Advanced International Journal for Research (AIJFR)
E-ISSN: 3048-7641 ● Website: www.aijfr.com ● Email: editor@aijfr.com

Forecasting on Temperature and Dew Point based on Agricultural Field data using Transformer Based Model

Subhash Kumar[1], Dr. SP Singh[2], Dr. Dheerendra Pratap Singh[3]

[1]Research Scholar, NIMS University Jaipur-Delhi Highway (NH-11C), Jaipur, Rajasthan,
[2,3] Professor, NIMS University Jaipur-Delhi Highway (NH-11C), Jaipur, Rajasthan,

Abstract

A IoT based project in a village of Maharashtra Narayangaon was as a pilot study. In this village the farmers of this village mainly do the farming of grapes and sugarcane along with seasonal vegetables like cabbage, tomato etc. Through remote sensing the data is collected on wind speed, relative humidity, dew point etc.

This project is divided in various modules like data collection module, statistical model for timeseries forecasting, deep learning model in forecasting, agricultural field coverage design module Out of various modules this paper concentrates on the new time series model using Transformer. Advantages of Transformer are Parallel Processing where All words processed together and it works faster on GPUs due to parallelism. Another advantage is Long Dependency Learning through which It understands the distant relationships.

The current paper discusses the use of Transformer architecture for forecasting in Time Series on air temperature, wind speed, humidity based on the historical information collected from the sensor installed in the agricultural field. Transformer introduces the concept of self-attention and Transformer uses parallel computation of sequential ordered data such as in time series for faster processing. Proposed field of study is related to providing solution for the farmers in terms of minimizing loss due to heavy rains, winds and other weather conditions. Especially when the crop is ready for harvesting and unexpected rains come which leads to financial loss. A better prediction on atmospheric condition will help the farmers to minimize crop loss.

Index terms: Relative Humidity, Dew Point, Remote Sensing, Transformer, Self Attention, Long Dependency, Parallel Computation.

1. Introduction

Farmers' main focus was to avoid crop loss due to severe weather conditions. To achieve this, it was necessary to monitor the soil and weather conditions on the farm field. Thus, the need for the sensor device was recognized. After exploring different sensors, the required sensors were ultimately installed in the Agricultural Field in village Narayangaon. The initial tasks involved installing devices and assessing their feasibility, ensuring they are safe and capable of detecting soil, air, and other data. Thus, remote sensors were positioned at the center of the field, enclosed by a fence. Due to the instability of electricity in the village, a solar panel was incorporated into the device. Transformer architecture is applied in Time Series data for forecasting HC Air temperature [°C] and Dew Point [°C].

2. Literature Review

As per Ashish Vaswani et al. [1], the Transformer architecture introduced in Attention Is All You Need replaced recurrent neural networks with self-attention mechanisms for sequence modeling tasks. The proposed model achieved superior performance in machine translation while reducing training time through parallel processing and multi-head attention mechanisms.

As per Jacob Devlin et al. [2], BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding introduced bidirectional Transformer encoders for language understanding tasks. The model used masked language modeling and next sentence prediction techniques to improve contextual learning and achieved state-of-the-art performance on various NLP benchmark datasets.

As per Alec Radford et al. [3], Improving Language Understanding by Generative Pre-Training proposed a generative pre-trained Transformer model using unsupervised learning on large text corpora. The model demonstrated strong capabilities in text generation, question answering, and language understanding tasks through transfer learning approaches.

As per Yinhan Liu et al. [4], RoBERTa: A Robustly Optimized BERT Pretraining Approach improved the BERT architecture by optimizing training strategies, increasing training data, and removing next sentence prediction tasks. The model achieved improved performance across multiple natural language processing benchmarks.

As per Colin Raffel et al. [5], Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer proposed a unified text-to-text Transformer framework where all NLP problems were converted into text generation tasks. The approach simplified model architecture and improved transfer learning efficiency across various NLP applications.

As per Alexey Dosovitskiy et al. [6], An Image is Worth 16x16 Words extended Transformer architectures to computer vision applications by treating image patches as sequential tokens. The Vision Transformer model achieved competitive image classification accuracy compared to convolutional neural networks.

As per Zihang Dai et al. [7], Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context addressed the fixed-length context limitation of standard Transformers using segment-level recurrence and memory mechanisms. The proposed model demonstrated improved long-term dependency learning and better language modeling performance.

As per Zhilin Yang et al. [8], XLNet: Generalized Autoregressive Pretraining for Language Understanding introduced permutation-based autoregressive pretraining to overcome limitations of masked language models. The model achieved superior performance compared to BERT on several NLP benchmark tasks.

As per Victor Sanh et al. [9], DistilBERT, a distilled version of BERT applied knowledge distillation techniques to reduce the size and computational requirements of BERT models while maintaining competitive performance. The proposed model enabled faster inference and reduced memory usage for practical deployment.

As per Uluocak et al [10], for air temperature (AT) forecasting, hybrid models that of convolutional neural network (CNN) with a long short-term memory (LSTM) or GRU-CNN for one day ahead prediction gives the best result for accuracy. Various statistical criteria like MAE, RMSE, NSE, and R^2 and visual comparison were made when evaluating the performance of model.

Zongjun Wu et al. [11] proposed a hybrid PSO-LSTM model for soil moisture estimation based on the particle swarm optimization (PSO) and long short-term memory (LSTM) network at a large scale. Their results showed that the accuracy of hybrid PSO-LSTM model to be greater than the standalone LSTM model at different depths.

It predicted the soil moisture content at different depths (5 cm, 10 cm, 20 cm and 40 cm) in the citrus orchard located at GPS (104°517′ 6″E, 29°55′7″N, 519.0 m).

K. Venkatachalam et al. [12] developed DWFH which stands for an improved data-driven deep weather forecasting hybrid model using Transductive Long Short Term Memory (T-LSTM). The experiments are carried out on HHWD and Jena Climate datasets. The dataset comprises 14 weather features like humidity, temperature, etc. The T-LSTM method performs better than other methodologies, producing 98.2% accuracy in forecasting the weather.

Mohamed Khalid et al [13] used Support Vector Regression (SVR), Regression Tree (RT). In predicting the temperature forecasting North America they took data from for 2000 to 2015 as training and forecasted on remaining years from 2016 to 2021 using SVR and RT and the results were good.

Wang et al. [14], according to this paper, the deep learning model perform better than the statistical models like ARIMA in temperature forecasting.

Aghelpour et al. [15] proposed SVR-FA models in temperature forecasting. Here they used firefly optimization over the Support Vector Regression (SVR). Five stations indicating different climates of Iran were taken and the data from 1951–2011 were used as training and then forecast was done for the period of 2012–2017 using Support Vector Regression (SVR) merged with Firefly optimization. It gave good result.

Xu et al. [16] proposed a wavelet transform-based support vector machine (SVM) for prediction of air temperature, which gives improved accuracy in seasonal temperature trends. The region where there is high variability of seasonal temperatures this method is effective.

Alencar et al.[17], in this paper deep learning methods, Long Short-Term Memory (LSTM) neural networks, was used for predicting air temperature. The deep learning models can handle complex temperature trends and seasonal patterns much better than conventional statistical methods.

Zhang et al.[18] proposed a hybrid model that combines ARIMA (Auto-Regressive Integrated Moving Average) and neural networks for time series forecasting, including temperature. The hybrid outperforms traditional models and it captures complex patterns in temperature data.

3. Research Methodology

a) Data Collection

This sensor is positioned at a latitude of 19.1212 and a longitude of 73.9742 in Narayangaon village, Maharashtra. As illustrated in Fig 1, the sensor harnesses sunlight to charge the battery, eliminating the need for electricity. A 3-meter sensor rod is buried beneath the soil surface, where it monitors various parameters such as soil moisture, volumetric ionic condition, and soil temperature at depths of 50cm, 100cm, 150cm, 200cm, 250cm, and 300cm. The field data, along with weather data, is transmitted as streaming data from the field. This data includes the following fields: Dew Point, Solar Radiation, HC Relative Humidity, Precipitation, Leaf Wetness, Ultrasonic wind speed, Volumetric Ionic Content 1, Volumetric Ionic Content 2, Volumetric Ionic Content 3, Volumetric Ionic Content 4, Volumetric Ionic Content 5, Volumetric Ionic Content 6, Soil temperature 1, Soil temperature 2, Soil temperature 3, Soil temperature 4, Soil temperature 5, and Soil temperature 6.

b) Transformer Architecture

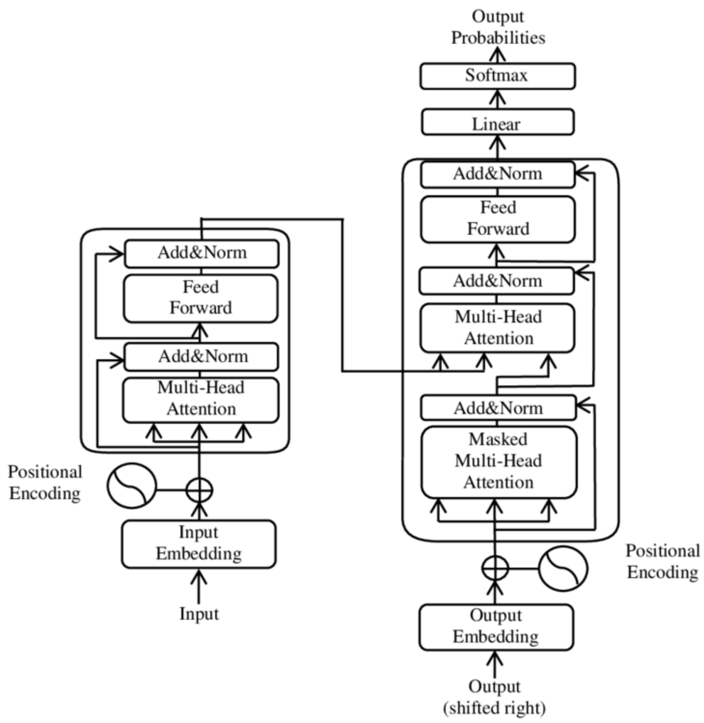

Fig 3: The Transformer - model architecture

Transformer has two major parts the Encoder and Decoder. Encoder converts the data input into the meaningful vector. The Embedding goes through the following steps.
Input Embedding→Positional Encoding→ Multi-Head Self Attention→ Add & Normalize
→ Feed Forward Network→ Add & Normalize

Positional encoding is responsible for telling which word comes first or later.

For the positional encoding sine and cosine functions are used.

The attention mechanism is the main part of the Transformer. It uses Query (Q), Key(K) and Value (V) and using these the attention score is calculated.

$$\text{Attention}(Q,K,V)=\text{softmax}(QK^T / \sqrt{dk}\,)\, V$$

Where dk is dimension of vector embedding

Also multi attention architecture is used in Transformer that is it uses many heads.

For Example:

- Head 1 → grammar check in the sentence
- Head 2 → semantic meaning in the sentence
- Head 3 → subject-object relation between various parts of sentence

The advantage of multi head is it learns multiple relationships simultaneously

And it gives better contextual understanding

After attention each token passes through a neural network.

This introduces non-linearity.

Decoder consists mainly of:

1. Masked Self-Attention
2. Encoder–Decoder Attention
3. Feed Forward Network

Masking is done to ensure the model doesn’t see future words during training and it helps in calculation of error in prediction on the masked word.

Decoder focuses on important encoder outputs.

Example:

Input: "The book is nice"

In generating translation for “book”, decoder attends strongly to encoder representation of “book”.

Advantages of Transformer are:

a) Parallel Processing

1. All words processed together.
2. Works faster on GPUs due to parallelism.

b) Long Dependency Learning

It understands the distant relationships.

Transformer uses self-attention, enables parallel learning of relationships between all words in a sequence which is optimized and efficient. It is extendible to Time Series data with embedding self-attention encoding and decoding.

c) Necessary python codes:

Building Transformer Model

```
input_layer = Input(shape=(X_train.shape[1], X_train.shape[2]))

# Multi-head attention

attention_output = MultiHeadAttention(
    num_heads=4,
    key_dim=32
)(input_layer, input_layer)

# Create model

model = Model(inputs=input_layer, outputs=output_layer)
model.compile(
    optimizer=Adam(learning_rate=0.001),
    loss='mse',
    metrics=['mae']
)
model.summary()
```

Model: "functional"

Layer (type)	Output Shape	Param #	Connected to
input_layer (InputLayer)	(None, 72, 9)	0	-
multi_head_attenti… (MultiHeadAttentio…	(None, 72, 9)	5,001	input_layer[0][0… input_layer[0][0]
add (Add)	(None, 72, 9)	0	input_layer[0][0… multi_head_atten…
layer_normalization (LayerNormalizatio…	(None, 72, 9)	18	add[0][0]
dense (Dense)	(None, 72, 128)	1,280	layer_normalizat…
dropout_1 (Dropout)	(None, 72, 128)	0	dense[0][0]
dense_1 (Dense)	(None, 72, 9)	1,161	dropout_1[0][0]
add_1 (Add)	(None, 72, 9)	0	layer_normalizat… dense_1[0][0]
layer_normalizatio… (LayerNormalizatio…	(None, 72, 9)	18	add_1[0][0]
global_average_poo… (GlobalAveragePool…	(None, 9)	0	layer_normalizat…
dense_2 (Dense)	(None, 128)	1,280	global_average_p…
dropout_2 (Dropout)	(None, 128)	0	dense_2[0][0]
dense_3 (Dense)	(None, 9)	1,161	dropout_2[0][0]

Total params: 9,919 (38.75 KB)
Trainable params: 9,919 (38.75 KB)
Non-trainable params: 0 (0.00 B)

Table 1: The summary of the Transformer Model

4. Result and Discussion

Forecast for 2 days

```
forecast_df = pd.DataFrame(
    forecast_actual,
    columns=features
)
forecast_df.insert(0, "Date", future_dates)
print(forecast_df.head())
```

```
     Date                  HC Air temperature [°C]   Dew Point [°C]  \
0 2024-09-18 09:00:00        23.554495      19.563560
1 2024-09-18 10:00:00        23.571192      19.532053
2 2024-09-18 11:00:00        23.570145      19.491289
3 2024-09-18 12:00:00        23.568205      19.450510
4 2024-09-18 13:00:00        23.589203      19.429256
```

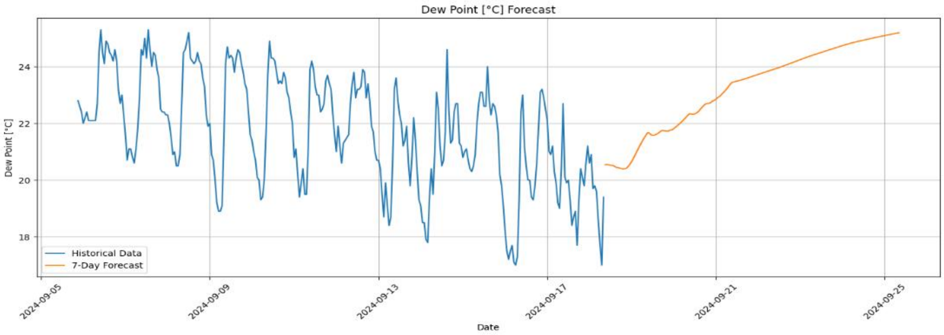

Fig 4: Forecasting on Dew Point for Next Two Days

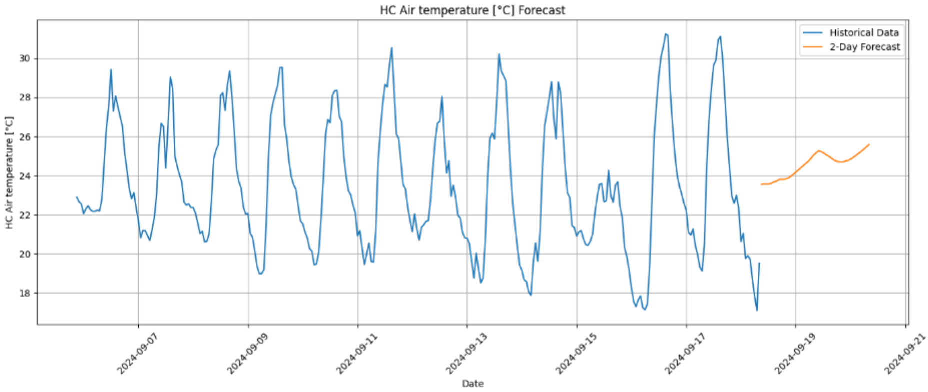

Fig 5: Forecasting on HC Air Temperature

The transformer based model uses attention mechanism to find the vectors which are close by and understands the context and it encodes it then for prediction it uses decoder part of Transformer and predicts for the next 2 days.

Training Loss Graph

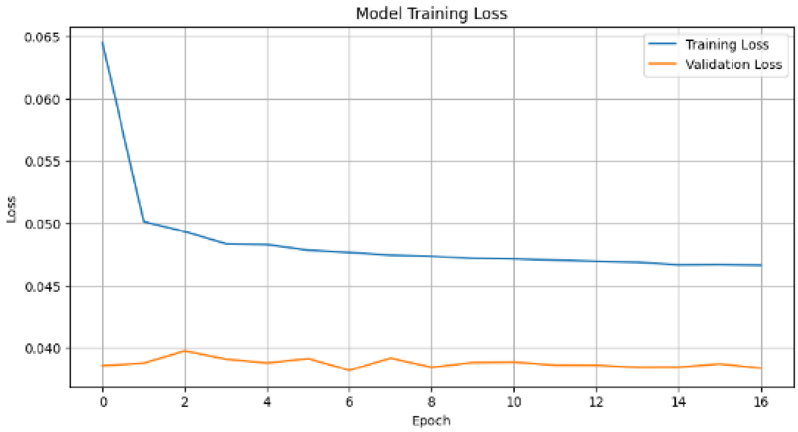

Fig 6: Model Training Loss

5. Limitations and Future Work

Although this paper contains Transformer based model some hybrid models and also be tested with the new models which recently came like minLSTM, minGRU. Use of N-Beats and N-Hits for prediction can also be used. Also, Large Language Model can be explored for Time Series problems. Use of Gen AI can be also thought of.

6. Conclusion

This paper discusses the use of Transformer model for Time Series forecasting and it takes advantage of parallelism feature of transformer to attain the speed, also it uses attention mechanism to understand the relation between token embedding and does prediction to a good extent.

References

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). *Attention is all you need*. In *Advances in neural information processing systems* (pp. 5998–6008).
2. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies* (pp. 4171–4186).
3. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving language understanding by generative pre-training*. OpenAI.
4. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach (arXiv Preprint No. arXiv:1907.11692). arXiv.
5. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21, 1–67.
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., & Dehghani, M. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In International Conference on Learning Representations.
7. Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 2978–2988).
8. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. In Advances in neural information processing systems (pp. 5753–5763).
9. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter (arXiv Preprint No. arXiv:1910.01108). arXiv.
10. Uluocak, M., Taplamacioglu, M. C., & Sayan, H. H. (2022). Air temperature forecasting using hybrid deep learning models. Applied Artificial Intelligence, 36(1).

11. Wu, Z., Lan, Y., Yan, J., & Li, X. (2021). Soil moisture prediction based on PSO-LSTM model for citrus orchard. Computers and Electronics in Agriculture, 183, Article 106093.
12. Venkatachalam, K., Manikandan, R., & Sangeetha, S. (2022). Data-driven deep weather forecasting hybrid model using transductive long short-term memory. Journal of Ambient Intelligence and Humanized Computing, 13, 421–435.
13. Khalid, M., Mahmoud, A., & El-Mougy, S. (2022). Temperature forecasting using support vector regression and regression trees. International Journal of Advanced Computer Science and Applications, 13(2).
14. Wang, J., Wang, J., & Zhang, Z. (2015). Deep learning approaches for temperature forecasting compared with statistical models. Energy, 93, 1456–1468.
15. Aghelpour, P., Kisi, O., & Varshavian, M. H. (2019). Temperature forecasting using support vector regression combined with firefly algorithm. Theoretical and Applied Climatology, 138, 1205–1217.
16. Xu, W., Peng, H., & Ding, J. (2017). Wavelet transform-based support vector machine for air temperature prediction. Atmospheric Research, 192, 1–10.
17. Silva, A., Gomes, R., & Pereira, J. (2020). Air temperature prediction using deep learning and long short-term memory networks. Environmental Modelling & Software, 134, Article 104840.
18. Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model.