Analysis of LSTM Networks for Reduced Environmental Impact in Time Series Forecast

Analysis of LSTM Networks for Reduced Environmental Impact in Time Series Forecast

Aurora Martiny
Department of Electronics and Telecommunications
Politecnico di Torino, Italy
aurora.martiny@polito.it

Michela Meo
Department of Electronics and Telecommunications
Politecnico di Torino, Italy
michela.meo@polito.it

Greta Vallero
Department of Electronics and Telecommunications
Politecnico di Torino, Italy
greta.vallero@polito.it

***Abstract*—The increasing adoption of Deep Learning (DL) algorithms for time series forecast has led to a significant environmental concern due to the high computational demands and associated carbon footprint. This study investigates the environmental impact of DL models, particularly Long Short-Term Memory (LSTM) networks, for time series forecasting tasks where frequent retraining of models is essential. We conduct an empirical analysis of carbon emissions produced by LSTM models trained on two distinct time series datasets. By systematically varying model hyperparameters (epochs, train-test split, number of layers and neurons per layer), and by reducing the number of models or the number of input features, we aim to understand the impact of these changes on carbon emissions and model accuracy. Our contributions include a comprehensive analysis of carbon emissions during model training and the identification of possible trade-offs between emissions and accuracy. The findings indicate that strategic adjustments can significantly reduce environmental impact while maintaining satisfactory accuracy levels.**

***Index Terms*—Machine Learning, Carbon Emissions, LSTM**

I. Introduction

The rapid and expansive growth of Machine Learning (ML) algorithms has profoundly transformed the landscape of contemporary problem-solving domains, including network traffic and energy production forecasting. This explosion is largely attributed to advancements in technology – particularly with the advent of Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs) – and the unprecedented availability of data [19]. Indeed, ML algorithms are capable of processing extensive datasets and extracting meaningful insights. Within this huge application spectrum, time series analysis is particularly noteworthy due to its ability to forecast future events based on historical data, thereby facilitating informed strategic planning.

Despite their transformative potential, ML algorithms have a considerable environmental impact, especially in the context of time series analysis where models require continuous or periodic retraining to remain updated on more recent record information, with the goal of maintaining accuracy and relevance. The computational demands of training sophisticated models, such as Long Short-Term Memory (LSTM) networks [8], necessitate significant energy consumption, leading to increased carbon emissions. As the deployment of ML technologies scales, addressing their environmental impact becomes increasingly imperative [14].

This study addresses the environmental implications associated with the deployment of Deep Learning (DL) algorithms, with a specific focus on time series analysis. We conduct an empirical investigation into the carbon emissions produced by an LSTM model trained on two distinct time series datasets. Our approach involves systematically varying several factors, including training hyperparameters and model inputs to observe the resultant changes in carbon emissions and model performance.

Our contributions are twofold: (i) we present a detailed quantification of carbon emissions associated with training LSTM models, providing a clear understanding of the environmental footprint produced under different training and architectural configurations; (ii) we identify optimal trade-offs between carbon emissions and model performance, offering critical insights into strategies for minimizing environmental impact while preserving model efficacy.

The structure of the paper is organized as follows: Section II reviews existing literature on the environmental impact of ML algorithms; Section III provides datasets description; Section IV introduces the baseline LSTM model, detailing modifications to training hyperparameters and model inputs; Section V describes the experimental setup for reproducibility and analyzes the main results, highlighting trade-offs between carbon emissions and model performance; finally, Section VI summarizes main findings and suggests directions for future research.

II. Related Work

In recent years, there has been an increasing interest in the environmental impact of using DL algorithms [14], [9]. The main concern regards the energy consumption associated with training and deploying these models, which can significantly contribute to the release of carbon emissions. Researchers have particularly focused on highlighting the substantial energy demands of training large DL models, emphasizing the need for more sustainable AI practices [13]. While the environmental impact of tasks such as Natural Language Processing (NLP) has gained attention, as seen in studies like [15], and [4], similar considerations are lacking in areas like network traffic and PV panel energy forecasting using LSTM networks. This gap highlights the need to consider these issues in a broader

TABLE I
COMMONLY USED OPEN ACCESS CARBON TRACKING TOOLS.

Python Libraries	Online Tools
CodeCarbon [12]	Green Algorithms [11]
Carbontracker [3]	
Eco2AI [5]	
experiment-impact-tracker [7]	ML CO2 Impact [10]
Cumulator [16]	
energyusage	

range of applications, including those in this paper. These tasks are relevant in the context of optimizing fifth-generation (5G) networks and beyond, where data traffic is highly dynamic and voluminous and where Base Stations (BSs) are powered by PV panels to balance energy consumption with renewable energy production. Existing studies focused solely on improving prediction accuracy [17]. However, these algorithms require periodic retraining to adapt to changes in traffic data, which significantly increases the energy consumption needed for their deployment. This poses substantial sustainability challenges due to the heightened computational demands and associated carbon emissions. Similarly, research on PV panel energy production forecasting has been directed towards enhancing the reliability of predictions, while overlooking the associated environmental costs [1].

Tools for emissions calculation are well-established in the literature, providing various options tailored to different study requirements. These tools can be integrated into Python code or accessed online, and they cater to both Windows and Linux environments. Examples are reported in Table I. These tools are crucial for researchers aiming to quantify and mitigate the environmental impact of their DL models.

III. DATASETS DESCRIPTION

In this work, we employ two distinct real-world datasets. The first dataset consists of traffic volume data, while the second dataset collects PV panel energy production data. Both datasets belong to the domain of time-series data but differ significantly in terms of stability. Indeed, the production data derived from photovoltaic (PV) panels clearly demonstrated a more predictable trend compared to the inherently volatile nature of traffic data.

1) Traffic Data from Italian MNO: In this work, we use the traffic data employed in [18], confidentially provided by a Network Operator. The dataset collects the traffic volume, in bit, from 1 420 BSs spanning two months in 2015, with 15 minutes granularity. We divide the city into seven distinct zones, listed and described in Table II. Each zone is chosen for its specific typical activities within an urban environment, exemplified in Fig. 1.

2) PVWatts Energy Production Data: The second dataset is provided by [6]. It collects the hourly solar energy production data of a PV panel located in Turin. For each hour, in addition

TABLE II
ZONES IN MILAN AND RELATIVE ACTIVITIES

	Zone	Activities
1	Business	Traffic peaks often from 11:00 AM to 6:00 PM, Monday through Friday.
2	Residential	Increased traffic in the evening (after 6:00 PM, when people return home from work or school).
3	Train station	High activity coinciding with the start and end of typical working hours (around 7:00 AM to 9:00 AM and around 5:00 PM to 7:00 PM).
4	San Siro	The operating hours for soccer stadiums can vary depending on match schedules and events.
5	Politecnico di Milano	Area frequented by students, experiencing heightened activity levels from 8:00 AM to 7:00 PM.
6	Industrial	Industrial operations can vary based on sector-specific needs.
7	Rho Fiere	Traffic in this area can vary according to the schedules of events and exhibitions.

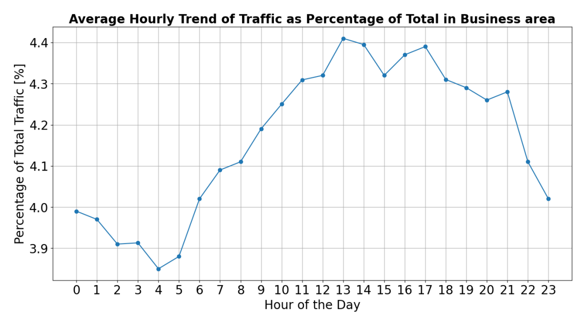

Fig. 1. Average hourly distribution of traffic as a percentage of daily total, specifically considering the Business area.

to the produced energy, in Wh, many features are given. We categorize them into three main groups, each representing a distinct aspect of the solar energy generation environment. These categories include:

1) **Solar Radiation:**
   - Beam Irradiance (W/m^2)
   - Diffuse Irradiance (W/m^2)
   - Plane of Array Irradiance (W/m^2)
2) **Temperature Conditions:**
   - Ambient Temperature ($^{\circ}C$)
   - Cell Temperature ($^{\circ}C$)
3) **Wind Dynamics:**
   - Wind Speed (m/s)

Fig. 2 shows a typical day for each month, highlighting how production varies across different periods. This variation is evident both in the number of active hours (production hours) and in the peak production achieved during the day.

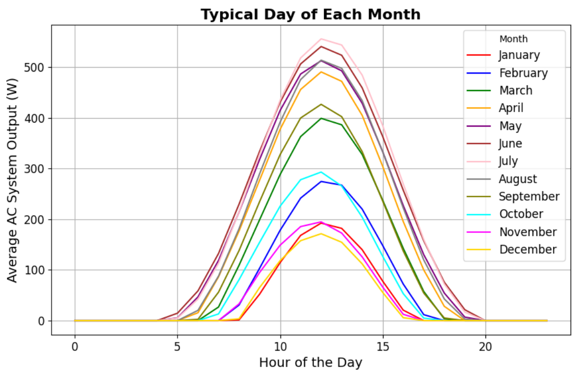

Fig. 2. Variation in PV production across months: average production days per month highlighting shifts in active hours and maximum output between seasons.

IV. Methodology

In this section, we discuss the methodology for time series prediction and carbon emission monitoring used in this work.

A. LSTM-Based Prediction

In this work, we use an LSTM-based recurrent neural network, commonly employed in literature for time series prediction tasks [8]. LSTMs have proven effective in handling sequential data and retaining long-term dependencies, as noted in [17], [18].

For PV panel energy forecasting, it is crucial to account for daily and seasonal patterns, as sunlight variability throughout the day and across seasons directly impacts energy generation. Similarly, network traffic prediction relies heavily on temporal dependencies. Network usage behavior exhibits recurring patterns, such as peak usage hours, weekday versus weekend fluctuations, and periodic events like promotions or holidays that trigger increased traffic. These dependencies make historical observations essential for accurate future predictions.

For traffic volume forecasting, at time t, the model predicts the next 4 samples, i.e., the traffic volume from $t+1$ to $t+4$ (representing 1 hour). To do this, it receives as input the 10 previous traffic volume samples (covering 2.5 hours), from $t-10$ to $t-1$. The model is composed of 4 LSTM layers, each with 64 nodes. The output layer includes a dense layer with a linear activation function, aligning with the regression nature of the task. We use a testing period of 14 days and a training period of 46 days. The model is trained for 500 epochs. We refer to this configuration as *baseline*.

For predicting PV panel production at time t, the model outputs a single sample and receives as input the 24 previous samples. For this dataset, the *baseline* configuration consists of 3 LSTM layers, each with 256 nodes. As with the traffic prediction model, the output layer uses a linear activation function. We use 25% of the whole dataset as the test set and train for 250 epochs. We then post-process the predictions for the PV panel dataset to avoid unrealistic negative values.

We train both models using Xavier initialization, the Adam optimizer, and the Mean Absolute Error (MAE) as the loss function. For both datasets, the learning rate of the Adam optimizer is 0.001.

B. Manipulating Model Input

The inputs of the model are strongly correlated with computational demand, which directly impacts energy usage, as energy depends on both power and time. Since our goal is to define a trade-off between model accuracy and its environmental impact, we evaluate the influence of the size of the training and test sets on performance.

We explored this impact on two distinct levels:

- Variation in the training set size with a fixed test set: In this configuration, we keep the test set size constant while modifying the size of the training set. Thus, we either add or remove rows from the training set, increasing or decreasing the amount of historical data available to the model. Since each row represents traffic data captured in 15-minute intervals, the temporal window considered by the model is also affected.
- Variation in the partitioning between training and testing sets: In this second type of experiment, we vary the way the entire dataset is split between the training and test sets, modifying the model's generalization ability.

C. Training Hyperparameters

In addition to varying the input data, our analysis focuses on modifying critical components of LSTM architectures. Specifically, we investigate the effects of varying the number of LSTM layers, the number of nodes per layer, and the number of epochs. By varying these parameters separately, we aim to understand their respective impacts on model performance and sustainability, and ultimately find an optimal configuration that balances these factors effectively.

D. Number of Models

The overall ecological footprint of a model is influenced not only by the training hyperparameters and input model, but also by the number of trained models. To address this in the context of the network dataset, we consider two different approaches:

- A single model for each individual BS (M4BS): In this case, for each BS, we train a single model as described in previous sections, using the corresponding BS traffic trace.
- A distinct model for each zone (M4Z): Here, we aggregate the traffic traces of the BSs belonging to a given zone and we train a single model, as detailed above, which we use for traffic predictions for all those BSs.

E. Feature selection

For the PV panel prediction problem, we study the impact of the variation of the input features. For the prediction of the amount of generated energy at time t, we use the three feature categories (solar radiation, temperature conditions, and wind dynamics) or a subset of them, collected between time stamps $t-24$ and $t-1$.

F. Carbon Emission Monitoring

For monitoring carbon emissions, we use a Python library called CodeCarbon[1]. This open-source tool calculates the carbon emissions by tracking power usage associated with the CPU and applying regional electricity carbon intensity factors. Specifically, it considers the energy mix typical of the specified country and provides the carbon emissions in kg. Our monitoring accounts for both the training and testing phases.

V. Experimental Results

The purpose of this investigation is to explore the relationship between carbon emissions and the accuracy of the LSTM-based neural networks tailored to time-series-related tasks and find a trade-off between these two aspects.

A. Experimental setup

We use the CodeCarbon python library to track the carbon emissions, selecting Italy as the location when running the tracker. We perform every experiment on a machine equipped with an Intel Core i7-1165G7 CPU operating at 2.80GHz and 16 GB of available RAM.

B. Impact of hyperparameters and inputs

We first examine the impact of the training hyperparameters on both model accuracy and carbon emissions. Fig. 3 and Fig. 4 present the results obtained by varying on the x-axis the number of epochs for the Polimi zone of network traffic data. The figures report the emissions, in Kg of CO_2-equivalents, and the accuracy. In both cases, confidence intervals have been obtained by repeating each experiment 20 times. It is worth noting that the accuracy values are relatively low, which can likely be attributed to the inherent noise in the data. This is a common challenge when dealing with network time series data, where fluctuations and irregularities in the traffic can reduce model performance, as observed in similar related studies [2].

In both figures, the x-axis represents the number of epochs, ranging from 50 to 500. The y-axis reports the median and mean values, with the median shown in red and the mean in purple dashed lines. As the number of epochs increases, we observe a general trend of improving accuracy, stabilizing around 300 epochs. This suggests that training beyond 300 epochs yields diminishing returns in terms of accuracy improvements. Overall, the most interesting aspect is the evident linear increase in emissions over the number of epochs. Figs. from 5 to 13 illustrate the model accuracy and training emissions, varying several parameters considering the two datasets discussed in section III. In particular, in Fig. 9 we vary, on the x-axis, the number of epochs, in Figs. 7, 12 the number of layers, in Figs. 8, 13 the number of nodes per layer, in Figs. 6, 11 the training set size and in Figs. 5, 10 the testing set size. It is evident from Figs. 5 to 8 that accuracy slightly improves or even plateaus when increasing the model complexity, while emissions always increase. Indeed, increasing the number of training days results in higher emissions due to the required extended computational effort. Conversely, increasing the number of testing days initially leads to higher emissions, but beyond a certain threshold, emissions start to decrease. This is because the increase in testing days implicitly reduces the number of training days, thereby lowering the overall training emissions. Findings with the second dataset (Figs. 9-13) follow a similar trend. Emissions increase linearly with model complexity, while accuracy shows only marginal improvement with increases in epochs, layers, and nodes per layer. However, variations in the test and training set sizes significantly impact accuracy, highlighting that a trade-off between accuracy and emissions is impractical.

C. Impact of the number of used models

In this part of the study, we analyze the impact of employing the M4BS and M4Z approaches, which utilize a single model per BS and per zone, respectively. For this analysis, we employ a single-layer LSTM network trained for 100 epochs, because of the marginal gain with large number of both epochs and layer, while increasing the carbon emissions, as discussed above.

We compare the accuracy and carbon emissions of this architecture, referred to as *Updated*, with the baseline, with the M4BS and M4Z configurations. The results are presented in Fig. 14, where different zones are shown on the x-axis, with M4BS configurations in blue and green bars, and M4Z configurations in orange and purple bars. The histograms illustrate a significant reduction in emissions—more than fourfold—achieved through the reduction in the number of layers and/or epochs. Particularly noteworthy is the substantial improvement observed with the M4Z approach, where a single model is trained for each zone encompassing 8 BSs.

Furthermore, these emission reductions do not entail significant accuracy losses across any combination. These findings indicate that substantial emission reductions can be achieved by using a single model for multiple BSs within a zone, with minimal impact on accuracy.

D. Impact of the feature selection

Fig. 15 depicts the accuracy (left y-axis) and carbon emissions (right y-axis) while varying the input features of the model used for predicting energy generation. Each color represents a distinct feature configuration, with the *Updated* model utilizing a single-layer LSTM network trained for 100 epochs instead of the 500 epochs in the baseline model. Reducing the number of epochs is advantageous for minimizing carbon emissions. Interestingly, selecting features excluding wind conditions shows improved accuracy, nearly matching that of the baseline model.

Overall, the experimental results highlight that considerations regarding model complexity should be carefully tailored to achieve lower emissions, as increasing complexity often

[1] https://codecarbon.io/

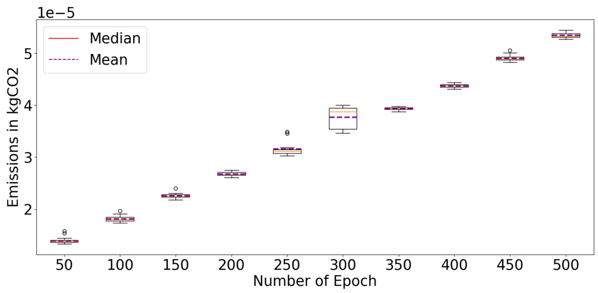

Fig. 3. Emissions confidence intervals across epochs for Polimi zone network traffic data: results from 20 experiment repetitions.

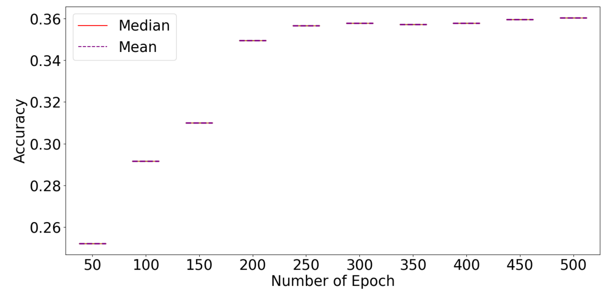

Fig. 4. Accuracy confidence intervals across epochs for Polimi zone network traffic data: results from 20 experiment repetitions.

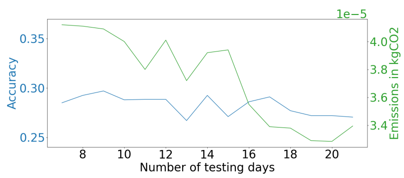

Fig. 5. Accuracy and emissions variation over the number of testing days for FS Zone in network traffic dataset.

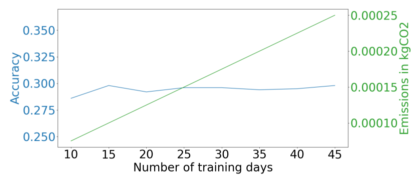

Fig. 6. Accuracy and emissions variation over the number of training days for FS Zone in network traffic dataset.

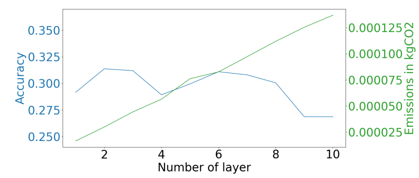

Fig. 7. Accuracy and emissions variation over the number of layers for Polimi Zone in network traffic dataset.

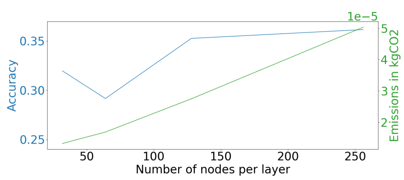

Fig. 8. Accuracy and emissions variation over the number of nodes per layer for Polimi Zone in network traffic dataset.

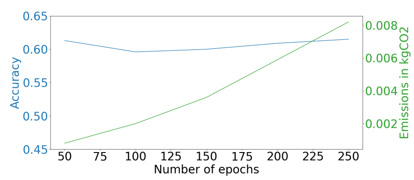

Fig. 9. Accuracy and emissions variation over the number of epochs in PV panel dataset.

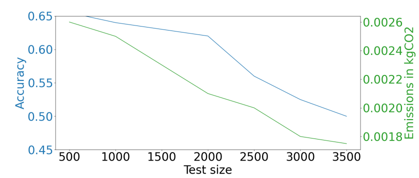

Fig. 10. Accuracy and emissions variation over the test size in PV panel dataset.

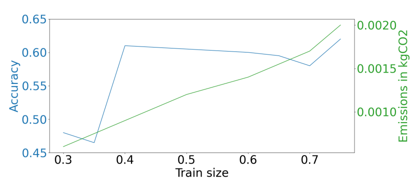

Fig. 11. Accuracy and emissions variation over the train size in PV panel dataset.

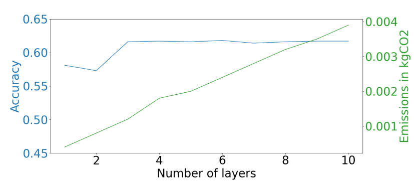

Fig. 12. Accuracy and emissions variation over the number of layers in PV panel dataset.

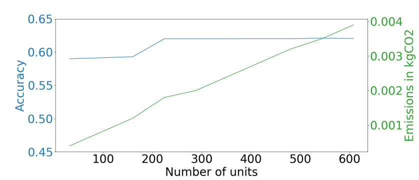

Fig. 13. Accuracy and emissions variation over the number of nodes per layer in PV panel dataset.

results in minimal accuracy gains while significantly raising the environmental impact.

VI. Conclusion

Due to the high penetration of DL models, the investigation of their environmental impact has become urgent. This study emphasizes this aspect within the time-series domain, where frequent retraining makes the emissions-accuracy trade-off particularly significant. Our research underscores the necessity of a monitoring phase to identify key parameters for optimizing the emissions-accuracy balance. We found that significant emission reductions can be achieved with minimal accuracy loss by adjusting parameters such as the number of epochs, layers, and number of trained models. We consider a dataset, reporting the hourly Radio Access Network traffic volume and another, collecting the hourly PV panel generation. With the former, we achieve an average of 94.9% reduction in emissions with a corresponding average accuracy decrease no larger than 2.8%, by adjusting the number of epochs and layers and by reducing the number of used models. Similarly, for the latter, emissions are reduced at most by 93.6%, with only a 0.16% reduction in accuracy, by modifying

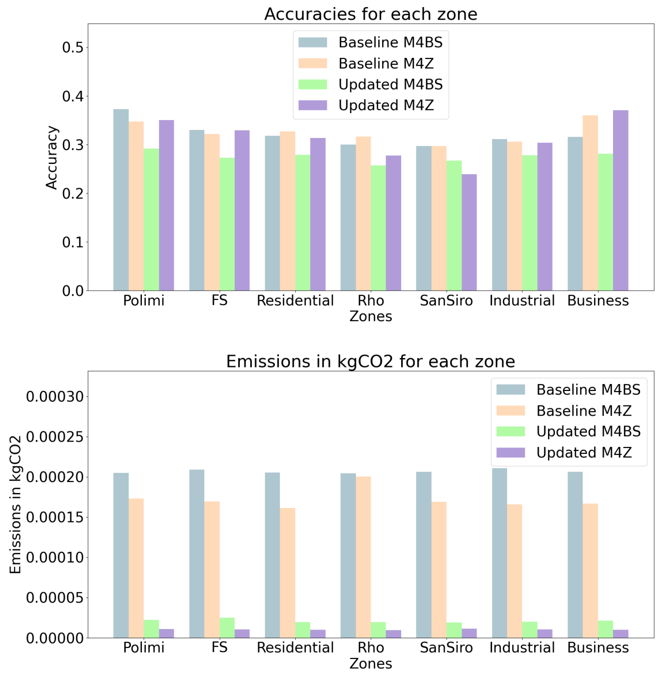

Fig. 14. Comparison between M4Z and M4BS values for testing accuracy and training emissions. The histograms illustrate four distinct cases: (1) Baseline architecture, (2) Baseline architecture with M4Z, (3) Architecture with updated hyperparameters and M4BS, and (4) Architecture with updated hyperparameters and M4Z.

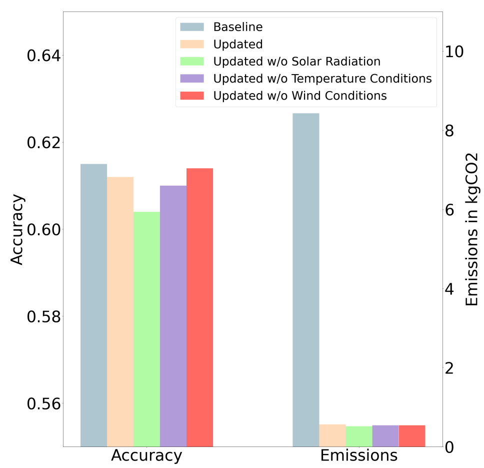

Fig. 15. Testing accuracy and training emissions for different model formulations: baseline, model with updated parameters, and updated models excluding solar radiation, temperature conditions, and wind conditions.

the number of epochs and selecting the most relevant input features. However, even if the study provides valuable insights into the emissions-accuracy trade-offs of LSTM networks, future research is needed to extend these findings. Specifically, repeating the same analysis using a broader set of DL models could yield more generalized conclusions. Moreover, testing the models in diverse hardware environments, including GPUs and cloud-based infrastructures, will help assess the feasibility of the proposed methods in real-world scenarios where energy consumption and carbon emissions vary considerably based on hardware configurations.

Acknowledgment

This paper was supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART", Focused Project R4R).

This paper was supported by the project "National Center for HPC, Big Data and Quantum Computing", CN00000013 (Bando M42C – Investimento 1.4 – Avviso Centri Nazionali" – D.D. n. 3138 of 16.12.2021, funded with MUR Decree n. 1031 of 17.06.2022).

References

[1] R. Ahmed, V. Sreeram, Y. Mishra, and M. Arif. A review and evaluation of the state-of-the-art in pv solar power forecasting: Techniques and optimization. *Renewable and Sustainable Energy Reviews*, 2020.

[2] D. Andreoletti, S. Troia, F. Musumeci, S. Giordano, G. Maier, and M. Tornatore. Network traffic prediction based on diffusion convolutional recurrent neural networks. In *IEEE INFOCOM WKSHPS 2019*, pages 246–251, 2019.

[3] L. F. W. Anthony, B. Kanding, and R. Selvan. Carbontracker: Tracking and predicting the carbon footprint of training deep learning models, 2020.

[4] N. Bannour, S. Ghannay, A. Névéol, and A.-L. Ligozat. Evaluating the carbon footprint of NLP methods: a survey and analysis of existing tools. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 11–21, Virtual, Nov. 2021. ACL.

[5] S. Budennyy, V. Lazarev, N. Zakharenko, A. Korovin, O. Plosskaya, D. Dimitrov, V. Akhripkin, I. Pavlov, I. Oseledets, I. Barsola, et al. Eco2ai: carbon emissions tracking of machine learning models as the first step towards sustainable ai. In *Doklady Mathematics*, pages 1–11. Springer, 2023.

[6] A. P. Dobos. Pvwatts version 5 manual. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2014.

[7] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau. Towards the systematic reporting of the energy and carbon footprints of machine learning, 2022.

[8] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] C. C. J. Kuo and A. M. Madni. Green learning: Introduction, examples and outlook, 2022.

[10] A. Lacoste, A. Luccioni, V. Schmidt, and T. Dandres. Quantifying the carbon emissions of machine learning, 2019.

[11] L. Lannelongue, J. Grealey, and M. Inouye. Green algorithms: Quantifying the carbon footprint of computation, 2020.

[12] K. Lottick, S. Susai, S. A. Friedler, and J. P. Wilson. Energy usage reports: Environmental awareness as part of algorithmic accountability, 2019.

[13] D. Patterson, J. Gonzalez, Q. Le, C. Liang, L.-M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean. Carbon emissions and large neural network training, 2021.

[14] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni. Green ai. *Commun. ACM*, 63(12):54–63, nov 2020.

[15] E. Strubell, A. Ganesh, and A. McCallum. Energy and policy considerations for deep learning in nlp, 2019.

[16] M. J. T. Trebaol, M.-A. Hartley, and H. S. Ghadikolaei. A tool to quantify and report the carbon footprint of machine learning computations and communication in academia and healthcare. Infoscience EPFL: record 278189, 2020.

[17] H. D. Trinh, L. Giupponi, and P. Dini. Mobile traffic prediction from raw data using lstm networks. In *IEEE 29th PIMRC 2018*, pages 1827–1832, 2018.

[18] G. Vallero, D. Renga, M. Meo, and M. A. Marsan. Greener ran operation through machine learning. *IEEE Transactions on Network and Service Management*, 16(3):896–908, 2019.

[19] J. Xu, W. Zhou, Z. Fu, H. Zhou, and L. Li. A survey on green deep learning, 2021.