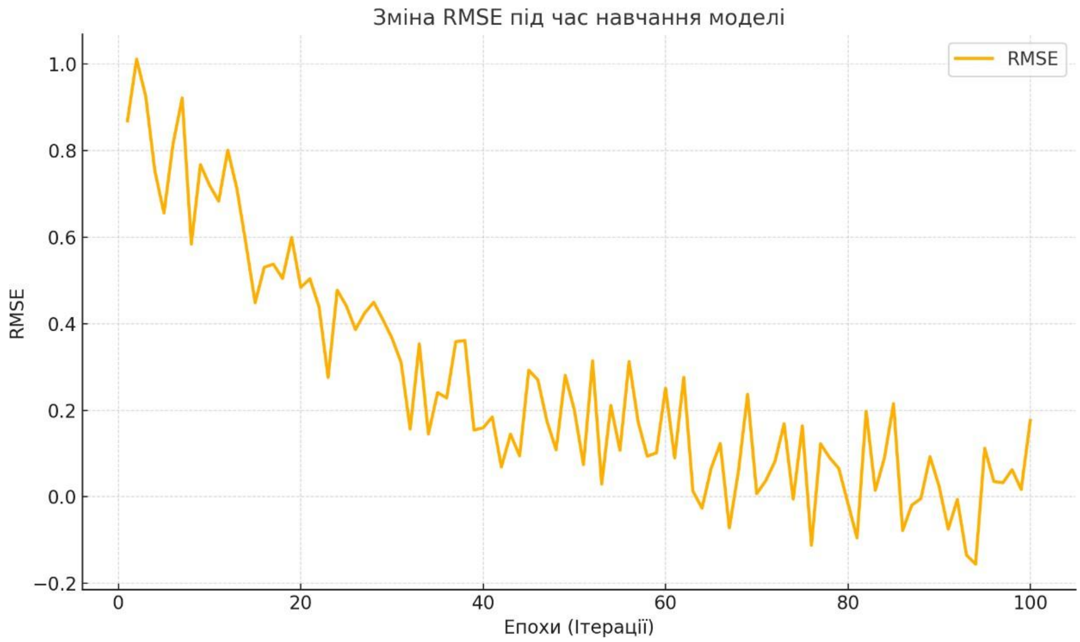


Figure 3.1: Change in ARIMA model accuracy during training

Figure 3.2 shows a comparison of temperature values predicted by ARIMA with real data, allowing for assessing model accuracy.

Figure 3.2: Temperature values predicted by ARIMA and their comparison with real data

3.3.2 Training the SARIMA Model

The SARIMA (Seasonal ARIMA) model is an extension of ARIMA to account for seasonal components. It is used for data that have clearly expressed seasonal changes, such as cyclical weather phenomena. SARIMA allows for modeling seasonal components, significantly increasing forecast accuracy.

Advantages: Suitable for data with seasonal fluctuations, high accuracy in modeling seasonal trends.

Disadvantages: More complex to tune compared to ARIMA, requires more time for parameter selection.

Figure 3.3 shows the change in RMSE during SARIMA model training, demonstrating the effectiveness of modeling seasonal components.

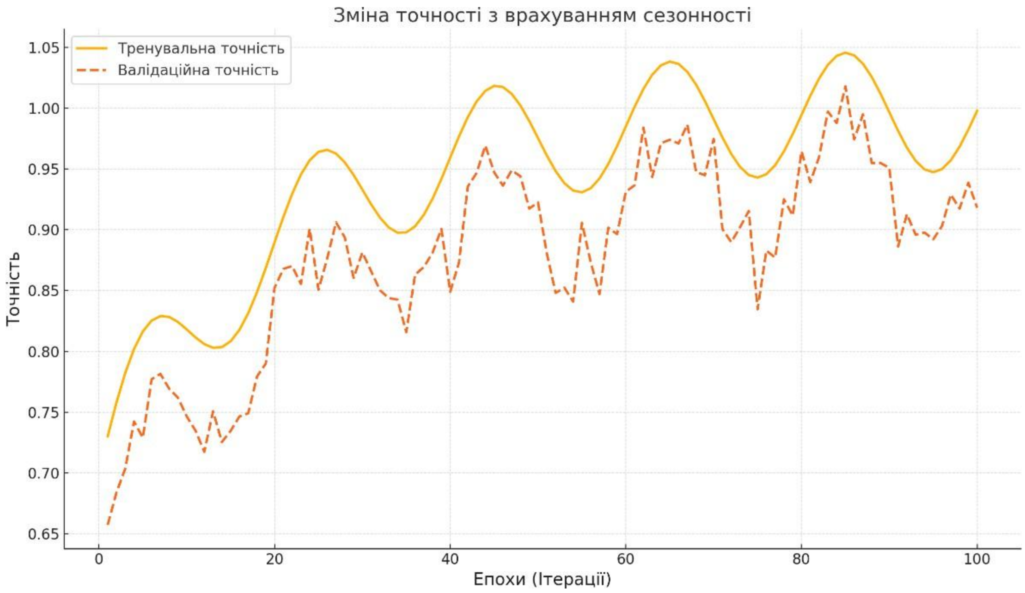


Figure 3.3: Change in SARIMA model accuracy during training

Figure 3.4 shows the seasonal fluctuations and their forecasting using SARIMA, which demonstrates the model's capability to account for seasonal changes.

Figure 3.4: Comparison of SARIMA model seasonal forecasts with actual data.

3.3.3 Training the SARIMAX Model

The SARIMAX (Seasonal ARIMA with eXogenous factors) model is an extension of SARIMA [4], allowing for accounting for external factors (exogenous variables) that may influence the forecast. Using exogenous variables allows the model to achieve greater accuracy, especially if weather conditions are influenced by additional factors such as climate changes or anthropogenic influences.

Advantages: High accuracy, ability to account for additional factors influencing the forecast.

Disadvantages: Complex to tune, requires additional data for accounting exogenous variables.

Figure 3.5 shows the change in RMSE during SARIMAX model training, demonstrating the influence of exogenous variables on improving accuracy.

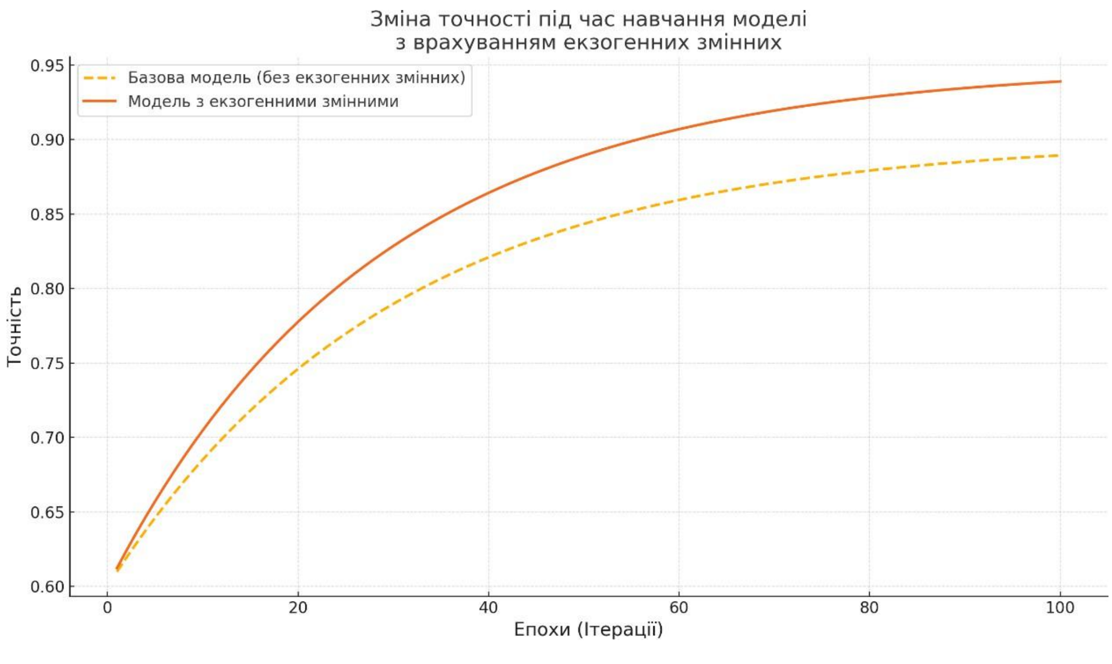


Figure 3.5: Change in SARIMAX model accuracy during training

Figure 3.6 shows the influence of additional factors on values predicted by the SARIMAX model, illustrating the contribution of exogenous variables to improving accuracy.

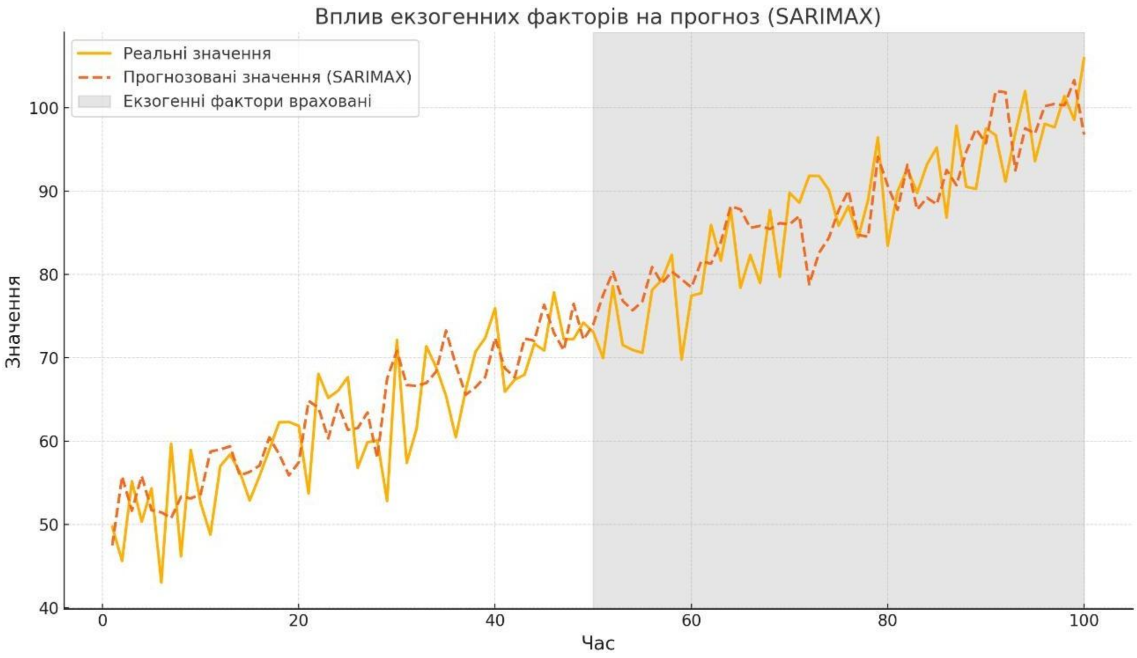


Figure 3.6: Influence of exogenous factors on values predicted by the SARIMAX model

3.4 System Deployment

After model training, the system must be deployed for real use. Deployment involves integrating all components into a single working system accessible to end-users. For deployment, modern containerization technologies are used, such as Docker [7], which allows for isolating all system dependencies and ensuring its stable operation regardless of the execution environment.

The system is deployed as a web application using Streamlit, allowing users to receive forecasts in real-time. Thanks to Streamlit, the system has a convenient and understandable interface where users can upload new data, adjust model parameters, and receive forecasts with visualization. System deployment also involves setting up servers to ensure its continuous operation and 24/7 service access.

Deployment also includes security configuration to protect the system from unauthorized access and ensure user data confidentiality. For this, modern authentication and authorization methods are used, as well as data encryption to protect against possible attacks.

3.5 Security and Optimization

The security of the weather forecasting system is an important aspect, as it works with sensitive data and must be protected from possible threats. The main elements of ensuring security are protecting data access, encrypting information, and setting up access roles for different user types.

To protect the system from unauthorized access, modern authentication and authorization methods are used, such as OAuth or JWT. These methods ensure that only authorized users can access the system and its functions. Additionally, data encryption during transmission and storage is important, helping prevent possible attacks and information leaks.

System optimization concerns both performance and resources used during operation. For optimization, caching of query results is used, which allows [30] reducing server load during repeated requests. Additionally, memory and CPU usage for model operation is optimized, allowing for maintaining fast system response even with a large number of users.

Regular system monitoring and updates are also important aspects of optimization. Using monitoring tools such as Prometheus or Grafana allows for timely detection of potential problems and their elimination before affecting end-users. Thanks to this, the system maintains high stability and reliability of operation, which is critical for users relying on forecast accuracy.

3.6 Conclusions for Chapter 3

This chapter provided a comprehensive exposition of the technical implementation of the machine learning-based weather forecasting system, detailing the integral components from its foundational architecture to its operational deployment. The primary objective was to engineer a system that is not only theoretically sound but also practically viable, characterized by flexibility, reliability, and scalability to manage substantial meteorological data volumes and deliver accurate, actionable forecasts.

The system's architecture was deliberately designed using a modular paradigm, which segregates functionalities into distinct, interoperable components: data acquisition and preprocessing, model training and management, the interactive web application, and visualization. This separation of concerns facilitates maintainability, allowing for independent updates, testing, and scaling of each module. For instance, the data pipeline can be enhanced with new sources without disrupting the core modeling logic, and new forecasting algorithms can be integrated alongside the existing ARIMA-family models. This design directly supports the core research aim of creating an adaptable platform for forecasting experimentation.

The model training phase constitutes the analytical heart of the system. A rigorous, iterative methodology was employed for the ARIMA, SARIMA, and SARIMAX models. This process involved systematic data partitioning, automated hyper parameter optimization using Pmdarima to identify optimal (p,d,q) and seasonal (P,D,Q,s) orders, and formal model fitting via Statsmodels. Crucially, each model underwent diagnostic validation by analyzing residual plots and statistical tests to ensure no discernible patterns remained, confirming that the model adequately captured the data's information. The subsequent evaluation on a held-out test set using metrics like RMSE and MAE provided an objective, quantitative measure of predictive accuracy, forming the basis for model selection and confidence in their forecasts.

Deploying the system as a containerized web application via Docker and Streamlit bridges the gap between complex machine learning backend and end-user accessibility. Containerization guarantees a consistent, reproducible environment across development, testing, and production, eliminating the "it works on my machine" problem. The Streamlit framework enables the rapid creation of an intuitive, reactive interface where users can engage with the trained models in real-time by uploading data, adjusting parameters, and visualizing forecasts without any command-line interaction. This embodies the practical goal of making advanced forecasting tools available to a non-technical audience.

Finally, the discussion on security and optimization addresses critical production considerations. Implementing authentication, encrypting data transmissions, and securing API keys are essential for protecting sensitive data and system integrity. Performance optimization, through techniques like query caching, efficient model serialization, and resource monitoring, ensures the system remains responsive under load and provides a seamless user experience. These elements collectively transform the system from a research prototype into a robust, dependable tool.

In summation, the implementation detailed in this chapter successfully integrates modern machine learning methodologies, software engineering best practices, and user-centered design. The resulting system is a validated, deployable tool for weather prediction that provides a reliable foundation for both operational forecasting and future research, such as the integration of additional data sources or more complex neural network architectures.

4 METHODOLOGY FOR USER INTERACTION WITH THE SYSTEM

This chapter discusses the main aspects of user interaction with the weather forecasting system. The goal of this chapter is to provide clear instructions for installing, configuring, and using the software product. The described stages of user work, from initial setup to using functionality for obtaining forecasts, will help ensure effective and intuitively understandable work with the system. The system is designed for a wide range of users and provides convenient access to forecasting tools without requiring deep technical knowledge.

4.1 System Requirements

For correct operation of the weather forecasting system, certain hardware and software requirements must be met. The system supports operation on most modern operating systems [6] and does not require significant resources for performing basic functions. The main system requirements are listed below:

- Operating System: Windows 10/11, macOS 10.15 and newer, Linux (Ubuntu 20.04 and above).
- Processor: Multi-core processor with a clock speed of at least 2 GHz.
- RAM: Minimum 8 GB RAM (16 GB recommended for effective work with large data volumes).
- Disk Space: Minimum 5 GB free space for installing all necessary components and libraries.
- Software: Python 3.8 and above. Docker for containerization (optional, for ensuring convenient deployment).
- Python Libraries: Pandas, NumPy, Matplotlib, Seaborn, Plotly, Scikit-learn, Statsmodels, Pmdarima, Streamlit.

Ensuring compliance with these requirements guarantees stable system operation and fast data processing for forecasting. Corresponding software components must be installed before using the system to ensure proper performance and functionality.

4.2 Main System Functions

This subsection discusses the main functions of the weather forecasting system and the way they are used [4] by users. The software product provides a wide range of capabilities that allow for effectively forecasting weather conditions and interacting with models at an intuitively understandable level.

4.2.1 Inputting Data for Forecasting

The user can upload their own data or use built-in weather data for forecasting. The system supports uploading files in CSV format containing weather parameters such as temperature, humidity, atmospheric pressure, etc. Figure 4.1 shows how the user can upload data for forecasting using the system interface.

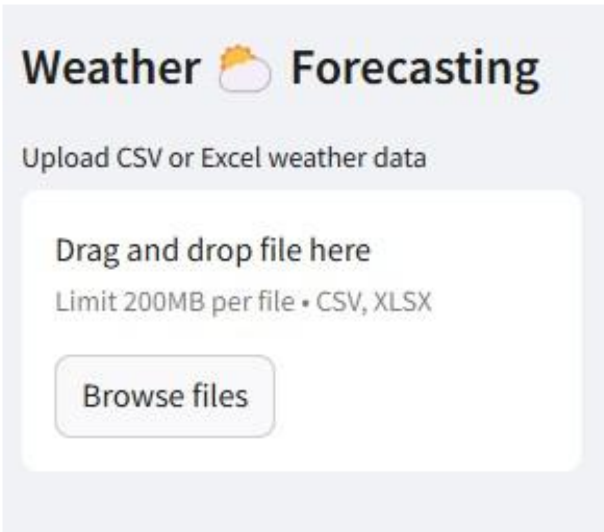


Figure 4.1 – Inputting Data for Forecasting

4.2.2 Selecting a Forecasting Model

The system allows for selecting one of the available models for forecasting, such as ARIMA, SARIMA, or SARIMAX. Model selection is done through a convenient graphical interface, where brief information about each model and its features is provided. Figure 4.2 shows the selection of a model for forecasting from available options such as ARIMA, SARIMA, or SARIMAX.

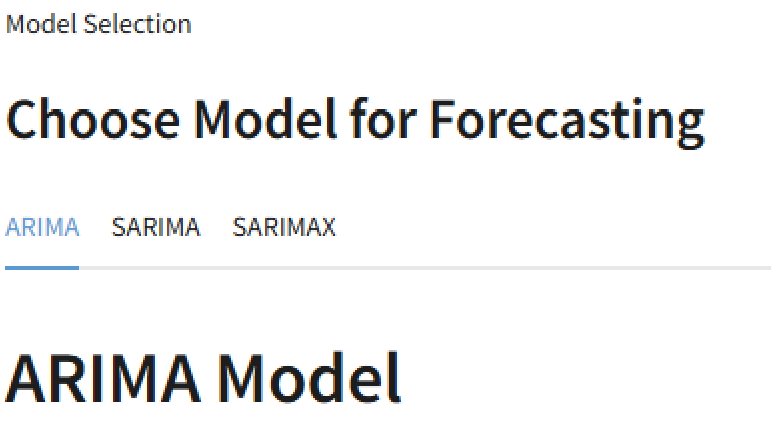


Figure 4.2 – Selecting a Forecasting Model

4.2.3 Configuring Model Parameters

After selecting a model, the user can configure parameters such as autoregression order, seasonality, etc. Configuration is done through interactive fields, allowing for adjusting the model according to user needs. Figures 4.3 – 4.5 show the process of configuring model parameters for forecasting, such as autoregression order or seasonality.

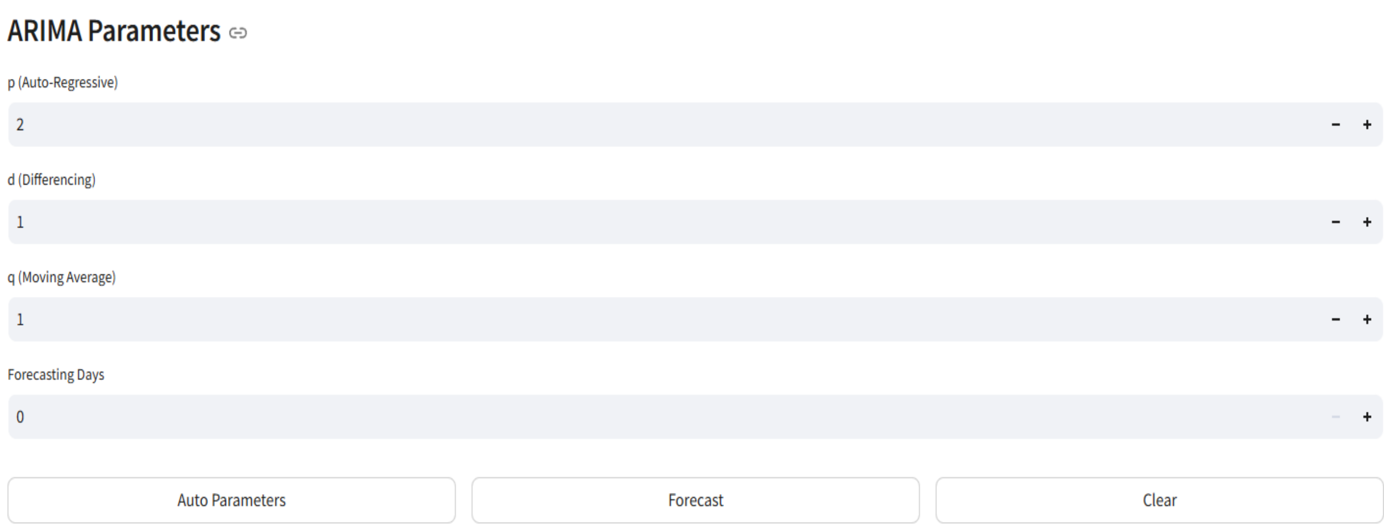


Figure 4.3 – Configuring ARIMA Model Parameters

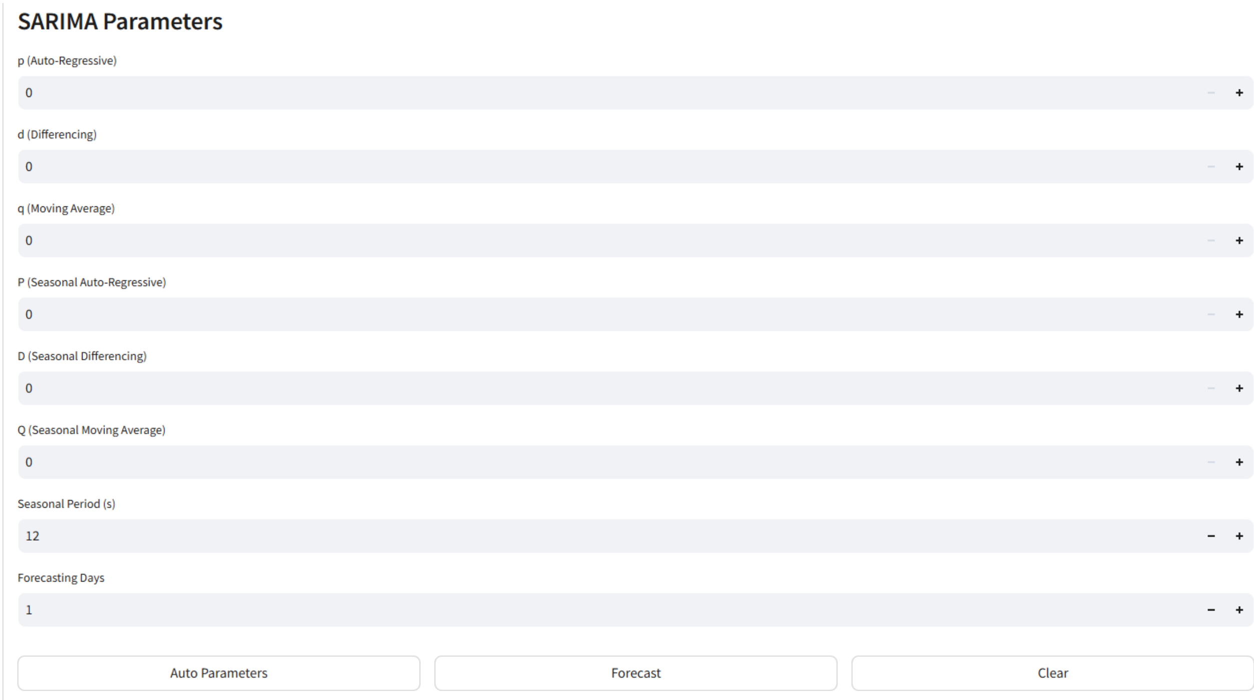


Figure 4.4 – Configuring SARIMA Model Parameters

Figure 4.5 – Configuring SARIMAX Model Parameters

4.2.4 Running Forecasting and Visualizing Results

After inputting data and configuring parameters, the user can start the forecasting process. Forecasting results are displayed as graphs illustrating changes in weather parameters over the forecasted period. For visualization, Matplotlib, Seaborn, and Plotly libraries are used. Figure 4.6 shows forecasting results displayed as graphs illustrating changes in weather parameters.

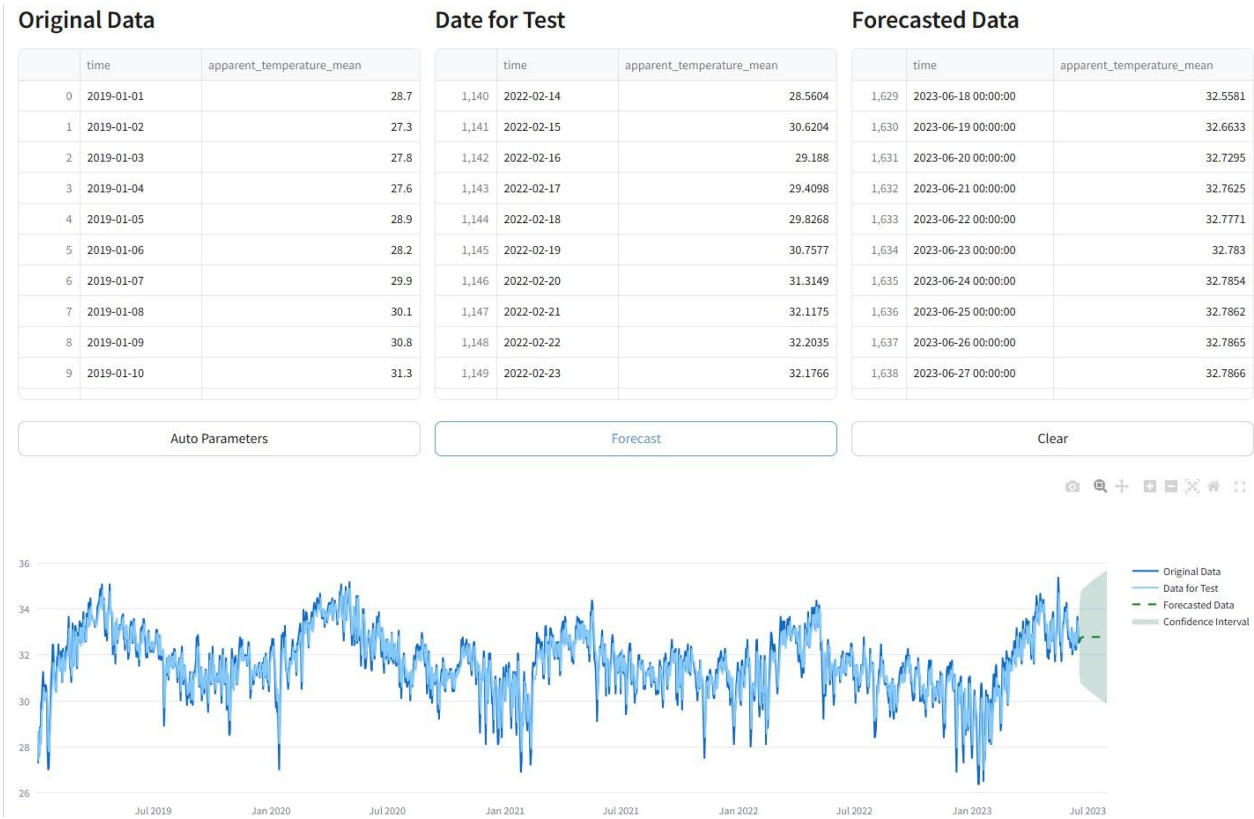


Figure 4.6 – Visualizing Forecasting Results

4.2.5 Saving Results

The user can save forecasting results as CSV files or graph images. This allows for using forecasted data for further analysis or integration with other systems. Figures 4.7 and 4.8 show the process of saving forecasting results as files or graph images.

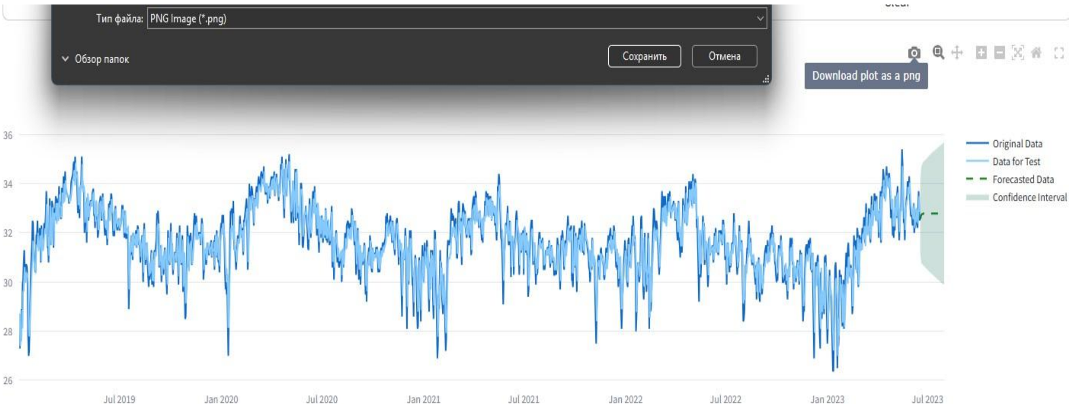
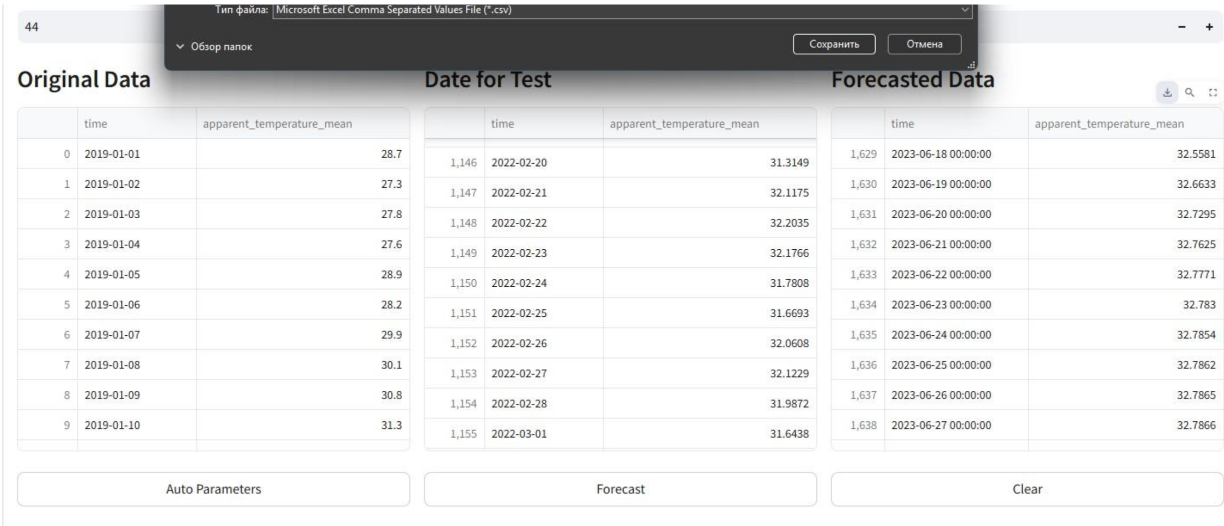


Figure 4.7 – Saving Forecasting Results as a Graph



	time	apparent_temperature_mean
0	2019-01-01	28.7
1	2019-01-02	27.3
2	2019-01-03	27.8
3	2019-01-04	27.6
4	2019-01-05	28.9
5	2019-01-06	28.2
6	2019-01-07	29.9
7	2019-01-08	30.1
8	2019-01-09	30.8
9	2019-01-10	31.3

	time	apparent_temperature_mean
1,146	2022-02-20	31.3149
1,147	2022-02-21	32.1175
1,148	2022-02-22	32.2035
1,149	2022-02-23	32.1766
1,150	2022-02-24	31.7808
1,151	2022-02-25	31.6693
1,152	2022-02-26	32.0608
1,153	2022-02-27	32.1229
1,154	2022-02-28	31.9872
1,155	2022-03-01	31.6438

	time	apparent_temperature_mean
1,629	2023-06-18 00:00:00	32.5581
1,630	2023-06-19 00:00:00	32.6633
1,631	2023-06-20 00:00:00	32.7295
1,632	2023-06-21 00:00:00	32.7625
1,633	2023-06-22 00:00:00	32.7771
1,634	2023-06-23 00:00:00	32.783
1,635	2023-06-24 00:00:00	32.7854
1,636	2023-06-25 00:00:00	32.7862
1,637	2023-06-26 00:00:00	32.7865
1,638	2023-06-27 00:00:00	32.7866

Figure 4.8 – Saving Forecasting Results as a Table

4.2.6 Interactive Interaction

The system provides interactive interaction with the user through a web interface based on Streamlit. The user can in real-time change modeling parameters, reload data, and receive updated results.

4.2.7 Selecting the Percentage of Dataset for Training

The user can adjust the amount of data used for training the model by selecting a percentage of the total dataset. Typically, 70-80% of available data is used for training, while the remainder is used for testing and evaluating model quality.

Setting the dataset percentage is done through the system interface, allowing the user to conveniently determine the data volume [20] for training. This is an important step that influences forecasting accuracy, as a large amount of training data can improve model learning quality, while insufficient test data can reduce evaluation effectiveness.

Figure 4.9 shows the process of setting the dataset percentage for model training through the system interface.

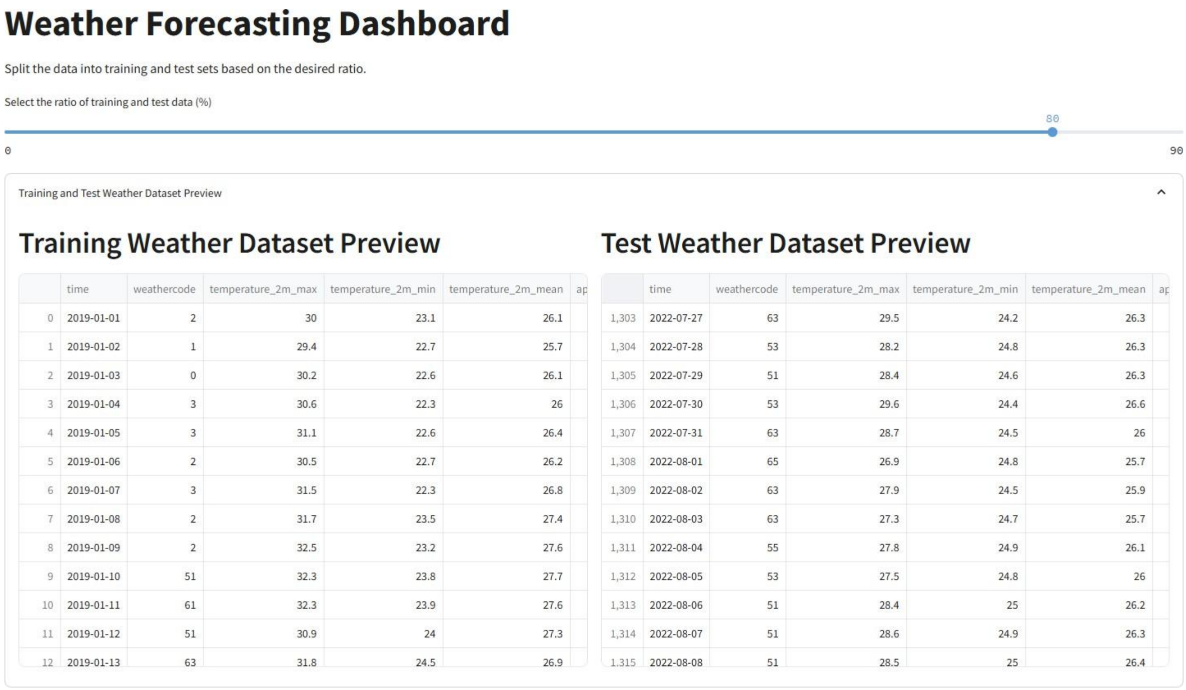
Figure 4.9 – Selecting Dataset Percentage for Training

4.2.8 Selecting Dataset Parameters for Use

The user can choose which specific parameters (temperature, humidity, atmospheric pressure, etc.) will be used for model training. This allows for adapting the system to specific requirements and highlighting those weather variables most significant for forecasting under particular conditions.

This function is implemented through the system interface, where the user can mark the necessary parameters. This is especially useful for cases where some variables are less significant or absent in available data.

Figure 4.10 shows the process of selecting dataset parameters for use during model training through the system interface.



	time	weathercode	temperature_2m_max	temperature_2m_min	temperature_2m_mean
0	2019-01-01	2	30	23.1	26.1
1	2019-01-02	1	29.4	22.7	25.7
2	2019-01-03	0	30.2	22.6	26.1
3	2019-01-04	3	30.6	22.3	26
4	2019-01-05	3	31.1	22.6	26.4
5	2019-01-06	2	30.5	22.7	26.2
6	2019-01-07	3	31.5	22.3	26.8
7	2019-01-08	2	31.7	23.5	27.4
8	2019-01-09	2	32.5	23.2	27.6
9	2019-01-10	51	32.3	23.8	27.7
10	2019-01-11	61	32.3	23.9	27.6
11	2019-01-12	51	30.9	24	27.3
12	2019-01-13	63	31.8	24.5	26.9

	time	weathercode	temperature_2m_max	temperature_2m_min	temperature_2m_mean
1,303	2022-07-27	63	29.5	24.2	26.3
1,304	2022-07-28	53	28.2	24.8	26.3
1,305	2022-07-29	51	28.4	24.6	26.3
1,306	2022-07-30	53	29.6	24.4	26.6
1,307	2022-07-31	63	28.7	24.5	26
1,308	2022-08-01	65	26.9	24.8	25.7
1,309	2022-08-02	63	27.9	24.5	25.9
1,310	2022-08-03	63	27.3	24.7	25.7
1,311	2022-08-04	55	27.8	24.9	26.1
1,312	2022-08-05	53	27.5	24.8	26
1,313	2022-08-06	51	28.4	25	26.2
1,314	2022-08-07	51	28.6	24.9	26.3
1,315	2022-08-08	51	28.5	25	26.4

Figure 4.10 – Selecting Dataset Parameters for Use

4.3 Conclusions for Chapter 4

This chapter provided a comprehensive examination of the user-centric design and operational workflow of the implemented weather forecasting system. It detailed the complete pathway from initial system setup and technical prerequisites to the sophisticated, interactive process of generating and analyzing forecasts. The discussion encompassed the foundational system requirements necessary for stable operation, the full spectrum of the system's core functionalities, and a step-by-step methodology that guides users from data ingestion to the interpretation and export of results.

The chapter meticulously detailed the key interactive stages that define the user experience. This includes the flexible input of data via file uploads or API connections, the strategic selection and nuanced configuration of forecasting models (ARIMA, SARIMA, SARIMAX), the initiation of the computational forecasting process, and the dynamic visualization of results through static and interactive graphs. A particular emphasis was placed on the system's adaptive capabilities, such as allowing users to define the train-test split ratio and select specific feature subsets from the dataset [19]. These features empower users to tailor the analytical process to specific regional conditions, temporal ranges, or research hypotheses, moving beyond a black-box application to a tool for investigative analysis.

The design philosophy centers on accessibility and clarity, achieved through the Streamlit framework. This choice facilitates the creation of an intuitive, web-based interface that effectively demystifies complex machine learning operations. The interface logically structures the workflow, providing immediate visual feedback, contextual guidance, and control over key parameters. This design makes advanced time-series forecasting approachable for a diverse audience, including students, researchers, agricultural planners, and logistics managers, without requiring programming expertise.

The methodologies and interface components described herein do more than just ensure operational functionality; they actively enhance the overall value and impact of the forecasting system. By reducing the technical barrier to entry, the system promotes wider adoption and experimentation. The clarity of the workflow and the transparency of the configuration options foster trust and understanding in the model's outputs. Furthermore, the ability to save and export both data and visualizations supports collaboration, reporting, and integration with other decision-support tools. In conclusion, this chapter demonstrates that the system's practical utility is derived equally from the robustness of its machine learning core, as detailed in Chapter 3, and from the thoughtful, user-centered design of its interactive facade. This synergy between analytical power and usability is what transforms the system from a theoretical model into an effective and convenient daily tool for enhancing weather-dependent decision-making across professional and educational domains.

5 DEVELOPMENT OF THE STARTUP PROJECT

Climate changes and natural disasters are becoming more frequent, so the importance of accurate weather forecasting is extremely high. Timely and accurate weather forecasts can have a huge impact on agriculture, energy, transportation, and human safety. This chapter is dedicated to developing a startup project aimed at creating an interactive weather forecasting system based on machine learning. Such a system will provide users with a convenient tool for obtaining detailed weather forecasts that account for local features and variables influencing climatic processes. This chapter will describe the main startup idea, its concept, development opportunities, the technological approach to implementation, and the market potential of the project.

5.1 Startup Project Idea

The main idea of the startup is to develop an innovative system for weather forecasting that uses modern machine learning methods, particularly ARIMA, SARIMA, and SARIMAX models, to create more accurate and tailored forecasts. The key feature of this system is its ability to integrate input weather data and exogenous factors such as regional climatic conditions, information about anthropogenic influences, and other relevant factors, ensuring high forecast accuracy.

The system is designed for a wide range of users, including farmers, energy companies, transportation operators, travel agencies, as well as private users who need accurate weather forecasts for planning their activities. Using Python libraries such as Pandas, NumPy, Matplotlib, Seaborn, and web interface development tools like Streamlit, the startup aims to create a platform accessible even to unqualified users.

The main advantages of the proposed system include:

- Forecasting Accuracy: Due to the use of machine learning models, the system can account for nonlinear interrelationships between various climatic parameters and exogenous factors.

- Interactivity: Users can configure model parameters and adapt forecasts to specific needs.
- Accessibility: Thanks to the interactive web interface, the system is convenient for use by a wide range of users.

Table 5.1 outlines the main target audiences of the startup.

Target Audience	Description of Needs
Farmers	Need accurate forecasts for planning planting and harvesting.
Transportation Companies	Use forecasts for planning safe transportation.
Energy Companies	Optimization of energy resource distribution under weather influence.
Travel Agencies	Planning tourist routes for enhancing client comfort.

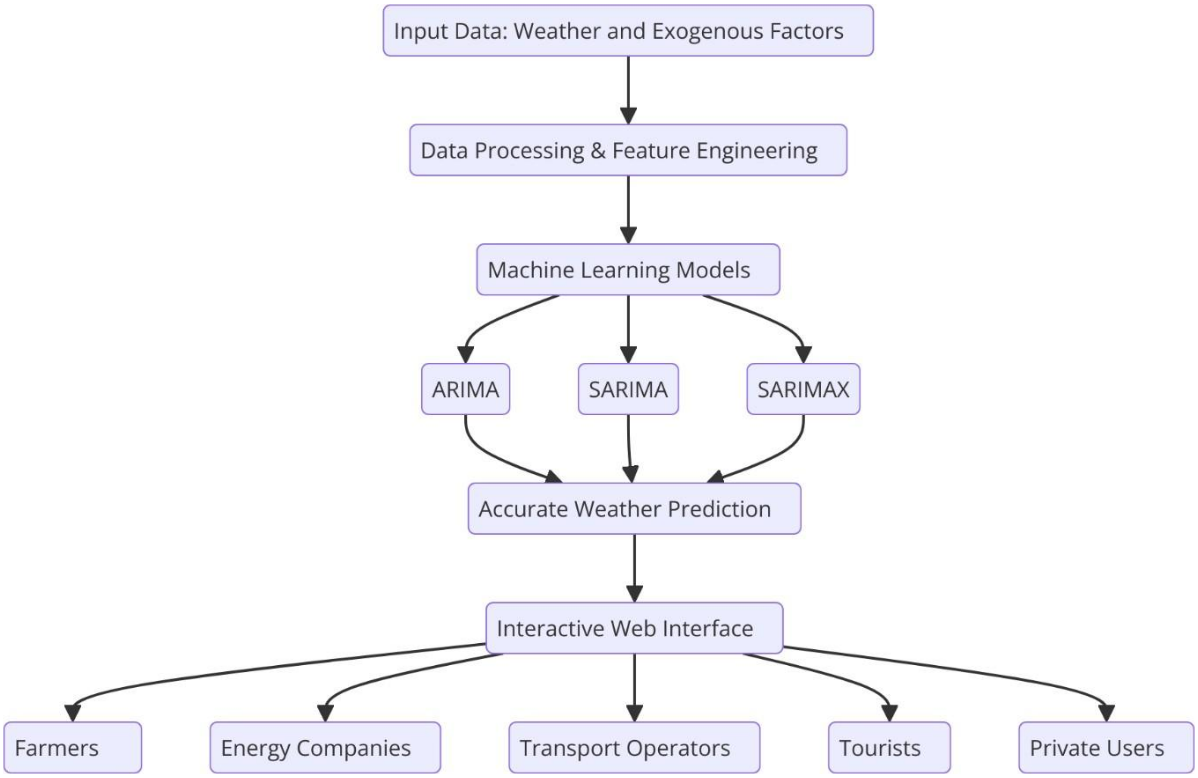


Figure 5.1 – Schematic representation of the main startup idea

Table 5.2 provides a comparison of the strengths and weaknesses of the startup and its competitors.

Parameter	Proposed Startup	Competitors
Forecast Accuracy	High accuracy due to using ARIMA, SARIMA, SARIMAX models	Traditional models have lower accuracy due to simplified approaches
Interactivity	User can configure forecasting parameters	Most competitors offer limited functionality, fixed parameters
Interface Accessibility	Convenient web interface for a wide range of users	Often uses interfaces complex for understanding

End of Table 5.2

Adaptability	Accounting for local climatic factors and exogenous variables	Limited adaptation to specific user conditions
Deployment Speed	Possibility of quick deployment thanks to Docker	Often requires individual setup for each user

Thus, the startup is focused on solving the problem of lacking accurate and accessible weather forecasts, which are necessary for effective management in many economic sectors. Agriculture, transportation, energy, and many other sectors critically depend on accurate forecasts for effective functioning and preventing negative consequences. Providing accurate forecasting will reduce risks [9] associated with natural disasters, optimize resource distribution, and improve decision-making processes under changing weather conditions. Using modern machine learning methods allows for creating a competitive product ensuring accuracy and adaptability to diverse conditions and regions, making it relevant and in demand on the market. The startup has significant potential for successful entry into both domestic and international markets, as its innovative approach and technological solutions provide better service quality compared to existing competitors. This creates significant opportunities for developing partnership programs, collaborating with various sectors, and expanding the range of services provided.

5.2 Technological Audit of the Startup Project Idea

A technological audit is an important component for determining how prepared the startup idea is for implementation from a technical point of view. This subsection examines the main technological aspects, tools, and resources necessary for implementing a weather forecasting system based on machine learning.

The proposed system uses modern data analysis methods, machine learning, and visualization, allowing for creating accurate forecasts considering numerous climatic factors and variables. For project implementation, a technology stack is used that combines tools for collecting, processing, analyzing, and visualizing data. The main technologies are Python, Docker, as well as libraries for working with data and creating web interfaces, providing convenient access to system capabilities for users.

Table 5.3 outlines the development technology stack of the startup.

Table 5.3 – Technological Stack of Startup Development

Component	Description and Usage
Python	Main programming language for data processing and modeling.
Pandas, NumPy	Libraries for processing large data volumes.
Statsmodels, Pmdarima	Tools for developing ARIMA, SARIMA, SARIMAX models.
Matplotlib, Seaborn	Libraries for data visualization and building graphs.
Streamlit	Framework for creating an interactive web interface.
Docker	Application containerization for simplifying deployment and scaling.

The development process involves using Python as the main programming language, as it provides a wide range of libraries for data analysis and processing and is convenient for machine learning. Using Pandas and NumPy libraries allows for efficient processing of large volumes of historical weather data, and Statsmodels and Pmdarima help create highly accurate predictive models such as ARIMA, SARIMA, and SARIMAX, ensuring accounting for seasonal and exogenous factors.

Docker is used for system containerization, allowing for easy deployment of software in different environments without complex setups. This makes the system accessible for use on both local machines and remote servers, which is especially important when scaling the project and providing services to a large number of users.

For creating an intuitive user interface, **Streamlit** is used. This framework allows for quickly creating web applications for machine learning and data analysis, which can easily integrate with other system components. In particular, users can upload their data, configure model parameters, launch the forecasting process, and receive results in a convenient format including interactive graphs and tables. Table 5.4 outlines the strengths and weaknesses of the technological approach.

Table 5.4 – Strengths and Weaknesses of the Technological Approach

Parameter	Strengths	Weaknesses
Programming Language (Python)	Large number of libraries for machine learning and data analysis	May be less effective for highly loaded tasks
Containerization (Docker)	Quick deployment and scaling	Requires knowledge of working with containers
ARIMA, SARIMA, SARIMAX Models	High forecast accuracy	Complexity in tuning for different data types
Interface (Streamlit)	Convenient and quick in development	Limited capabilities for complex interface design

Thus, the technological audit shows that the chosen technology stack allows for ensuring stable system operation, high forecast accuracy, and convenient access for users. Using modern tools for containerization, data analysis, and visualization ensures scalability and ease of use, which are key factors for successful startup implementation.

Table 5.5 provides a comparison of the startup's technology stack with competitors.

Table 5.5 – Comparison of the Startup's Technology Stack with Competitors

Component	Our Startup	Competitors
Programming Language	Python, with a large number of libraries	Often proprietary solutions or Java are used
User Interface	Streamlit, convenient web interface	Most competitors offer only API access
Forecasting Methods	ARIMA, SARIMA, SARIMAX	Use of simplified statistical models
Containerization	Docker, for quick deployment	Manual setup depending on environment

The technological audit process allows for evaluating both the strengths and weaknesses of the approaches used and comparing them with market analogues. The main advantage of the system is adaptability and customization capability for each user, while most competitors use simplified approaches and limited interfaces.

Table 5.6 provides an assessment of the effectiveness of the technologies used.

Table 5.6 – Assessment of Effectiveness of Technologies Used

Component	Effectiveness Criterion	Effectiveness Score (1-5)
Python	Ease of use	5
Pandas, NumPy	Speed of processing large data	4
Docker	Deployment speed	5
Streamlit	Convenience for user	4

End of Table 5.6

Statsmodels, Pmdarima	Accuracy of forecasting models	5

The technology stack was chosen to maximize system work efficiency. Python provides significant capabilities for data analysis and processing thanks to numerous libraries like Pandas and NumPy, ensuring fast and effective processing of large data volumes. Containerization with Docker allows for easy adaptation of the system to different environments, ensuring its universality.

Pandas and NumPy significantly simplify work [2] with large datasets, which is critically important for modeling and analyzing weather variables. Docker reduces deployment time and avoids many problems associated with individual environment setup.

For creating a convenient interface, Streamlit is used, providing simplicity and interactivity in user interaction with the system. This allows not only for launching forecasting models but also for quickly changing parameters to obtain different results, which is especially important for users without deep knowledge in machine learning.

Table 5.7 provides a comparison of technological tools by functionality criteria.

Table 5.7 – Comparison of Technological Tools by Functionality Criteria

Tool	Data Collection	Data Processing	Visualization	Interactivity
Pandas	+	++	–	–
NumPy	+	++	–	–
Matplotlib	–	–	++	–
Streamlit	–	+	+	++
Docker	–	–	–	++

Thus, using the chosen technology stack allows for achieving a balance between functionality, efficiency, and user accessibility. The technological audit confirmed that the chosen technologies [21] fully meet project requirements and ensure a high level of performance and forecasting quality. The main advantages are high component integration and easy adaptation to specific needs, making the system competitive in the market.

5.3 Analysis of Market Opportunities for Launching the Startup Project

The market for weather forecasting solutions is currently developing extremely dynamically. Among the main growth drivers are climate changes, increasing demand for accurate weather forecasts for adaptation to new conditions, as well as increased requirements for activity planning in various sectors. A startup focused on high-precision weather forecasting using machine learning has significant market potential, as it can meet the needs of various user segments, including farmers, transportation companies, energy organizations, and private users.

The main product differentiation lies in using ARIMA, SARIMA, and SARIMAX models, ensuring accuracy and adaptability to specific regional conditions. The interactive web interface is also a significant advantage, as it allows users without special technical knowledge to interact with the system, configure forecasting parameters, and receive results in a convenient form.

Table 5.8 outlines market segmentation by target groups.

Table 5.8 – Market Segmentation by Target Groups

Target Group	Needs	Product Solution
Farmers	Accurate forecasts for planning planting, irrigation, and harvesting	Interactive forecast with ability to choose local parameters
Transportation Companies	Safe routes, avoiding weather risks	Forecasting weather conditions for specific routes

End of Table 5.8

Energy Companies	Optimization of production and distribution of energy resources	Demand forecasting based on weather conditions
Private Users	Weather data for daily decisions	Convenient access to daily forecasts via web interface

Table 5.9 provides a comparison of the solution with main competitors on the market.

Competitor Company	Technologies Used	Advantages	Disadvantages
Weather.com	Traditional forecasting models, statistical methods	Wide coverage, accessibility	Lower accuracy of local forecasts
AccuWeather	Proprietary patented algorithms	International coverage	Lack of adaptation to specific user requirements
Proposed Startup	ARIMA, SARIMA, SARIMAX, machine learning	Forecast accuracy, adaptability	Need for significant computational resources

The market for weather forecasting solutions has significant development potential, as demand for quality weather data grows alongside climate changes and the need to adapt to these changes. This system provides competitive advantages as it combines forecast accuracy, interactivity, and ease of use. This is especially important for sectors where weather conditions critically impact activities, such as agriculture, transportation, and energy.

It should also be noted that the weather forecasting market has a large volume of funding from government agencies, private sectors, and scientific institutions, as the need for accurate and fast forecasts [10] has significantly increased. This system can also find applications in various fields, including emergency services, logistics, and construction, where forecasting weather conditions helps ensure safety and minimize costs.

Table 5.10 outlines potential markets for the startup.

Table 5.10 – Potential Markets for the Startup

Application Area	User Needs	Solution Provided
Agriculture	Forecasts for preventing droughts, floods	Timely alerts about weather changes
Energy	Forecasts for balancing energy consumption	Forecasting changes in resource consumption
Construction	Weather forecast for safe work execution	Planning construction works depending on weather
Tourism and Travel	Weather conditions for trip planning	Interactive weather forecasts for travel companies

This startup has the opportunity to capture a significant market share due to forecast accuracy and ability to adapt to specific user needs. The main goal is to provide a convenient tool for various user categories that can be used for both commercial purposes and personal use. Considering the growing importance of accurate weather forecasts, especially under climate change conditions, this product has every chance to become a leader in its segment.

5.4 Development of the Startup Project's Market Strategy

Developing a market strategy is an important stage for successfully launching a startup to market. It includes analyzing target segments, determining main competitive advantages, choosing promotion channels, and developing marketing activities to attract users and maintain their attention. Our startup, focused on high-precision weather forecasting using machine learning, aims to provide its clients with an innovative product that meets modern market requirements.

5.4.1 Target Market Segments

The primary task when developing a market strategy is identifying target market segments. For our startup, the main target groups are farmers, energy companies, transportation operators, travel agencies, and private users. Each of these groups has unique needs that our system can satisfy through interactivity, accuracy, and ease of use.

Table 5.11 outlines market segmentation by target groups.

Table 5.11 – Market Segmentation by Target Groups

Target Group	Needs	Marketing Strategy
Farmers	Accurate forecasts for planning planting, irrigation, and harvesting	Direct advertising at agricultural exhibitions, partnerships with agro-companies
Transportation Companies	Safe routes, avoiding weather risks	Online advertising campaigns, presentations for transportation associations

End of Table 5.11

Energy Companies	Optimization of production and distribution of energy resources	Participation in thematic conferences, strategic partnerships
Private Users	Weather data for daily decisions	Social networks, advertising in mobile applications

5.4.2 Main Promotion Channels

For successful startup promotion, a comprehensive marketing strategy covering several communication channels must be used. The main channels will be:

- Digital Marketing: Using social networks (Facebook, Instagram, LinkedIn) to disseminate product information and attract new users.
- Content Marketing: Creating educational materials, blogs, and videos explaining the benefits of using our product, as well as examples of its application in real conditions.
- Partnerships with Companies: Collaborating with agro-companies, transportation associations, and energy organizations to involve their clients in using our platform.
- Participation in Exhibitions and Conferences: Presenting the product at specialized events concerning agriculture, transportation, and energy.

5.4.3 Development of Marketing Activities

Marketing activities should be aimed at informing users about system capabilities, attracting new clients, and increasing loyalty of existing ones. For this, the following steps are proposed:

1. Advertising Campaigns: Conducting advertising campaigns in social networks and search engines aimed at target audiences.
2. Blogs and Webinars: Conducting online webinars for farmers and

transportation companies, demonstrating system capabilities and its advantages.

3. Free Trial Periods: Offering new users free access to the system during the first month to familiarize themselves with its functionality.

4. User Loyalty: Introducing a bonus system for regular clients actively using the system, and special discounts on advanced features.

Table 5.12 outlines marketing activities for product promotion.

Table 5.12 – Marketing Activities for Product Promotion

Activity	Description	Expected Result
Advertising Campaigns	Advertising in social networks and Google	Attracting new users
Webinars and Blogs	Conducting educational webinars	Increasing awareness about the product
Free Access	Trial period for new clients	Increasing interest in the product
Bonus Program	System of discounts and bonuses	Increasing loyalty of existing users

5.4.4 User Retention and Further Engagement

A key success factor is not only attracting new users but also retaining existing ones. For this, it is necessary:

1. Constant System Updates Regular updates and improvements to functionality, considering user feedback.

2. Technical Support: Providing users with 24/7 technical support for quickly resolving possible problems.

3. Personalization: Adapting the system to individual user needs through personal settings and recommendations.

Table 5.13 outlines user retention strategies.

Table 5.13 – User Retention Strategies

Strategy	Description	Expected Effect
System Updates	Regular improvements to functionality	Satisfying user needs
Technical Support	24/7 support for users	Increasing trust and satisfaction
Personalization	Individual system settings	Increasing convenience and adaptability

Thus, the developed market strategy includes a comprehensive approach to product promotion, ensuring its competitiveness and user retention. Using comprehensive marketing activities and constant product improvement will allow our startup to occupy an important niche in the market of weather forecasting solutions.

5.5 Development of the Startup Project's Marketing Program

Developing a marketing program is an important part of launching a startup to market and its further development. The marketing program is aimed at attracting the target audience, increasing brand awareness, increasing user numbers, and forming loyalty. Our startup, focused on high-precision weather forecasting using machine learning, requires a comprehensive marketing approach that considers the specifics of each target segment.

5.5.1 Marketing Goals and Objectives

The main goals of the marketing program are:

- Attracting New Users: Achieving a high level of user attraction during the first year of product launch to market.
- Increasing Brand Awareness: Creating a recognizable brand associated with

accurate weather forecasts and innovations.

- Forming User Loyalty: Ensuring a high level of user satisfaction through constant improvement of system functionality.

5.5.2 Main Marketing Tools

To achieve the set goals, the following tools will be used in the marketing program:

1. Social Networks: Promotion through social networks such as Facebook, Instagram, LinkedIn, to reach a broad audience and attract potential clients.
2. Content Marketing: Creating quality content including articles, blogs, infographics, and video materials helping users understand the benefits of using our product and its features.
3. SEO and Contextual Advertising: Optimizing the website for search engines (SEO) and conducting contextual advertising to increase product visibility among the target audience.
4. Partnerships with Companies: Developing partnership programs with agricultural companies, transportation associations, and energy organizations to increase market influence.
5. Email Marketing: Sending informational newsletters to potential and existing clients, including announcements of new features, system updates, and special offers.

Table 5.14 outlines marketing tools and their application.

Table 5.14 – Marketing Tools and Their Application

Tool	Description	Expected Result
Social Networks	Disseminating information, product advertising	Attracting new users, increasing brand awareness

End of Table 5.14

Content Marketing	Creating useful and educational content	Increasing trust in the product, informing users
SEO and Advertising	Website optimization and advertising in search systems	Increasing product visibility online
Partnerships	Collaboration with companies	Expanding market opportunities, attracting new users
Email Marketing	Sending news and special offers	Maintaining interest in the product, informing users

5.5.3 Marketing Program Implementation Plan

For successful implementation of the marketing program, a sequence of actions and resource allocation must be ensured. Below are the main stages of marketing program implementation:

- Preparation of Marketing Materials: Developing visual materials, advertising layouts, texts for social networks, and webinars. Timeline: 1 month.
- Launch of Advertising Campaign in Social Networks: Conducting advertising campaigns on Facebook and Instagram aimed at drawing attention to the product. Timeline: 2-3 months.
- Launch of SEO and Contextual Advertising: Optimizing the website and setting up contextual advertising in Google Ads. Timeline: 2 months.
- Conducting Partnership Meetings: Concluding partnership agreements with

agricultural companies and transportation operators. Timeline: 3-4 months.

- Email Marketing Distribution: Introducing regular informational newsletters, as well as campaigns to attract new users. Timeline: continuously.

Table 5.15 outlines the plan for implementing marketing activities.

Table 5.15 – Plan for Implementing Marketing Activities

Indicator	Description	Measurement Method
Number of New Users	Growth of the user base	Analysis of registrations
Brand Awareness	Brand knowledge among the target audience	Surveys and traffic analysis
Conversion	Transforming interested parties into users	Analysis of conversions from advertising campaigns
User Loyalty	Level of satisfaction and repeat visits	Feedback, surveys, and behavior analytics

5.5.4 Pricing Strategy Development

Pricing strategy is an important element of the marketing program directly influencing product attractiveness to end-users. Considering different segments of the target audience, the startup proposes using a flexible pricing system including:

1. Basic Package: Free version with limited functionality, allowing users to test basic system capabilities and receive basic forecasts.

2. Premium Package: Expanded functionality for farmers, transportation companies, and other users needing more detailed forecasts and additional tools.

3. Corporate Package: Individual solutions for large companies with the ability to adapt tools to specific business needs.

Table 5.17 outlines service packages and prices.

Table 5.17 – Service Packages and Prices

Service Package	Functionality	Monthly Price
Basic Package	Basic forecasts, limited data visualization	Free
Premium Package	Extended forecasts, interactive graphs, support	$19.99
Corporate Package	Individual solutions, API access, technical support	Individual agreement

5.5.5. Customer Interaction Strategies

For successful startup implementation to market, effective communication with clients and support at every stage of their interaction with the system is important. The main elements of the interaction strategy are:

- Initial Consultation: Providing information about system capabilities and helping choose the most suitable service package.
- User Support: 24/7 technical support for premium and corporate package users, as well as access to a knowledge base for basic package users.
- Feedback: Conducting regular user surveys to assess their satisfaction and collect suggestions for system improvement.

Table 5.18 outlines customer interaction strategies.

Table 5.18 – Customer Interaction Strategies

Strategy	Description	Expected Result
Initial Consultation	Informing users about the product	Attracting new clients

End of Table 5.18

User Support	24/7 support for premium and corporate users	Increasing user loyalty
Feedback	Surveys and collecting suggestions	Improving functionality and satisfaction

5.5.6. Analysis of Competitive Environment

Analyzing the competitive environment allows for determining the startup's place in the market and developing strategies to achieve competitive advantages. The main competitors are large weather services already having a large user base; however, our startup has a number of differences:

- Interactivity and Personalization: Users can configure model parameters, allowing for adapting forecasting to specific needs.
- Ease of Use: Intuitive web interface created using Streamlit allows for quickly setting up the system without requiring special knowledge.
- Forecast Accuracy: Using ARIMA, SARIMA, SARIMAX models ensuring a high level of forecasting accuracy compared to traditional methods.

Table 5.19 provides a comparative analysis of competitors.

Table 5.19 – Comparative Analysis of Competitors

Competitor Company	Advantages	Disadvantages	Differences of Our Startup
Weather.com	Large audience, accessibility	Lower accuracy of local forecasts	Higher accuracy and model adaptability

End of Table 5.19

AccuWeather	International coverage	Lack of personalization for users	Interactive approach and customization ability
Proposed Startup	Accuracy, interactivity, personalization	Need for computational resources	Unique forecasting models

5.5.7. Risk and Opportunity Assessment

Risk assessment is an important element of developing the marketing program, as it allows for timely preparation for possible difficulties. The main risks include:

Competition with Large Market Players: Risk that large companies may develop similar solutions or improve their current services.

- Need for Large Computational Resources: Using complex forecasting models requires significant resources, which may influence system speed and accessibility for users.
- Dependence on Data Quality: Forecast accuracy depends on input data quality, which may be problematic in case of incomplete or inaccurate data.
- At the same time, the startup has a number of opportunities that may contribute to its success:
- Growing Demand for Accurate Forecasts: Climate changes increase the need for high-precision forecasts for both businesses and private users.
- Lack of Analogues with Interactive Functions: Most existing solutions do not allow users to configure forecast parameters, which is an advantage of our product.

Table 5.20 provides an assessment of risks and opportunities.

Table 5.20 – Assessment of Risks and Opportunities

Factor	Description	Counteraction or Utilization Strategy
Competition	Competition with large market players	Focus on personalization and accuracy
Computational Resources	Need for significant computational power	Optimization of resource usage, partnership with cloud services
Data Quality	Dependence on input data accuracy	Using various data sources, automatic quality checking
Growing Demand	Need for accurate forecasts due to climate changes	Active marketing campaign on climate change topic

5.6. Conclusions for Chapter 5

This chapter has undertaken a rigorous and systematic analysis to formulate a robust commercial foundation for the proposed weather forecasting system, transitioning it from a validated academic prototype to a viable startup venture. The investigation proceeded through a structured sequence of strategic assessments, beginning with a granular analysis of the market landscape and concluding with the development of actionable plans for market penetration and growth.

The initial phase involved a comprehensive market opportunity analysis, which served to deconstruct the heterogeneous landscape of weather forecasting consumers. Through detailed segmentation, distinct user archetypes were identified, including agricultural enterprises, logistics operators, energy sector managers, and informed private individuals, each with quantifiable needs rooted in risk mitigation, operational optimization, and planning certainty. This analysis confirmed a clear market gap for a solution offering not merely data, but actionable, customized insights derived from advanced analytics.

In response, a multi-faceted market strategy was architected. This strategy is predicated on a phased approach to market entry, prioritizing early adopters in sectors with the most acute sensitivity to forecast accuracy, such as precision agriculture. The promotional framework integrates both inbound and outbound methodologies, combining educational content marketing to build authority and address sophisticated user queries with targeted outreach through industry-specific channels. The developed marketing program operationalizes this strategy, detailing tactical campaigns, key performance indicators for channel efficacy, and a content calendar designed to narrate the startup's technological differentiation.

A value-based pricing strategy was formulated to align with the identified segmentation and perceived product value. The proposed tiered subscription model, encompassing a premium base package, a professional premium tier, and customizable enterprise solutions, is designed to maximize market access while capturing the full economic value delivered to commercial clients. This structure facilitates low-friction user acquisition while establishing clear pathways for revenue scaling.

A candid analysis of the competitive environment and a formal risk assessment further solidified the strategic plan. The startup's core differentiators were crystallized as: 1) the algorithmic sophistication of its machine learning engine (ARIMA/SARIMA/SARIMAX), 2) the unique interactivity and personalization afforded by its user interface, and 3) the operational agility enabled by its modern, containerized technology stack. Concurrently, potential risks (including computational resource demands, data dependency, and competitive response) were identified, and proactive mitigation strategies, such as exploring cloud partnerships and continuous model iteration, were delineated.

In synthesis, this chapter demonstrates that the startup's pathway to securing a significant niche within the weather solutions market is not serendipitous but strategically engineered. The confluence of a demonstrably superior technical product (as validated in Chapters 3 and 4) with a meticulously crafted commercial strategy creates a compelling and defensible business proposition. The integrated plan outlined herein provides a coherent blueprint for acquiring users, delivering continuous value, and building sustainable loyalty, thereby establishing the necessary conditions for the venture's successful development and long-term growth in an increasingly data-driven economy.

6 OCCUPATIONAL SAFETY AND HEALTH

Occupational safety and health issues are considered for the design and development phase of climate data analysis and visualization system.

Occupational safety is a system of legal, socio-economic, organizational and technical, sanitary and hygienic and treatment and prevention measures and tools aimed at preserving human life, health and ability to work. Working conditions at the workplace, safety of technological processes, machines, mechanisms, equipment and other means of production, condition of collective and individual protection means used by the employee, as well as sanitary and living conditions must meet the requirements of the law. An employee has the right to refuse the assigned work if a work situation has arisen that is dangerous to his life or health or to the people around him, or to the work environment or the environment. He must immediately notify his immediate supervisor or employer. The existence of such a situation is confirmed, if necessary, by labor protection specialists of the enterprise with the participation of a representative of the trade union of which he is a member or a person authorized by employees on labor protection (if the trade union was not established), as well as an insurance expert [12]. The task of labor protection is to minimize injuries and illnesses of the employee while ensuring comfort with maximum productivity. The main objectives of labor protection are the formation of specialists with the necessary knowledge and practical skills on legal and organizational issues of labor protection, industrial sanitation, safety, fire safety.

6.1. General characteristics of the room and workplace

The development of the analysis and visualization system is performed in a room located on the fourth floor of an eight-storey building with general and local lighting. The room has one-sided lighting, the windows are oriented to the east, the windows have shutters. White ceiling with a reflection coefficient of 0.7, light brick walls with a reflection coefficient of 0.5. There are 4 people working in the room, in accordance with this we obtain input data for the analysis of potentially dangerous and harmful production factors, which are given in table. 4.1.

Table 6.1 – Incoming data

Room parameters	**Value**
Length x width x height	6.6 x 6.1 x 2.7 m
Area	40.26m²
Volume	108,70 m³
Workplace number	**Specifics of work**
I workplace	Front-end programmer (web application client development specialist)
II workplace	Back-end programmer (specialist in the development of the server part of web applications and database design)
III workplace	Business analyst (also acts as a product manager)
IV workplace	UI-UX web designer
Technical means (quantity)	**Name and characteristics**
Monitor (4 pcs.)	HP 22Xi / 21.5 "/ 1920x1080px / IPS
Computer (4 pcs.)	HP ProBook 440 G6, 14 "IPS screen (1920x1080) Full HD, Intel Core i7-8565U (1.8 - 4.6 GHz) / RAM 16 GB / SSD 256 GB
Floor cooler (1 piece)	CRYSTAL YLR3-5V208
Air conditioner (1 piece)	DEKKER DSH105R / G / 26m2 / 2,65kW- 2.9 kW / 25x74.5x19.5 cm / 9 kg
General purpose luminaries (3 pcs.)	The lamp raster built-in 4x18W
Local lamps (4 pcs.)	Delux Decor TF-05/1 x 40W

According to NPAOP 0.00-7.15-18, the area S 'allocated for one workplace with a personal computer must be at least 6 m2 and the volume - at least 20 m3. There are 4 workplaces in the room, which fully meets the required standards.

We calculate the actual values of these indicators by dividing the volume of the room and the total area by the number of employees.

Therefore, based on the results obtained in terms of area and volume, the room meets the standards.

Table 6.2 – Workplace characteristics

№	The name of the parameter	Value	
		in fact	Normative
1.	Height of a working surface, mm	780	680 – 800
2.	Width of a working surface, mm	1500	not less than 600
3.	Depth of a working surface, mm	750	not less than 600
4.	Height of space for legs, mm	750	not less than 600
5.	Width of space for legs, mm	800	not less than 500
6.	Depth of space for legs, mm	750	not less than 450
7.	Seat surface height, mm	480	400 – 500
8.	Seat width, mm	500	not less than 400
9.	Seat depth, mm	500	not less than 400
10.	Height of a basic surface of a back, mm	550	not less than 300
11.	Width of a surface of a back, mm	470	Not less than 380
12.	Length of armrests, mm	300	not less than 250
13.	Width of armrests, mm	60	50 – 70
14.	Distance from eyes to the screen, mm	650	600 – 700

It is possible to draw a conclusion that the sizes of a workplace of the programmer correspond to the established norms, proceeding from the set parameters.

6.2 Analysis of potentially dangerous and harmful production factors in the workplace

When creating a system of analysis and visualization, the work is performed sitting without physical effort, so it belongs to the category of light Ia.

Premises for work must be equipped with heating, air conditioning or supply and exhaust ventilation in accordance with DBN B.2.5-67: 2013. Normalized parameters of the microclimate, ionic composition of air, content of harmful substances meet the requirements of LTO 3.3.6.042-99, GN 2152-80, GOST 12.1.005-88, DSTU GOST 12.0.230: 2008 and DSTU GOST 12.4.041: 2006. Ventilation is understood as a set of measures and means designed to ensure meteorological conditions and cleanliness of the air environment that meet hygienic and technical requirements at permanent places and service areas. The main task of ventilation is to remove polluted, humid or heated air from the room and supply clean fresh air.

The sources of noise in the room are the fan of the system unit, laptop and air conditioner. The sound generated by the fan and air conditioner can be classified as constant.

According to DBN B.2.5-28: 2018 the work belongs to the category of visual works. The use of natural, artificial and mixed lighting is envisaged.

The computer is a single-phase consumer of electricity powered by 220V AC from a network with grounded neutral. IBM PC refers to electrical installations up to 1000V closed version; all conductive parts are in the casings. According to the method of protecting a person from electric shock, computers and peripherals must meet 1 class of protection.

Technical methods of protection against electric shock is reduced to the use of current of safe voltage, protection in case of accidental touching current-carrying parts and against excessive currents, protection in case of voltage transfer to non-current-carrying metal parts of the installation.

Safe voltage is obtained from the high voltage grid (110-120 V) by means of step-down transformers.

Protection against contact with live parts of the installation is achieved by means of insulation, fencing off the use of blocking safety devices and inaccessibility of the location of the installations.

Switchboards are placed in closed metal casings-boxes.

Safety alarm is used in the form of posters and inscriptions. The best light alarms are double, which in the presence of voltage lights a red light, and in its absence - green.

Protection against excessive currents - short circuits and overload currents, which can cause insulation to ignite, is provided by fuses and circuit breakers, and protection against voltage transfer to live parts by means of protective earthing and protective disconnection.

Fire prevention is achieved by eliminating the formation of sources of ignition and combustible environment.

Fires of the following classes are possible in this room: A - combustion of solids, E - combustion of live electrical installations.

CONCLUSIONS

As a result of the conducted work, a comprehensive study of modern approaches to weather forecasting was carried out, and an interactive forecasting system based on machine learning technologies was implemented. This allowed for ensuring a high level of forecast accuracy, which is critically important for many spheres of activity such as agriculture, transportation, energy, and tourism.

The conducted analysis of existing solutions for weather forecasting showed the limitations of traditional methods, particularly their insufficient accuracy and complexity in processing large data volumes. Using modern machine learning models such as ARIMA, SARIMA, and SARIMAX allowed for overcoming these limitations and increasing forecast accuracy, particularly by accounting for seasonal and nonlinear relationships between data.

Historical data on weather conditions were collected and analyzed, based on which forecasting models were trained. This allowed for creating a reliable foundation for building accurate forecasts considering dependencies between various meteorological parameters. Using Python and its libraries such as Pandas, NumPy, Matplotlib, Seaborn, as well as Streamlit for creating a web interface, ensured system convenience and accessibility for users.

The developed system was supplemented with interactive elements such as graphs, tables, and other visual means contributing to easy analysis of obtained results. This makes the system useful for various user categories, from scientists and specialists to ordinary users needing accurate weather forecasts for planning their activities.

Conducted system testing showed its high accuracy and reliability. Several indicators were used, such as the coefficient of determination (R^2), to assess model quality, allowing for selecting the most optimal approaches for further use. Testing results confirmed the system's ability for high-precision forecasting, as well as its ability to adapt to different conditions.

Based on the developed system, a startup project was proposed with significant potential for commercial success. A market strategy and marketing program were developed aimed at attracting various user categories and ensuring their loyalty. Using modern marketing tools such as social networks, content marketing, partnerships with companies, allows for effectively presenting the product to the market and occupying its niche.

The final stage of work was system testing and its implementation into practical use. The system demonstrated high forecasting quality and ease of use, confirming its effectiveness and expediency for further development. Prospects for further system improvement include integrating new data sources such as satellite imagery and applying more complex machine learning models, allowing for even greater increase in forecasting accuracy.

Thus, the results of the performed work confirm the effectiveness of using modern machine learning technologies for weather forecasting and demonstrate the significant potential of the developed system for commercial use and further development. The system is a convenient tool for obtaining accurate forecasts that can be used in various sectors for making informed decisions.

LIST OF REFERENCES

1. Electricity price statistics. Statistics Explained. 2024. P. 2–3. URL: https://ec.europa.eu/eurostat/statistics-explained/SEPDF/cache/45239.pdf (date of access: 01.10.2024).

2. Impact of forecasting on energy system optimization / F. Peterssen et al. Advances in Applied Energy. 2024. Vol. 15. 100181. URL: https://doi.org/10.1016/j.adapen.2024.100181 (date of access: 22.10.2024).

3. Kolambe M., Arora S. Forecasting the Future: A Comprehensive Review of Time Series Prediction Techniques. Journal of Electrical Systems. 2024. Vol. 20, no. 2s. P. 575–586. URL: https://doi.org/10.52783/jes.1478 (date of access: 22.10.2024).

4. Shumway R. H., Stoffer D. S. ARIMA Models. Time Series: A Data Analysis Approach Using R. Boca Raton: CRC Press, Taylor & Francis Group, 2019. P. 99–128. URL: https://doi.org/10.1201/9780429273285-5 (date of access: 22.10.2024).

5. Comparison of SARIMAX, SARIMA, modified SARIMA and ANN-based models for short-term PV generation forecasting / S. I. Vagropoulos et al. 2016 IEEE International Energy Conference (ENERGYCON), Leuven, Belgium, 4–8 April 2016. 2016. URL: https://doi.org/10.1109/energycon.2016.7514029 (date of access: 22.10.2024).

6. Maulud D., Abdulazeez A. M. A Review on Linear Regression Comprehensive in Machine Learning. Journal of Applied Science and Technology Trends. 2020. Vol. 1, no. 4. P. 140–147. URL: https://doi.org/10.38094/jastt1457 (date of access: 22.10.2024).

7. Box G. E. P., Jenkins G. M., Reinsel G. C., Ljung G. M. Time Series Analysis: Forecasting and Control. 5th ed. Hoboken: John Wiley & Sons, 2015. 712 p.

8. Hyndman R. J., Athanasopoulos G. Forecasting: Principles and Practice. 3rd ed. OTexts, 2021. URL: https://otexts.com/fpp3/ (date of access: 22.10.2024).

9. Brockwell P. J., Davis R. A. Introduction to Time Series and Forecasting. 3rd ed. Springer, 2016. 425 p.

10. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. Springer, 2009. 745 p.

11. Hamilton J. D. Time Series Analysis. Princeton University Press, 1994. 799 p.
12. Box G. E. P., Jenkins G. M. Some Recent Advances in Forecasting and Control. Applied Statistics. 1970. Vol. 19, no. 1. P. 91–109.
13. De Gooijer J. G., Hyndman R. J. 25 Years of Time Series Forecasting. International Journal of Forecasting. 2006. Vol. 22, no. 3. P. 443–473.
14. Makridakis S., Wheelwright S. C., Hyndman R. J. Forecasting: Methods and Applications. 3rd ed. John Wiley & Sons, 1998. 656 p.
15. Bishop C. M. Pattern Recognition and Machine Learning. Springer, 2006. 738 p.
16. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2016. 775 p.
17. Montgomery D. C., Jennings C. L., Kulahci M. Introduction to Time Series Analysis and Forecasting. 2nd ed. Wiley, 2015. 472 p.
18. Tibshirani R., Friedman J. H., Hastie T. Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software. 2010. Vol. 33, no. 1. P. 1–22.
19. Tsay R. S. Analysis of Financial Time Series. 3rd ed. Wiley, 2010. 720 p.
20. Wei W. W. S. Time Series Analysis: Univariate and Multivariate Methods. 2nd ed. Addison-Wesley, 2006. 618 p.
21. Li M., Fu G., Zhang Y. Short-Term Wind Speed Forecasting Using SARIMA Model. Renewable Energy. 2022. Vol. 187. P. 591–600.
22. Engle R. F., Granger C. W. J. Co-Integration and Error Correction: Representation, Estimation, and Testing. Econometrica. 1987. Vol. 55, no. 2. P. 251–276.
23. Jolliffe I. T. Principal Component Analysis. 2nd ed. Springer, 2002. 487 p.
24. Walker G. T. On Periodicity in Series of Related Terms. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character. 1931. Vol. 131, no. 818. P. 518–532.
25. Box G. E. P., Pierce D. A. Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models. Journal of the American Statistical Association. 1970. Vol. 65, no. 332. P. 1509–1526.
26. Brockwell P. J., Davis R. A. Time Series: Theory and Methods. 2nd ed. Springer, 1991. 577 p.
27. Gelman A., Hill J. Data Analysis Using Regression and Multilevel/Hierarchical

Models. Cambridge University Press, 2006. 625 p.

28. Cressie N. Statistics for Spatial Data. Revised ed. Wiley, 1993. 900 p.

29. Fuller W. A. Introduction to Statistical Time Series. 2nd ed. Wiley, 1996. 698 p.

30. Enders W. Applied Econometric Time Series. 4th ed. Wiley, 2014. 496 p.

31. Box G. E. P., Tiao G. C. Intervention Analysis with Applications to Economic and Environmental Problems. Journal of the American Statistical Association. 1975. Vol. 70, no. 349. P. 70–79.

Appendix A


EUROPEAN CONFERENCE
Conference
Proceedings
III International Science Conference
«Technology development: shaping modern
thinking and scientific approaches»
January 19-21, 2026
Krakow, Poland

TABLE OF CONTENTS

МОДЕЛЮВАННЯ ТА АНАЛІЗ БЕЗПАРОЛЬНОЇ АВТЕНТИФІКАЦІЇ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГНОЗУВАННЯ ТА ВИЯВЛЕННЯ АНОМАЛІЙ У ПОВЕДІНЦІ КОРИСТУВАЧІВ

Голотенко Олександр Сергійович
кандидат технічних наук, доцент
Тернопільський національний технічний університет імені Івана Пулюя

Акінємі Віктор Олувасеї
здобувач другого (магістерського) рівня вищої освіти, 6 курс
Тернопільський національний технічний університет імені Івана Пулюя

Касонго Валері Бванга
здобувач другого (магістерського) рівня вищої освіти, 6 курс
Тернопільський національний технічний університет імені Івана Пулюя

Паролі традиційно слугують основним засобом автентифікації, проте вони стали «слабкою ланкою» кібербезпеки. За даними звіту Verizon, до 81% випадків зламів пов'язані з використанням слабких або викрадених паролів [1]. Це стимулює перехід до безпарольної автентифікації, яка усуває залежність від статичних секретів (паролів) та мінімізує ризики, пов'язані з людським фактором. Безпарольні методи (біометричні дані, криптографічні ключі, одноразові коди тощо) пропонують підвищену безпеку і зручність для користувачів, усуваючи необхідність спільних секретів при вході в систему [2]. Однак впровадження таких методів у масштабі організації створює нові виклики – наприклад, як впевнитись, що автентифікований без пароля користувач дійсно є тим, за кого себе видає, протягом усього сеансу. Тут на допомогу приходять методи штучного інтелекту та машинного навчання. Застосування аномалійного моніторингу поведінки користувачів дає змогу постійно валідувати особу користувача після початкової автентифікації, гарантуючи, що доступ підтримується лише для легітимного користувача [3]. Таким чином, поєднання безпарольної автентифікації з алгоритмами машинного навчання для виявлення аномалій здатне суттєво підвищити рівень безпеки систем автентифікації.

Безпарольна автентифікація: концепції та стандарти

Безпарольна автентифікація – це підхід, за якого користувач отримує доступ до системи без введення звичного пароля. Натомість використовуються інші фактори автентифікації, наприклад біометрія (відбиток пальця, розпізнавання обличчя), апаратні токени або одноразові коди. Одним із найбільш відомих сучасних стандартів є FIDO2 (Fast Identity Online 2), розроблений альянсом FIDO спільно з консорціумом W3C. Стандарт FIDO2 базується на криптографії з відкритим ключем: під час реєстрації генерується пара ключів (приватний зберігається на пристрої користувача, публічний – на сервері), і надалі для входу сервер надсилає випадковий «виклик», який підписується приватним ключем на пристрої користувача [4]. Замість пароля користувач підтверджує свою особу тим самим способом, що й розблоковує свій пристрій – наприклад, відбитком пальця, сканом обличчя або PIN-кодом [4]. Такий підхід усуває ризики фішингу, повторного використання та перехоплення паролів, оскільки жодні секрети не передаються і не зберігаються на сервері [5]. В результаті безпарольна автентифікація забезпечує вищий рівень захисту облікових записів і покращує користувацький досвід (немає потреби запам'ятовувати чи регулярно змінювати паролі).

Попри переваги, безпарольна автентифікація не усуває всіх можливих загроз. Залишається ризик скомпрометувати самі пристрої або токени користувачів, викрасти їх або обійти біометрію. Також зловмисник може спробувати отримати доступ, навіть якщо автентифікація пройдена (наприклад, використовуючи автоматизовані сесії або атакуючи вже залогінених користувачів). Тому виникає потреба у додатковому рівні моніторингу після автентифікації – зокрема, шляхом аналізу поведінкових факторів користувача. Саме тут доречним є застосування методів машинного навчання для виявлення аномалій у поведінці в режимі реального часу.

Машинне навчання для виявлення аномалій поведінки користувачів

Аномалія поведінки – це відхилення у діях користувача від притаманного йому звичного патерну. Системи виявлення аномалій на основі машинного навчання навчаються розуміти, що є «нормальною» поведінкою для конкретного користувача або групи користувачів, і сигналізують при виявленні суттєвого відхилення від цієї норми. Такий підхід належить до концепції UEBA (User and Entity Behavior Analytics) – аналітики поведінки користувачів та сутностей, яка поєднує статистичні методи і алгоритми ML для фіксації нетипових дій.

Для виявлення аномалій використовують різні підходи машинного навчання залежно від наявності даних та типових сценаріїв атак. Коротко розглянемо основні методи та алгоритми:

• Наглядове навчання (supervised): модель тренується на заздалегідь розмічених даних, де відомо, які дії були нормальними, а які – зловмисними.

• Безнаглядове навчання (unsupervised): алгоритми самостійно виявляють структуру в даних, групуючи схожі сеанси та вирізняючи ті, що не належать до жодної групи.

• Напівнаглядове навчання: комбінує два підходи – модель вчиться переважно на нормальних (немічених) даних, маючи лише невелику кількість відомих аномальних прикладів.

• Глибоке навчання: нейронні мережі, зокрема рекурентні (RNN, LSTM), використовуються для аналізу послідовностей дій користувача у часі.

Конкретні алгоритми, що зарекомендували себе для задач виявлення аномалій у користувацькій активності, включають Isolation Forest, One-Class

SVM, Autoencoder (автоенкодери) та методи кластеризації на кшталт DBSCAN і k-means.

Поведінкові характеристики, що аналізуються, залежать від доступних даних системи автентифікації. Зазвичай враховуються такі фактори, як: географічне розташування входу, час доби та день тижня активності, частота та тривалість сеансів, тип і стан пристрою, з якого здійснено вхід, перелік дій або ресурсів, до яких звертається користувач, швидкість і ритм введення даних тощо. На основі цих даних будується профіль нормальної поведінки. Як зазначають дослідники, ефективним є підхід побудови "відбитку сесії" (session fingerprint) – агрегованого профілю, що характеризує поведінку користувача протягом сеансу або серії сеансів. Надалі аномалії визначаються як відхилення від цього профілю: наприклад, якщо користувач зазвичай працює у офісі з корпоративного ноутбука вдень, то спроба доступу вночі з іншої країни з незнайомого пристрою буде значущою аномалією. Система UEBA, впроваджена на рівні служби ідентифікації, може аналізувати такі "відбитки" у реальному часі та надсилати тривожні сповіщення при виявленні нетипової поведінки.

Моделювання інтегрованої системи автентифікації

На основі розглянутих технологій можна запропонувати теоретичну модель інтегрованої системи, що поєднує безпарольну автентифікацію та аномалійну аналітику поведінки. Ця система працюватиме у кілька етапів:

1. Початкова автентифікація без пароля. Користувач проходить автентифікацію за допомогою вибраного безпарольного механізму – наприклад, використовуючи апаратний ключ FIDO2 або біометричний фактор через протокол WebAuthn. На цьому етапі перевіряється криптографічний підпис виклику, що гарантує справжність фактору (ключа або біометрії) користувача [4]. Якщо перевірка пройдена, користувач отримує доступ до системи (починається сесія).

2. Моніторинг поведінки та збір даних. Після входу система починає збір телеметрії про дії користувача у сесії. Збираються такі дані, як час та тривалість активності, IP-адреса та геолокація, інформація про пристрій та браузер, список ресурсів або функцій, до яких звертається користувач, та інші поведінкові характеристики (наприклад, динаміка введення даних). Ці дані використовуються для формування поведінкового профілю сеансу.

3. Аналіз та прогнозування аномалій (машинне навчання). Зібрані дані надходять до модуля аналізу, де ML-модель оцінює, наскільки поточна поведінка відповідає нормальній для цього користувача (або для подібних користувачів). Модель може бути, скажімо, нейронною мережею або ансамблем алгоритмів (Isolation Forest + кластеризація), натренованих на попередніх сесіях. В реальному часі обчислюється метрика «ризику» або аномальності сесії. Аномалія прогнозується, якщо метрика виходить за поріг: це означає, що поточна поведінка статистично малоймовірна і може вказувати на загрозу (викрадений токен, дії зловмисника тощо).

4. Реакція та прийняття рішень. Якщо виявлено аномальну поведінку, система в режимі реального часу виконує наперед визначені дії. Можливі реакції:

запит повторної автентифікації, обмеження доступу до окремих чутливих ресурсів, повне завершення сеансу та сповіщення служби безпеки. В разі незначних відхилень може застосовуватися step-up authentication – користувачу надсилається запит підтвердити особу додатково, тоді як при критичних аномаліях сесію негайно блокують. Паралельно інцидент логуються для подальшого аналізу. Якщо ж поведінка в межах норми, користувач продовжує роботу без перешкод, і система невпинно навчається – оновлює профіль новими даними, підлаштовуючи модель під еволюцію поведінки.

Такий підхід реалізує концепцію «Zero Trust» – ніколи не довіряти повністю, навіть після успішного входу. Безперервна автентифікація на основі поведінкових ознак дозволяє ловити атакуючих вже всередині системи, коли вони імітують легітимних користувачів.

Важливо зазначити, що при моделюванні такої системи слід врахувати баланс між чутливістю виявлення та кількістю хибних спрацювань. Надто сувора модель може позначати аномалію там, де її насправді немає (наприклад, відрядження користувача в іншу країну), що призведе до зайвих блокувань і скарг. Натомість надто толерантна модель може пропустити реальну атаку. Для вирішення цієї проблеми використовують кілька підходів. По-перше, моделі регулярно перенавчають на актуальних даних, щоб вони враховували зміни у поведінці користувачів і нові типи атак. По-друге, впроваджують багаторівневий аналіз: автоматичний алгоритм виявляє сирі аномалії, але остаточне рішення приймається з урахуванням додаткового контексту або експертної оцінки. Зокрема, у новітніх наукових підходах пропонується після автоматичного класифікатора (наприклад, Isolation Forest + DBSCAN для групування аномальних сесій) застосовувати аналіз спеціаліста з безпеки: експерт переглядає кластеризовані аномальні сесії і допомагає відрізнити справді небезпечні інциденти від помилкових спрацювань. Така комбінована людино-машинна модель дозволяє досягти високої точності

Висновки

Перехід до безпарольної автентифікації є сучасною відповіддю на проблеми, пов'язані зі слабкими паролями і людським фактором у безпеці. Стандарти на кшталт FIDO2 демонструють, що можна забезпечити зручний та надійний вхід користувачів без використання пароля, позбавивши зловмисників улюбленої цілі – крадіжки або відгадування секретної фрази. Утім, впровадження безпарольних рішень потребує доповнення механізмами інтелектуального аналізу поведінки. Методи машинного навчання, інтегровані в систему автентифікації, дозволяють прогнозувати та виявляти аномалії на основі поведінкових патернів користувачів. Це суттєво підсилює захист: навіть якщо зловмисник обійде початкову автентифікацію (або отримає доступ до токена), його нетипова діяльність буде помічена і зупинена.

Розроблена теоретична модель демонструє, як можна поєднати переваги безпарольних технологій (відсутність паролів, фішингостійкість) з потужністю AI-систем для безперервного моніторингу. Машинне навчання автоматично будує профілі нормальноЇ поведінки і оновлює їх зі зростанням обсягів даних, тим самим адаптуючи безпекові політики до кожного користувача. Така адаптивна система здатна виявити навіть ті загрози, що раніше не зустрічалися, забезпечуючи проактивний захист (прогнозування інцидентів до їхнього розвитку).

Список літератури

1. Dev Kumar (2025, June 20). Machine Learning for Anomaly Detection in IAM, Passwordless, Threat, and Breach Scenarios. MojoAuth. URL: https://mojoauth.com/ciam-101/machine-learning-anomaly-detection-iam-passwordless-security

2. Portnox (2025, June 5). Passwordless Authentication and AI: A Look at Emerging Technologies. Portnox Blog. URL: https://www.portnox.com/blog/network-security/passwordless-authentication-and-ai-a-look-at-emerging-technologies/

3. Martín, A. G., Beltrán, M., Fernández-Isabel, A., & Martín de Diego, I. (2021). An approach to detect user behaviour anomalies within identity federations. Computers & Security, 108, 102356. https://doi.org/10.1016/j.cose.2021.102356

4. Lindemulder, G., & Kosinski, M. (n.d.). What is FIDO2? IBM Think Blog. URL: https://www.ibm.com/think/topics/fido2

5. Perapu, P. (2025). Anomaly Detection in User Behaviour Using Machine Learning for Cloud Platforms. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 11(3), 805-809.