**ABSTRACT**

Weather prediction aims to help people and minimize weather-related damage and improve overall well-being. It helps ensure public health and safety and supports economic growth. Weather prediction plays a crucial role in various sectors. It includes agriculture, transportation, energy management, and biological field. Nowadays methods of weather forecasting have limitations in accurately capturing the tough dynamics of the atmosphere.

In recent years, the collaboration of machine learning techniques has shown promising results in improving the accuracy and solidity of weather predictions. This survey report provides an overview of the application of machine learning in weather prediction, including the techniques used, data sources put to use, Advanced mathematics applied, and real-world applications. Additionally, the report discusses the challenges faced in leveraging machine learning for weather forecasting and explores emerging trends and future directions in the field. Through case studies and a literature review, this report highlights the significant potential of machine learning in revolutionizing weather prediction, ultimately contributing to more precise and actionable forecasts for societal benefit.

Earlier the models were made up using a high number of nodes and they used to consume a very high amount of energy to predict weather conditions but our model predict it in very short periods using modern advanced mathematics and calculation. We collect data from the neighboring regions and train our model on that data keeping in mind that RMSE(Root Mean Square Error) is minimized.

In this paper, we will predict the temperature by feeding weather features along with the temperature that comes under **Supervised Learning** as the feature we want to predict is present in the dataset.

In Temperature prediction –**Linear Regression Algorithm**, **Random Forest Algorithm, Polynomial Regression Algorithm and Support Vector Machine Algorithm (SVM)** are used.

The results of these models are compared based on the Root Mean Squared Error (RMSE) between the actual and predicted values. Graphs are plotted for each model to provide a clearer depiction of accuracy.

Python, NumPy, Anaconda Navigator, Jupiter Notebook, Matplotlib, Scikit-learn, and Panda will be used in this project. After the research, we have found that the **Random Forest Regression** performs best in weather prediction with an accuracy of 86%.

Keywords: *Machine Learning,Regression, Forecasting techniques, challenges, applications, case studies, Real-world impact, Support Vector Machine, Random Forest, Supervised Learning.*

**1. INTRODUCTION**

Predicting the weather conditions is challenging. It has been a critical aspect of human life that influences various sectors such as transportation, agriculture, and natural disaster management. Being able to predict weather conditions of the coming days will be helpful for us to protect lives and our economic damage, reliable crop bowing, and events related to weather.

In recent years, Machine Learning models have trained through very complex mathematical equations, which results in weather prediction with good accuracy. Machine learning is a subset of Artificial intelligence. It involves the algorithm and statistics to draw the graph and analyze it.

ML reads the historical weather data and patterns by itself. This survey report will explore the application of machine learning techniques in predicting weather conditions. It will cover all the algorithms used in weather forecasting, advantages, and disadvantages. In this model, we will train the historical weather data of a region and then test it. We will find accuracy by predicting the data through each model set and the model set providing the maximum accurate data will be taken into account.

**2. LITERATURE SURVEY**

**2.1 PREVIOUS WORK**

Traditionally, weather predictions are performed with the help of large complex models of physics, which utilize different atmospheric conditions over a long period. These conditions are often unstable because of perturbations of the weather system, causing the models to provide inaccurate forecasts. In recent years, weather prediction has seen a variety of approaches. However, many of these fail to capture the complex relationships between various factors that affect weather.

**2.2 METHODOLOGY**

The following steps were performed to implement the Machine Learning model:

**1. Setup**

**2. Data Collection**

**3. Data preprocessing**

**4. Splitting dataset into Train and Test Data**

**5. Model Deployment.**

**6. Comparison of Model.**



**2.2.1 SETUP**

**1. SOFTWARE USED**

* **Anaconda Navigator:** Anaconda Navigator is a desktop graphical user interface (GUI) included in the Anaconda distribution, which allows users to easily manage conda packages, environments, and launch applications without the need to use command line commands.
* **Jupyter Notebook:** It is an open-source web-based interactive computing notebook environment. It is widely used in data science, machine learning, and scientific computing for interactive data analysis, and visualization.

**2. LANGUAGE USED**

* **Python 3.12** : Python is an interpreted, high-level, general-purpose programming language known for its clear and readable syntax. It offers a vast library ecosystem, making it highly suitable for data mining and predictive analytics.

**3. LIBRARIES USED**

* **Numpy**: For numerical computations and handling arrays.
* **Pandas**: For data manipulation and analysis.
* **Matplotlib**: For data visualization.
* **Seaborn:** For statistical data visualization.
* **Scikit-learn:** For implementing machine learning algorithms and models.
  + 1. **DATA COLLECTION**

To implement the Machine Learning model, we collected weather dataset of Kolkata city from **Kaggle.com.** The dataset consists of several attributes such as precipitation, wind speed, maximum and minimum temperature, humidity, wind direction, etc. The dataset has been downloaded in a CSV file. The weather data was collected in Kolkata from 2017 to 2022.

2.2.3 **DATA PREPROCESSING**

This involves transforming raw, unorganized, and unstructured data into a structured, usable format. The steps are as follows:

**1.Data Cleaning**

Data cleaning is crucial as datasets can be messy, with empty values, missing or additional values, invalid fields, etc.

The Data in the Excel sheet was incomplete, with many empty values and columns, text needed to be mapped with numbers before passing the dataset through different models.

Wind Chill, Wind Gust, Info, and snow depth columns are have significant numbers of NaN values, hence we dropped these columns.

And dropping the rows for features having lesser Null values.

**2. Feature Selection**

The data we have collected has many unwanted attributes which will not be needed in our project. Hence, we will select the significant features for temperature and rainfall prediction .

**3. Data Normalization**

The goal of normalization is to change the values of numeric columns in the dataset to a common scale without affecting differences in the ranges of values or losing any information. Normalization helps in faster training of models. We normalized the dataset, except for the temperature, to a 0-1 range.

**2.2.4 TRAINING AND TESTING OF DATA**

Once everything is done, we will split the data in train and test set.

● Training dataset (usually 80%) is used to train the machine learning model and to get the best-fit shape.

● Testing dataset (usually 20%) is used to test the model formed.

This process of feeding data helps in knowing the accuracy of the model. The data cleaning and all these processes are carried out before separating data for training and testing

**Code to Test and Train Data**



**2.2.5 ALGORITHMS USED**

**1. Linear Regression Algorithm:**

Linear Regression can be simple or multiple, Linear Algorithm Regression is a method of modeling a target value based on independent prediction. This method is mostly used for forecasting and finding out cause and effect relationship between variables.

The first method we used for temperature prediction is Multiple Linear Regression, a Supervised learning algorithm that derives a relationship between a dependent variable and one or more independent variables. The variable to be predicted is called the **dependent variable**, while the variables used for prediction are called **independent variables**. In our case, we aim to predict the apparent temperature based on some features like precipitation, wind speed, relative humidity visibility etc. Linear regression fits a straight line that minimizes the discrepancies between predicted and actual output values using a “least squares” method to discover the best-fit line.

**y = mx+ C**

Where ‘y’ is a dependent variable and ‘x’ is an independent variable.

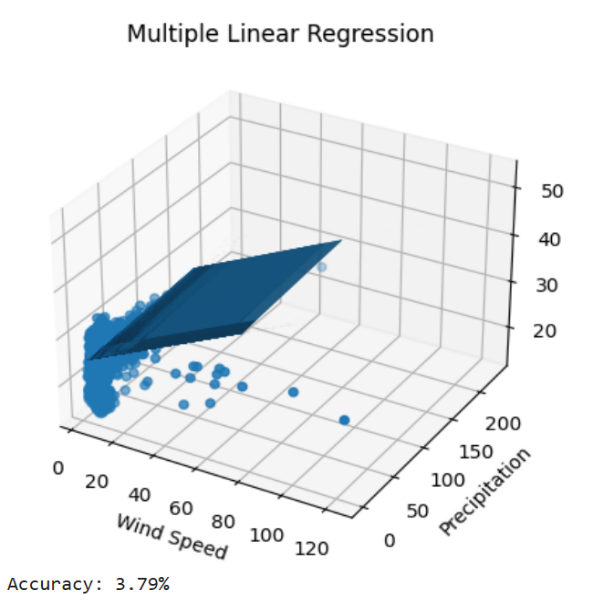
M is the slope but there are certain disadvantages of using simple linear regression

1. The target depends on one attribute, while the weather depends on many factors.

2. The relationship between the factor and target does not change linearly.

To overcome this problem we switched to multiple Linear Regression.

**diagram Graph of Multiple linear regression**



**2. Polynomial Regression Algorithm**

**3. Support Vector Regression Algorithm**

**4. Random Forest Algorithm.**

**3. OPTIMIZATION TECHNIQUES**

Following are some approaches till now implemented for this network reconstruction using the micro-array time series data set. As the “No Free Lunch” (NFL) theory, any improved performance on one class of problems can only be balanced by degraded or equal performance over another class for every algorithm. So according to this NFL, we there are many ways of solving this meta heuristic problem but no one is good enough to identify regulation in a GRN with sufficient accuracy.

**3.1 Particle Swarm Optimization**

However, the Particle Swarm Optimization, which is a stochastic optimization method based on population, was introduced by Eberhart and Kennedy in 1995. It is based on a principle of operation similar to fish schooling or flocking of birds. PSO is an optimization process that is evolutionary where the candidate solutions are improved iteratively. In the year of 2007, Rui Et Al started the work for the hybrid of PSO and differential evolution (DEPSO) to optimize the gene regulatory network. They found that DEPSO achieved a better overall performance compared to the RNN for their particular test case. In the year of early 2009, Zhang et al had developed a noble hybrid model of particle swarm optimization and recurring neural network, namely PSO-RNN for gene inference method. Chien-Pang Lee et al in the year 2011 attempted to reconstruct the gene regulatory networking through the aid of microarray data set employing GA/PSO. For local search, PSO is very efficient while it fails in exploration on the other hand. Therefore, in the year of 2017 Liu, et al. present the MPSO that is able in improving the search exploration of PSO by the memory structure modification of the canonical PSO. In this case, the particles decide who among them will take the roles of the game’s leaders as opposed to a random selection process.

In the theoretical concept of PSO, a number of particles participating in the search bounded to a particular area are said to be incorrigible of food’s position. The above proposition holds that there will be one among the particles that is nearest to the food supply center. The particle is said to occupy the best position in the globe for all the particles existing at a given location. It will vanguard the other particles. Each particle holds the best location known so far by the particle during the local search phase and is notorious as the global best position. Until the stopping condition gets met, this goes on.

In order to solve a problem using PSO, the first step includes assigning an initial position vector along with the initial velocity vector that are consisting of the same number of parameters and it also has to be in the same range. At each iteration of set all the randomly initialized parameters generate accurate and actual candidate solution. All of them should not exceed ci, should not lower co, and all of them have to be equal for all particles. It is important to know that the range of each of the featured parameters may differ. This will also lead to the existence of an objective function which is responsible for the computation of the fitness value of each particle, based on the corresponding particles. Political Science Research As mentioned earlier, we will now determine the fitness value using the objective function and then look for the global best which will be the position of the particle up to where it has minimum fitness so far. This position is retained as personal best (pbest) position of each particle in the solutions. Until this level of the search the PSO will do the searching at a local level and in accordance to the preceding parameters. After that, we will move to the international form of PSO. In this case we will determine the most appropriate global best position of the entire search space, that is the particle’s position that has came closest to the position of food. The start and end of variables that track the PM of each swarm are noted by the or symbol. The particle’s position and velocity are then updated in the subsequent generation using the following equations:

where is inertia weight, c1 and c2 are called learning factor and are initialized as c1 =c2 =2, r1 and r2 are two generated random numbers between 0 and 1, (t) and (t) represent the velocity vector and position vector of the particle at the t instance.

**3.2 Bat Algorithm Inspired PSO (BAPSO)**

Bat mined PSO (BAPSO) is another newly developed approach for the optimization problem which is the combined form of BAT algorithm and Particle Swarm Optimization (PSO). Yang et al. employed the search operation of operating echolocation behavior of the micro bat. These are small bats that use echolocation; a sort of a sonar system to locate their foods and have an avoiding mechanisms. The microbats on the other hand are able to determine the direction and distances of an object on reception of the waves. The clutter in this BAT algorithm can be presumed to move around randomly, and several Bats are dispatched for local search and probing every iteration. Another modification, as pointed out by Khan et al. [35], which was made by the creator of the BAPSO algorithm is that, specifically in this scenario, each microbat is initially at rest. Both the inertia weight w and r that is used in PSO are evenly distributed random numbers generated and selected over the interval [0,1]. Due to this, the BAT algorithm can search for new solution space as well as exploit the solution space in the search. The last condition for the stop is that either the error is the smallest possible, or the number of repetitions is the biggest. The mathematical representation of the BAPSO becomes as follow:

**3.3 Grey Wolf Optimization Inspired Particle Swarm Optimization (GWPSO)**

Based on the social leadership hierarchy and hunting behavior of the grey wolf, another natural-inspired algorithm was brought into light by Mirjalili et al. known as Grey Wolf Optimization. Typically, grey wolves are known to occupy areas and they are the superior predators and though they move in groups, the pack size ranges from five to twelve. There are four different varieties of wolves and there both split equal in each group. The choice of what to hunt is by and large made by the alphas; the senior independent male and female of the pack. The next layer that makes up the SV hierarchy is the beta wolves, the secondary wolves in the pack that assists alphas in decision-making and responsibilities. These wolves can be of any gender; that is, they can be males or females of the less dominant packs betas. After alpha, the beta wolves are the second best option in the group, in terms of the the social hierarchy, the solution. The next type of pack candidates are those falling within the beta group and classified as delta.

Another meta heuristic that has been proposed by Khan is the GWPSO, in particular, a hierarchical GW approach was incorporated into the PSO procedure. The particles in this suggested method, unlike the previous one, memorize the second and the third best solutions in addition to the current best one during each iteration. This method is following by the following: This is an improvised global best solution for equation:

Where is the first best solution, is the second best solution, is the third best solution and so on. Thus the equation becomes,

**3.4 Artificial Bee Colony Optimization (ABC)**

Artificial Bee Colony optimization is another meta-heuristic nature-inspired algorithm which is based on the forging behaviour of the honey bee. It was first proposed by Karaboga , for the unconstrained optimization problem. It has been proved that it is superior to any other existing heuristic algorithm for unconstrained problems. In the proposed algorithm there are three types of bees in a colony: Employed bees, Onlookers, and scouts. Half of the colony consists of employed bees and the remaining half includes the onlookers. For every employed bee, there is only one food source, i.e. the number of food sources is equal to the number of employed bees in the hive. The scouts are employed bees whose food supply has run out. The location of the food source, or the position of the working bee, is one potential solution to the optimization problem, and the amount of nectar is the best solution, or the optimization problem’s fitness value. The following equation defines the probability value by which the onlooker bee chooses the food position:

where P is the number of the food source, pi is the position of the food source, is the fitness value of the position. In order to produce the candidate solution or the food position from the old one ABC used the following equation:

where j is the randomly selected index and k ∈ 1, 2, 3 · · · .P and is the randomly chosen number in between [1,-1]. It manages the process of producing potential food sources. The found food source is presumed to be abandoned if it is determined that it cannot be improved, and the scout bee reinitializes it using the equation below:

here xi is the position of the discarded food source and j ∈ 1, 2, · · · P Babayigit has modified this ABC algorithm by providing a probability function for improving the exploration mechanism of the onlooker bee. The author proposed that the onlooker bee will choose the food source according to the likelihood function which is as follows:

where the fitness value normalized within [0,1] is denoted by , and ρ is the governing parameter of the ABC algorithm. It signifies that the higher the fitness there is the more probability for the selection of the food source by the onlooker bees. The position of the onlooker bee is also improved by the authors for increasing the diversity. Thus the equation for the onlooker bee becomes:

In the equation is the best solution in the current population, and is the present position exploited by the onlooker bee. This is known as the ABC best algorithm. In the year 2015 Forghany used ABC and modified ABC in GRN construction with S-system as objective function. They set the population size as 50 and applied their proposed work to three networks. The maximum iteration was 2,000,000 for the first two networks and 5,000,000 for the third network. They proposed that modified ABC gave a superior result in the context of the GRN rather than any other evolutionary algorithm.

**4. DISCUSSION**

the weather forecasting analysis done in our report compares various seven day period forecasting models .It mainly focusses on the main differences between a weather forecasting services and the model that has been created .It also check the performance difference between linear regression and functional regression models.This result shows that the professional weather forecasting services are more better in finding the output than the created model throughout the week, with a larger discrepancy in the early days .this can mainly occur because of high accuracy in the short term predictions than long term ,whereas machine learning algorithms also deals with initial(starting) conditions better and might excel over longer periods.The linear regression model showed up better results than the functional regression model .This can be proved through the fact that the forecasts were based on the weather data for the last two days, which are too short to capture significant results in the weather trends.

After analysing the results we can say that the linear regression would be a better model than functional regression and it was also mentioned by Occam’s razor that the simpler lin- ear regression model is better than the funtional regression .If we increase the number of days in the forecast from two days to four to five days then ,funtional regression should have outperformed linear regression and then perhaps there would be evident trends in the weather that functional regression could capture.

**5. CONCLUSION**

In this paper, we have introduced a technology that applies machine learning techniques to provide accurate weather forecasts. These machine-learning models are simpler and less resource-intensive than traditional physical models, and they can be run on any computer or mobile device. Our evaluation results indicate that these models are accurate enough to compete with traditional models. We also incorporate historical data from surrounding areas to predict weather patterns for a specific location. This approach is more effective than focusing solely on the area in question. Weather forecasts can be used to save lives, such as by making arrangements for local residents in areas where famine is predicted. AI may not be able to prevent disasters, but new scientific tools like machine learning, image recognition, and computational modeling can help us prepare for them. In the future, we plan to use low-cost IoT devices, such as temperature and humidity sensors, to collect weather data from various parts of a city. The use of multiple sensors would increase the number of local features in the training dataset, which would further enhance the performance of our prediction models.

**6. REFERNCES**