



DAerosol-NTM: applying deep learning and neural Turing machine in aerosol prediction

Zahra-Sadat Asaei-Moamam^{1,2} · Faramraz Safi-Esfahani^{1,2} · Seyedali Mirjalili³ · Reza Mohammadpour⁴ · Mohamad-Hosein Nadimi-Shahraki^{1,2}

Received: 1 November 2022 / Accepted: 10 July 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

The pollution caused by aerosol (particulate matter) has a detrimental impact on urban environments, particularly in terms of socio-economic factors and public health. Aerosol particles, ranging in size from 1 nm to 100 μm , can easily penetrate organic tissues, carrying toxic gaseous compounds and minerals such as carbon monoxide, ozone, nitrogen dioxide, and sulfur dioxide. Recent advancements in neural network technology, combined with deep learning techniques, have made it possible to predict surges in aerosol pollution. In this study, we introduce DAerosol-NTM, a deep learning framework that utilizes the latest developments in neural Turing machines (NTMs) to access external memory. When compared with four baseline studies that employ multilayer perceptron (MLP), deep neural networks (DNNs), long short-term memory (LSTM), and deep LSTM (DLSTM), DAerosol-NTM significantly improves prediction accuracy by 8–31% and precision by 46–91% and reduces the root mean square error (RMSE) by 24–85%. Additionally, DAerosol-NTM incorporates up to 20 years of particulate matter data in its external storage, making it the first model capable of predicting aerosol pollution surges. By analyzing the data from the previous 96 h, the optimal time interval before and after the aerosol event (TIBAAE) enables the prediction of aerosol events within the following 24 h.

Keywords Aerosol · Neural Turing machines (NTMs) · Long short-term memory (LSTM) · Deep learning · Air quality index (AQI) · Particulate matter 2.5 ($\text{PM}_{2.5}$) · Particulate matter 10 (PM_{10})

1 Introduction

Particulate matter, commonly known as aerosol, comprises fine airborne particles ranging from approximately 1 nm to 100 μm [1, 2]. These particles are large enough to carry toxic gaseous compounds, minerals, and infectious diseases, yet small enough to penetrate the lungs and organic

tissue, introducing harmful elements into the body and causing adverse reactions. The air quality index (AQI) serves as a widely accepted indicator that takes into account key parameters shaping air quality and safety. The AQI incorporates six air pollutants: carbon monoxide (CO), ozone (O_3), nitrogen dioxide (NO_2), sulfur dioxide (SO_2), particulate matter of 2.5 microns ($\text{PM}_{2.5}$), and

✉ Faramraz Safi-Esfahani
fsafi@iaun.ac.ir

Zahra-Sadat Asaei-Moamam
zahraasaei@sco.iaun.ac.ir

Seyedali Mirjalili
seyedalii.mirjalili@griffith.edu.au

Reza Mohammadpour
Re.Mo564@iau.ac.ir

Mohamad-Hosein Nadimi-Shahraki
nadimi@iaun.ac.ir

¹ Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran

² Big Data Research Center, Najafabad Branch, Islamic Azad University, Najafabad, Iran

³ Centre for Artificial Intelligence Research and Optimisation, Torrens University, Melbourne, Australia

⁴ Department of civil engineering, Estahban Branch, Islamic Azad University, Estahban, Iran

particulate matter of 10 microns (PM_{10}) [3, 4]. The pollutants with the highest concentrations determine the AQI value, with $PM_{2.5}$ and PM_{10} typically playing prominent roles. Evaluation involves considering 22 meteorological parameters in addition to CO, O₃, NO₂, and SO₂, making a total of 26 parameters. The AQI is presented on a continuous scale from 0 to 500 and is divided into six pollution categories for easy public comprehension. This classification system effectively communicates the impact of air quality on human health and provides a straightforward numerical reference for managing the risks associated with outdoor activities. Previous endeavors have sought to estimate and forecast the key parameters of the AQI or the AQI value itself, employing machine learning and neural network methods while comparing the performance of various approaches.

Initially, Kim [5] and Xayasouk [6] implemented and compared frameworks for forecasting AQI parameters in the Seoul Metropolitan Area, specifically $PM_{2.5}$ and PM_{10} , using deep autoencoder (DAE), deep neural network (DNN), and deep long short-term memory (DLSTM) models. Data were collected from 25 pollutant and weather stations. The DLSTM method demonstrated superior performance compared to the DAE and DNN models due to improved normalization and transparent data processing. However, both studies utilized meteorological data from a single weather station, which may be insufficient for accurately predicting aerosol parameters across a broader district in Seoul. Moreover, focusing solely on $PM_{2.5}$ and PM_{10} among the six air pollutants does not provide a comprehensive understanding of the AQI.

In contrast, the article by Sharma [7] deviates from the previous studies by considering the seasonal patterns of AQI parameters. It incorporates a substantial amount of time-series data from the pollution control center of India, estimating the AQI value as a whole rather than its individual components. The article employs a deep neural network architecture, comparing recurrent neural networks (RNN) and DLSTM with the autoregressive integrated moving average (ARIMA) model [8]. Once again, the DLSTM model outperforms other approaches, showcasing its advantage in handling data gaps. Similar to the aforementioned baseline methods, this framework does not utilize meteorological data from multiple districts without clear preprocessing and data normalization.

The approach presented by Pengfei et al. [9] involves applying a DNN model after precise data processing and normalization to forecast $PM_{2.5}$ levels in the Wuhan district. The study analyzes the data using super vector machine (SVM) models and a traditional backpropagation network with three hidden layers. The results demonstrate that the DNN model outperforms shallow learning methods in terms of prediction accuracy, error consideration, and

regression rate calculation. However, this study solely relies on Wuhan's pollution data and does not incorporate meteorological data. Furthermore, the selected model proves inadequate for time-series data due to weight disappearance.

Ultimately, all baseline studies [5–7, 9] exhibit certain shortcomings, which can be detailed as follows: (1) low model accuracy due to the lack of long-term memory for preserving data history; (2) utilization of a single meteorological station for a large city without representing local data; (3) failure to utilize both meteorological and local pollution parameters as effective indicators for estimating air quality; and (4) inability to predict future aerosol events.

To address the aforementioned limitations and achieve accurate predictions of aerosol and AQIs ahead of time, this research article presents the following hypotheses: The first hypothesis suggests that incorporating external memory into a deep neural network enhances the prediction accuracy, precision, and root mean square error (RSME) of PM_{10} , $PM_{2.5}$, and the AQI quality control index by considering a long history of meteorological information. The second hypothesis proposes that incorporating external memory into a deep neural network, along with meteorological and local pollution data, improves the time intervals before and after the aerosol event (TIBAAE) in relation to PM_{10} , $PM_{2.5}$, and the AQI.

The objective of this study is to introduce a novel framework named DAerosol-NTM (deep aerosol with NTM) that leverages long short-term memory (LSTM) and neural Turing machines (NTMs) to retain long-term data. DAerosol-NTM incorporates the aforementioned 26 independent variables and predicts four dependent variables in aerosol (PM_{10} , $PM_{2.5}$, and AQI quality control index): accuracy, precision, mean square error (RSME), and time interval before and after aerosol event (TIBAAE).

Ultimately, DAerosol-NTM aims to enhance our comprehension of aerosol pollution in relation to the air quality index (AQI) by maximizing the utilization of available meteorological data and employing modern framework approaches such as NTM to surpass previous baseline models. Furthermore, it contributes to the mitigation of other detrimental weather events that significantly impact the socio-economic and public health aspects of urban environments.

The remainder of this paper is organized as follows: Sect. 2 provides a comprehensive overview of the theoretical concepts, while Sect. 3 reviews relevant literature and prior works in the field. The framework developed in this study is presented in Sect. 4, followed by a detailed description of the experimental design in Sect. 5. Finally, Sect. 6 presents the conclusions drawn from this research and offers suggestions for future investigations.

143

2 Literature review

144 Figure 1 depicts the chronological literature categorized
 145 into sections A and B. The former section encompasses
 146 studies focusing on the neural Turing machine (NTM),
 147 while the latter encompasses research specifically related to
 148 the prediction of air quality parameters.

149

2.1 Section A: neural Turing machine (NTM)

150 Research on intelligent machines has a history spanning
 151 over 70 years. Concurrent with the development of the first
 152 electronic computers [10], the design of algorithms has
 153 consistently been a fundamental and long-term objective of
 154 artificial intelligence [11]. Alan Turing's invention of the
 155 Turing machine in 1936 marked one of the earliest
 156 endeavors to create an intelligent machine. Initially, the
 157 Turing machine significantly influenced the fields of psy-
 158 chology and cognitive philosophy, as it presented a func-
 159 tional model of the brain. However, neuroscience quickly
 160 dismissed this metaphor due to the dissimilar architecture
 161 between Turing machines and humans [12].

162 Artificial intelligence, often abbreviated as AI, refers to
 163 machine intelligence or the capacity of a machine to per-
 164 ceive and learn. In this context, the term "machine"
 165 encompasses any smart device equipped with a processor
 166 capable of receiving input data from the environment,

167 processing it, and potentially making decisions. The pri-
 168 mary and ultimate goal of artificial intelligence is to sim-
 169 ulate and understand human behavior, with the eventual
 170 development of robots or devices capable of replacing
 171 humans in tasks that are difficult or dangerous for humans
 172 to perform. Machine learning, which lies at the core of
 173 artificial intelligence, encompasses a range of methods and
 174 algorithms that operate in distinct ways.

175 The earliest instances of convolutional neural networks,
 176 for example, were presented in the 1990s, with the first
 177 successful example being designed in 1993. Subsequently,
 178 recurrent neural networks (RNNs) emerged, including the
 179 development of long short-term memory (LSTM), although
 180 they faced challenges such as limited input data, hardware
 181 requirements, and training issues.

182 The year 2006 was heralded as a turning point in
 183 overcoming these challenges, as a new method for training
 184 deep networks was introduced [13]. By 2014, numerous
 185 scientists were actively involved in this field, leading to a
 186 rapid growth in the number of articles and books. Deep
 187 learning projects saw a significant surge from 2012 to
 188 2020. Neural Turing machines (NTMs) were first intro-
 189 duced by Gravé et al. in 2014 [14]. NTMs enable neural
 190 networks to read from and write to an external memory
 191 matrix, akin to random access memory in a conventional
 192 computer. This feature empowers NTMs to manipulate and
 193 display complex data structures while simultaneously

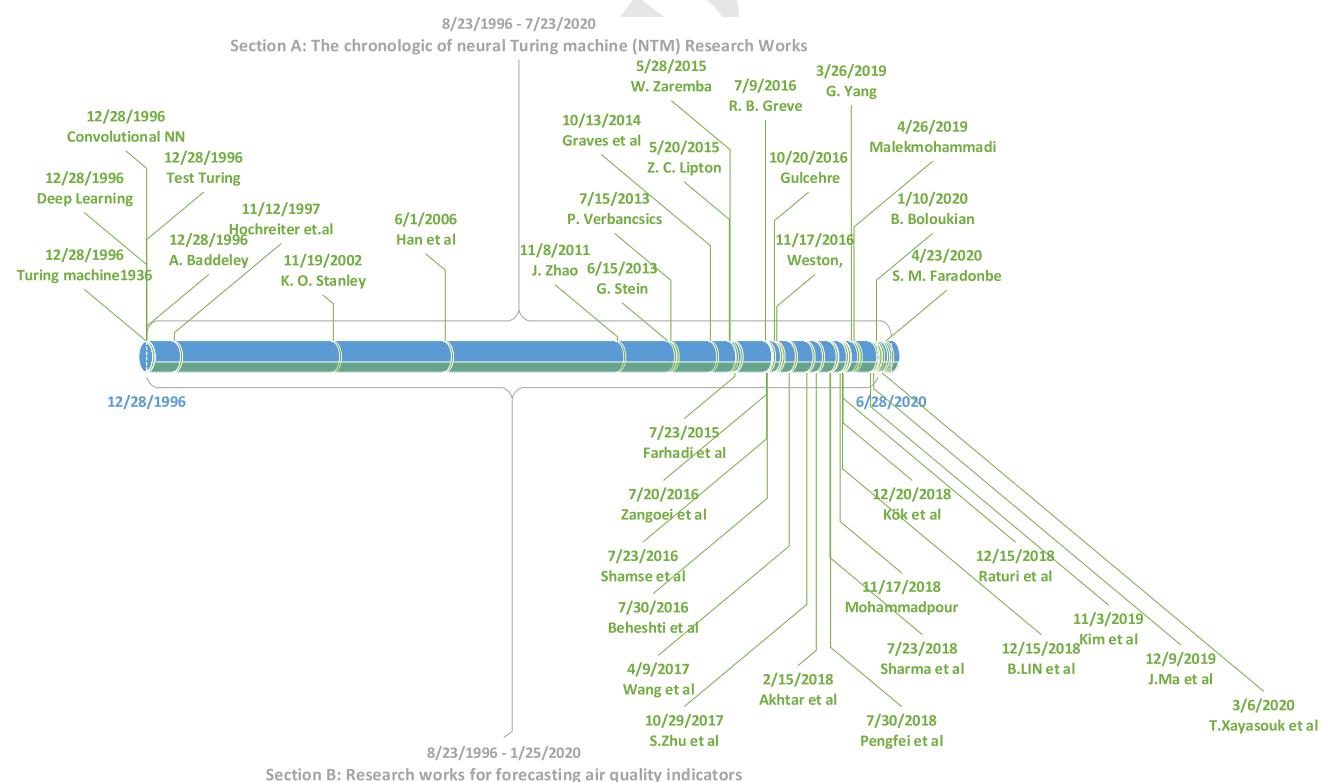


Fig. 1 Chronological of neural Turing machines and air quality prediction indicators

	Journal : Large 521	Dispatch : 13-9-2023	Pages : 37
	Article No. : 8868		
	MS Code : NCAA-D-22-05363R2	<input type="checkbox"/> LE <input checked="" type="checkbox"/> CP	<input type="checkbox"/> TYPESET <input checked="" type="checkbox"/> DISK

learning from the data. The subsequent section provides a comprehensive review of related works on neural Turing machines.

Numerous studies conducted in the past decade have utilized machine learning and deep learning methods [14, 15]. In a study by Baddeley [16], a Turing machine and a neural network were employed, combining recursive neural networks (RNNs) with external memory to perform algorithmic tasks. In comparison with the Turing machine, the neural Turing machine (NTM) utilizes a neural controller to direct the read and write heads to an external memory tape [17]. From the NTM, a machine learning model was derived, resulting in the differentiable neural computer (DNC), which modifies the memory architecture. The DNC employs the same architecture as the neural network controller, with read–write head access to the memory matrix, albeit with different access mechanisms and memory relationships. Information is retained based on a specified number of repetitions. Through supervised learning, the DNC has demonstrated the ability to address complex questions, simulating natural language problems and inference using the bAbI database. In a paper by Graves [18], the DNC showcased its capacity to perform tasks such as finding the shortest path between specified points (e.g., transportation networks) and deducing missing links (e.g., family genealogies) in randomly generated graphs. Various approaches have been proposed that modify the NTM method controller. These include the lie access NTM [19], RL NTM [20], and dynamic NTM, which are applied to tasks involving copying and pasting.

However, these methods have not demonstrated significant improvements in solving more complex tasks in neuroevolution and lack the necessary accuracy and speed. In a study by Greve et al. [21], it was shown that the neural Turing machine (NTM) can be enhanced through the evolutionary neural Turing machine (ENTM) topology. The NTM method was developed in conjunction with the neural evolution of augmenting topologies (NEAT¹) algorithm [22]. NEAT serves as a notable example of a genetic algorithm [23]. Unlike conventional approaches, NEAT does not require the design of artificial neural network (ANN) topologies; instead, it can create and transform ANN topologies and weights dynamically [24]. The NEAT method starts with a small population of simple neural networks and increases their complexity by adding new nodes and connections through mutations. In essence, NEAT is an evolutionary method that discovers the optimal network topology and weights to maximize task performance [25]. The ENTM method [21] has also demonstrated success in solving simple tasks such as copying. Notably, the trial version of the double T-Maze introduces a

reinforcement learning challenge. In the T-Maze learning task, the agent utilizes memory to adapt its behavior, often accomplished through a neural network. However, this method suffers from high computational requirements and memory limitations.

Hence, this study aims to address these challenges and provide a practical model for predicting air quality parameters. By doing so, it seeks to mitigate the issues of weight retention and disappearance in deep networks during updates, which can greatly impact their practicality and effectiveness.

2.2 Section B: forecast of air quality parameters

In recent years, extensive research has been conducted to investigate the impact of air pollution on human health and the environment. The primary objective of this study is to predict air quality parameters for the period spanning from 2015 to 2020. Accurate air quality information plays a crucial role in safeguarding human health and managing air pollution effectively [7]. The proliferation of vehicles and factories in recent years has contributed to the escalating levels of air pollution, and this trend is projected to persist in the near future. Furthermore, air pollution significantly affects not only human health but also the architectural landscape and the overall environment.

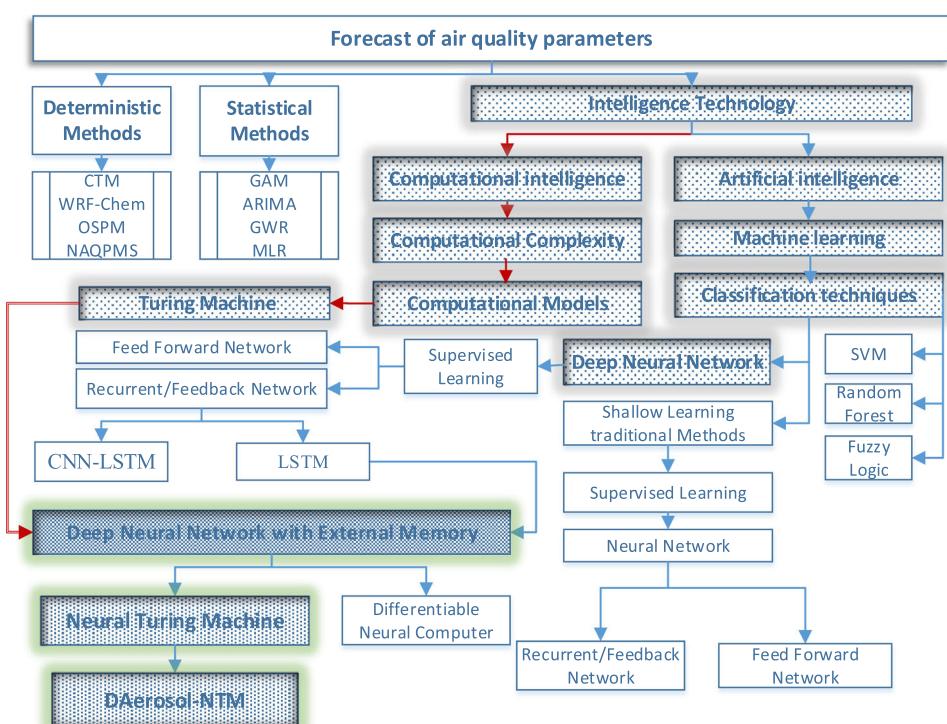
Furthermore, air pollution has detrimental effects on water bodies such as rivers and lakes, posing a threat to the well-being of animals and plants. The environmental challenges are further exacerbated by the rapid urbanization witnessed in cities due to overpopulation. Researchers hold divergent perspectives regarding the relationship between urbanization and the urban environment. Some argue that increasing urbanization exacerbates environmental pollution within cities. Urbanization has led to significant transformations in human lifestyles. Firstly, the utilization of natural resources for the production of goods has resulted in the generation of numerous pollutants. Secondly, urbanization has created a surge in the demand for transportation and infrastructure systems, leading to increased household energy consumption. This heightened energy consumption contributes to the emission of various pollutants, ultimately exacerbating the environmental conditions in urban areas [26].

Figure 2 depicts the mind map of this study, providing a concise overview of the methodologies employed thus far in predicting air quality parameters.

Figure 2 illustrates the categorization of research on predicting air quality parameters into three sections: deterministic, statistical, and machine learning. The highlighted areas represent the predominant focus of this research, which involves the utilization of external-memory-equipped neural networks for aerosol prediction. The

¹ Neuro evolution of augmenting topologies (NEAT).

Fig. 2 Mind-map diagram of this research and the position of this research among the primary methods



296 proposed method employs a deep learning approach that
297 combines the Turing machine and LSTM recursive neural
298 networks with addressable external memory to forecast air
299 quality parameters prior to the occurrence of aerosol
300 pollution.

301 The deterministic branch encompasses methods such as
302 CTMs,² WRF-Chem,³ OSPM,⁴ and NAQPMS⁵ which
303 construct models to simulate atmospheric chemistry dis-
304 perssion and transport. In statistical methods (e.g., GAM,⁶
305 ARIMA,⁷ GWR,⁸ and MLR⁹), the concentration of pol-
306 lutants is observed using a substantial amount of data. These
307 methods overcome the limitations of deterministic
308 approaches but are subject to statistical constraints.
309 Machine learning techniques, including SVM,¹⁰ ANNs,¹¹
310 FL,¹² and RF,¹³ have been employed to predict air quality
311 factors, among which neural networks have gained

312 significant recognition [8]. The subsequent paragraphs
313 provide an overview of previous research conducted on
314 predicting air quality parameters.

315 In the articles by Kim et al. [5] and Xayasouk et al. [6],
316 the objective was to predict the concentration of PM10 and
317 PM2.5 solid particles. Deep learning models, particularly
318 the DLSTM model, exhibited superior performance com-
319 pared to DAE and DNN. Sharma et al. [7] demonstrated the
320 accurate prediction of the air quality index (AQI) for a
321 specific location using deep learning models, taking into
322 consideration seasonal patterns and a substantial volume of
323 data. The suitability of the LSTM model for time-series
324 data was also highlighted in their study.

325 Another investigation by Pengfei et al. [9] focused on
326 forecasting the concentration of PM2.5 solid particles.
327 Bekkar et al. [27] proposed a hybrid model based on CNN
328 and LSTM, utilizing 24-h (daily) time intervals to predict
329 PM2.5 in Beijing's urban area by capturing spatiotemporal
330 characteristics of the data. This article presented a com-
331 parative analysis of various deep learning methods such as
332 LSTM, Bi-LSTM, GRU, Bi-GRU, CNN, and a hybrid
333 CNN-LSTM model. The convolutional neural network (CNN)
334 model was employed to capture spatial character-
335 istics, while the LSTM network extracted temporal fea-
336 tures. The deep learning model was rigorously evaluated,
337 and the results demonstrated the superiority of the DNN
338 model for prediction.

339 Akhtar et al. [28] predicted the PM10 value using the
340 multilayer perceptron algorithm and SVM. Raturi and

2FL01² Chemical transport models (CTMs).

3FL01³ Weather research and forecasting (WRF) model coupled with
3FL02 chemistry (Chem).

4FL01⁴ Operational street pollution models (OSPM).

5FL01⁵ Nested air quality prediction modelling system (NAQPMS).

6FL01⁶ Generalized additive models (GAMs),

7FL01⁷ Autoregressive integrated moving average (ARIMA).

8FL01⁸ Geographically weighted regression (GWR).

9FL01⁹ Multi-layer regression (MLR).

10FL01¹⁰ Support vector machine (SVM).

11FL01¹¹ Artificial neural networks (ANNs).

12FL01¹² Fuzzy logic (FL).

13FL01¹³ Random forest (RF).

341 Prasad [29] employed an artificial intelligence model to
 342 forecast the future air quality index (AQI). Wang et al. [30]
 343 developed a predictive system for timely AQI forecasting.
 344 Kök et al. [31] presented a comprehensive deep learning
 345 model for IoT events in smart cities, specifically designing
 346 and implementing an LSTM-based forecasting model to
 347 address future air quality challenges. Beheshti et al. [32]
 348 predicted the concentration of CO, NO₂, and SO₂ in the air
 349 of Tabriz metropolis using three multilayer feedforward
 350 neural networks and four error post-propagation training
 351 algorithms. Shams et al. [33] employed a multivariate
 352 regression model to determine the AQI air pollution index.

353 Furthermore, Zangouei et al. [34] conducted a study in
 354 which they predicted and analyzed the concentration of
 355 PM10 using artificial neural networks and Markov chain
 356 models. In a similar work by Farhadi et al. [35], the authors
 357 employed an artificial neural network model to model air
 358 pollution index pollutants in relation to climatic factors and
 359 stable and unstable indices, specifically focusing on PM10
 360 and AQI pollutants. Additionally, Table 1 provides a
 361 comparative analysis of the methods utilized in previous
 362 studies found in the literature and the approach employed
 363 in the present study.

3 Proposed method

364 Figure 3 illustrates the deep aerosol with neural Turing
 365 machines (DAerosol-NTM) framework, which incorporates
 366 the use of NTM with an LSTM-based controller
 367 specifically designed for handling time-series data. In this
 368 framework, each training step involves the utilization of
 369 external memory to store and retrieve network weights to
 370 and from the NTM external memory. More precisely, these
 371 weights are stored in the NTM memory with a specific
 372 coefficient assigned to each training step. During each
 373 update cycle, new weights are calculated based on the
 374 weights stored in memory. The DAerosol-NTM framework
 375 comprises four key functions, each serving a distinct purpose.
 376 These functions will be elaborated upon individually
 377 in the subsequent sections, while the dataflow section
 378 presents an overview of the interrelation between these
 379 functions.

380 Figure 4 depicts the pseudo-code implementation of
 381 DAerosol-NTM, which integrates an LSTM deep neural
 382 network with NTM external memory. The algorithm follows
 383 a series of steps to ensure the effective training and
 384 utilization of the network.

385 Initially, in Function 1, the input data undergoes pre-
 386 processing and modification to conform to the require-
 387 ments of the neural network. Subsequently, Function 2 sets
 388 the initial network parameters by randomly initializing the
 389 weights. The initial state of each cell is set to zero, and the

390 values of the gates and their activation functions are
 391 computed based on the formulas outlined in Appendix
 392 Section “Deep short-term neural network”.
 393

394 In Function 3, the algorithm reads the network’s weights
 395 from the external memory using the read head and adjusts
 396 the network’s weights accordingly. Prior to this step, it
 397 verifies whether weights exist in the memory. If weights
 398 are present, they are read from external memory and set
 399 within the network. It is important to note that this step
 400 does not involve the LSTM deep neural network. The
 401 network then commences training with new data to gen-
 402 erate output, and the network error is calculated for each
 403 input.

404 Next, the weights are stored in the external memory
 405 through the write head. Again, it is important to emphasize
 406 that this step does not involve the LSTM deep neural
 407 network without external memory. The algorithm then
 408 proceeds back to the previous step, reading the previously
 409 stored weights in the NTM memory, provided that the
 410 external memory is not empty (as specified in Eqs. 11 and
 411 12). The network output is computed for each test data
 412 based on the trained network.

413 Finally, Function 4 assesses the prediction accuracy,
 414 precision, and root mean squared error (RMSE) of the test
 415 data, while also measuring the best time before and after
 416 the occurrence of the aerosol event (referred to as
 417 TIBAAE).

418 Figure 5 illustrates the distinction between the deep
 419 LSTM network and the DAerosol-NTM network devel-
 420 oped in this study, which lies in the utilization of NTM to
 421 establish a semantic correlation across different training
 422 repetitions. Following the training of the network in X₀ and
 423 the generation of output Y₀, the weights of the DAerosol-
 424 NTM network are stored in the NTM memory using the
 425 write head. At time t₁, the network retrieves its weights
 426 from the memory using the read/write heads, and new
 427 weights are generated for further training. This iterative
 428 process persists until the network output approaches the
 429 target with high accuracy. The normalized weights
 430 obtained from previous training iterations influence the
 431 current training phase. Furthermore, it is assumed that a
 432 deep neural network comprises multiple hidden layers with
 433 long-term external memory. The controller effectively
 434 manages this long-term memory and employs two heads
 435 for reading and writing operations.

3.1 Function 1: dataset preprocessing

436 Figure 6 presents the dataset preprocessing procedures
 437 outlined in Function 1. Upon gaining a comprehensive
 438 understanding of the environment and the data, the data
 439 cleaning phase is initiated to replace missing information
 440 with accurate values. Additionally, noise data are smoothed
 441

Table 1 A comparison of the most related research works on pollution prediction

Research work	Data	Size of dataset	No. of independent variables	Method	Evaluated parameters	Accuracy	Precision	RMSE	The time interval before and after the aerosol event (TIBAAE)
Bekkar et al. [27]	Quality and Meteorological Data (Beijing)	March 1, 2013, to February 28, 2017	12	DLSTM, Bi-LSTM, GRU, Bi-GRU, CNN, CNN-LSTM, CNN-GRU,	PM _{2.5}	—	—	For DLSTM: — PM _{2.5} = 16%, For Bi-LSTM: PM _{2.5} = 16%, For GRU: PM _{2.5} = 17%, For Bi-GRU: PM _{2.5} = 17%, For CNN: PM _{2.5} = 17%, For CNN-LSTM: PM _{2.5} = 15%, For CNN-GRU: PM _{2.5} = 17%, For DLSTM: — PM _{2.5} = 11%, PM ₁₀ = 11% For DAE: PM _{2.5} = 84%, PM ₁₀ = 85% PM ₁₀ = 15% For DNN: — PM _{2.5} = 6%, For DLSTM: PM _{2.5} = 5%, PM ₁₀ = 5% For DNN: — PM _{2.5} = 9% For SVM: PM _{2.5} = 66% For BP: — PM _{2.5} = 67%	—
Xayasouk et al. [6]	Quality and Weather Data (Seoul)	January 1, 2015, to December 31, 2018 (25 quality control stations and one weather station)	12	DLSTM, DAE	PM _{2.5} , PM ₁₀	For DLSTM: PM _{2.5} = 88%, PM ₁₀ = 89%	For DAE: PM _{2.5} = 84%, PM ₁₀ = 85%	For DNN: — PM ₁₀ = 6%	For DLSTM: PM _{2.5} = 17%, For GRU: PM _{2.5} = 11%, PM ₁₀ = 11% For DAE: PM _{2.5} = 15%, PM ₁₀ = 15% For DNN: — PM _{2.5} = 6%, For DLSTM: PM _{2.5} = 5%, PM ₁₀ = 5% For DNN: — PM _{2.5} = 9% For SVM: PM _{2.5} = 66% For BP: — PM _{2.5} = 67%
International Conference [5]	Quality and Weather Data (Seoul)	One year (39 quality control stations and one weather station)	17	DNN, DLSTM	PM _{2.5} , PM ₁₀	—	—	For DNN: — PM _{2.5} = 90% For SVM: PM _{2.5} = 66% For BP: — PM _{2.5} = 67%	For DNN: — PM _{2.5} = 90% For SVM: PM _{2.5} = 66% For BP: — PM _{2.5} = 67%
Pengfei et al. [9]	Quality Data (Wuhan)	One quality control station	8	DNN, SVM, BP	PM _{2.5}	For DNN: — PM _{2.5} = 90% For SVM: PM _{2.5} = 66% For BP: — PM _{2.5} = 67%	For DNN: — PM _{2.5} = 90% For SVM: PM _{2.5} = 66% For BP: — PM _{2.5} = 67%		

Table 1 (continued)

Research work	Data	Size of dataset	No. of independent variables	Method	Evaluated parameters	Precision	RMSE	The time interval before and after the aerosol event (TIBAAE)
Sharma et al. [7]	Quality Data (RK Puram, New Delhi)	Two years (quality control station)	7	DLSTM	AQI	For DLSTM: AQI = 63%	—	For DLSTM: AQI = 37%
Akhtar et al. [28]	Quality Data (Center (CPCB))	Two years (quality control station)	8	MLP	PM ₁₀	For MLP: PM ₁₀ = 95%	—	—
Wang et al. [30]	Quality Data (China)	Two years	6	ANFIS, AHP	AQI	For ANFIS: AQI = 76%, For AHP: AQI = 68%	—	—
	http://113.108.142.147:20035/encipublish/							
This study (DAerosol-NTM) 2023	Quality and Weather Data (Tehran)	Twenty years of data with 58,922 samples (four meteorological stations and 15 air pollution sensors)	29	DAerosol-NTM	PM _{2.5} , PM ₁₀ , AQI, TIBAAE	PM _{2.5} = 95%, PM ₁₀ = 96%, AQI = 81% PM _{2.5} = 86%, PM ₁₀ = 87%, AQI = 72% PM _{2.5} = 5%, PM ₁₀ = 4% AQI = 19% PM ₁₀ = 96%, AQI = 96%	TIBAAE _{before} = 96(H), TIBAAE _{after} = 24(H), PM _{2.5} = 96%, AQI = 96%	TIBAAE _{before} = 96(H), TIBAAE _{after} = 24(H), PM _{2.5} = 96%, AQI = 96%

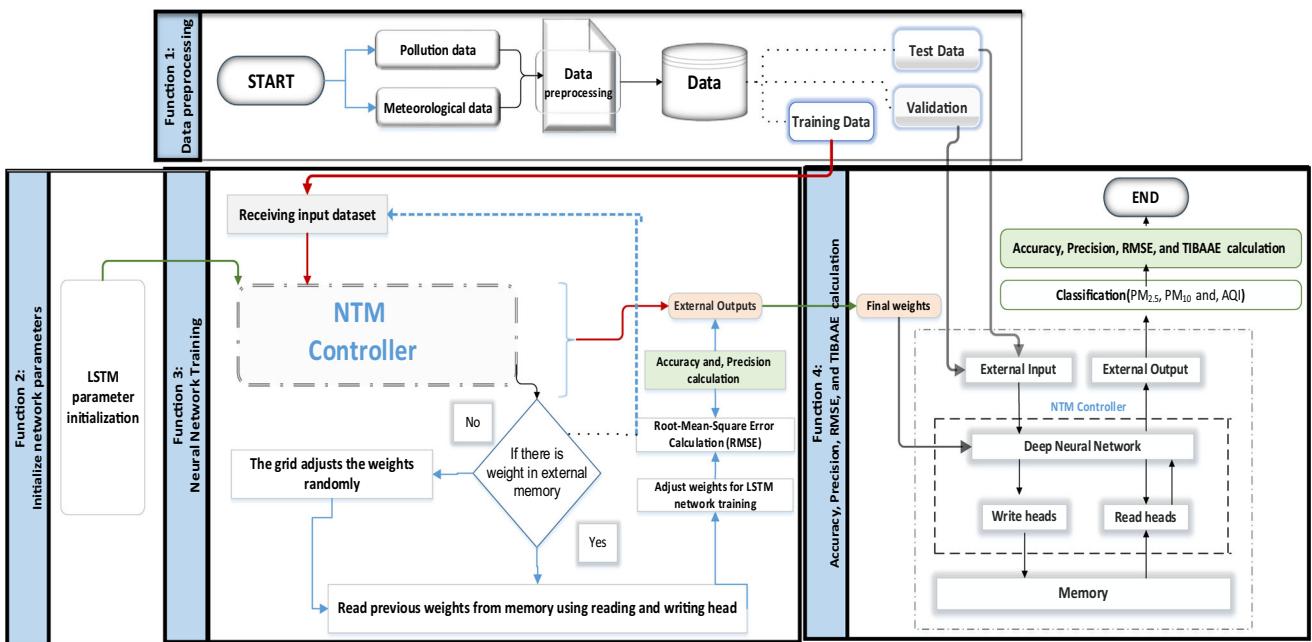


Fig. 3 DAerosol-NTM: applying deep learning and neural Turing machine in aerosol prediction

Total: DAerosol-NTM algorithm
Input: Dataset
Output: Accuracy percent, Precision percent, RMSE percent

```

1 Call Function 1(); // Data preprocessing;
2 Call Function 2(); // Initialize network parameters;
3 Call Function 3(); // Neural Network Training;
4 Call Function 4(); // Calculate accuracy, precision, RMSE, and TIBAAE
END

```

Fig. 4 Dataflow in DAerosol-NTM

by considering neighboring data points, and outlier data points are detected and subsequently eliminated. Subsequently, scattered data from various sources are consolidated and combined. In the data conversion step, normalization techniques are employed to transform the values into a specific range. This study further incorporates the expertise of technical professionals to guide the selection of high-priority features and assess their impact on meteorological parameters related to air pollution. Validation is performed through correlation and sensitivity analysis techniques to ensure data accuracy. Finally, to optimize data processing, the data volume is reduced by applying compression techniques such as data averaging. For a more detailed explanation of the data preprocessing procedures, please refer to Sect. 4.1.

3.2 Function 2: initializing LSTM network parameters

Figure 7 illustrates the initialization function, which is responsible for assigning initial values to the network. The weights of the neurons are randomly generated during this

process. The selection of the number of neurons and layers is determined through trial and error, ensuring an optimal configuration. Furthermore, the number of neurons in the input and output layers is determined in proportion to the length of the input data and the number of classes, respectively, that the network aims to classify the data into.

3.3 Function 3: neural network training for DLSTM and DAerosol-NTM methods

The purpose of this function is to facilitate network training, as depicted in Fig. 8. Firstly, the function receives the weights obtained from the previous step and initializes the network accordingly. It takes the input data in the form of a vector and generates the corresponding output. Subsequently, it compares the network's output with the desired result, allowing for the calculation of the network error using Eq. 1. Finally, the function updates the network weights based on the calculated error.

3.4 Function 4: calculating prediction accuracy, precision, RMSE, and TIBAAE

The forecasting process is conducted in three distinct sections: training, testing, and validation. These sections are assessed using various evaluation metrics such as root mean square error (RMSE), accuracy, precision, and a novel parameter known as TIBAAE.

RMSE, as described in previous studies [36] (Pengfei et al. [9]), is calculated by measuring the difference

462
463
464
465
466
467

468
469

470
471
472
473
474
475
476
477
478

481
482
483
484
485
486
487

Fig. 5 Training structure (DAerosol-NTM)

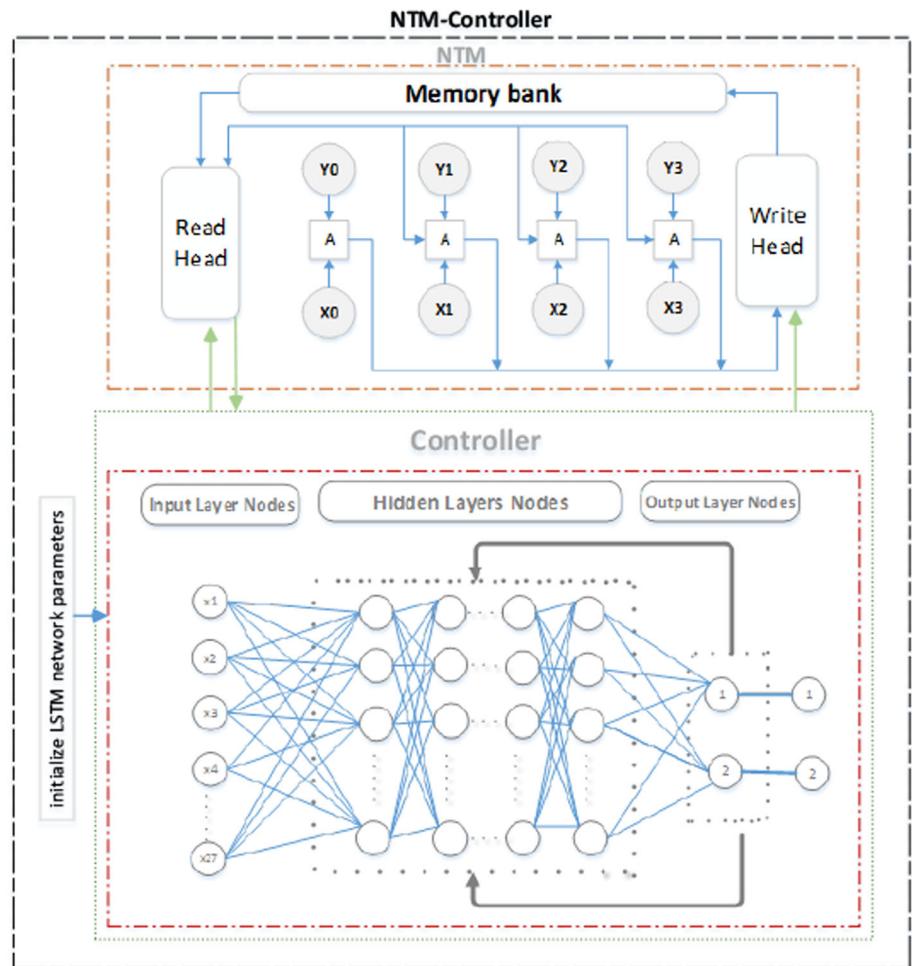


Fig. 6 Dataset preprocessing in DLSTM and DAerosol-NTM methods

Function 1: Dataset preprocessing
Input = Pollution data & Meteorological dataset
Output = Dataset
Description: The combination of two Pollution data && Meteorological data to achieve a single dataset

1. Environment understanding;
2. Data understanding;
3. Data cleaning for each one of the datasets (Pollution & Meteorological data)
4. Data preparation (Pollution & Meteorological data)
5. Data Modeling;

Fig. 7 The pseudo-code for initializing DLSTM and DAerosol-NTM methods

Function 2: Initialize LSTM network parameters
Input: number of layers, number of input nodes, number of hidden nodes, number of output nodes
Output: Adjusted parameters

- 1 Number of Layers = L
- 2 Number of input nodes = Length of sequence vector
- 3 Number of hidden nodes = H
- 4 Number of output nodes = Number of classes

End

488 between the known output values and the machine-generated
489 output values during training. It is expressed as a
490 percentage and can be computed using Eq. 1.

Accuracy is defined as the percentage of correct predictions made by the machine for the test data, as referenced in the work of Kellman and Hansen [37]. It can be easily calculated based on the known output values and the

491
492
493
494

Fig. 8 The pseudo-code of implementing NTM with the LSTM controller

```

Function 3: DAerosol-NTM algorithm for network training
Input: Dataset
Output: Best weights (Final weights)

1   for each training input, do:
2     decide for the forget gates applying Equation 7;
3     train the LSTM network by using NTM external memory
4   {
5     if (the NTM external memory is not empty)
6       use reader-head for reading previous weights from external Memory with (Equation 11, Equation 12);
7     else
8       set weights of the network randomly;
9     decide for the output gates applying Equation 8;
10    adjust the output of the neural network training function (value of weights, row of sequence vectors);
11    check the error with root-mean-square error (RMSE) with Equation 1;
12    use write-head for writing best weights to external memory with (Equation 13, Equation 14);
13  }
14 end for (Until met the criteria; Return the best output)
15 return (Train accuracy; Train precision) with (Equation 2, Equation 3);

END

```

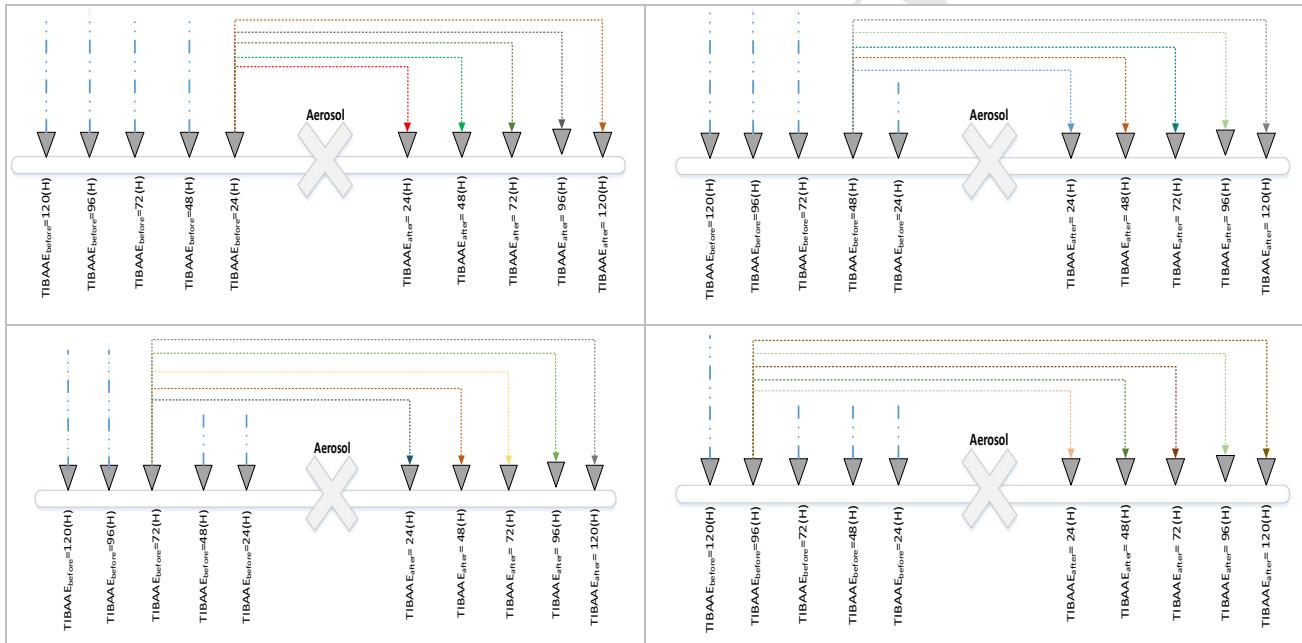


Fig. 9 TIBAAE by one /two/three/four/five days data before the aerosol event calculates the possibility of the aerosol event for one /two/three/four/five days later

Fig. 10 Accuracy, precision, RMSE, and TIBAAE calculation pseudo-code used in DLSTM and DAerosol-NTM methods

Function 4: Validation and Test Evaluation

Input = Validation and Test dataset

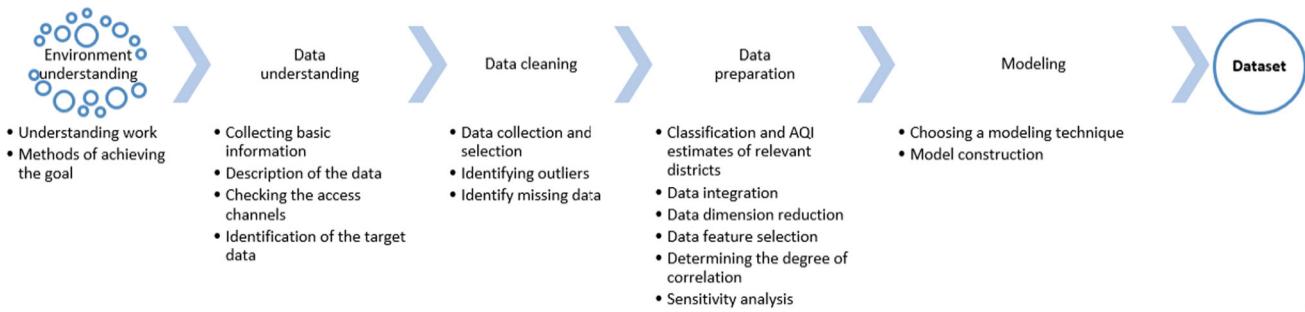
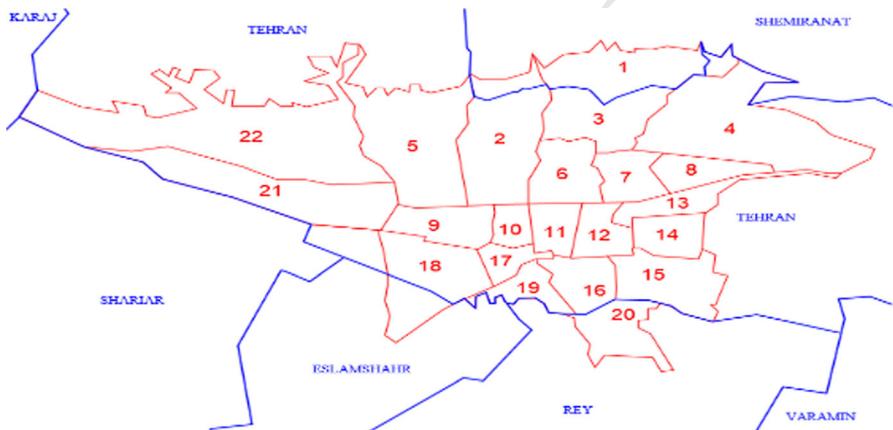
Output = Classification accuracy, precision, RMSE, TIBAAE;

```

1   for each testing or validation input, do:
2     set the testing or validation dataset as an input to the trained LSTM network;
3     get LSTM network output;
4   end for
5   calculate RMSE, Accuracy, Precision, and TIBAAE;
6   return (RMSE, Accuracy, Precision, TIBAAE);

END

```

**Fig. 11** Data preprocessing steps**Fig. 12** Urban areas of Tehran [40]

495 machine-generated output values obtained during training,
496 using Eq. 2.

497 Precision, on the other hand, represents the fraction of
498 relevant examples correctly identified by the machine
499 among all examples predicted to belong to a particular
500 class. It is evaluated by comparing the machine's output
501 with the known output values, and its assessment is carried
502 out according to Eq. 3.

$$\text{RMSE} = \frac{1}{n} \left(\sum_{i=1}^{\text{lengthofinput}} (\text{known.Output}[i] - \text{Machine.Output}[i])^2 \right)^{1/2} \quad (1)$$

$$504 \text{Accuracy} = 1 - \text{RMSE} \quad (2)$$

$$506 \text{Precision} = 1 - \left(\left(\frac{1}{n} \sum_{i=1}^{\text{lengthofinput}} \frac{1}{(\text{Known.Output}[i])} \times (\text{Known.Output}[i] - \text{Machine.Output}[i]) \right) \times 100 \right) \quad (3)$$

508 TIBAAE, which stands for time interval before and after
509 aerosol event, refers to the duration before and after the
510 prediction of an aerosol event that characterizes the event.
511 It aims to determine the predictability of subsequent days
512 based on the data available prior to the aerosol event.
513 Specifically, it assesses the accuracy of predicting the

514 occurrence of the aerosol event in the next one, two, three,
515 etc., days given the data from one day before the event.
516 Similarly, it evaluates the accuracy of predicting the event
517 in the following one, two, three, etc., days based on the
518 data from two days before the event. Figure 9 illustrates
519 this concept.

520 Figure 10 illustrates the methodology for computing the
521 RMSE, accuracy, precision, and TIBAAE metrics for
522 training, validation, and test predictions.

4 Experimental dataset, environment, and design

4.1 Data preprocessing

523 The cross-industry standard process for data mining
524 (CRISP-DM) data mining methodology [38] is employed
525 to carry out the data preprocessing steps. This methodology
526 provides a systematic approach for data processing and
527 divides the data preprocessing lifecycle into six distinct
528 phases, as depicted in Fig. 11.

Table 2 A small sample of datasets used in the experiments

	The lowest visible cloud	Horizontal view	Cloudiness	Wind direction	Wind speed	Temperature	Dew point	Sea level pressure	Station pressure	The amount of cloud is the lowest	Bottom Layla cloud	Medium Layla cloud	The amount of coverage of the first layer of the cloud	Type the first layer of the cloud
Date	H	vv	n	dd	ff	t	td	p	p0	nh	cl	cm	nl	tl
43,589.25	0.45	0.46	0.1	0.12	0.1	0.1	0.15	0.3	0.1	0.1	0.1	0.1	0.11	0.11
43,589.29	0.41	0.31	1	1	1	1	1	1	1	1	1	1	1	1
43,589.33	0.41	0.31	1	1	1	1	1	1	1	1	1	1	1	1
43,589.38	0.45	0.55	0.1	0.14	0.1	0.1	0.14	0.3	0.1	0.1	0.1	0.1	0.1	0.1
43,589.42	0.41	0.31	1	1	1	1	1	1	1	1	1	1	1	1
43,589.5	0.45	0.32	0.1	0.21	0.1	0.1	0.15	0.3	0.1	0.1	0.1	0.1	0.1	0.1
43,589.63	0.45	0.55	0.1	0.12	0.1	0.1	0.15	0.3	0.1	0.1	0.1	0.1	0.1	0.1
	The base height of the first cloud layer	The amount of coverage of the second cloud layer	The base height of the second type of cloud	Relative humidity	Wet temperature	Steam pressure	QNH pressure	Ozone pressure	Carbon monoxide	Nitrogen dioxide	Sulfur dioxide	Solid particles with a diameter less than or equal to 10 mm	Solid particles with a diameter less than or equal to 2.5 mm	Air quality index
Date	hl1	nl2	tl2	hl2	u	twet	ew	pqnh	o3	co	no2	pm 10	pm 2.5	AQI
43,589.25	0.44	0.1	0.1	0.38	0.12	0.12	0.13	0.94	1	1	1	0.54	0.71	0.49
43,589.29	0.33	1	1	0.16	1	1	1	0.1	0.1	0.18	0.13	0.22	0.15	0.22
43,589.33	0.33	1	1	0.16	1	1	1	0.11	0.1	0.18	0.1	0.19	0.14	0.19
43,589.38	0.4	0.1	0.1	0.19	0.13	0.12	0.13	0.93	0.13	0.1	0.16	0.1	0.17	0.14
43,589.42	0.33	1	1	0.16	1	1	1	0.1	0.15	0.1	0.14	0.1	0.17	0.13
43,589.5	0.4	0.1	0.1	0.19	0.17	0.12	0.13	0.95	1	1	1	0.54	0.71	0.49
43,589.63	0.4	0.1	0.1	0.19	0.15	0.12	0.13	0.94	1	1	1	0.54	0.71	0.49

Table 3 Range of air pollution variables

Parameter description	Scale	Parameters	Min	Max	Mean	Standard deviation
The lowest visible cloud	Meter	h	50	2500	1275	0.03998234875
Horizontal view	Degree	W	100	20,000	10,050	0.103439528
Cloudiness	Degree	n	2	3172	1587	0.409721572
Wind direction	Degree	dd	10	2428.631	1219.3155	0.378808213
Wind speed	Degree	ff	1	1943.608	972.304	0.409732491
Temperature	Degree	t	- 9.8	1625.968	808.083892	0.402200747
Dew point	Degree	td	- 17.7	1394.558	688.4290375	0.403780867
Sea level pressure	kg/cm ²	p	989.1679	1346.518	1167.843169	0.385220176
Station pressure	Degree	p0	761.6	1293.615	1027.607462	0.325946986
The amount of cloud is the lowest layer	Degree	nh	761.6	1164.305	962.9526135	0.409606825
Bottom layer cloud	Degree	cl	1	1058.625	529.8126735	0.409200627
Medium layer cloud	Degree	cm	1	970.6095	485.8047451	0.408983303
The amount of coverage of the first layer of the cloud	Degree	nl1	1	895.9941	448.4970519	0.409823982
Type the first layer of the cloud	Km	tl1	1	832.4384	416.7191842	0.407124661
The base height of the first cloud layer	Meter	hl1	90	3000	1545	0.038858835
The amount of coverage of the second layer of the cloud	millibar	nl2	1	794.466	397.7329872	0.408709075
The second type of cloud layer	Km	tl2	2	747.9518	374.9758996	0.408854975
The base height of the second cloud layer	Meter	hl2	240	9000	4620	0.090861262
Relative humidity	Degrees celsius	u	5	790.8312	397.9156215	0.388401697
Wet temperature	Degree	twet	- 10.2	751.9397	370.8698612	0.399058489
Steam pressure	Degree	ew	- 18.85572	716.4769	348.8105916	0.395297729
QNH pressure	Degree	pqnh	730.1052	1034.4	882.2525861	0.385501341
Ozone	ppb	O3	1	660.1358	330.5679177	0.400348919
Carbon monoxide	ppm	CO	0.1	601.2626	300.6813031	0.408890217
Nitrogen dioxide	ppb	NO2	1	555.443	278.2215109	0.380254769
Sulfur dioxide	ppb	SO2	1	513.9647	257.4823508	0.404108716
Air quality index	Good, average, unhealthy for sensitive groups, unhealthy, very unhealthy, hazardous	AQI	11	985	498	0.148421883
Solid particles with a diameter less than or equal to 10 mm	ug/m ³	PM ₁₀	1	985	493	0.170669391
Solid particles with a diameter less than or equal to 2.5 mm	ug/m ³	PM _{2.5}	1	675	338	0.257377418

532 4.1.1 Environment understanding

533 Figure 12 shows the description of the study area; due to
 534 the absence of standardized data on climate, pollution, and
 535 diverse climatic and geographical conditions, this study

536 exclusively focuses on the parameters derived from data
 537 collected in the City of Tehran, Iran. According to the 2018
 538 estimate provided by the United Nations, Tehran holds the
 539 38th position among the world's most populous cities and
 540 stands as the most populous city in West Asia, the second

Table 4 Name of selected statistical samples

Row	Areas subject to air pollution	Meteorological stations
1	District 1—Aqdasiyeh	District 1—Shemiranat
2	District 2—Sharif	District 6—Geophysics
3	District 2—Shahrdari	District 9—Mehrabad
4	District 3—Doroos	District 22—Chitgar
5	District 4—Shahrdari	
6	District 6—Tarbiat Modares	
7	District 7—Setade Bohran	
8	District 9—Shahrdari	
9	District 10—Shahrdari	
10	District 11—Shahrdari	
11	District 16—Shahrdari	
12	District 18—Shahrdari	
13	District 19—Shahrdari	
14	District 21—Shahrdari	
15	District 22—Shahrdari	

most populated metropolis in the Middle East, and the capital of Iran [39, 40]. Notably, Tehran consistently ranks among the ten most polluted cities globally based on the global air quality index [41].

The capital city of Tehran is situated on the southern slopes of the Alborz Mountain, encompassing an area of approximately 730 square kilometers and nestled between two mountainous and desert valleys. To facilitate effective city management, Tehran Municipality has divided the city into 22 municipal districts and 122 urban districts. Within the administrative framework of the Ministry of Interior, Tehran is centrally located in Tehran province. Its neighboring cities include Karaj and Shemiranat to the north, Damavand to the east, and Varamin, Rey, and Islamshahr to the south, while Shahryar and Karaj lie to the west [42].

4.1.2 Data understanding: a partial view of the dataset

According to experts in the field of meteorology and air quality control, various factors play a significant role in predicting air pollution. These factors include sky conditions such as cloud cover, cloud type, cloud layer, cloud height, horizontal visibility, atmospheric pressure, wind direction and speed, relative humidity, temperature, and concentrations of carbon monoxide, ozone, nitrogen dioxide, and sulfur dioxide [43].

The analysis of the data revealed that among the pollutants, PM_{2.5} and PM₁₀ exhibit the highest air quality index (AQI) values [4]. A subset of the dataset is presented in Table 2, while Table 3 illustrates the range of the

collected parameters [44]. Furthermore, the data have been normalized to a range of 0–1 [29].

4.1.3 Data cleaning

The dataset used in this study was collected from a total of 25 air pollution sensors and four meteorological stations located in various areas of Tehran. From this dataset, a suitable subset was chosen, consisting of data obtained from 15 active sensors, spanning the period from December 30, 1999, to May 4, 2019. This subset comprises 882,735 air pollution records and 226,330 meteorological records, resulting in a total of 1,109,065 records.

Table 4 provides details about the four selected meteorological stations and the 15 active sensors situated in areas prone to air pollution.

4.1.4 Data preparation

4.1.4.1 Classification and AQI estimates of relevant districts This section presents an analysis of the average air quality index (AQI) for all categories across the entire city of Tehran. The preliminary analysis of the data and the correlations between the AQI and meteorological indicators for different time periods are illustrated in Fig. 13. Subsequently, a deep learning model was employed using hourly data to forecast air pollution levels in the entire city of Tehran. To facilitate processing in MATLAB software, the data for each day were encoded, as demonstrated in Table 5.

Figure 13A depicts the average air quality index (AQI) for 24-h days over a span of 20 consecutive years. A total of 20 graphs were plotted for each day, resulting in a total of 140 charts. The x-axis represents the hours of the day (0, 3, 6, 9, 12, 15, 18, 21, and 24), while the y-axis denotes the average AQI. Notably, the most polluted time of day is observed at 21:00, whereas the minimum pollution occurs around 2:30 a.m. Pollution levels increase from noon onwards and reach their peak at 9 p.m.

On the other hand, Fig. 13B shows the official holidays and their corresponding average daily pollution levels up to one week before each holiday. With an assumption of approximately 27 official holidays per year, a total of 540 diagrams were generated over the course of 20 consecutive years ($20 \times 27 = 540$). Figure 13B provides a summary of these diagrams. It can be observed that pollution levels start declining six days before an official holiday and reach their lowest point just before the holiday. However, pollution tends to rise again in the final hours of the holidays. In other words, air pollution decreases prior to official holidays but increases toward the end of the holiday period.

Figure 13C illustrates the average levels of pollutants on weekends throughout the year, up to one week before each

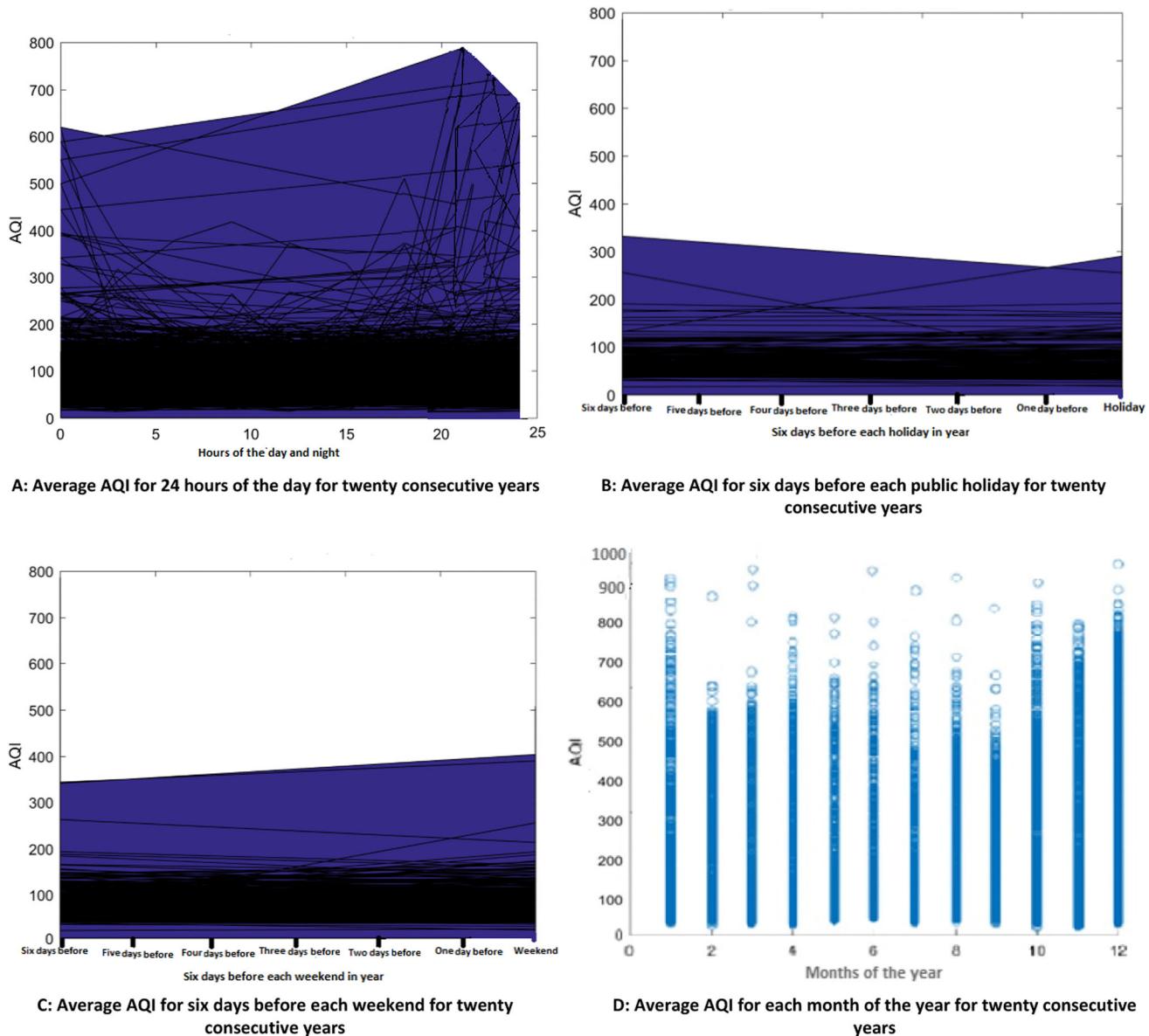


Fig. 13 AQI average air quality index of all studied categories in Tehran

Table 5 Numbers considered for the days of the week in the data table

Days of the week	Number
Monday	1
Tuesday	2
Wednesday	3
Thursday	4
Friday	5
Saturday	6
Sunday	7

618 holiday. Assuming 52 weekends per year, a total of 1040
619 diagrams were generated over 20 consecutive years. These
620 diagrams are summarized in Fig. 13C. It can be observed

that pollution levels do not decrease on weekends; instead, they exhibit a relatively small increase.

Moving on to Fig. 13D, it demonstrates the average monthly pollution levels for the entire year. With 12 months in a year, 20 data points per month were recorded over 20 consecutive years. The months of January, October, and December stand out as the most polluted, with air pollution increasing primarily from early autumn to mid-winter. The changes in air pollution over the years remain relatively consistent unless there are extraordinary events.

4.1.4.2 Data integration The Tehran Meteorological Organization has established four stations that are equipped with quality control sensors. These stations are categorized

Table 6 Category of areas in terms of meteorological stations

	Name of stations			
	Shemiranat station Category 1	Geophysics station Category 2	Mehrabad station Category 3	Chitgar station Category 4
1	District 1	District 2	District 9	District 21
2	District 3	District 3	District 10	District 22
3	District 4	District 6	District 16	–
4	–	District 7	District 18	–
5	–	District 8	District 19	–
6	–	District 10	District 21	–
7	–	District 11	District 2	–

Table 7 The degree of correlation between independent variables and dependent variables

Parameter description	Parameter	Correlation PM ₁₀	Correlation PM _{2.5}	Correlation AQI
The lowest visible cloud	h	– 0.626856278	– 0.59053065178	– 0.6697409903
Horizontal view	w	– 0.607703276	– 0.57248446361	– 0.6492776859
Cloudiness	n	– 0.623519478	– 0.58738665652	– 0.6661759105
Wind direction	dd	– 0.558102351	– 0.52575119044	– 0.5962834462
Wind speed	ff	– 0.537909738	– 0.50672655177	– 0.5747094481
Temperature	t	– 0.527803925	– 0.49720533264	– 0.5639122357
Dew point	td	– 0.253686353	– 0.23896763935	– 0.2710416335
Sea level pressure	p	– 0.648549046	– 0.61097019671	– 0.6929178131
Station pressure	p0	– 0.648429866	– 0.61085789546	– 0.6927904738
The amount of cloud is the lowest layer	nh	– 0.564882591	– 0.53213933215	– 0.6035275445
Bottom layer cloud	cl	– 0.580665947	– 0.54701005421	– 0.6203906691
Medium layer cloud	cm	– 0.552014289	– 0.5200152228	– 0.5897788765
The amount of coverage of the first layer of the cloud	nl1	– 0.565542989	– 0.5327614680	– 0.6042330393
Type the first layer of the cloud	tl1	– 0.632773011	– 0.5961056055	– 0.6760625894
The base height of the first cloud layer	hl1	– 0.639093125	– 0.6020605133	– 0.6828149926
The amount of coverage of the second layer of the cloud	nl2	– 0.565368337	– 0.5325969809	– 0.6040465128
The second type of cloud layer	tl2	– 0.590450388	– 0.5562288318	– 0.6308444865
The base height of the second cloud layer	hl2	– 0.561194828	– 0.5286648232	– 0.5995874856
Relative humidity	u	– 0.562479899	– 0.5298755763	– 0.6009604690
Wet temperature	twet	– 0.530023709	– 0.4992966861	– 0.5662838703
Steam pressure	ew	– 0.588278379	– 0.5541823374	– 0.6285238308
QNH pressure	pqnh	– 0.648566165	– 0.6109863247	– 0.6929361010
Ozone	O3	0.212247926	0.0937665708	0.2439902903
Carbon monoxide	CO	0.508057475	0.5595096392	0.5435975215
Nitrogen dioxide	NO2	0.535740491	0.6306802232	0.6179278878
Sulfur dioxide	SO2	0.465257284	0.5112756430	0.5090234275
Solid particles with a diameter less than or equal to 10 mm	PM₁₀	1	0.799675092	0.9799183616
Solid particles with a diameter less than or equal to 2.5 mm	PM_{2.5}		1	0.797171295
Air quality index	AQI			1

634 based on the areas they cover, as depicted in Fig. 12,
 635 According to the guidelines set by the World Meteorological Organization (WMO-1160), each meteorological
 636

station on a flat surface typically covers an area of approximately 150 km. Therefore, the selection of nearest neighbors is determined using geographical information

637
 638
 639

Table 8 Percentage of the effect of input variables on output

Input variable Solid particles with a diameter less than or equal to 10 mm	Output PM₁₀	Impact factor	Input variable Solid particles with a diameter less than or equal to 2.5 mm	Output PM_{2.5}	Impact factor
QNH pressure	pqnh	0.648566165	Solid particles with a diameter less than or equal to 10 mm	PM₁₀	0.799671345
Sea level pressure	p	0.648549046	Nitrogen dioxide	NO₂	0.630680223
Station pressure	p0	0.648429866	QNH pressure	pqnh	0.610986325
The base height of the first cloud layer	hl1	0.639093125	Sea level pressure	p	0.610970197
Type the first layer of the cloud	tl1	0.632773011	Station pressure	p0	0.610857895
The lowest visible cloud	h	0.626856278	The base height of the first cloud layer	hl1	0.602060513
Cloudiness	n	0.623519478	Type the first layer of the cloud	tl1	0.596105606
Horizontal view	w	0.607032760	The lowest visible cloud	h	0.590530652
The second type of cloud layer	tl2	0.590450388	Cloudiness	n	0.587386657
Steam pressure	ew	0.588278379	Horizontal view	w	0.572484464
Bottom layer cloud	cl	0.580665947	Carbon monoxide	CO	0.559509639
The amount of coverage of the first layer of the cloud	nl1	0.565542989	The second type of cloud layer	tl2	0.556228832
The amount of coverage of the second layer of the cloud	nl2	0.565368337	Steam pressure	ew	0.554182337
The amount of cloud is the lowest layer	nh	0.564882591	Bottom layer cloud	cl	0.547010054
Relative humidity	u	0.562479899	The amount of coverage of the first layer of the cloud	nl1	0.532761468
The base height of the second cloud layer	hl2	0.561194828	The amount of coverage of the second layer of the cloud	nl2	0.532596981
Wind direction	dd	0.558102351	The amount of cloud is the lowest layer	nh	0.532139332
Medium layer cloud	cm	0.552014289	Relative humidity	u	0.529875576
Wind speed	ff	0.537909738	The base height of the second cloud layer	hl2	0.528664823
Nitrogen dioxide	NO₂	0.535740491	Wind direction	dd	0.52575119
Wet temperature	twet	0.530023709	Medium layer cloud	cm	0.520015223
Temperature	t	0.527803925	Sulfur dioxide	SO₂	0.511275643
Carbon monoxide	CO	0.508057475	Wind speed	ff	0.506726552
Sulfur dioxide	SO₂	0.465257284	Wet temperature	twet	0.499296686
Dew point	td	0.253686353	Temperature	t	0.497205333
Ozone	O₃	0.212247926	Dew point	td	0.238967639
Solid particles with a diameter less than or equal to 2.5 mm	PM_{2.5}	0	Ozone	O₃	0.093766571
Air quality index	AQI	0	Air quality index	AQI	0

640 system (GIS) data, including latitude, longitude, and
 641 altitude.

642 Table 6 provides information on the categorization of
 643 areas in terms of air pollution at the respective meteorological
 644 stations: Shemiranat station in district 1 (category
 645 1), geophysics station in district 6 (category 2), Mehrabad
 646 station in district 9 (category 3), and Chitgar station in
 647 district 22 (category 4). To obtain unit data, the air pollution
 648 data from these areas were averaged. It should be
 649 noted that certain areas may fall under multiple categories
 650 simultaneously due to their proximity to two or three

651 meteorological stations. The following table lists each
 652 category along with its associated districts.

653 **4.1.4.3 Data dimension reduction** As is customary, the
 654 air pollution data are synchronized with the meteorological
 655 data on an hourly basis. However, it is necessary to convert
 656 these data from an hourly resolution to an average every
 657 three hours. After integrating the meteorological data with
 658 the corresponding quality control data, the total dataset
 659 consists of 58,921 samples with 82 attributes.

Table 9 Sensitivity analysis for PM_{2.5}

4.1.4.4 Data feature selection and determining the degree of correlation In collaboration with technical experts, a subset of 29 features was selected from the initial pool of 82 meteorological parameters known to impact air pollution. The correlation coefficient, a statistical measure, is employed to determine the type and strength of the relationship between two quantitative variables. This coefficient serves as a criterion for assessing the correlation between the variables, ranging from -1 to 1 . A value of zero indicates no relationship between the variables. Table 7 displays the correlation coefficients for the independent and dependent variables. The closer the coefficient is to 1 , the stronger the relationship, while values closer to zero suggest a weaker relationship. A positive coefficient signifies a direct relationship, whereas a negative coefficient implies an inverse relationship between the variables. These findings reveal a direct correlation between the quality control data and an indirect correlation between the meteorological data.

4.1.4.5 Sensitivity analysis Sensitivity analysis involves assessing and estimating the behavior of a system, specifically how the system's output is influenced by the values of its independent variables, which serve as inputs, Tables 7 and 8 hold significant importance within the article. Table 7 serves the purpose of identifying, examining, and investigating the correlations of features used in data analysis. It lists various features applied to the data separately. Subsequently, the degree of correlation between these features is determined in the adjacent column. This correlation may be positive or negative, indicating the degree of interrelation and predictiveness between the paired features. In this study, the impact percentage, which represents the ratio of input-to-output variables, is used to prioritize the factors that affect the size of PM2.5 and PM10 solid particles. The results of this analysis are presented in Table 8. Table 8 is designated for sensitivity analysis. Its purpose in the article is to assess the influence of parameters or distinct features on the outcomes under examination. Each column in this table represents a

Table 10 Sensitivity Analysis for PM₁₀

Scenarios	Ozone	Dew point	Sulfur dioxide	Carbon monoxide	Temperature	Wet temperature	Nitrogen dioxide	Wind speed	Medium layer cloud	Wind direction	The base height of the second cloud layer	Relative humidity	The amount of cloud is The lowest layer	The amount of coverage of the second layer of the cloud	Bottom layer cloud	The amount of coverage of the first layer of the cloud	cl	ew	Steam pressure	The second type of cloud layer	Horizontal view	Cloudiness	Type the first layer of the cloud	The base height of the first cloud layer	Station pressure	Steam pressure	QNH pressure	Results (PM10)
	O3	td	SO2	CO	t	twet	NO2	ff	cm	dd	hl2	u	nh	nl2	nl1	v	v	v	v	v	v	v	h	tl1	h1	p0	p	pqnh
1	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.7648	
2	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.7593		
3	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.9232		
4	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.2580		
5	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.9448		
6	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.6445		
7	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.5125		
8	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.3420		
9	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.0001		
10	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.9656		
11	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.1302		
12	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.3664		
13	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	v	v	0.9469		
14	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	v	0.7841	
15	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	v	0.3938	
16	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	v	0.7538	
17	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	0.2453	
18	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	v	0.9437	
19	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	0.3188	
20	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	v	0.7841	
21	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	0.3008	
22	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	v	v	v	v	0.5483	
23	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	0.2618	
24	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	0.2411	
25	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	v	v	0.0026	
26	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	v	0.2936	

parameter or feature, showcasing how result variations correspond to changes in these attributes. This table enhances comprehension of feature impact and their significance in the study, facilitating direct comparisons between them. Additionally, Tables 9 and 10 provide the sensitivity analysis for PM10 and PM2.5 outputs, respectively. The presence of gray shading in Tables 9 and 10 serves to denote the features exerting the most significant influence on the model. In this sensitivity analysis, the shallow LSTM model incorporates all independent variables simultaneously. Subsequently, variables with lesser impact are progressively eliminated until they attain the maximum threshold. Consequently, the features highlighted in gray demonstrate the foremost sway on the model and its prognostication of PM2.5 and PM10 concentrations. The underlying purpose of this procedure is to discern the variables bearing the greatest impact on prediction. Furthermore, as the ultimate outcome has been subjected to editorial refinement and distinction, it is now denoted by

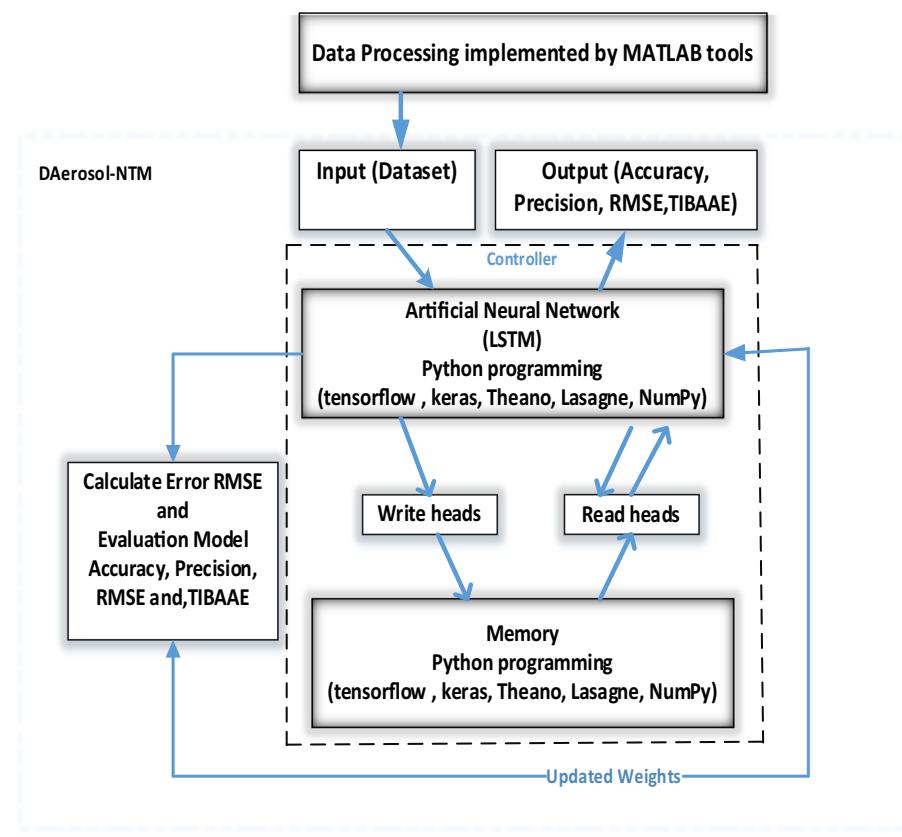
being both bold and italicized, superseding the previous gray color representation.

The first step of the analysis involved sorting the features in a descending order based on their impact percentages, as shown in Table 8. Subsequently, a sensitivity analysis was conducted to assess the model's accuracy in terms of consistency. Initially, all independent variables were included in the shallow LSTM model simultaneously. Then, variables with relatively minor effects were gradually removed one by one until the maximum value was achieved. The features of the gray color represent the ones that had the greatest impact on the model. This process aimed to identify the most influential variables in the prediction of PM2.5 and PM10 concentrations.

4.2 Experimental environment

Figure 14 shows the data preprocessing phase utilizing MATLAB R2018a software [46] and the experimental environment created in the Anaconda environment using

Fig. 14 Experimental environment



the Python programming language version 3.6 [46]. The implementation was performed on a desktop computer equipped with 8GB of memory, a Core7 CPU, and GeForce435m graphics. For the development of deep neural networks, the Keras library [46] was utilized, which relies on the GPU-based TensorFlow framework [47] and [45]. The coding platform for this implementation involved Google Colab and Spyder [48]. To facilitate the implementation of the neural Turing machine (NTM), several libraries were required, including:

- (1) Theano [47] (Jason [49]): This Python library enables the evaluation of mathematical operations, including multi-dimensional arrays.
- (2) Lasagne (Jason [49]): A lightweight library built on top of Theano that facilitates the construction and training of neural networks.
- (3) NumPy: An open-source Python library widely used in various scientific and engineering fields for numerical computations.

It is important to note that the distinguishing aspect of this experimental environment is the utilization and implementation of the neural Turing machine (NTM), setting it apart from conventional deep neural networks.

4.3 Experimental design and setup

This study employs the cross-validation method [44] to assess the model's effectiveness. The training dataset is randomly partitioned into k folds of equal size, where $K - 1$ folds are utilized for training and one fold serves as the validation set. Generally, the dataset is divided as follows: 60% for training, 20% for cross-validation to prevent overfitting, and 20% for testing purposes. The experimental design and setup are presented in Tables 11 and 12, respectively.

The experiments are described as follows:

- Experiment-1 aims to evaluate the performance of DAerosol-NTM in comparison with the baseline articles, focusing on the prediction accuracy, precision, and root mean square error (RMSE) of the PM2.5, PM10, and AQI parameters.
- Experiment-2 involves assessing DAerosol-NTM against DLSTM specifically regarding the time interval before and after the occurrence of an aerosol event (TIBAAE).

Table 11 Experimental design

Experiment no.	Experiment description	Dataset	Evaluated parameters	Parameter initialization	Hypothesis Support
Experiment-1	Comparing DAerosol-NTM with baseline research articles [5–7, 9, 27]	The provided dataset (training, test, and validation)	Evaluating prediction accuracy, precision, and RMSE for PM2.5, PM10, and AQI	Initial network values	Hypothesis 1
Experiment-2	Comparing the time interval before and after the prediction of the aerosol (TIBAAE) event for DAerosol-NTM and DLSTM [5–7, 9]	The provided dataset (training, test, and validation)	Evaluating TIBAAE for PM2.5, PM10, and AQI	Using the parameters set in the previous experiments	Hypothesis 2

Table 12 Experimental setup

Method	Adjustment parameters
Neural Turing machine (NTM)	Controller = feedforward, memory size = 128, learning rate = 10^{-4} , iterations = 4000, error = 0
Long short-term memory (LSTM) neural network	Four-layer, activation function = tanh, [20, 15, 9, 6] hidden neurons
MLP	Feedforward net, activation function = ReLU, topology = nine-layer [45, 45, 45 ... 45]

778 4.4 Experiments

779 4.4.1 Experiment-1: evaluating the proposed DAerosol- 780 NTM compared to the baseline articles in terms 781 of PM2.5, PM10, and AQI

782 Due to the unavailability of the data used in the baseline
783 articles [5–7, 9, 27], this study applies the baseline methods
784 to the provided dataset for the evaluation of PM2.5, PM10,
785 and AQI parameters, as described in Tables 13, 14, and 15.
786 These tables demonstrate how the proposed model effec-
787 tively predicts PM10, PM2.5, and AQI in the four target
788 areas in Tehran, in comparison with the baseline models.

789 4.4.2 Experiment-2: comparing the time interval 790 before and after predicting the aerosol event 791 (TIBAAE) of DAerosol-NTM with DLSTM

792 Aerosol forecasting can be conducted on a daily (24-h) or
793 hourly basis, although hourly forecasts are typically less
794 common [50–52]. Similarly, the General Directorate of
795 meteorology [53] primarily focuses on daily forecasts
796 rather than hourly ones. In alignment with these practices,
797 this study also employs daily forecasting to present the

results displayed in Tables 16, 17, and 18. For the prediction of aerosol events in the next 24 h, the time interval considered encompasses data from the preceding five days up to one day prior to the event. The bar chart depicted in Fig. 15 illustrates the accurate prediction of aerosol concentration and future air quality control index ($TIBAAE_{after} = 24(H)$) for four days leading up to the occurrence of the aerosol event ($TIBAAE_{before} = 96(H)$). Both the DAerosol-NTM and DLSTM models exhibit commendable accuracy in these predictions.

Given that the AQI serves as the primary focus of this research, the subsequent experiments will solely concentrate on AQI. Tables 19, 20, 21, and 22 present the time intervals preceding the prediction of the future quality control index (AQI) for the subsequent 48–120 h, considering data from one to five days before the occurrence of the aerosol event. Figure 16 demonstrates that as the time frame progresses from the upcoming 24–120 h, there is a decline in both accuracy and precision, accompanied by an increase in forecast error.

Table 13 Comparison of DAerosol-NTM and baseline articles in terms of PM_{2.5}

Obtained results	Method	Train / Accuracy	Train / Precision	Train / RMSE	Validation / Accuracy	Validation / Precision	Validation / RMSE	Test / Accuracy	Test / Precision	Test / RMSE
PM _{2.5} [9]	MLP	0.80	0.59	0.20	0.79	0.48	0.21	0.72	0.45	0.28
PM _{2.5} [9]	DNN (MLP)	0.87	0.85	0.13	0.82	0.50	0.18	0.74	0.49	0.26
PM _{2.5} [6]	LSTM	0.85	0.69	0.15	0.84	0.50	0.16	0.73	0.48	0.27
PM _{2.5} [6]	DLSTM	0.87	0.71	0.13	0.86	0.56	0.14	0.80	0.55	0.20
PM _{2.5} [27]	CNN-LSTM	0.88	0.73	0.12	0.86	0.57	0.14	0.80	0.56	0.20
PM _{2.5} (this research)	DAerosol-NTM	0.97	0.90	0.03	0.96	0.88	0.03	0.95	0.86	0.05

Table 14 Comparison of DAerosol-NTM and baseline articles in terms of PM₁₀

Obtained results	Method	Train / Accuracy	Train / Precision	Train / RMSE	Validation / Accuracy	Validation / Precision	Validation / RMSE	Test / Accuracy	Test / Precision	Test / RMSE
PM ₁₀ [6]	DLSTM	0.88	0.72	0.13	0.86	0.57	0.14	0.81	0.55	0.20
PM ₁₀ [27]	CNN-LSTM	0.89	0.73	0.12	0.86	0.57	0.14	0.81	0.56	0.19
PM ₁₀ (This research)	DAerosol-NTM	0.98	0.91	0.03	0.97	0.89	0.03	0.96	0.87	0.04

Table 15 Comparison of DAerosol-NTM and baseline articles in terms of AQI

Obtained results	Method	Train / Accuracy	Train / Precision	Train / RMSE	Validation / Accuracy	Validation / Precision	Validation / RMSE	Test / Accuracy	Test / Precision	Test / RMSE
AQI [7]	DLSTM	0.85	0.50	0.15	0.83	0.49	0.17	0.75	0.48	0.25
AQI [27]	CNN-LSTM	0.86	0.60	0.14	0.83	0.51	0.17	0.76	0.49	0.25
AQI (this research)	DAerosol-NTM	0.98	0.94	0.02	0.88	0.81	0.12	0.81	0.72	0.19

818 4.5 Experimental discussion

819 The present study utilized climate and pollution data from
820 Tehran, the capital city of Iran. The proposed DAerosol-
821 NTM framework focused on investigating parameters such
822 as PM10, PM2.5, AQI, and TIBAAE. In Experiment-1, the
823 first hypothesis presented in Table 23 was supported, as the
824 framework improved the prediction accuracy, precision,
825 and RMSE of PM10, PM2.5, and the AQI quality control

index. This improvement was attributed to the incorporation of extensive historical meteorological information into the external memory of the deep learning neural network.

In comparison with the baseline articles, the introduced DAerosol-NTM framework demonstrated a significant enhancement in accuracy ranging from 8 to 31%, precision ranging from 46 to 91%, and root mean square error (RMSE) ranging from 24 to 85%.

826
827
828
829
830
831
832
833

Table 16 The next following one-day forecast of future PM2.5 based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /Accuracy	Train /Precision	Train /RMSE	Validation /Accuracy	Validation /Precision	Validation /RMSE	Test /Accuracy	Test /Precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 24(H))
CNN-DLSTM	PM _{2.5}	0.83	0.52	0.17	0.81	0.52	0.19	0.79	0.49	0.21	TIBAAE _{before} = 120(H)
		0.88	0.53	0.12	0.86	0.54	0.14	0.82	0.52	0.18	TIBAAE _{before} = 96(H)
		0.81	0.50	0.19	0.80	0.50	0.20	0.76	0.47	0.24	TIBAAE _{before} = 72(H)
		0.80	0.49	0.20	0.78	0.49	0.22	0.73	0.47	0.27	TIBAAE _{before} = 48(H)
		0.77	0.47	0.23	0.77	0.48	0.23	0.71	0.46	0.29	TIBAAE _{before} = 24(H)
DAerosol-NTM	0.96	0.94	0.04	0.96	0.94	0.04	0.95	0.95	0.87	0.05	TIBAAE _{before} = 120(H)
	0.97	0.95	0.03	0.97	0.95	0.03	0.96	0.90	0.90	0.04	TIBAAE _{before} = 96(H)
	0.96	0.94	0.04	0.96	0.93	0.04	0.95	0.95	0.87	0.05	TIBAAE _{before} = 72(H)
	0.95	0.92	0.05	0.95	0.93	0.05	0.93	0.93	0.85	0.07	TIBAAE _{before} = 48(H)
	0.95	0.90	0.05	0.94	0.92	0.06	0.92	0.84	0.84	0.08	TIBAAE _{before} = 24(H)

Table 17 The next following one-day forecast of future PM10 based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /accuracy	Train /precision	Train /RMSE	Validation /accuracy	Validation /precision	Validation /RMSE	Test /accuracy	Test /precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 24(H))
DLSTM	PM ₁₀	0.85	0.55	0.15	0.84	0.54	0.16	0.79	0.51	0.21	TIBAAE _{before} = 120(H)
		0.90	0.57	0.10	0.89	0.55	0.11	0.85	0.54	0.15	TIBAAE _{before} = 96(H)
		0.86	0.55	0.14	0.85	0.53	0.15	0.79	0.52	0.21	TIBAAE _{before} = 72(H)
		0.84	0.52	0.16	0.80	0.51	0.20	0.75	0.49	0.25	TIBAAE _{before} = 48(H)
		0.80	0.51	0.20	0.79	0.50	0.21	0.73	0.47	0.27	TIBAAE _{before} = 24(H)
DAerosol-NTM	0.97	0.95	0.04	0.96	0.95	0.03	0.95	0.95	0.89	0.05	TIBAAE _{before} = 120(H)
	0.98	0.96	0.02	0.98	0.96	0.02	0.96	0.96	0.90	0.04	TIBAAE _{before} = 96(H)
	0.96	0.94	0.04	0.96	0.95	0.04	0.94	0.94	0.86	0.06	TIBAAE _{before} = 72(H)
	0.95	0.93	0.05	0.95	0.94	0.05	0.93	0.85	0.07	0.07	TIBAAE _{before} = 48(H)
	0.94	0.92	0.06	0.94	0.92	0.06	0.92	0.93	0.84	0.07	TIBAAE _{before} = 24(H)

Table 18 The next following one-day forecast of future AQI quality control index based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /Accuracy	Train /Precision	Train /RMSE	Validation /Accuracy	Validation /Precision	Validation /RMSE	Test /Accuracy	Test /Precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 24(H))
DAerosol-NTM	AQI	0.84	0.53	0.16	0.82	0.53	0.18	0.80	0.50	0.20	TIBAAE _{before} = 120(H)
		0.89	0.54	0.11	0.87	0.55	0.13	0.83	0.53	0.17	TIBAAE _{before} = 96(H)
		0.83	0.52	0.17	0.81	0.52	0.19	0.78	0.49	0.22	TIBAAE _{before} = 72(H)
		0.82	0.51	0.18	0.79	0.50	0.21	0.75	0.49	0.25	TIBAAE _{before} = 48(H)
		0.79	0.50	0.21	0.78	0.50	0.22	0.73	0.48	0.27	TIBAAE _{before} = 24(H)
		0.97	0.95	0.04	0.97	0.95	0.03	0.95	0.88	0.05	TIBAAE _{before} = 120(H)
		0.98	0.96	0.02	0.98	0.96	0.02	0.96	0.90	0.04	TIBAAE _{before} = 96(H)
		0.97	0.95	0.03	0.96	0.95	0.04	0.95	0.87	0.05	TIBAAE _{before} = 72(H)
		0.95	0.93	0.05	0.95	0.93	0.05	0.94	0.86	0.06	TIBAAE _{before} = 48(H)
		0.94	0.90	0.06	0.94	0.92	0.06	0.93	0.85	0.07	TIBAAE _{before} = 24(H)

Table 24 provides support for the second hypothesis regarding TIBAAE. By considering the data from the preceding four days, the DAerosol-NTM model exhibited the most favorable performance in predicting the next 24 h. In comparison with DLSTM, which previously demonstrated the best results, the DAerosol-NTM framework showcased significant improvements. Specifically, it achieved an accuracy increase ranging from 12 to 29%, a precision enhancement ranging from 66 to 88%, and an RMSE reduction ranging from 72 to 77%.

5 Conclusions and future directions

This study introduces the DAerosol-NTM (deep aerosol with neural Turing machine) framework, which combines DNN and NTM models to accurately predict aerosol activity in terms of AQI values and six principal pollutants (SO₂, NO₂, PM2.5, PM10, O₃, and CO). Comparative analysis with previous studies utilizing MLP, DNN, LSTM, deep LSTM (DLSTM), and CNN-LSTM reveals that DAerosol-NTM achieves significant improvements in accuracy (8–31%), precision (46–91%), and root mean square error (RMSE) (24–85%). The incorporation of NTM external storage enables the utilization of up to 20 years of pollutant and particulate matter data, along with 22 independent variables from various meteorological stations across different districts in Tehran, Iran.

This capability distinguishes DAerosol-NTM as the first model capable of predicting aerosol pollution and poor AQI values 24–120 h in advance, surpassing DLSTM as the superior method in advanced air quality prediction.

However, it is important to acknowledge certain limitations and shortcomings of this study. Firstly, the focus is primarily on the weather and geographical conditions of Tehran, Iran, due to the lack of standardized data sources worldwide. Future endeavors should involve global partners and governmental bodies to collect data and implement similar models in other urban environments. Secondly, the unavailability of raw data from previous baseline studies prevents a direct comparison between the given methods and DAerosol-NTM within the exact geographic location. Lastly, incomplete or unavailable data from the Tehran meteorological and air quality stations have limited the scope of this study.

There are several approaches that can be implemented to further enhance the accuracy and precision of the DAerosol-NTM model. One potential avenue for improvement is the application of generative adversarial network (GAN) networks and Turing networks to recover lost data, enrich the dataset, and generate future data without relying solely on LSTM networks. Machine learning models derived from NTM, which modify memory architecture and

834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883

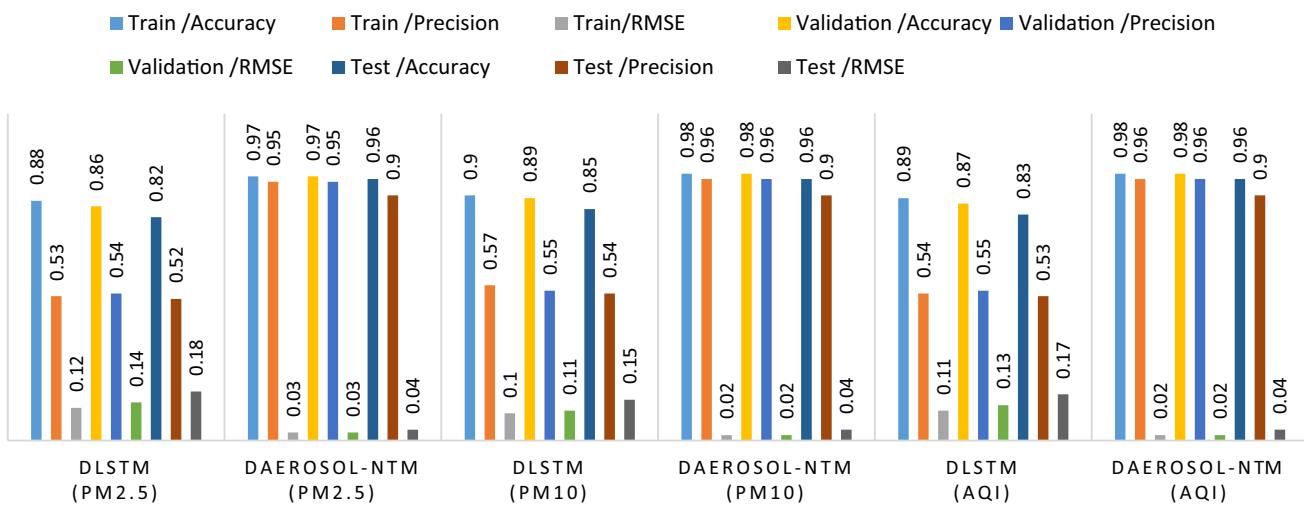


Fig. 15 Comparison of $TIBAAE_{before} = 96(H)$ and $TIBAAE_{after} = 24(H)$ for DAerosol-NTM with DLSTM to the daily forecast of the aerosol concentration (PM2.5, PM10) and future quality control index (AQI) based

produce neural computer models that can be separated by differential neural computers (DNCs), have shown greater strength compared to NTM models alone. By combining GAN networks with DNC-based models, it is possible to alter memory architecture and enable the prediction of contamination parameters.

Additionally, the evolution of the control neural network using algorithms such as NeuroEvolution of Augmenting Topologies (NEAT) and particle swarm optimization (PSO), or similar optimization techniques, can further optimize NTM- and DNC-based models. Furthermore, investigating the time complexity of the DAerosol-NTM framework and exploring new aerosol-related tasks can enrich the capabilities of NTM and DNC models.

Despite its limitations and possibilities, DAerosol-NTM has demonstrated enhanced performance within the Tehran meteorological context. Integrating previously successful time-series models with NTM has the potential to make DAerosol-NTM a crucial tool for addressing air quality prediction concerns, which have significant implications for human health and the environment. It is worth exploring the integration of different machine learning approaches with DAerosol-NTM to achieve accurate predictions of air quality indices.

Appendix A: research concepts

This section addresses the theoretical and technical nuances and specialized terms used in the study and the advancements of DAerosol-NTM, including the AQI, deep learning (DL), deep short-term neural networks (DSTNN), and the neural Turing machine. For direct remarks on the

materials and methods and the results of this paper, please visit the segments following “Research Concepts” section.

AQI index

AQI is an air quality indicator that reflects and evaluates the air quality status. Although the AQI scale is continuous, different descriptive categories have been implemented to ease public communication, as given in Table 25 [3, 54].

One crucial factor to note about AQI is its calculation method. The concentrations are independently measured using six reported pollutant parameters (SO₂, NO₂, PM2.5, PM10, O₃, and CO). Any pollutant's highest value is taken as the AQI value [4]. For this reason, the calculation of AQI is driven by a single parameter, and the accurate prediction of all six pollutants is essential. In most urban areas, PM2.5 and PM10 are the leading drivers of AQI calculations and the dominating reason behind pollution and erosion.

Deep learning

Deep learning is machine learning, but it functions more similarly to the human brain in a deeper and more advanced form. In other words, deep learning is part of a more prominent family of machine learning that focuses on methods based on artificial neural networks. This type of learning is an essential element in data science that receives raw inputs and extracts high-level features in several layers, including statistics and forecasting modeling. In-depth learning is very beneficial for data scientists to collect, analyze, and interpret large amounts of data. In general, it makes the process faster and easier. In other words, deep learning models process data more accurately

Table 19 The next following two-day forecast of future AQI quality control index based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /Accuracy	Train /Precision	Train /RMSE	Validation /Accuracy	Validation /Precision	Validation /RMSE	Test /Accuracy	Test /Precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 48(H))
DAerosol-NTM	DLSTM	AQI	0.79	0.48	0.21	0.78	0.48	0.22	0.77	0.45	0.23
			0.83	0.49	0.17	0.82	0.50	0.18	0.79	0.48	0.21
			0.78	0.48	0.22	0.76	0.47	0.24	0.74	0.44	0.26
			0.77	0.46	0.23	0.75	0.45	0.25	0.72	0.44	0.28
			0.74	0.45	0.26	0.73	0.45	0.27	0.69	0.42	0.31
			0.92	0.90	0.08	0.92	0.90	0.08	0.91	0.83	0.09
			0.93	0.91	0.07	0.93	0.91	0.07	0.93	0.85	0.07
NTM	DLSTM	AQI	0.91	0.90	0.09	0.91	0.91	0.09	0.90	0.82	0.10
			0.90	0.88	0.10	0.90	0.89	0.10	0.89	0.81	0.11
			0.89	0.87	0.11	0.89	0.88	0.11	0.88	0.79	0.12

Table 20 The next following three-day forecast of future AQI quality control index based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /Accuracy	Train /Precision	Train /RMSE	Validation /Accuracy	Validation /Precision	Validation /RMSE	Test /Accuracy	Test /Precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 72(H))
DAerosol-NTM	DLSTM	AQI	0.74	0.43	0.26	0.73	0.43	0.27	0.72	0.41	0.28
			0.80	0.45	0.20	0.80	0.45	0.20	0.74	0.43	0.26
			0.73	0.43	0.27	0.71	0.43	0.29	0.70	0.40	0.30
			0.72	0.42	0.28	0.69	0.41	0.31	0.68	0.40	0.32
			0.70	0.40	0.30	0.67	0.40	0.33	0.65	0.41	0.35
			0.88	0.85	0.12	0.88	0.84	0.12	0.86	0.79	0.14
			0.90	0.87	0.10	0.89	0.87	0.11	0.88	0.81	0.12
NTM	DLSTM	AQI	0.88	0.85	0.12	0.87	0.86	0.13	0.85	0.78	0.15
			0.87	0.83	0.13	0.85	0.84	0.15	0.84	0.77	0.16
			0.86	0.82	0.14	0.84	0.83	0.16	0.82	0.75	0.18

Table 21 The next following four-day forecast of future AQI quality control index based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /Accuracy	Train /Precision	Train /RMSE	Validation /Accuracy	Validation /Precision	Validation /RMSE	Test /Accuracy	Test /Precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 96(H))
DAerosol-NTM	DLSTM	AQI	0.70	0.38	0.30	0.68	0.38	0.32	0.68	0.37	0.32
			0.76	0.40	0.24	0.75	0.40	0.25	0.69	0.39	0.31
			0.67	0.37	0.33	0.66	0.37	0.34	0.66	0.36	0.34
			0.66	0.36	0.34	0.64	0.36	0.36	0.65	0.35	0.35
			0.65	0.35	0.35	0.62	0.35	0.38	0.63	0.34	0.37
			0.83	0.80	0.17	0.82	0.84	0.18	0.81	0.74	0.19
			0.86	0.83	0.14	0.83	0.82	0.17	0.83	0.76	0.17
			0.82	0.80	0.18	0.82	0.81	0.18	0.81	0.73	0.19
			0.82	0.78	0.18	0.80	0.79	0.20	0.79	0.72	0.21
			0.80	0.77	0.20	0.79	0.77	0.21	0.78	0.70	0.22

Table 22 The next following five-day forecast of future AQI quality control index based on the hours before (24H to 120H) the aerosol occurrence

Method	Parameter	Train /Accuracy	Train /Precision	Train /RMSE	Validation /Accuracy	Validation /Precision	Validation /RMSE	Test /Accuracy	Test /Precision	Test /RMSE	Prediction interval (TIBAAE _{after} = 120(H))
DAerosol-NTM	DLSTM	AQI	0.65	0.32	0.35	0.62	0.32	0.38	0.61	0.32	0.39
			0.69	0.35	0.31	0.69	0.34	0.31	0.62	0.34	0.38
			0.64	0.32	0.36	0.61	0.32	0.39	0.60	0.31	0.40
			0.63	0.31	0.37	0.59	0.30	0.41	0.59	0.30	0.41
			0.60	0.30	0.40	0.58	0.30	0.42	0.57	0.30	0.43
			0.80	0.78	0.20	0.78	0.77	0.22	0.77	0.70	0.23
			0.82	0.79	0.18	0.81	0.78	0.19	0.78	0.71	0.21
			0.79	0.77	0.21	0.77	0.76	0.23	0.75	0.68	0.25
			0.78	0.74	0.22	0.78	0.74	0.22	0.74	0.66	0.26
			0.75	0.73	0.25	0.74	0.72	0.26	0.73	0.65	0.27

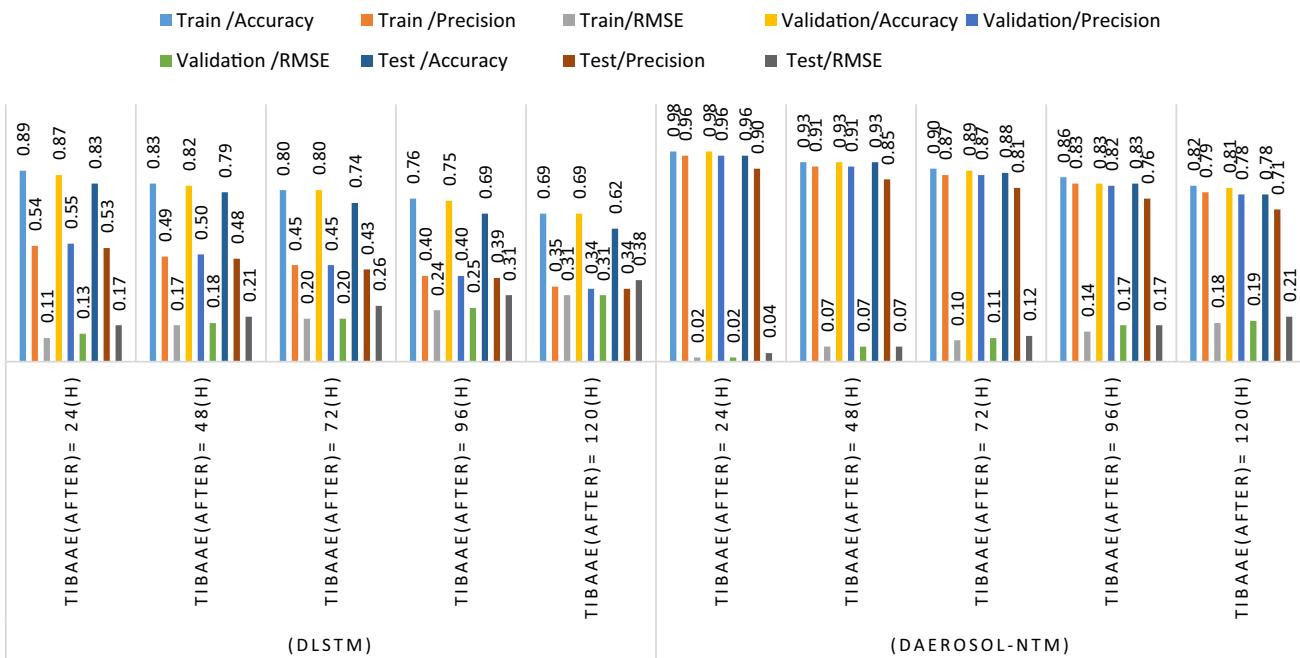


Fig. 16 Comparing TIBAAE_{after} for DAerosol-NTM and DLSTM based on four days before data (TIBAAE_{before} = 96(H))

Table 23 Summary of experimental result (Experiment-1) in support of Hypothesis 1

Method									
DAerosol-NTM			CNN- LSTM			DLSTM			
Test/ Accuracy	Test/ Precision	Test / RMSE	Test/ Accuracy	Test/ Precision	Test / RMSE	Test/ Accuracy	Test/ Precision	Test / RMSE	
PM _{2.5}	0.95	0.86	0.05	0.80	0.56	0.20	0.80	0.55	0.20
PM ₁₀	0.96	0.87	0.04	0.81	0.56	0.19	0.81	0.55	0.20
AQI	0.81	0.72	0.19	0.75	0.49	0.25	0.75	0.48	0.25

Method									
LSTM			DNN (MLP)			MLP			
Test/ Accuracy	Test/ Precision	Test / RMSE	Test/ Accuracy	Test/ Precision	Test / RMSE	Test/ Accuracy	Test/ Precision	Test / RMSE	
PM _{2.5}	0.73	0.48	0.27	0.74	0.49	0.26	0.72	0.45	0.28
PM ₁₀	0.74	0.49	0.26	0.75	0.51	0.25	0.73	0.46	0.28
AQI	0.70	0.46	0.30	0.71	0.47	0.29	0.69	0.44	0.31

944 and quicker due to the complexity and high ability to learn,
945 especially in big data in research fields such as image
946 processing, pattern recognition, and computer vision [55].
947 Deep learning is a powerful machine learning method that
948 provides approximation, classification, and predictability
949 [57–59] and [56].

950 See Fig. 17.

Deep short-term neural network

951
952 The LSTM architecture [55] works better than conventional
953 neural networks for long-term tasks [58], including a
954 deep and short-term neural network. Thus, the lower the
955 short-term neural network output sequence, the higher the
956 short-term neural network input sequence [61]. The deep
957 LSTM architecture used in Refs. [7, 63] can be described
958 by $(\sigma(x), h_t^l, i_t^l, f_t^l, s_t^l, o_t^l)$, where l represents the layer index

Table 24 Summary of experimental result (Experiment-2) in support of Hypothesis 2

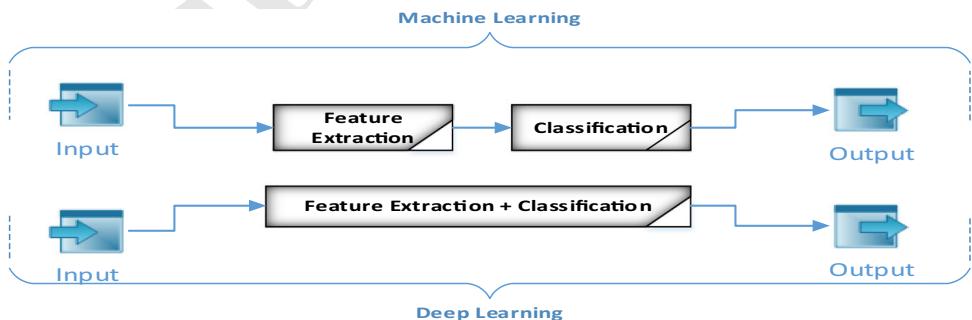
TIBAAE after	DAerosol-NTM					Average	DLSTM					Average		
	Test/Accuracy						Test/Accuracy							
	120(H)	96(H)	72(H)	48(H)	24(H)		120(H)	96(H)	72(H)	48(H)	24(H)			
TIBAAE before	120(H)	0.77	0.80	0.87	0.91	0.95	0.86	0.61	0.69	0.72	0.77	0.79	0.72	
	96(H)	0.78	0.84	0.89	0.93	0.96	0.88	0.63	0.70	0.74	0.80	0.82	0.74	
	72(H)	0.75	0.81	0.85	0.90	0.95	0.85	0.60	0.66	0.70	0.75	0.76	0.69	
	48(H)	0.74	0.79	0.84	0.89	0.93	0.84	0.59	0.66	0.69	0.72	0.73	0.68	
	24(H)	0.73	0.78	0.82	0.88	0.92	0.83	0.57	0.63	0.65	0.69	0.71	0.65	
	120(H)	0.77	0.81	0.86	0.91	0.95	0.86	0.60	0.69	0.72	0.77	0.79	0.72	
	96(H)	0.78	0.84	0.88	0.93	0.96	0.88	0.62	0.69	0.74	0.79	0.85	0.74	
	72(H)	0.76	0.82	0.86	0.91	0.94	0.86	0.60	0.66	0.70	0.74	0.79	0.70	
	48(H)	0.74	0.79	0.84	0.89	0.93	0.84	0.58	0.65	0.66	0.70	0.75	0.67	
	24(H)	0.74	0.79	0.83	0.89	0.93	0.84	0.57	0.63	0.65	0.69	0.73	0.65	
PM2.5	120(H)	0.77	0.81	0.86	0.91	0.95	0.86	0.61	0.68	0.72	0.77	0.80	0.72	
	96(H)	0.78	0.83	0.88	0.93	0.96	0.88	0.62	0.69	0.74	0.79	0.83	0.73	
	72(H)	0.75	0.81	0.85	0.90	0.95	0.85	0.60	0.66	0.70	0.74	0.78	0.70	
	48(H)	0.74	0.79	0.84	0.89	0.94	0.84	0.59	0.65	0.68	0.72	0.75	0.68	
	24(H)	0.73	0.78	0.82	0.88	0.93	0.83	0.57	0.63	0.65	0.69	 0.73 	0.65	
TIBAAE after	DAerosol-NTM					Average	DLSTM					Average		
	Test/Precision						Test/Precision							
	120(H)	96(H)	72(H)	48(H)	24(H)		120(H)	96(H)	72(H)	48(H)	24(H)			
TIBAAE before	120(H)	0.70	0.74	0.79	0.82	0.87	0.78	0.32	0.36	0.40	0.44	0.49	0.40	
	96(H)	0.71	0.79	0.81	0.84	0.90	0.81	0.32	0.36	0.42	0.46	0.52	0.42	
	72(H)	0.68	0.74	0.78	0.83	0.87	0.78	0.30	0.35	0.39	0.44	0.47	0.39	
	48(H)	0.67	0.71	0.76	0.82	0.85	0.76	0.30	0.33	0.40	0.43	0.47	0.39	
	24(H)	0.65	0.70	0.75	0.78	0.84	0.74	0.30	0.34	0.39	0.42	0.46	0.38	
	120(H)	0.70	0.73	0.78	0.82	0.89		0.32	0.37	0.40	0.44	0.51		
	96(H)	0.71	0.77	0.80	0.84	0.90	0.80	0.33	0.37	0.42	0.48	0.54	0.43	
	72(H)	0.67	0.73	0.78	0.81	0.86	0.77	0.30	0.35	0.39	0.44	0.52	0.40	
	48(H)	0.65	0.71	0.76	0.80	0.85	0.75	0.30	0.34	0.40	0.43	0.49	0.39	
	24(H)	0.65	0.70	0.75	0.78	0.84	0.74	0.30	0.34	0.40	0.42	0.47	0.39	
PM10	120(H)	0.70	0.74	0.79	0.83	0.88	0.79	0.32	0.37	0.41	0.45	0.50	0.41	
	96(H)	0.71	0.76	0.81	0.85	0.90	0.81	0.34	0.39	0.43	0.48	0.53	0.43	
	72(H)	0.68	0.73	0.78	0.82	0.87	0.78	0.31	0.36	0.40	0.44	0.49	0.40	
	48(H)	0.66	0.72	0.77	0.81	0.86	0.76	0.30	0.35	0.40	0.44	0.49	0.40	
	24(H)	0.65	0.70	0.75	0.79	0.85	0.75	0.30	0.34	0.41	0.42	0.48	0.39	
TIBAAE after	DAerosol-NTM					Average	DLSTM					Average		
	Test/RMSE						Test/RMSE							
	120(H)	96(H)	72(H)	48(H)	24(H)		120(H)	96(H)	72(H)	48(H)	24(H)			
TIBAAE before	120(H)	0.23	0.20	0.13	0.09	0.05	0.14	0.39	0.31	0.28	0.23	0.21	0.28	
	96(H)	0.22	0.16	0.11	0.07	0.04	0.12	0.37	0.30	0.26	0.20	0.18	0.26	
	72(H)	0.25	0.19	0.15	0.10	0.05	0.15	0.40	0.34	0.30	0.25	0.24	0.31	
	48(H)	0.26	0.21	0.16	0.11	0.07	0.16	0.41	0.34	0.31	0.28	0.27	0.32	
	24(H)	0.27	0.22	0.18	0.12	0.08	0.17	0.43	0.37	0.35	0.31	0.29	0.35	
	120(H)	0.23	0.19	0.14	0.09	0.05	0.12	0.40	0.31	0.28	0.23	0.21	0.29	

Table 24 (continued)

TIBAAE after	DAerosol-NTM					Average	DLSTM					Average		
	Test/RMSE						Test/RMSE							
	120(H)	96(H)	72(H)	48(H)	24(H)		120(H)	96(H)	72(H)	48(H)	24(H)			
AQI	96(H)	0.22	0.16	0.12	0.07	0.04	0.12	0.38	0.31	0.26	0.21	0.15	0.26	
	72(H)	0.24	0.18	0.14	0.09	0.06	0.14	0.40	0.34	0.30	0.26	0.21	0.30	
	48(H)	0.26	0.21	0.16	0.11	0.07	0.16	0.42	0.35	0.34	0.30	0.25	0.33	
	24(H)	0.26	0.21	0.17	0.11	0.07	0.16	0.43	0.37	0.35	0.31	0.27	0.35	
	120(H)	0.23	0.19	0.14	0.09	0.05	0.14	0.39	0.32	0.28	0.23	0.20	0.28	
	96(H)	0.21	0.17	0.12	0.07	0.04	0.12	0.38	0.31	0.26	0.21	0.17	0.27	
	72(H)	0.25	0.19	0.15	0.10	0.05	0.15	0.40	0.34	0.30	0.26	0.22	0.30	
	48(H)	0.26	0.21	0.16	0.11	0.06	0.16	0.41	0.35	0.32	0.28	0.25	0.32	
	24(H)	0.27	0.22	0.18	0.12	0.07	0.17	0.43	0.37	0.35	0.31	0.27	0.35	

Table 25 Different levels (AQI) [51]

Air Quality Index (AQI) Values	Levels of Health Concern	Colors
AQI	Air quality conditions:	Symbol
0 to 50	Good	Green
51 to 100	Moderate	Yellow
101 to 150	Unhealthy for Sensitive Groups	Orange
151 to 200	Unhealthy	Red
201 to 300	Very Unhealthy	Purple
301 to 500	Hazardous	Maroon

Fig. 17 The difference between machine learning and deep learning 6

followed by Eqs. 4–9. Equation 10 defines the Softmax activity function for a simple N-dimensional problem.

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \quad (4)$$

$$h_t^l = o_t^l \tanh(s_t^l) \quad (5)$$

$$i_t^l = \sigma(W_i^l[X_t; h_{t-1}^l; h_t^{l-1}] + b_i^l) \quad (6)$$

$$f_t^l = \sigma(W_f^l[X_t; h_{t-1}^l; h_t^{l-1}] + b_f^l) \quad (7)$$

$$s_t^l = f_t^l s_{t-1}^l + i_t^l \tanh(W_s^l[X_t; h_{t-1}^l; h_t^{l-1}] + b_s^l) \quad (8)$$

$$o_t^l = \sigma(W_o^l[X_t; h_{t-1}^l; h_t^{l-1}] + b_o^l) \quad (9)$$

$$S_N = \left\{ a \in R^N : a_i \in [0, 1], \sum \sum^N a_i = 1 \right\} \quad (10)$$

968

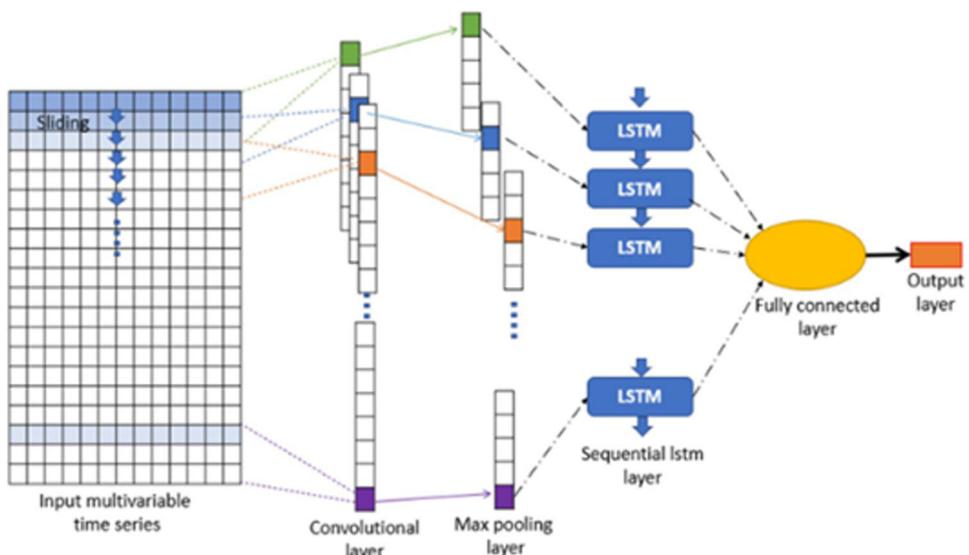
970

972

974

966

Fig. 18 Architecture of the CNN-LSTM model [27]



975 CNN-DLSTM

976 The use of classical CNN architecture is the best choice
 977 when input networks are 2-D or 3-D tensors like images or
 978 videos [64]. Since LSTMs architectures were more adapted
 979 for 1-D data, a new variant of LSTM called convolutional
 980 LSTM or ConvLSTM [65] is designed. In this architecture,
 981 the LSTM cell, which contains a convolution operation and
 982 input dimension of data, is kept in the output layer instead
 983 of just a 1-D vector. A convolution operation has replaced
 984 matrix multiplication at each gate of classical LSTM. The
 985 ConvLSTM architecture applies the capabilities of CNN
 986 and LSTM neural networks. It is normally developed for
 987 2-D spatiotemporal data such as satellite images. In the first
 988 part of this model, convolutional layers extract essential
 989 features of the input data, and the results were flattened in a
 990 1-D tensor so that they can use as input for the second part
 991 of the model (LSTM). Finally, before passing data in the
 992 last hidden layer, information has to be reshaped in the
 993 original form of input data. The architecture of CNN-
 994 LSTM is shown in Fig. 18.

995 Neural Turing machine

996 It is a method derived from the Turing machine and neural
 997 networks. This model consists of recurrent neural networks
 998 (RNNs) [60] with an addressable external memory along
 999 with the ability of the recursive neural networks to perform
 1000 algorithmic tasks such as sorting, copying, and N-gram.
 1001 Generally speaking, a memory bank, a controller, and read
 1002 and write heads are the main components of this method.
 1003 The controller's job is to receive data from the outside and
 1004 generate outputs during the update cycle. In addition, the
 1005 neural Turing machine method guides the read and write

heads directly into the external memory in the form of a tape [57, 66]. Figure 19 shows the structure of the neural Turing machine in general in section (A) and its components in section (B) [10] and [62].

1006 The actions that the controller performs through the heads

1007 The controller consists of five main functions: read, write,
 1008 erase, move to the next memory cell, and move to the
 1009 previous one. During each update cycle, the network con-
 1010 troller receives inputs from the external environment and
 1011 publishes the outputs in response. The network also starts
 1012 reading and writing from a memory matrix using parallel
 1013 read and write heads. The dotted line in Fig. 19A shows the
 1014 division between the NTM circuit and the outside world.
 1015 Each part of the structure is recognizable and distinct. This
 1016 feature makes the network easier to train with gradient
 1017 descent by defining “blurry or unclear” reading and writing
 1018 operations. It also communicates more or less with all
 1019 memory elements (instead of considering one element as a
 1020 typical Turing machine or digital computer). A “focus”
 1021 mechanism determines the degree of blur.

1022 Each operation forces the read/write heads to commu-
 1023 nicate with the small memory and ignore the rest. Because
 1024 the interaction with memory is low and fragmented, NTM
 1025 is based on data storage without interference. The output of
 1026 the heads determines which memory location gets the most
 1027 attention. These outputs define a normalized weighting on
 1028 the rows of the memory matrix (referring to memory
 1029 locations). At each assigned weight, each read/write head
 1030 establishes the degree to which the head reads/writes in
 1031 each area. Each head focuses precisely on a single memory
 1032 location or multiple memory locations. Figure 20 displays

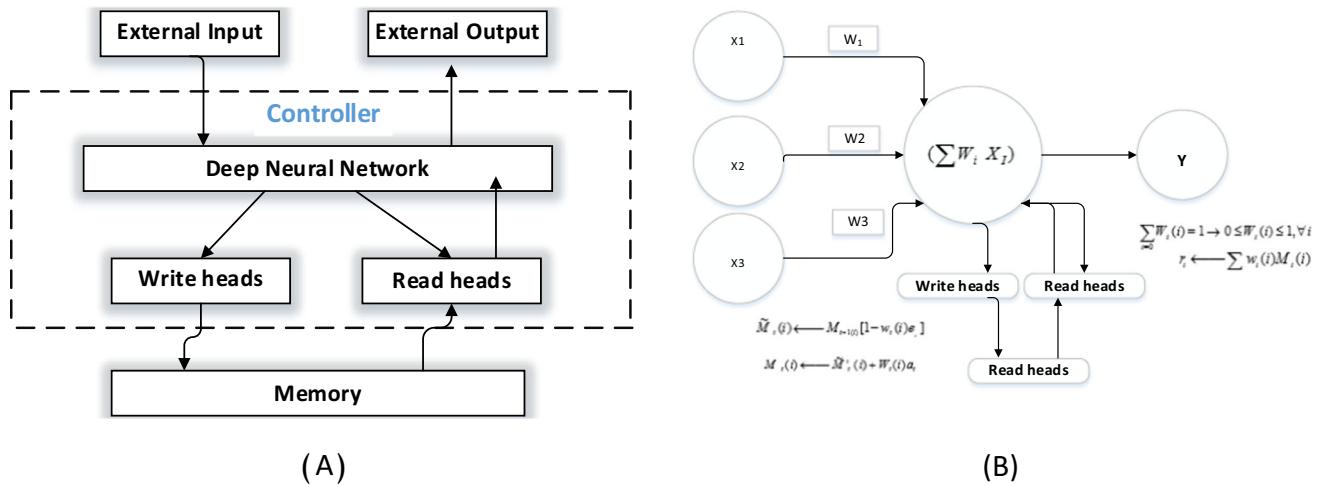


Fig. 19 Structure of the neural Turing machine **A** in general and **B** in part [10]

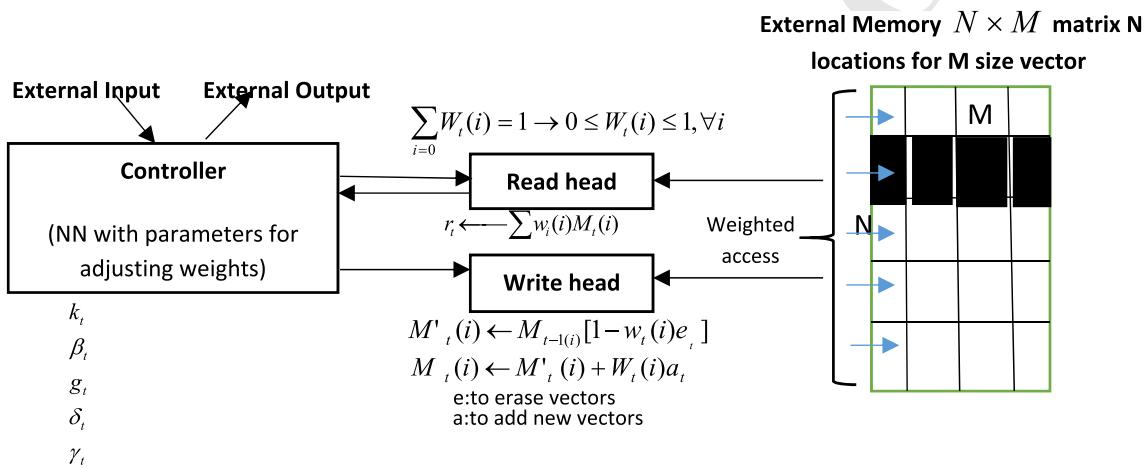


Fig. 20 How to access external memory [60]

1037 the conceptual model of the controller's actions through the
1038 heads.

1039 **Reading and writing**

1040 The read and write operations are normalized weighting
1041 functions over the memory locations, similar to attention
1042 mechanisms. These weightings define a continuous distribution
1043 over the memory locations to make the operation
1044 differentiable. The reading operation is a simple linear
1045 combination of the memory locations:
1046

Figure 21 shows the read and write operation in NTM [66] in sections (A) and (B).

Figure 21A shows the Nth element of $W_t(i)$ according to the limits of Eq. 11:

$$\sum_{i=0}^N W_t(i) = 1 \rightarrow 0 \leq W_t(i) \leq 1, \forall i \quad (11)$$

1051 M_t is the contents of the $M \times N$ memory matrix at time t ,
1052 where N is the number of memory locations and M is the
1053 size of the vector in each location. W is the vector of
1054 weights on the N locations propagated by a head reading at
1055 time t since all weights are normalized.

1056 The read vector is a weighted convex combination of the
1057 memory location. The length M reads the return r_t vector
1058 by the head defined as a combination of row vectors $M_t(i)$
1059 in memory, as shown in Eq. 12, separable in terms of
1060 memory and weight.

$$r_t \leftarrow \sum w_i M_t(i) \quad (12)$$

1062 Figure 21B shows that the writing operation is a convex
1063 combination of erasing and writing to the memory locations.
1064 The write head outputs both erase (e) and add (a)
1065 vectors. Writing to the memory will then be made by
1066 erasing the locations defined by the write weighting vector
1067 and adding the locations specified by the same weighting

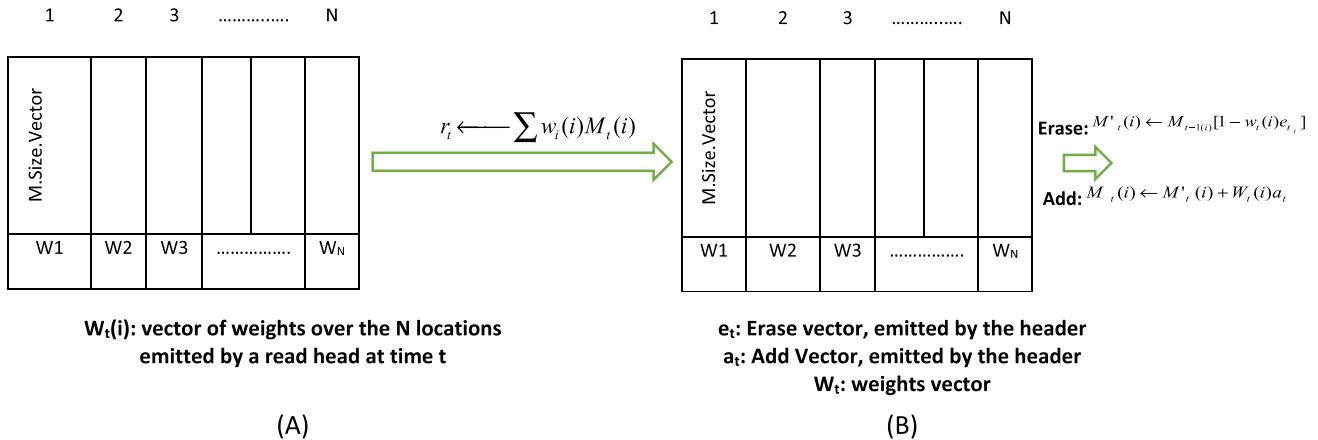


Fig. 21 Read and write operation in NTM [60]

vector. Again, notice that erasing and writing locations in different proportions make the operation differentiable, where parts of memory are erased according to the weighting vector, as shown in Eq. 13:

$$M'_t(i) \leftarrow M_{t-1}(i)[1 - w_t(i)e_{r_t}] \quad (13)$$

Multiplying it in the memory location works as a peer-to-peer. Therefore, the elements of the memory location are zeroed where the weight is and the clearing element is 1. If the weight or clearance element is zero, the memory remains unchanged. When there are multiple writing heads, cleansing can be done in any order, where new information is added to locations defined by the weightings, as shown in Eq. 14:

$$M_t(i) \leftarrow M'_t(i) + W_t(i)a_t \quad (14)$$

The combine, clear, and add operations produce the final memory content at time t. Because delete and add are different, compound writing operations are also distinguishable. Note that both addition and subtraction vectors have M-independent components. Again, adding the sequence of vectors by multiple heads is also trivial. It allows precise control over the modified elements in each memory location.

Addressing mechanism

In the previous section, the equations of reading and writing were shown and examined. However, no explanation was given on how the weights are produced. These weights are created by combining two addressing mechanisms with complementary features. The first mechanism is content-based addressing, which looks at locations based on the similarity between current values and values

published by the controller that addresses the content of Hopfield networks. The advantage of content-based addressing is that the retrieval is easy. It only needs a controller to estimate a portion of the stored data to compare it to memory for the accurate stored value. However, content-based addressing is not suitable for solving all problems. In some works, the content of a variable is as desired [57]. However, the variable still needs a recognizable name or address. Computational problems are as follows: Variables X and Y can take both values, but the $f(x, y) = x \times y$ trend must still be defined. A controller initializes the x and y variables, stores them in different addresses, retrieves them, and performs a multiplication algorithm. In this case, the variables are addressable by location, not content. This form of addressing is called location-based addressing. Content-based addressing is more commonly used than location-based addressing, as the content of a memory location can contain location information within it. However, it is necessary to provide location-based addressing as a primary operation for some generalized forms in experiments. As a result, both mechanisms are used together. The flow diagram of the addressing mechanism indicates the sequence of operations to construct a weight vector while reading and writing [11] (see Fig. 22).

Network controller

The NTM structure is described in the section. It has several free parameters: memory size, number of read/write heads, and allowable location changes. Perhaps, the most important structure is the type of neural network used as the controller, especially when deciding whether to use a recurrent or a feedforward neural network (FNN). A

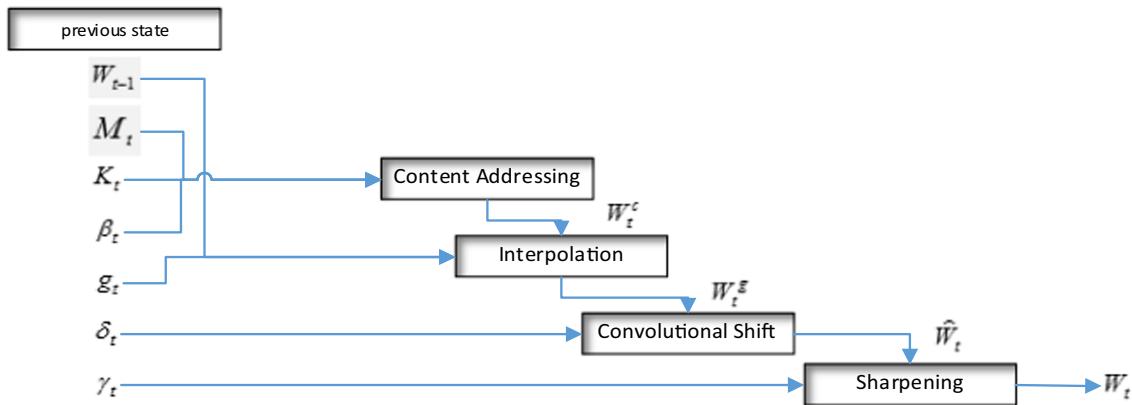


Fig. 22 Flow diagram of the addressing mechanism [10]

recursive controller such as LSTM has internal memory to complete a larger matrix. Suppose a central processing unit controller on a digital computer (albeit with adaptive instructions instead of predefined ones) is compared to the RAM matrix. In that case, the hidden activities of the recursive controller are similar to the processor's registers. They allow the controller to combine information throughout the various time stages of the operation.

On the other hand, a feedforward controller mimics an RNN network by reading and writing to the specific memory location at each step. In addition, the feedforward controllers often give more transparency to network operations, because the read and write patterns in the memory matrix are usually easier to interpret than the internal state of an RNN. However, one of the limitations of a feedforward controller is that the number of read/write heads simultaneously causes a constraint (limiting operation) on the type of computation that NTM can perform. With a single read head, only a single conversion can be performed on a single memory vector at each time step. With two reading heads, it can do binary conversions, etc. Recurrent controllers can store reading vectors from previous efforts, so they do not suffer from this limitation.

Data availability This research also uses well-known data from public repositories that can be shared based on request.

Declarations

Conflict of interest There are no conflicts of interest to disclose for this submission titled "DAerosol-NTM: Applying deep learning and neural Turing machine in aerosol prediction."

References

- Nakata M, Sano I, Mukai S (2015) Relation between aerosol characteristics and impact factors on climate and environment. In:

- International geoscience and remote sensing symposium (IGARSS), 2015-November, pp 2342–2345
- Qin Y, Yin Y, Wu Z, Shi L (2010) An observational study of atmospheric Aerosol in the Shijiazhuang area. In: 2010 2nd IITA international conference on geoscience and remote sensing, IITA-GRS 2010, 2, pp 328–331
 - Diro AA, Chilamkurti N (2018) Distributed attack detection scheme using deep learning approach for Internet of Things. *Futur Gener Comput Syst* 82:761–768. <https://doi.org/10.1016/j.future.2017.08.043>
 - Zhu S, Lian X, Liu H, Hu J, Wang Y, Che J (2017) Daily air quality index forecasting with hybrid models: a case in China. *Environ Pollut* 231:1232–1244. <https://doi.org/10.1016/j.envpol.2017.08.069>
 - Kim S, Lee JM, Lee J, Seo J (2019) Deep-dust: predicting concentrations of fine dust in Seoul using LSTM. *arXiv Preprint arXiv:1901.10106*, pp 8–10
 - Xayasouk T, Lee HM, Lee G (2020) Air pollution prediction using long short-term memory (LSTM) and deep auto encoder (DAE) models. *Sustainability*. <https://doi.org/10.3390/su12062570>
 - Sharma A, Mitra A, Sharma S, Roy S (2018) Estimation of air quality index from seasonal trends using deep. *Int Conf Artif Neural Netw* 2018:511–521. <https://doi.org/10.1007/978-3-030-01424-7>
 - Ma J, Cheng JC, Lin C, Tan Y, Zhang J (2019) Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmos Environ* 214:116885. <https://doi.org/10.1016/j.atmosenv.2019.116885>
 - Pengfei Y, Juanjuan H, Xiaoming L, Kai Z (2018b) Industrial air pollution prediction using deep neural network. *Commun Comput Inf Sci* 951:173–185. https://doi.org/10.1007/978-981-13-2826-8_16
 - Gulcehre C, Chandar S, Cho K, Bengio Y (2016) Dynamic neural Turing machine with soft and hard addressing schemes. *arXiv Preprint arXiv:1607.00036*
 - Turing AM (1950) A quarterly review of psychology and philosophy I. Computing machinery and intelligence. *Mind* 59:433–460
 - Siegelmann HT, Sontag ED (1991) Turing computability with neural nets. *Appl Math Lett* 4(6):77–80. [https://doi.org/10.1016/0893-9659\(91\)90080-F](https://doi.org/10.1016/0893-9659(91)90080-F)
 - Han W, Cha S, Ha H-J (2006) Method and apparatus for multi-layered video encoding and decoding. <https://patents.google.com/patent/US20060120450A1/en>
 - Graves A, Wayne G, Danihelka I (2014) Neural Turing machines. *arXiv preprint arXiv:1410.5401*, pp 1–26

- 1211 15. Malekmohammadi Faradonbeh S, Safi-Esfahani F (2019) A
1212 review on neural Turing machine. [https://arxiv.org/abs/1904.
1213 05061](https://arxiv.org/abs/1904.05061)
- 1214 16. Baddeley A (1996) Working memory and executive control.
1215 Philos Trans R Soc Lond Ser B Biol Sci 351(1346):1397–1404.
1216 <https://doi.org/10.1098/rstb.1996.0123>
- 1217 17. Weston J, Bordes A, Chopra S, Rush AM, Van Merriënboer B,
1218 Joulin A, Mikolov T (2016). Towards AI-complete question
1219 answering: a set of prerequisite toy tasks. In: 4th International
1220 conference on learning representations, ICLR 2016—conference
1221 track proceedings. <https://arxiv.org/abs/1502.05698>
- 1222 18. Graves A, Wayne G, Reynolds M, Harley T, Danihelka I,
1223 Grabska-Barwińska A, Colmenarejo SG, Grefenstette E,
1224 Ramalho T, Agapiou J, Badia AP (2016) Hybrid computing using
1225 a neural network with dynamic external memory. Nature
1226 538(7626):471–476. <https://doi.org/10.1038/nature20101>
- 1227 19. Yang G, Rush AM (2019) Lie-access neural Turing machines. In:
1228 5th International conference on learning representations, ICLR
1229 2017—conference track proceedings. [http://arxiv.org/abs/1602.
1230 08671](http://arxiv.org/abs/1602.08671)
- 1231 20. Zaremba W, Sutskever I (2015) Reinforcement learning neural
1232 Turing machines—revised. arXiv Preprint <arXiv:1505.00521>
- 1233 21. Greve RB, Jacobsen EJ, Risi S (2016) Evolving neural Turing
1234 machines for reward-based learning. In: GECCO 2016—pro-
1235 ceedings of the 2016 genetic and evolutionary computation
1236 conference, pp 117–124. [https://doi.org/10.1145/2908812.
1237 2908930](https://doi.org/10.1145/2908812.2908930)
- 1238 22. Stanley KO, Miikkulainen R (2002) Evolving neural networks
1239 through augmenting topologies. Evol Comput 10(2):99–127
- 1240 23. Stein G, Gonzalez AJ, Barham C (2013) Machines that learn and
1241 teach seamlessly. IEEE Trans Learn Technol 6(4):389–402.
1242 <https://doi.org/10.1109/TLT.2013.32>
- 1243 24. Zhao J, Peng G (2011) NEAT versus PSO for evolving autonomous
1244 multi-agent coordination on pursuit-evasion problem.
1245 Lecture Notes in Electrical Engineering, vol 2, 711–717
- 1246 25. Verbancsics P, Harguess J (2013) Generative NeuroEvolution for
1247 deep learning. arXiv Preprint <arXiv:1312.5355>
- 1248 26. Lin B, Zhu J (2018) Changes in urban air quality during urbaniza-
1249 tion in China. J Clean Prod 188:312–321. [https://doi.org/10.
1250 1016/j.jclepro.2018.03.293](https://doi.org/10.1016/j.jclepro.2018.03.293)
- 1251 27. Bekkar A, Hssina B, Douzi S, Douzi K (2021) Air-pollution
1252 prediction in smart city, deep learning approach. J Big Data
1253 8(1):1–21. <https://doi.org/10.1186/s40537-021-00548-1>
- 1254 28. Akhtar A, Masood S, Gupta C, Masood A (2018) Prediction and
1255 analysis of pollution levels in Delhi using multilayer perceptron.
1256 Adv Intell Syst Comput 542:563–572. [https://doi.org/10.1007/
1257 978-981-10-3223-3_54](https://doi.org/10.1007/978-981-10-3223-3_54)
- 1258 29. Raturi R, Prasad JR (2018) Recognition of future air quality index
1259 using artificial neural network. Int Res J Eng Technol (IRJET)
1260 5:2395–0056
- 1261 30. Wang J, Zhang X, Guo Z, Lu H (2017) Developing an early-
1262 warning system for air quality prediction and assessment of cities
1263 in China. Expert Syst Appl 84:102–116. [https://doi.org/10.1016/j.
1264 eswa.2017.04.059](https://doi.org/10.1016/j.eswa.2017.04.059)
- 1265 31. Kök I, Şimşek MU, Özdemir S (2017) A deep learning model for
1266 air quality prediction in smart cities. In: Proceedings—2017 IEEE
1267 international conference on big data, big data 2017, 2018-Jan,
1268 1983–1990
- 1269 32. Beheshti S, Khosroshahi K (2017) Study of the ability of various
1270 types of artificial neural networks in predicting the amount of CO,
1271 NO₂, and SO₂ pollutants in the metropolitan area of Tabriz. In:
1272 Fourth international conference on planning and management
- 1273 33. Shams R, World A (2017) Assessing the accuracy of multiple
1274 regression model in forecasting air quality index (AQI) in Tehran.
1275 Int Conf Res Civil Eng Urban Manage Environ. <https://civilica.com/doc/711061/>
- 1276 34. Zangouei H, Asdaleh F (2017) Prediction of PM10 contamination
1277 in Mashhad using MLP artificial neural networks and Markov
1278 chain model. J Appl Res Geogr Sci 17(47):39–59. <https://iran.journals.nlai.ir/handle/123456789/578038>
- 1279 35. Farhadi R, Hadavifar M (2016) Prediction of air pollutant con-
1280 centrations in Tehran based on climatic factors using artificial
1281 neural network. In: National conference on research and tech-
1282 nology findings in natural and agricultural ecosystems
- 1283 36. Li X, Peng L, Hu Y, Shao J, Chi T (2016) Deep learning archi-
1284 tecture for air quality predictions. Environ Sci Pollut Res
1285 23(22):22408–22417. <https://doi.org/10.1007/s11356-016-7812-9>
- 1286 37. Kellman P, Hansen MS (2014) T1-mapping in the heart: accuracy
1287 and precision. J Cardiovasc Magn Reson 16(1):1–20. [https://doi.org/10.
1288 1186/1532-429X-16-2](https://doi.org/10.1186/1532-429X-16-2)
- 1289 38. Azevedo A, Santos MF (2008) DD, SEMMA and CRISP-DM: a
1290 parallel overview. In: MCCSIS’08—IADIS multi conference on
1291 computer science and information systems; proceedings of
1292 informatics 2008 and data mining 2008, pp 182–185
- 1293 39. Castillo Esparcia A, López Gómez S (2021) Public opinion about
1294 climate change in United States, partisan view and media cov-
1295 erage of the 2019 United Nations climate change conference
1296 (COP 25) in Madrid. Sustainability 13(7):3926. [https://doi.org/10.
1297 3390/su13073926](https://doi.org/10.3390/su13073926)
- 1298 40. Organización de las Naciones Unidas. (2018). World urbaniza-
1299 tion prospects 2018. In: Department of economic and social
1300 affairs. World Population Prospects 2018
- 1301 41. Hosseini V, Shahbazi H (2016) Urban air pollution in Iran. Iran
1302 Stud 49(6):1029–1046. [https://doi.org/10.1080/00210862.2016.
1303 1241587](https://doi.org/10.1080/00210862.2016.1241587)
- 1304 42. Nazmfar H, Saredeh A, Eshgi A, Feizizadeh B (2019) Vulne-
1305 rability evaluation of urban buildings to various earthquake inten-
1306 sities: a case study of the municipal zone 9 of Tehran. Hum Ecol
1307 Risk Assess Int J 25(1–2):455–474. [https://doi.org/10.1080/
1308 10807039.2018.1556086](https://doi.org/10.1080/10807039.2018.1556086)
- 1309 43. Vallero D (2014) Fundamentals of air pollution—Daniel Val-
1310 llero—Google Books. Academic Press
- 1311 44. Mohammadpour R, Asaie Z, Shojaeian MR, Sadeghzadeh M
1312 (2018) A hybrid of ANN and CLA to predict rainfall. Arab J
1313 Geosci. <https://doi.org/10.1007/s12517-018-3804-z>
- 1314 45. Brownlee J (2018) How to develop LSTM models for time series
1315 forecasting. Mach Learn Mastery 14:1–77
- 1316 46. Zocca V, Spacagna G, Slater D, Roelants P (2017) Python deep
1317 learning—Google Books. Packt Publishing
- 1318 47. Brownlee J (2016) Deep learning with Python: develop deep
1319 learning models on Theano and TensorFlow using Keras in deep
1320 learning with Python
- 1321 48. Vasilev I, Slater D, Spacagna G, Roelants P, Zocca V (2019)
1322 Python deep learning: exploring deep learning techniques and
1323 neural network—Ivan Vasilev, Daniel Slater, Gianmario Spa-
1324 cagna, Peter Roelants Valentino Zocca—Google Books. Packt
1325 Publishing
- 1326 49. Brownlee J (2020) Deep learning with Python: develop deep
1327 learning models on Theano and Jason Brownlee—Google Sách
- 1328 50. Hossain E, Shariff MAU, Hossain MS, Andersson K (2020) A
1329 novel deep learning approach to predict air quality index. In:
1330 Proceedings of international conference on trends in computa-
1331 tional and cognitive engineering, pp 367–381
- 1332 51. Jamal A, Nodehi RN (2017) Predicting air quality index based on
1333 meteorological data: a comparison of regression analysis, artifi-
1334 cial neural networks and decision tree. J Air Pollut Health 2(1)
- 1335 52. Wu Q, Lin H (2019) A novel optimal-hybrid model for daily air
1336 quality index prediction considering air pollutant factors. Sci
1337 Total Environ 683:808–821. [https://doi.org/10.1016/j.scitotenv.
1338 2019.05.288](https://doi.org/10.1016/j.scitotenv.2019.05.288)
- 1339 53. Battan LJ (1979) Fundamentals of meteorology. Fundam Meteo-
1340 rol. <https://doi.org/10.1007/978-3-030-52655-9>

- 1343 54. Jassim MS, Coskuner G (2017) Assessment of spatial variations
1344 of particulate matter (PM₁₀ and PM_{2.5}) in Bahrain identified by
1345 air quality index (AQI). *Arab J Geosci* 10(1):1–14. <https://doi.org/10.1007/s12517-016-2808-9>
- 1346 55. Hochreiter S, Urgen Schmidhuber J (1997) Long short-term
1347 memory. *Neural Comput* 9(8):17351780
- 1348 56. Kim P (2017) Machine learning. MATLAB Deep Learning
1349 130:1–18. https://doi.org/10.1007/978-1-4842-2845-6_1
- 1350 57. Boloukian B, Safi-Esfahan F (2020) Recognition of words from
1351 brain-generated signals of speech-impaired people: application of
1352 autoencoders as a neural Turing machine controller in deep
1353 neural networks. *Neural Netw* 121:186–207. <https://doi.org/10.1016/j.neunet.2019.07.012>
- 1354 58. Gers FA, Schraudolph NN, Schmidhuber J (2003) Learning
1355 precise timing with LSTM recurrent networks. *J Mach Learn Res*
1356 3(1):115–143. <https://doi.org/10.1162/153244303768966139>
- 1357 59. Mohammadi M, Al-Fuqaha A, Guizani M, Oh JS (2018)
1358 Semisupervised deep reinforcement learning in support of IoT
1359 and Smart City services. *IEEE Internet Things J* 5(2):624–635.
1360 <https://doi.org/10.1109/JIOT.2017.2712560>
- 1361 60. Lipton ZC, Berkowitz J, Elkan C (2015) A critical review of
1362 recurrent neural networks for sequence learning. arXiv Preprint
1363 [arXiv:1506.00019](https://arxiv.org/abs/1506.00019)
- 1364 61. Yao K, Peng B, Zhang Y, Yu D, Zweig G, Shi Y (2014) Spoken
1365 language understanding using long short-term memory neural
1366 networks. In: 2014 IEEE Spoken Language Technology Workshop
1367 (SLT), pp. 189–194
- 1368 62. Gulcehre C, Chandar S, Cho K, Bengio Y (2018) Dynamic neural
1369 Turing machine with continuous and discrete addressing
1370 schemes. *Neural Comput* 30(4):857–884. https://doi.org/10.1162/NECO_a_01060
- 1371 63. Graves A, Mohamed A, Hinton G (2013) Speech recognition with
1372 deep recurrent neural networks. In: 2013 IEEE international
1373 conference on acoustics, speech and signal processing, pp 6645–6649
- 1374 64. Ji S, Xu W, Yang M, Yu K (2013) 3D Convolutional neural
1375 networks for human action recognition. *IEEE Trans Pattern Anal
1376 Mach Intell* 35(1):221–231. <https://doi.org/10.1109/TPAMI.2012.59>
- 1377 65. Liu Y, Zheng H, Feng X, Chen Z (2017) Short-term traffic flow
1378 prediction with Conv-LSTM. In: 2017 9th international conference
1379 on wireless communications and signal processing, WCSP
1380 2017—proceedings, 2017-Jan, 1–6
- 1381 66. Faradonbe SM, Safi-Esfahani F (2020) A classifier task based on
1382 neural Turing machine and particle swarm algorithm. *Neurocomuting*
1383 396:133–152. <https://doi.org/10.1016/j.neucom.2018.07.097>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.