Enhancing Data Imputation in Weather IoT Sensor Networks

Generative AI (autoencoder) for meteorological data imputation

Veldhuis, T.G.
Microdata Analysis Department
Dalarna University, DU
Borlänge, Sweden
h22twave@du.se

Ubani, C.W
Microdata Analysis Department
Dalarna University, DU
Borlänge, Sweden
h19chiub@du.se

Abstract — **Weather data plays an important role across different sectors, with IoT sensor networks facilitating automated data collection of the weather statistics (like temperature). Despite these advancements, it still results in these networks producing datasets that are full with missing values. To address this challenge, this research will use generative AI, specifically autoencoders, for the imputation of missing temperature data across Quito's districts. By training autoencoders as denoisers, their performance will be compared with traditional imputation methods which have been previously used for imputing the missing weather data. The results of this research present an improvement in imputation performance when using autoencoders, specifically for larger datasets with missing temperature entries.**

Keywords - Generative AI; Data gaps; Autoencoder; Data imputation; Meteorology

I. Introduction

The accuracy of weather forecasts holds an important influence across various sectors, like agriculture, transportation, and urban planning [1]. In recent years, the integration of Internet of Things (IoT) sensor networks, have opened up opportunities for real-time data collection, analysis, and forecasting, leading to better-informed reactions to weather related challenges [25]. Despite the high potential of IoT sensor networks, they often still struggle with data gaps due to sensor failures, connectivity disruptions, and other issues [2].

Traditional methods have been implemented to address these data gaps and these methods have primarily revolved around imputation techniques, such as using mean, median, or random values, to fill in missing data [28]. Although these methods are effective to an extent, the criticality of historical data and the importance of weather predictions using this historical data demands even more accurate solutions.

This is where generative AI comes into play, a domain of artificial intelligence that produces new data based on machine learning insights [27]. Among its models, autoencoders – a form of neural network optimized for unsupervised learning – stands out for its capability to identify complex, non-linear relationships within data [2]. This research explores the potential of denoising autoencoders to improve imputation of missing weather data in IoT sensor networks. By training on complete datasets, the autoencoder would learn to "predict" values for missing data points, improving the dataset's overall quality [25].

By doing so, this research seeks to contribute to the field of meteorology, providing methodologies that could be applied to meteorological data worldwide. By using generative AI, specifically autoencoders, our goal for this research is to enhance the reliability of historical data imputation, supporting informed decision-making and contributing to the resilience of urban environments.

II. Literature Review

Drawing insights from existing literatures within the field of meteorology, machine learning and IoT sensor networks, this literature review discusses an overview of key aspects in meteorology, including the role of IoT sensor networks, machine learning in meteorological predictions, the importance of historical weather data in predictive modeling, case studies of weather predictions in urban planning, and data quality and preprocessing. It also discusses the impact of advanced weather forecasting in various sectors

Overview of IoT Sensor Networks in Meteorology:

Meteorology, the study of atmosphere, atmospheric phenomena, and atmospheric effects on the weather has witnessed significant advancements with the Internet of Things (IoT) sensor networks, largely in the area of weather monitoring and forecasting [1]. These networks consist of distributed sensors that collect data on various meteorological parameters such as temperature, humidity, wind speed, and precipitation [1]. The data generated by these sensors is invaluable for meteorologists in understanding and predicting weather patterns. The sensors within these networks are designed to transmit data wirelessly to centralized data centers, where it can be analyzed and processed [2]. The real-time data flow from IoT sensor networks allows meteorologists to continuously monitor and assess weather conditions, providing insights into local and global weather patterns and enabling the development of accurate forecasts for various applications [2].

Challenges associated with data gaps and missing information in IoT sensor networks are a critical concern for

meteorologists. Data gaps and missing information are common issues in these networks, which can significantly impact the accuracy of weather forecasts [2]. Data gaps can occur due to sensor failures, connectivity issues, or power outages, making it essential to develop effective data imputation techniques to enhance the reliability of weather predictions [2].

Addressing data gaps is a pressing concern, especially in meteorology, where precision is paramount. Incomplete data can result in less reliable weather forecasts, potentially causing disruptions to daily life and activities. Thus, effective data imputation techniques are essential for filling these gaps and ensuring the quality of weather predictions [28].

Machine Learning in Meteorological Predictions:

Machine learning algorithms are widely applied in weather forecasting to predict key meteorological parameters, including temperature. Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), and ensemble methods like Random Forests have been commonly used for these purposes [3]. These algorithms leverage historical meteorological data, including temperature records, to create predictive models that can make accurate temperature predictions.

Evaluating the performance of machine learning algorithms in meteorological predictions is essential to determine their accuracy, efficiency, and reliability. S. Y. Lin, C. C. Chiang, J. B. Li, Z. S. Hung, and K. M. Chao [4] introduced a dynamic fine-tuning stacked auto-encoder neural network, outperforming traditional machine learning techniques in terms of both accuracy and efficiency. This innovative approach demonstrated the significant potential of more advanced neural network architectures in improving the accuracy of temperature predictions [5].

Machine learning has demonstrated its effectiveness in handling missing or incomplete data in meteorological predictions [6]. Data gaps are common in IoT sensor networks, and machine learning algorithms can be employed to impute missing data and generate more complete and accurate weather forecasts [5].

J. Kwon, C. Cha, & H. Park [6] developed a multilayered Long Short-Term Memory (LSTM) model with parameter transfer for data imputation, effectively addressing data gaps and improving the quality of predictions. Machine learning-based data imputation techniques, mainly when dealing with temperature data, have contributed to more reliable and comprehensive weather forecasts, crucial for various applications, from agriculture to disaster preparedness [5].

Importance of Historical Weather Data in Predictive Modeling:

Guo et al. [7] emphasized the importance of historical weather data in improving the accuracy of solar power output forecasts. By analyzing historical weather patterns and their impact on solar power generation, the researchers developed more reliable models for predicting solar energy production. The historical context provided by this data significantly enhances the precision of such forecasts.

As employed by N. Bogdanovs et al. [8], the Kalman filter is an effective tool for assimilating historical data into weather prediction algorithms. By considering historical and current weather states, the Kalman filter enables more accurate forecasts by providing a holistic view of weather patterns.

S. Indhumathi, S. Aghalya, S. J. A., and P. Aarthi M [9] employed historical weather data and machine-learning algorithms for rainfall prediction. Their research showcased the importance of historical information in achieving timely and accurate weather forecasts, particularly for agriculture and flood control applications.

In the context of energy forecasting, Guo et al. [7] showed how historical weather data enhanced the accuracy of solar power output forecasts. Incorporating historical data into predictive models allowed for a more comprehensive understanding of the impact of weather conditions on energy production, ultimately leading to more reliable energy forecasts.

Case Studies of Weather Forecasting in Urban Planning:

The incorporation of weather forecasting data into urban planning and design is becoming increasingly important for urban areas' sustainability and climate resilience. Temperature measurements are essential given their substantial influence on urban liveability and climate adaptability. S. Indhumathi et al. [9] showed the importance of strategic urban design in reducing heat intensity through an analysis of urban heat islands, concentrating on the wise use of green spaces and building materials. This focus on physical characteristics is consistent with Kolokotroni, M. and R. Giridharan [10]'s findings, which showed a direct connection between urban density and local climatic changes in Melbourne, Australia.

United Nations Habitat has compiled additional global findings [12] highlighting the close relationship between cities and climate change and calling for incorporating climate data in urban frameworks to improve climate adaptation within urban planning. This is similar to Coutts, A.M., J. Beringer, and N.J. Tapper [11], which detailed the connections between urbanization, coastal climate influence, and ensuing urban design solutions for effective flood control and water management for the case of Houston.

Machine Learning Applications in Quito's Environmental Conditions:

Innovative weather forecasting techniques are essential when considering Quito, Ecuador, a city with unique topographical and climatic circumstances. Shepherd, J.M., Carter, M., Manyin, M., Messen, D. and Burian, S., [13] explored this area using deep learning, like Encoder-Decoder GRU, Encoder-Decoder Convolutional GRU, and stacked LSTM (with two layers), for statistical weather forecasting with errors between 0.74 °C and 2.24°C. This effort demonstrated the value of cutting-edge methods in understanding the complexity of Quito's

microclimates. In parallel, Cañar, R.L., Fontaine, A., Morillo, P.L. and El Yacoubi, S., [14] explored neural network-based weather forecasting for Quito, comparing several optimizers (Adam, AdaMax, and AdamW) on an LSTM structure and revealing the nuanced impact of optimizer selection on predicted precision with Adam W having an improvement of 1.3 °C MSE.

By extending the scope and using machine learning to forecast urban pollution levels, Llugsi, R., El Yacoubi, S., Fontaine, A. and Lupera, P. [15] started a creative exploration within Ecuador. This research demonstrates how machine learning, using a decision tree model, may be tailored to environmental forecasting and how it might help Quito's public health and urban planning sectors advance. In line with these researches, the use of machine learning in monitoring Quito's air quality at Park, M.-S. and K. Baek [16] demonstrated the usage of the k-Nearest Neighbour algorithm with promising numbers.

Data Quality and Preprocessing in Meteorological Machine Learning Models:

However, it is crucial to address the fundamental aspects of data quality and preprocessing in meteorological machine learning models before examining the sector-specific implications of advanced weather forecasting. This is especially important in regions with diverse climates like Quito.

The quality of meteorological data, which is increasingly coming from the Internet of Things devices, directly impacts how well weather forecasts work. Rosero-Montalvo, P.D., et al [17] emphasized the necessity of robust Quality Management Systems for IoT meteorological networks, highlighting common issues such as sensor errors and data transmission inconsistencies that could noticeably skew forecasting accuracy. Given the vastness and complexity of meteorological data, Park, M.-S. and K. Baek [18] argued for thorough data management rules to overcome these difficulties. A discourse on the complexity of weather data analysis and preprocessing, a requirement for improving the value of data in prediction models, was offered by Munappy, A., Bosch, J., Olsson, H.H., Arpteg, A. and Brinne, B., [19] to support this, focusing on approaches for data purification, normalization, and restructuring.

Impact of Advanced Weather Forecasting on Different Sectors:

Beyond improving model accuracy, data quality and preprocessing also enhance subsequent applications' dependability in various industries. Therefore, strict data management procedures are essential for releasing machine learning's full potential in weather forecasting and potentially resulting in more efficient decision-making.

A few industries have the potential to significantly improve through accurate weather forecasting, with agriculture and transportation standing out. The importance of cutting-edge weather analytics in improving agricultural decisions was underlined by Jayakumar, R. and Saravanan, R.A., [20] and Bendre, M., R. Thool, and V. Thool [21], highlighting the revolutionary potential of big data and machine learning in precision farming. According to studies by Jaybhaye, N., et al. [22] and Nurmi, P., A. Perrels, and V. Nurmi [23], improvements in weather forecasting show significant promise for enhancing road safety and productivity, mainly through integrating real-time meteorological data into Intelligent Transportation Systems.

In conclusion, integrating IoT sensor networks and machine learning has markedly enhanced meteorological monitoring and forecasting. While IoT provides extensive real-time data, machine learning, particularly advanced neural networks, addresses challenges such as data gaps, ensuring reliable predictions ground of existing research in this main field. However, the use of autoencoders for filling missing meteorological data is still an emerging field, and our research paper navigates this domain without heavily relying on existing literature. As the field continues to evolve, ongoing research and development are essential to fully leverage these technologies for more accurate and comprehensive weather forecasting.

Research Questions and Hypotheses:

This research aims to address the question: *How can generative AI techniques, specifically autoencoders, enhance the imputation of missing weather data in IoT sensor networks?*

Two hypotheses have been formulated in addressing this research question:

Testable Hypothesis: "If we utilize autoencoder-based generative AI to impute missing weather data in IoT sensor networks, then incorporating historical weather data as pre-training will significantly improve the imputation accuracy."

Null Hypothesis: "There is no significant improvement in the imputation accuracy when using autoencoder-based generative AI with historical weather data pretraining."

III. Method description

This chapter discusses the research's dataset, processes, and model used for research on data imputation applied on meteorological data.

A. The Dataset

The dataset is a collection of meteorological observations that has been gathered from an array of different IoT weather stations. The information was gathered within a broad time frame that goes from 2004 to January 2023. The temperature observations have been brought together by the municipal office of environmental quality in Quito, Ecuador. The data can be used as a valuable source for different research goals, and this amount of data already looks promising.

TABLE I. Description of Dataset

Column Names	*Description*	*Percentage of NaN values*
Date	Date and Time point information	0.00
Belisario	Weather data for Belisario in °C	1.26

Column Names	*Description*	*Percentage of NaN values*
Carapungo	Weather data for Carapungo in °C	2.60
Centro	Weather data for Centro in °C	78.40
Cotocollao	Weather data for Cotocollao in °C	2.34
El Camal	Weather data for El Camal in °C	13.32
Guamani	Weather data for Guamani in °C	68.85
Los Chillos	Weather data for Los Chillos in °C	0.95
San Antonio	Weather data for San Antonio in °C	69.62
Tumbaco	Weather data for Tumbaco in °C	1.29

The dataset contains ten different data points of which nine are of temperature values from different districts within Quito (Table I). Each row includes the date and time information along with temperature measurements in degree Celsius (°C) of the nine different areas: Belisario, Carapungo, Centro, Cotocollao, El Camal, Guamaní, Los Chillos, San Antonio, and Tumbaco. The dataset contains a years long historical dataset with exactly 167,304 rows, which could give great opportunities to investigate long-term climate trends, but in our case accurately predict missing data.

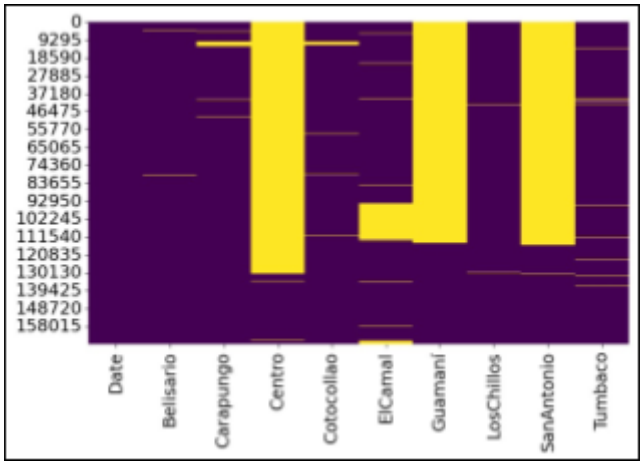

Figure 1. Dataset Heatmap (indicating missing NaNs in yellow and the complete data values in purple).

This dataset has missing values, represented as NaNs, these NaN values primarily are shown for the missing weather station in specific districts over a span of multiple years. This resulted in some of the weather stations only adding value 21% of the time over all these years. To be more specific regarding the NaN values, the dataset has 399,243 NaN values spread over all the different columns and 1,608,405 of non NaN values. This can be visualized in the heatmap of the dataset about (Figure 1), where the yellow color visualizes the parts of the dataset with missing NaN values, whilst the purple color shows the areas filled with the complete recorded data of the IoT sensor networks.

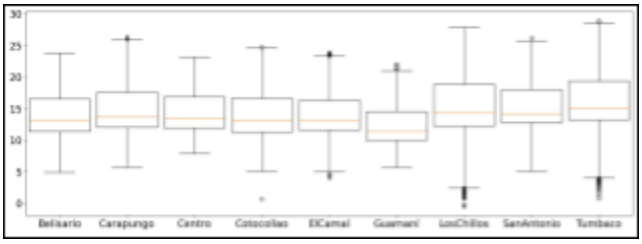

Figure 2. Box Plot distribution of the temperature values across regions.

The box plot in Figure 2 visualizes the distribution of the temperature value columns (of the different regions in Quito). The date column was removed because it was irrelevant, because it was not being a floating value and if it would have been translated to floating values then it would have a perfect distribution (it is like an index). Tumbaco has the highest median temperature and also has the highest average temperature (mean = 16.17°C). The standard deviation of this area (Tumbaco) is also the biggest (std = 4.13°C). Its whiskers extend to show the full range of temperatures from 0.87°C to 28.85°C. Whilst Guamani has the lowest average temperature (mean = 12.29°C) and a relatively narrow spread of temperatures (std = 2.82°C). Its whiskers extend from the box down to the minimum value (5.72°C) and up to the maximum value (22.02°C), showing the range of the majority of the data. Overall, there is only one extreme outlier which can be found in Cotocollao but we assumed that this will have little to no effect on the model as it is only one value over the whole dataset (which has 2 million values and NaN-values).

B. Data Pre-Processing

To be able to potentially make use of the date and time information, we needed to perform some feature engineering on the data and get a hold of data that could also be used in the model. So the date column has been transformed to a Year, Month, Day, Hour, Weekday, DayOfYear and IsWeekend column. Where Weekday is the number of the day within a week (from 0 for monday to 6 for sunday). DayOfYear is day's number within the year, which ranges from 1 to 365 and IsWeekend has the value 1 for saturdays and sundays and 0 for the rest of the week. This was done to see if there was any potential in utilizing date information in the "prediction" of missing data.

Then the data preprocessing starts off with some exploratory data analysis, of which the correlation between different columns is the biggest conclusion (Figure 3). This made us conclude that the 'Date' and its feature engineered co-features will be removed to train the model, because these have no correlation with the temperature.

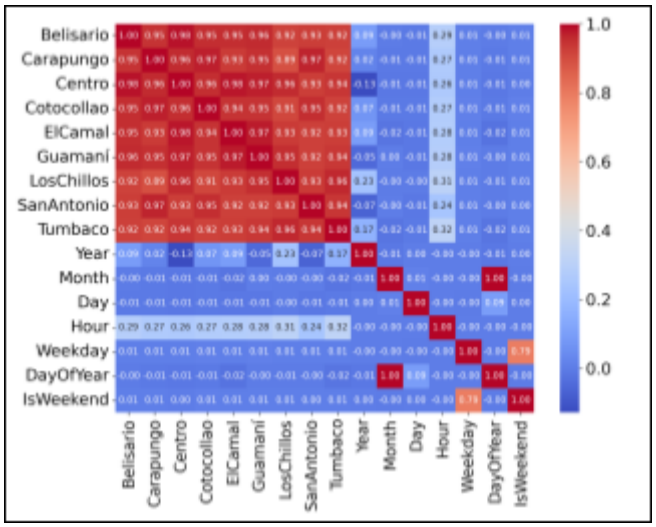

Figure 3. Correlation plot of all the columns (including the feature engineered columns)

After this the dataset is normalized to values that go from -1 to 1, this is done for two reasons: so all the different columns have the same amount of influence on the model and so the dataset and imputation goals will work with tanh (which has values from -1 to 1) that will be used in the autoencoder as the activation function. We also used ReLu as the activation function in the model, but this failed because when imputing data it showed a lot of 0 values. Our assumption is that this is because of the cut off at 0 that is built in ReLu functions.

One of the other main procedures that is executed before the data is feeded into the model, is masking temperature values. The data masking involves putting in fake numbers (in this case 0) on the locations of the real values and keeping a copy of the data, so the real value is known and the loss can be calculated. This step realizes a simulation of the real-world scenario where some data is missing. The autoencoder is then trained to fill in these gaps, based on the patterns it deciphers from the unmasked data, essentially teaching itself to "predict" these missing values. Masking will be executed on the rows that contain all the values (around 35,000 rows), and will be applied on the same missingness level as the leftover part of the dataset, so the rows that have missing values (around 130,000 rows), which have a value missing rate of 0.33.

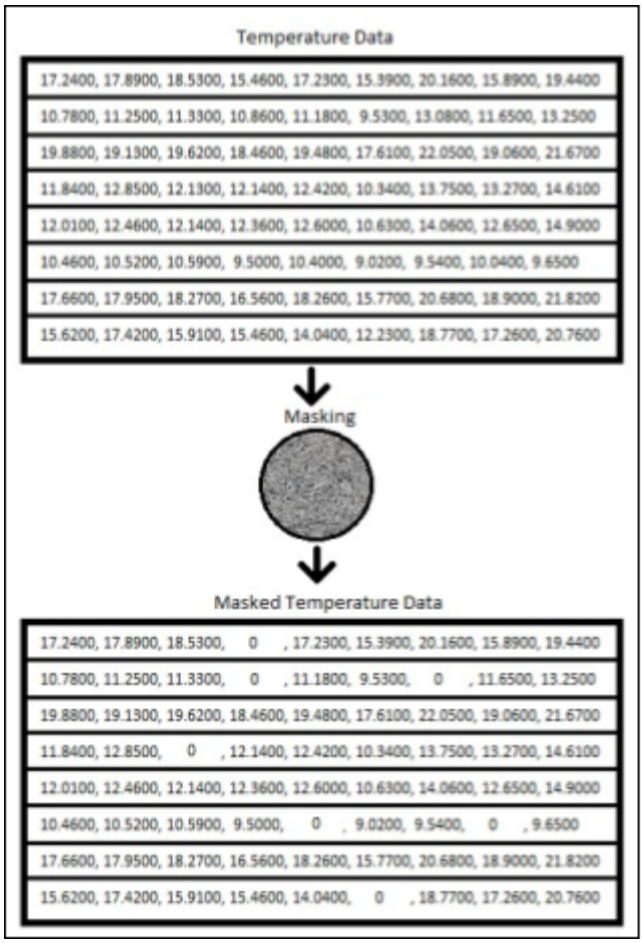

Figure 4. Simplified diagram of masking process.

Finally to be able to compare the autoencoder's performance against other traditional methods:

- Imputation using mean of row values
- Imputation using median of row values
- Imputation using mean of column values
- Imputation using median of column values
- Imputation using random value insertion from column values
- Imputation using normal distribution of the column values, based on mean and standard deviation

This way we have a thorough comparison of the already existing imputation methods, compared to the new auto encoder methods. Even though there are probably more methods, these methods should for now suffice as these are easy to use imputation methods.

C. Data Mining Method

At the core of this project will be the usage of an autoencoder, a type of artificial neural network used for unsupervised learning. The main goal for this model in our context is to understand the temperature time series of different areas in Quito. The model will be trained to understand the temperature features that go into the network, allowing the network to encode the data (Figure 5).

The same structure will be done mirrored in the output layer, aiming to reconstruct (decode) the original timeline input (Figure 5). This symmetry allows the model to "predict" or rather, reconstruct the masked temperature values using the unmasked data as reference.

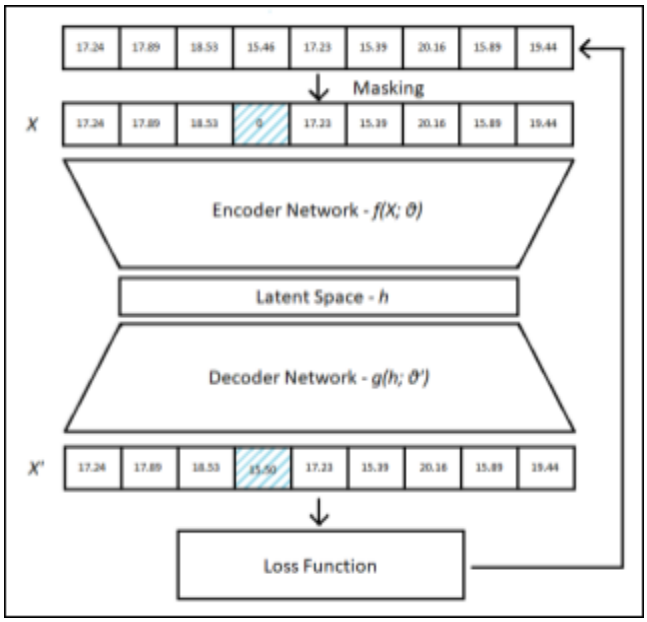

Figure 5. Autoencoder Model description

An autoencoder can be summarized by two nonlinear encoding and decoding functions f and g, respectively (Figure 5). From a given input, X, the encoder maps the input data to a hidden representation 'h', through the function f. Here the θ stands for all the parameters, which are the weights and bias for the encoder.

$$h = f(X; \theta)$$

The autoencoder has in this case a bottleneck (latent space), which is the layer that will contain the compressed knowledge of the data. And after this the decoder maps the hidden representation 'h' back to the space of X, resulting in a reconstruction X' through another mapping function 'g'. Here θ' represent the parameters, so the weights and biases of the layers in the decoder.

$$X' = g(h; \theta')$$

For an autoencoder, it is the goal to minimize the difference between the input 'X' and the reconstructed output 'X''. This done by a loss function L(X, X'), which is gonna be in this case the mean squared error (MSE) of input data:

$$L(X, X') = ||X - X'||^2$$

Also to be able to learn the autoencoder has to optimize its weights on the encoder and decoder functions, this is typically done through SGD (stochastic gradient descent) or Adam. The latter of the two will be used in this model, because of its fast convergence rate which can handle complex models. The gradients of the loss function in this model L are computed: $\nabla\_\theta L$ and $\nabla\_\theta' L$. These will be used to update the in the direction that minimizes the loss: $\theta = \theta - \alpha \nabla\_\theta L$ and $\theta' = \theta' - \alpha \nabla\_\theta' L$, where α is the learning rate.

Our autoencoder will be used as a denoising autoencoder, these models are trained to reconstruct the original input from a corrupted version. In this case, this 'corrupted' version of the data will be the missing data for the end model and the masking of the current data. This will force the autoencoder to learn the general aspects of the data distribution, rather than the noise.

The tuning of the model is an important step to improve its performance. We adjust various parameters, such as the number of layers in the autoencoder, the number of nodes in each layer, the learning rate, and the number of epochs (each epoch is an iteration over the entire dataset). These parameters are methodically varied to determine the most optimal combination that yields the highest accuracy in data reconstruction.

To make sure that the model will have improved performance and reliability, we use the k-fold cross-validation method. This to potentially improve its generalization capabilities on unseen data. Instead of splitting the dataset just once into a training and validation set, the data is divided into 'k' equally-sized "folds." In this iterative process, the model is trained 'k' times, with each fold taking its turn as the testing set while the remaining k-1 folds are used for training. By rotating the validation set across all folds, we acquire a better understanding of the model's performance across various data subsets, which helps with mitigation of the potential of overfitting. By averaging the results from all 'k' iterations, we derive a more accurate and consistent estimate of the model's performance, ensuring it is tuned to diverse scenarios within the data.

IV. Results and Analysis

In the methodology, we introduced the dataset that contains the temperature measurements from 2004 to January 2023 in Quito, Ecuador. Given the big number of missing values present in the data, an autoencoder, a type of artificial neural network, was created for unsupervised learning to decipher temperature trends across nine districts. After removing date-related columns due to their low correlation with temperature data, the dataset underwent normalization and data masking on the rows that did contain all the data to simulate scenarios with missing data. For improved model reliability, the k-fold cross-validation method was introduced, to ensure robustness against overfitting. The autoencoder's effectiveness in imputing missing values was then set to be benchmarked against

traditional imputation methods. Following this methodological approach, the following results and analysis chapter will present and discuss the model's performance in reconstructing missing data, delving into its comparison with the traditional imputation techniques.

Descriptive Statistics & Preliminary Analysis

Upon completing the imputation process, our dataset has no NaN values (missing values) left and shows a continuous representation of temperature data for the time frame in question. From the original dataset which contained around 26% missing values, post-imputation this percentage has been reduced to 0% of the values missing.

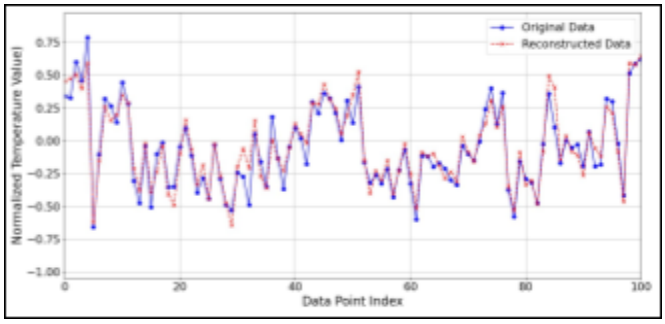

Figure 6. Comparison of Original vs Reconstructed Data Points for the first 100 masked points.

When comparing the original dataset with the reconstructed version, there were subtle variations in the temperature patterns across certain districts (Figure 6). The mean deviation between original and reconstructed data was 0.03°C, suggesting a relatively accurate approximation by the autoencoder. While on the other hand the standard deviation on average decreased with 0.04°C, so a deviation of -0.04°C.

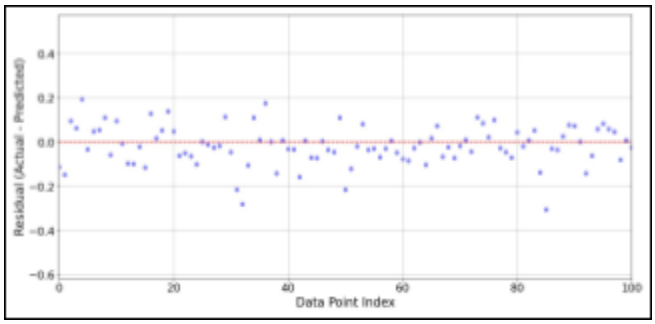

Figure 7. Residual Plot for the first 100 masked points.

It is also relevant to plot the residuals, this is the difference of the original data compared to the imputed data (Figure 7). This is plotted to check if there is any bias in the data and as can be seen in the plot above(Figure 7), this figure shows the first 100 imputed data points, the residuals are completely random and there looks like there is no bias in the dataset. This is only checked with checking the residuals, but won't be further checked because the main goal of this paper is the basic proof that autoencoders have potential in data imputation for meteorological data.

Model Evaluation Metrics

- **MSE (Mean Squared Error):** Our model achieved an MSE of 0.009 in testing and also a 0.009 in training, indicating the average squared difference between the estimated values and what is estimated. While the next lowest traditional imputation method achieved an MSE of 0.027 in training and testing.
- **RMSE (Root Mean Squared Error):** The RMSE value stands at 0.094 in testing and 0.094 in training, providing insights into the model's prediction accuracy in temperature units. While, again, the next lowest traditional imputation method a RMSE achieved 0.166 in training and testing.
- **Correlation between Original and Imputed:** A strong correlation of 0.96 was observed between the complete data (so the rows of the original dataset that contained completed rows, and where we simulated the missingness of data by masking the data) and the imputed values, emphasizing the model's ability to retain core data patterns. These parts are both from the training and testing phase,

TABLE II. MSE AND RMSE OF IMPUTATION METHODS

	MSE		*RMSE*	
Method	***Training***	***Testing***	***Training***	***Testing***
AutoEncoder Model	0.009	0.009	0.095	0.094
Mean Row Imputation	0.027	0.027	0.166	0.166
Median Row Imputation	0.032	0.032	0.179	0.179
Random Imputation from Column	0.187	0.187	0.433	0.433
Norm. Distribution Imputation from Column	0.187	0.188	0.433	0.433
Mean Column Imputation	0.094	0.094	0.306	0.306
Median Column Imputation	0.103	0.103	0.321	0.321

This table above explains all the different traditional imputation methods and the currently researched autoencoder (Table II). Here, it displays all the training and testing mean MSE and RMSE of the different methods under each other. However, there was not a high improvement in the testing over the training set because of how huge the dataset is, and the training and testing set are separated randomly. In both the MSE and RMSE cases, the most effective traditional imputation method was the mean row imputation, so the calculation of the mean of the row and using that as imputation.

Analysis & Discussion

Based on the analysis of the model, we undertook a comparison of the autoencoder. In this case we focused on an autoencoder configuration with a hidden layer of 3 nodes and a latent space of 2 nodes. This setup already shows a lot of promising findings, as it seems to already outperform the traditional methods when it comes to imputation of missing temperature values.

The quality of the autoencoder is further demonstrated through the comparison of the MSE and RMSE values, highlighting its improved prediction capabilities. These metrics suggest that the autoencoder is particularly adept at handling larger datasets with a lot of missing data. The precise prediction of the temperature is probably caused by the high correlation of the different temperature columns within the training data and the minimal variation across different areas within one timeframe.

Contrary to our initial expectations, the analysis didn't have any anomalies or outliers that could have impacted the results. These were for so far we can conclude, totally as expected. This is probably most likely because of the high volume of data that we had at our disposal.

Our findings align with the existing literature. Autoencoders have consistently been showcasing efficiency when handling large datasets with significant gaps. Specifically, our results are the same as the conclusions drawn in [4], which also shows that autoencoders have high accuracy.

In relation to our research objectives, we proposed the following hypotheses:

- Testable Hypothesis: "If we utilize autoencoder-based generative AI to impute missing weather data in IoT sensor networks, then incorporating historical weather data as pre-training will significantly improve the imputation accuracy."
- Null Hypothesis: "There is no significant improvement in the imputation accuracy when using autoencoder-based generative AI with historical weather data pretraining."

Taking the the outcomes of our study into consideration, the results seem to provide evidence that support our testable hypothesis, suggesting that autoencoder-based generative AI, especially when using historical weather data as pre-training, improves imputation accuracy.

Challenges & Limitations

In the end, even though the project turned out to be a success there were still some problems we had to tackle. One primary challenge faced was ensuring the autoencoder did not overfit, especially given the extensive dataset. This was to our knowledge solved by executing the autoencoder in a K-Fold cross validation method. Also the application of the masking was a challenge as we saw that the number that was used for the masking of the values impacted the prediction greatly. This was not taken into account during this assessment because it was not the goal of this research, but could be taken into consideration for future work.

Recommendations & Future Work

Talking about future work, we would also advise to start tweaking the autoencoder in different environments. This would mean the application of this autoencoder for different datasets with the same goals and aim.

It would also be interesting to see if the introduction of more external data (like humidity, altitude, etc.) could further enhance prediction accuracy. Additionally, investigating other machine learning techniques in tandem with autoencoders might present more robust solutions. Like using a RNN machine learning model, these models could also include the prediction of the missing data using the value from the previous row instead of only basing the imputation on the information from the current row.

V. CONCLUSION

The primary focus of this research was to explore the efficacy of autoencoders in the imputation of missing temperature data, specifically in the context of a dataset from Quito, Ecuador. Our results underline the superiority of autoencoders over traditional imputation methods. Not only did the autoencoder model yield a reduced mean squared error and root mean squared error, but it also showed a strong correlation between the original and imputed data. This highlights the model's proficiency in retaining core data patterns, reinforcing the notion that autoencoders can be pivotal in addressing data gaps, especially in expansive datasets like ours.

However, this research journey was not without its challenges. A significant hurdle faced was ensuring the autoencoder did not succumb to overfitting, a task made intricate by the sheer volume of our dataset. We tried to tackle this concern by integrating the K-Fold cross-validation method, even though this did not have a lot of impact on the evaluation metrics it should improve the overfitting issue.

Looking ahead, the promising results achieved here beckon a broader application of autoencoders in diverse environmental datasets. We advocate for experimenting with autoencoders across different terrains and climates to gauge their universal applicability. Additionally, the incorporation of external factors such as humidity and altitude could potentially enhance prediction accuracy. Furthermore, the synergy of other machine learning models, like recurrent neural networks, in tandem with autoencoders, could herald a new era of robust and reliable data imputation techniques. As climate patterns grow increasingly unpredictable, tools that offer precise meteorological data reconstruction, like the autoencoder, become ever more invaluable.

DISCLAIMER: *This paper's grammar has been improved using different AI techniques.*

REFERENCES

[1] L. S. Chandana and A. R. Sekhar, "Weather monitoring using wireless sensor networks based on IoT," Int. J. Sci. Res. Sci. Technol, vol. 4, pp. 525-531, 2018.

[2] B. S. Rao, K. S. Rao, and N. Ome, "Internet of Things (IoT) based weather monitoring system," International Journal of Advanced

Research in Computer and Communication Engineering, vol. 5, no. 9, pp. 312-319, 2016.
[3] B. Bochenek and Z. Ustrnul, "Machine learning in weather prediction and climate analyses—applications and perspectives," Atmosphere, vol. 13, no. 2, p. 180, 2022.
[4] S. Y. Lin, C. C. Chiang, J. B. Li, Z. S. Hung, and K. M. Chao, "Dynamic fine-tuning stacked auto-encoder neural network for weather forecast," Future Generation Computer Systems, vol. 89, pp. 446-454, 2018.
[5] Holmstrom, M., Liu, D., & Vo, C. (2016). Machine learning applied to weather forecasting. Meteorol. Appl, 10, 1-5.
[6] J. Kwon, C. Cha, and H. Park, "Multilayered LSTM with Parameter Transfer for Vehicle Speed Data Imputation," in 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 2021, pp. 1-5.
[7] Guo, J., You, S., Huang, C., Liu, H., Zhou, D., Chai, J., ... & Black, C. (2016, July). An ensemble solar power output forecasting model through statistical learning of historical weather dataset. In 2016 IEEE Power and Energy Society General Meeting (PESGM) (pp. 1-5). IEEE.
[8] N. Bogdanovs, V. Bistrovs, E. Petersons, A. Ipatovs, and R. Belinskis, "Weather Prediction Algorithm Based on Historical Data Using Kalman Filter," in 2018 Advances in Wireless and Optical Communications (RTUWO), Riga, Latvia, 2018, pp. 94-99.
[9] S. Indhumathi, S. Aghalya, S. J. A., and P. Aarthi M, "IoT-Enabled Weather Monitoring and Rainfall Prediction using Machine Learning Algorithm," in 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2023, pp. 1491-1495.
[10] Kolokotroni, M. and R. Giridharan, Urban heat island intensity in London: An investigation of the impact of physical characteristics on changes in outdoor air temperature during summer. Solar Energy, 2008. 82(11): p. 986-998.
[11] Coutts, A.M., J. Beringer, and N.J. Tapper, Impact of Increasing Urban Density on Local Climate: Spatial and Temporal Variations in the Surface Energy Balance in Melbourne, Australia. Journal of Applied Meteorology and Climatology, 2007. 46(4): p. 477-493.
[12] Cities and climate change: global report on human settlements 2011 / United Nations Human Settlements Programme. 2011
[13] Shepherd, J.M., Carter, M., Manyin, M., Messen, D. and Burian, S., 2010. The impact of urbanization on current and future coastal precipitation: a case study for Houston. Environment and planning B: Planning and Design, 37(2), pp.284-304.
[14] Cañar, R.L., Fontaine, A., Morillo, P.L. and El Yacoubi, S., 2020, October. Deep Learning to implement a Statistical Weather Forecast for the Andean City of Quito. In 2020 IEEE ANDESCON (pp. 1-6). IEEE.
[15] Llugsi, R., El Yacoubi, S., Fontaine, A. and Lupera, P., 2021, October. Comparison between Adam, AdaMax and Adam W optimizers to implement a Weather Forecast based on Neural Networks for the Andean city of Quito. In 2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM) (pp. 1-6). IEEE.
[16] Rybarczyk, Y. and R. Zalakeviciute. Machine learning approach to forecasting urban pollution. in 2016 IEEE Ecuador Technical Chapters Meeting (ETCM). 2016.
[17] Rosero-Montalvo, P.D., Caraguay-Procel, J.A., Jaramillo, E.D., Michilena-Calderón, J.M., Umaquinga-Criollo, A.C., Mediavilla-Valverde, M., Ruiz, M.A., Beltrán, L.A. and Peluffo, D.H., 2018, November. Air quality monitoring intelligent system using machine learning techniques. In 2018 International Conference on Information Systems and Computer Science (INCISCOS) (pp. 75-80). IEEE.
[18] Park, M.-S. and K. Baek, Quality Management System for an IoT Meteorological Sensor Network— Application to Smart Seoul Data of Things (S-DoT). Sensors, 2023. 23(5): p. 2384.
[19] Munappy, A., Bosch, J., Olsson, H.H., Arpteg, A. and Brinne, B., 2019, August. Data management challenges for deep learning. In 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) (pp. 140-147). IEEE.
[20] Jayakumar, R. and Saravanan, R.A., 2022. WEATHER DATA ANALYSIS DATA PREPROCESSING. Journal of Pharmaceutical Negative Results, pp.3149-3158..
[21] Bendre, M., R. Thool, and V. Thool. Big data in precision agriculture: Weather forecasting for future farming. In 2015 1st international conference on next-generation computing technologies (NGCT). 2015. IEEE.
[22] Jaybhaye, N., Tatiya, P., Joshi, A., Kothari, S. and Tapkir, J., 2022, January. Farming Guru:-Machine Learning Based Innovation for Smart Farming. In 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 848-851). IEEE.
[23] Nurmi, P., A. Perrels, and V. Nurmi, Expected impacts and value of improvements in weather forecasting on the road transport sector. Meteorological Applications, 2013. 20(2): p. 217-223.
[24] Dey, K.C., A. Mishra, and M. Chowdhury, Potential of intelligent transportation systems in mitigating adverse weather impacts on road mobility: A review. IEEE Transactions on Intelligent Transportation Systems, 2014. 16(3): p. 1107-1119.
[25] Yu, J., He, Y. and Huang, J.Z., 2021, December. A Two-Stage Missing Value Imputation Method Based on Autoencoder Neural Network. In 2021 IEEE International Conference on Big Data (Big Data) (pp. 6064-6066). IEEE.
[26] Fang, S., Da Xu, L., Zhu, Y., Ahati, J., Pei, H., Yan, J. and Liu, Z., 2014. An integrated system for regional environmental monitoring and management based on internet of things. IEEE Transactions on Industrial Informatics, 10(2), pp.1596-1605.
[27] Aydın, Ö. and Karaarslan, E., 2023. Is ChatGPT leading generative AI? What is beyond expectations?. What is beyond expectations.
[28] Wang, H., Tang, J., Wu, M., Wang, X. and Zhang, T., 2022. Application of machine learning missing data imputation techniques in clinical decision making: taking the discharge assessment of patients with spontaneous supratentorial intracerebral hemorrhage as an example. BMC Medical Informatics and Decision Making, 22(1), pp.1-14.