Deep Ensemble Learning Method to Forecast COVID-19 Outbreak

Nesrine Ben Yahia (nesrine.benyahia@ensi-uma.tn)
ENSI, Manouba University https://orcid.org/0000-0003-4788-4475
Mohamed Dhiaeddine Kandara
ENSI, University of Manouba
Narjes Bellamine Ben Saoud
ENSI, University of Manouba

Research Article

Keywords: Deep ensemble learning, COVID-19 outbreak, prediciton, DNN, LSTM, CNN, Stacking

DOI: https://doi.org/10.21203/rs.3.rs-27216/v1

Abstract

Due to the continuous spread of the novel coronavirus (COVID-19) worldwide, it is urgent to develop accurate decision-aided methods to support healthcare policymakers to control and early detect COVID-19 outbreak especially in the data science era. In this context, our main goal is to build a generic and accurate method that can predict daily conrmed cases which helps stake-holders to make and review their epidemic response plans. This method takes advantage of the complementarity of DNN (Deep Neuronal Networks), LSTM (Long Short Term Memory) and CNN (Convolutional Neuronal Networks) where their forecasted values represent the inputs of stacked ensemble meta-learners that will generate the nal outbreak predictions. To the best of our knowledge, this is the rst time that deep ensemble learning is used to deal with this issue. The proposed method is validated on three experimental scenarios, Tunisia case study, China case study and the third one is based on China data and models to predict Tunisia COVID-19 outbreak. Experiment results indicate that, compared with individual learners, the stacked-DNN meta-learner, whose input are forecasted values of DNN, LSTM and CNN, achieved the best accurate results in terms of accuracy as well as RMSE for the three scenarios. In conclusion, our ndings demonstrate that i) deep ensemble learning may be used as an accurate decision support tool for improving COVID-19 outbreak forecasting, ii) it is possible to reuse China learners and meat-learners to make prediction of the epidemic trend for other countries when preventive and control measures are comparable.

1. Introduction

COVID-19 [25], the official name of the 2019 novel coronavirus announced by the World Health Organization (WHO), has emerged recently as a severe acute respiratory syndrome with the current reference name SARS-CoV-2. Its outbreak was originally reported in Wuhan, China, but it has subsequently been spread rapidly across the world. Although, the disease has been almost contained and well controlled in China recently, it is persistently threatening public health worldwide, causing serious concern. Thus, it is urgent to develop accurate computer-aided method to assist healthcare policymakers to control COVID-19 for a successful public health response to the outbreak of this new infection. In this context, artificial intelligence (AI) has contributed in the battle against coronavirus disease (COVID-19) pandemic. In fact, various AI methods and applications have been proposed to fight against COVID-19 from medical data analytics, image processing, text mining, natural language pro- cessing and Internet of Things, to computational biology and medicine [16]. In this study, we focus on healthcare analytics, data analytics AI solutions, to deal with forecasting of COVID-19 outbreak. In fact, the growing healthcare industry is generating a large volume of useful medical, clinical and adminis- trative data attracting the attention of academicians and practitioners alike. Thus, in the data science era, healthcare analytics are introduced to provide tools and techniques to extract information from this complex and big data and to assist decision-making in healthcare [13]. More specifically, the role of data analytics in healthcare domain has grown rapidly in the last decade which has prompted increasing interests in the generation of data driven and analyti- cal models based on machine learning in health informatics [19]. In this paper, we mainly focus on key applications of deep learning advances, a sub-field of machine learning, in order to predict daily confirmed cases of COVID-19 and to early detect its epidemic outbreak. Indeed, with the daily increase in the number of newly diagnosed, suspected and confirmed cases, practitioners and clinicians need to forecast this number in order to make the necessary measures in terms of quarantine protocols, treatment measures and hospitals preparedness.

The goal of this paper is to propose a generic,data-independent, COVID-19 outbreak predictive method based on deep learning techniques that may be used as a healthcare decision support tool for different countries whose pre- ventive and control measures are comparable. Our motivation is to exploit the confirmed advantages of deep learning models both in infection disease outbreak prediction and health decision aided processes [15]. And, our main contribution is to use firstly three deep learning models, DNN (Deep Neu- ronal network), LSTM (Long Short-Term Memory) and CNN (Convolutional Neuronal Networks). Then, these models will be stacked in ensemble learning models in order to generate the most accurate results. The meta-learners use as input the forecasted values of these three learners in order to generate the final outbreak predictions. To the best of our knowledge, this is the first time that deep ensemble learning is used to deal with this issue.

The paper proceeds as follows: In Section 2, we give an overview of related works on the healthcare analytics and deep learning techniques used in the COVID-19 outbreaks prediction as well as healthcare decision-making sup- port tools. In section 3, we will briefly introduce the background knowledge of CNN, DNN and LSTM that we will use in this study. Then in Section 4, we detail our method and its main contributions. Section 5 is dedicated to present the experimental results and section 6 is devoted to discussing our method and to present its threats of validity. Finally, we conclude and present our future works in section 7.

2. Related Works

In this section, we aim to review some of the main recent literature on applying machine and deep learning advances to perform the prediction of COVID-19 outbreaks, especially those whose goal is to predict daily positive or confirmed cases. For instance, [14] proposed a machine learning approach for predicting the daily numbers of cumulative confirmed cases (CCCs), new cases (NCs), and death cases (DCs) of COVID-19 in China based on the data provided by the National Health Committee of China from Jan 20, 2020, to Mar 1, 2020. They used Eureqa, a machine learning algorithm that can automatically ob- tain a formula which perfectly shapes the relationship between daily numbers of CCCs or DCs and their corresponding days and build predictive models from these data.

In [11], authors investigated the performance of a modified stacked auto- encoder for modeling time series and real-time forecasting the confirmed cases of Covid-19 across China as an alternative to epidemiological models for its transmission dynamics. This model is applied on the dataset collected from January, 11 to February 27, 2020 given by the WHO and is based on the use of latent variables and clustering algorithms that help in investigating the trans- mission procedure by grouping the provinces/cities. In [1], Recurrent Neural Networks (RNN), that can model sequential (temporal) data prediction, are used for predicting positive (confirmed), negative, released and death cases of COVID-19. They proposed three models, a Long short-term memory (LSTM), a Gated Recurrent Unit (GRU) and a combined LSTM-GRU model. Exper- imental results on COVID-19 infection cases dataset in South Korea from 20th January 2020 to 12th March 2020 show that a high rate of accuracy is obtained by the combined model. In [12], a Convolutional Neural Network (CNN) method was proposed to analyze and predict the number of COVID- 19 confirmed cases in China using data on confirmed cases from January 23, 2020 to March 2, 2020 obtained from Surging News Network and WHO. Ex- periment results indicated that compared with MLP (Multilayer Perceptron), LSTM and GRU, the proposed CNN model is the best performing algorithm and its characteristic extraction is very helpful for predicting the number of confirmed cases of COVID-19. These deep learning solutions have proven their performance and accuracy in predicting COVID-19 confirmed cases number. However, these works are interested to predict this number for a particular country and are based on the data of this country.

Thus, the salient contribution of this study is to propose a generic and ac- curate predictive method that can be used to predict COVID-19 confirmed cases. By generic, we refer to the possible reuse of this method for different COVID-19 confirmed cases dataset of different countries whose preventive and control measures are similar (such as quarantine policies). In this method, we will test the ability of CNN, LSTM and DNN to overcome the issue of fore- casting COVID-19 outbreak. In fact, the selection of appropriate predictive models is a challenging task. Our motivation, when choosing these models, is to exploit the confirmed advantages of DNN and LSTM to predict infectious diseases spreading [4] and also the confirmed feasibility and practicality of CNN to deal with this issue [12]. Then, we will reflect on the use of ensemble learning to combine them in deep meta-learners that fuse the forecasted values of CNN, LSTM and DNN learners in order to give best accurate results and to improve the forecasting accuracy. In the next section, we will introduce the background knowledge of these three models.

3. Background

Artificial Neural Networks (ANN) are computational and machine learning modeling tools, inspired by information processing in the human brain, where the attractiveness comes from their ability in solving many complex real-world problems and facing nonlinearity, high parallelism, fault and noise tolerance, and generalization [3]. In the last years, with the spread of deep learning, Deep Neural Networks (DNN) have emerged as deep architectures composed of multiple levels of non-linear operations and many hidden layers which have the capacity to learn more complex models than shallow ones [10].

Recently, further improvements over DNN have been obtained with alternative types of deep neural network architectures, including Long Short Term Mem- ory Recurrent Neural Networks (LSTM) and Convolutional Neural Networks (CNN). These models, that we will use in this study, will be detailed in the following subsections.

3.1 Deep Neural Networks (DNN)

DNN, [10], are Artificial Neural Networks with multiple (at least two) hidden layers between the input and output layers where the ”deep” refers to the number of layers through which the data is transformed. In traditional DNN, each layer is fully connected and is composed of a set of neurons and an acti- vation function. Each neuron has a set of weights where each one is multiplied by one input into the neuron, these are then summed to form the output from the neuron after it has been fed through the activation function. Deep neural networks often require big data and huge numbers of training data to be able to give performing results [9].

3.2 Long Short Term Memory Recurrent Neural Networks (LSTM)

LSTM are an improvement from the recurrent neural networks (RNN) that have the ability to model sequential data and times series prediction [17]. In fact, for LSTM, a cell state is added to store long-term states and to develop a stable RNN for time series forecasting by capturing the long-term dependencies existing in the time series. The deep network architecture of the LSTM cells can provide a powerful model in temporal data processing. Recently, LSTM have attracted much interest in temporal data prediction of infection disease prediction such as in [22] where authors proposed a LSTM method to capture the temporal dynamics of seasonal flu and for real-time influenza forecasting.

3.3 Convolutional Neural Networks (CNN)

CNN, [8], are particular DNN based on the concept of weight sharing so that weights number does not have to be as large as that for a fully connected structure. CNN contain generally four levels in structure: an input layer, con- volutional layers, pooling layers, and fully connected layer (output). The con- volutional layer is the most important part of a CNN, in which the input is convoluted with several filters and each filter represents a smaller matrix, and corresponding feature maps can be obtained after the convolution operation. The pooling layer gives a summary statistic of the nearby outputs such as max- pooling and average-pooling, the most popular pooling layers, which outputs are respectively the maximum of a rectangular neighborhood and the aver- age of the rectangular neighborhood. The convolutional and pooling layers are generally used to extract features, and then one or more fully connected layers are usually adopted after one or more groups of convolutional and pooling layers. The fully connected layer can put the information from feature maps together, and then output them to latter layers.

To conclude, DNN are appropriate for mapping features to a more separable space, LSTM are good at temporal and times series modeling and CNN good at reducing frequency variations, so they are complementary in their modeling capabilities [21]. Their combination has been tested for speech recognition in [5] and [21] where experimental results demonstrate a significant increase in accuracy when applying their ensemble learning.

Thus, how can deep ensemble learning be applied to improve the COVID-19 outbreak prediction accuracy is the focus of this paper. In fact, we aim to take advantage of the complementarity of DNN, LSTM and CNN to forecast COVID-19 epidemic outbreak across the world by combining them into one unified method that will be presented in the next section.

4. Methodology

In this study, we aim to propose a generic data-driven method that may be used as a decision support tool for COVID-19 epidemic trend forecasting across the world. In this method, we adopt a stacking strategy to achieve better accuracy by fusing the forecasted values. In fact, we firstly build three learners (DNN, LSTM and CNN) that will individually forecast COVID-19 epidemic trend. Next, new combined and stacked meta-learners will be designed and trained using the outputs of the three learners in order to find the best stacked-model that improves the forecasting accuracy further. The pipeline of the method proposed in this study is shown in figure 1. Thus, this method consists of the following steps: first grid searching process, data preparation, individual training of the three learners CNN, DNN, and LSTM, second grid searching process, and finally the stacking process for the training of meta-learners. The details are show as following.

4.1 Grid searching process 1

Successful DNN applications usually rely on the appropriate choice of several modelling hyperparameters. These modelling hyperparameters can be related to the network configuration (number of hidden layers and of units or neurons in each layer, etc.) or related to data preparation such as the number of delays or the modelling window size to be used in time series applications [2].

In order to ensure that we choose the best and suitable hyperparameters mod- elling (the optimal ones), we use the grid search technique to establish op- timization decisions based on solid statistical criteria and many combination [18]. According to [2], grid search is the simplest (but also costliest) way to find optimal model parameters by generating an exhaustive search of all possi- ble combinations where it is helpful to select just a moderately small number of hyperparameters that are randomly selected and then will be tested and combined with all their possible values. So, grid search will be used to auto- mate the process of evaluating the selected learners on different combinations of modelling hyperparameters. In our case, for data preparation, we will grid search the window size hyperparameter and for models configuration, we will grid search the number of hidden layers and of units (neurons) in each layer, the optimizer whose role is to update the weight parameters to minimize the loss function, and the dropout layer value which will be putted before the output to selectively choose neurons to ignore during training and to prevent over-fitting.

Regarding data preparation, it is helpful to indicate here that the input of our method is times series dataset of COVID-19 confirmed cases presented as a sequence of vectors, x(t), where t represents elapsed time (i.e. days) and x represent COVID-19 confirmed cases number which varies continuously with

So, we need here to find the appropriately sized input window which means the number of data points which should be used in the input representation as the window size has an important impact on the quality of a neural network. In this context, sliding window method over the input sequence is commonly used in time series based neural networks forecasters [6]. In this method, a set of N-tuples is considered as inputs and a single output as the target value of the network so the N-tuple input slides over the full training set. Following the sliding window method, the general idea is to use N previous time step or N previous series values as input variables and the next step as the target value. We will vary the N value from 2 to 6 and grid search will be used in order to obtain the best N or lag to be used to reframe the data.

Concerning the models hyperparameters, the selected values to be tested, that are commonly used in the literature, are 3,4,5 and 6 for hidden layers number, 16, 32, 64, 128, 256 and 512 neurons for each layer. Concerning the optimizer, we will try Stochastic Gradient Descent (SGD), Root Mean Square Propaga- tion (RMSprop) and Adam optimizer. Finally, for dropout layer, the values of 0.1, 0.2, 0.3, 0.4 and 0.5 will be tested for this layer.

Then, by varying these hyperparameters values, grid search is applied for the three learners where for each one it recalculates the model for each possible combination of tested hyperparameters and selects the set of hyperparame- ters, for which the score is the best. In our case, we used the coefficient of determination R2 score that measures how good models might be constructed from these values combinations. By tuning hyperparameters according to grid search, the best window size will be used in the data preparation and the best configuration hyperparameters will be used to build the different learners that will be then trained.

4.2 Data preparation

In this step, the dataset 0 will be reframed to generate the new dataset 1 using the best window size recommended by the first grid searching process.

4.3 Training of CNN, DNN and LSTM learners

During this training step, our learners parameters i.e. weights of neural net- works will be learned. More specifically, we separately build the three chosen learners (DNN, LSTM and CNN) whose input are dataset 1 and their outputs will be collected in another dataset 2. Architectures configuration of these learners are based on the optimal hyperparameters selected by the first grid searching process.

4.4 Grid searching process 2

In order to give more insight into the quality of the meta-learners that will be trained in the next step, the goal of this step is to use again the grid search approach to tune the meta-learners hyperparameters and to find the best ones which go well with the new dataset 2.

4.5 The stacking process

In this phase, we aim to take advantage of the complementarity of DNN, LSTM and CNN by gathering the knowledge of each learner and combining them into one unified method using stacking technique as an ensemble learning method. Stacking is composed of two phases [7] : in the first phase, usually different models, called learners, are learned based on a dataset. Then, outputs of these learners are collected to create a new dataset englobing also for each row the real expected value. In the second phase, that new dataset is used with a new learning model, the so-called meta-learner in order to provide the final output. In this stacking method, results of a set of different learners at the level 0 are combined by a meta-learner at the level 1 in order to achieve better accuracy by fusing the learners forecasted values. Therefore, the stacking strategy is adopted here aiming to improve the forecasting accuracy further.

So, in our cases, forecasted values of DNN, LSTM an CNN will be collected in a new dataset 2 to be used for the training of the meta-learners. Then, in order to choose the best stacked-model that gives the best accurate prediction, we will test here three metalearners (a stacked-DNN, a stacked-LSTM and a stacked-CNN). The architectures of these meta-learners are depicted in figure 3 using the hyperparameters selected by the second grid searching process.

5. Experimental Results

In order to validate the proposed predictive method of COVID-19 forecast- ing outbreak, we have chosen two cases studies: Tunisia and China. In fact, we chose China, the first country where COVID-19 outbreaks and for which a huge amount of open data is available, to apply our method. We also intend to apply to Tunisia, our country, where containment measures are applied early similarly to China, and where we are following day to day the dynamics of the propagation.

Furthermore, in order to evaluate the prediction rates of the proposed leaners and meta-learners, we used the root mean squared error (RMSE) and the ac- curacy. RMSE is a common measurement for the difference between predicted and real values. It is usually used in the other fields as well as in the prediction of infectious diseases spreading [4] and the accuracy identifies the overall effectiveness of the models.

5.1 Tunisia case study

5.1.1 *Materials*

In this first experimental scenario, the ongoing COVID-19 outbreak is studied using the data provided by the official sources of the Tunisian National Obser- vatory1 of New and Emerging Diseases that represent official data published by the ministery of the public health. We also consider the verified sources of universal John Hopkins University [24] that includes the confirmed cases (CC) as well as the death cases (DC) and the recovered cases (RC) daily numbers across the world starting from January 22, 2020 until April 27,2020. From these data, including time series datasets and global situation reports, we extracted only the data of Tunisia.

5.1.2 *Method validation for Tunisia*

By tuning hyperparameters according to grid search process, the best ones for Tunisia are as follows. The best window size or lag value is 3 for the three learners. Thus, data will be prepared and reframed using three times inputs for each row. For instance, the target value Y for the Day 4 will be expressed 1 http://onmne.tn/fr/index.php by the three previous COVID-19 confirmed cases numbers for Day1, Day 2 and Day 3. So, in the novel dataset 1, which represents the input for our predictive leaners, we have four columns named and x(t-3), x(t-2), x(t-1) and x(t).

For networks configuration, hyperparameters that are recommended by grid search will be used to configure the architecture of each learner. Then, dur- ing the training step, models parameters i.e. weights of neural networks will be learned. More specifically, we separately build the three chosen learners (DNN, LSTM and CNN) whose input are dataset 1 and their outputs will be collected in another dataset 2. Architectures configuration of these learners are presented in figure 2 using the optimal hyperparameters selected by grid search.

Let’s start with the DNN, this learner contains three hidden layers with re- spectively 256, 64 and 16 neurons where the “relu” activation function is used for each one. The input layer contains three neurons and the output contains a one neuron with the “linear” activation function. About the optimizer, the Adam function is selected by grid search in addition dropout value is 0.2 and we use mean squared error as loss function.

Concerning the LSTM learner, it is equally based on three hidden layers with 256 units or neurons and “relu” activation function for each layer. The input layer contains equally 3 neurons and the output with only one neuron with the “linear” activation function preceded by a Dropout layer with 0.3 as value. Noting that the same optimizer and loss function of the DNN was used.

Next, regarding the CNN learner, we find an input layer with three neurons. After that, we find a convolutional layer with 128 units, a max-pooling layer with 128 units, a convolutional layer with 32 units, a convolutional layer with 64 units and an average-pooling layer with 64 units, a dropout layer with 64 units and 0.3 as value and finally a fully connected output layer with one unit. Noting that all layers used “relu” as activation function expect the output with “linear” function.

Finally, during the stacking process and in order to identify the best meta- learner to adopt, we will test the three meta-learners the stacked-DNN, stacked- LSTM and stacked-CNN using the forecasted values of the individual learn- ers DNN, LSTM and CNN of Tunisi cae study. Architectures of these meta- learners are depicted in figure 3 using the hyperparameters selected by grid search.

5.1. 3 *Results for Tunisia case study*

In table 1, we summarize the results of learners and meta-learners’ perfor- mances using Tunisia data. Accuracy and RMSE will be given separately in the first step for the models then for the stacked models. Then, in figure 5, we present the forecasted values of learners and meta-learners vs real values for the studied period.

Table 1 Performances of learners and meta-learners for Tunisia case study

Models	Accruacy	RMSE
DNN	0,9985531	10,3384
LSTM	0,626390087	80,9212
CNN	0,9995	7,8214
Stacked-DNN	**0,9999**	**2,396**
Stacked-LSTM	0,923103966	75,3678
Stacked-CNN	0,998492464	10,5528

5.2 China case study

In this second experimental scenario, the ongoing COVID-19 outbreak in China is studied using only the times series dataset for 33 provinces of China provided by the verified sources of John Hopkins University [24]. Thus, for each province, we apply our predictive method to build its three learners and its three meta-learners. By the end, we will have 33 LSTM, 33 DNN and 33 CNN, 33 stacked-LSTM, 33 stacked-DNN and 33 stacked-DNN. These learners and meta-learners' performances on the test set are reported below with optimal hyperparameters selected by grid search.

5.3 Augmented prediction: from China to Tunisia

In this third experimental scenario, we aim to achieve an augmented prediction i.e. we will build stacked models derived from the models of China case study for predicting the epidemic trend in Tunisia and then across the world. In fact, our goal here is to reuse the models already built and trained on china series times dataset to predict COVID-19 outbreak in Tunisia. Our motivation is to increase the size of training data as DNN often requires huge numbers of training data [9]. So in this augmented prediction, China data are used as training data and Tunisia data are considers as Test data. The process of this augmented prediction is depicted in figure 7. So, the 33 DNN, the 33 LSTM and the 33 CNN will be respectively stacked in a DNN, LSTM and CNN that will be equally stacked in one meta-learner (a DNN, a LSTM or a CNN).

We summarize in table 2 the performances of the stacked models (results of the stacking of 33 learners for each type) and the double stacked meta-learners (results of the stacking of the stacked models). For these meta-learners, we also tested the DNN, LSTM and CNN.

6. Discussion

In this study, we were interested in studying whether deep ensemble learning could produce valid and accurate predictions of COVID-19 confirmed cases number. To this end, we proposed a generic data-driven method that takes advantage of the complementarity of three deep learners DNN, LSTM and CNN to forecast COVID-19 epidemic outbreak across the world by combining them into stacked deep meta-learners. The proposed method is validated on different scenarios and the results show that the proposed stacked model can indeed improve the forecasting accuracy. In fact, experiment results indicated that compared with individual predictions of these three learners, the stacking method, which is an ensemble learning strategy, achieved better accuracy by fusing the forecasted values of DNN, LSTM and CNN. More specifically, the results shown in table 1 demonstrated that the stacked-DNN, whose input are forecasted values of DNN, LSTM and CNN, perform better than the stacked LSTM and the stacked CNN and it has the greatest prediction efficacy with an accuracy of 0.999 and a RMSE of 2,396 for the Tunisia case study. Thus, our study demonstrated the practicality and feasibility of deep ensemble learning models to assist healthcare decision process. By achieving a high performance on the prediction, the proposed method may enable an accurate prediction of COVID-19 outbreaks. Our findings indicated also that it is possible to reuse China data, learners and trained models to make prediction of the epidemic trend in Tunisia and then across the world where preventive and control mea- sures are similar to those adopted in China. In fact, the results shown in table 2 demonstrate that the stacked-DNN meta-learner, trained on China data and tested on Tunisia data, resulted 0,9997 for the accuracy and 2,4736 for the RMSE.

In conclusion, we successfully constructed a generic COVID-19 outbreak pre- dictive method that may be used as a decision support tool for improving its surveillance, controlling and managing epidemics. However, a limitation of this study is that we did not take into account during training other factors which are associated with the spread and outbreak of COVID-19 and that can provide more meaningful analyses and hopefully more reasonable predictions. These factors include for instance, politics, culture, education, minimizing out- door activities, health facilities, enforcement of wearing masks, geographical position, etc.

At this stage, we assume that there might be some validity threats of our research findings, and we have self-assessed them here in order to denote the trustworthiness of our results, to what extent they are true and not biased by our subjective point of view. In addition, these potential threats are addressed according to the classification proposed in [20]. Regarding the construct va- lidity, we assume that the provided measures could be biased regarding the researchers' expected results. However, we have used in this research, to vali- date and evaluate the performance of the adopted learners and metra-learners RMSE which is usually used in the prediction of infectious diseases spreading [4] and the accuracy which is considered among the standard metrics that reduce biases. Regarding the external validity, there might be some issues regarding generalization of our predictive method. To overcome this issue, this method is validated on two cases studies (Tunisia and China) which will provide more consistent feedback about the relevance of our results. Finally, regarding reliability, there might be a potential threat that concerns the depen- dency of data and analysis on the specific researchers. However, we are doing an effort towards trying to minimize this threat by proposing the augmented prediction experimental scenario where China data are used for training and Tunisia data for test to predict COVID-19 outbreak in Tunisia.

Table 2 Performances of learners and meta-learners for the augmented prediction

Models	Accruacy	RMSE
Stacked-DNN	0,99958	6,220065827
Stacked-LSTM	0.30192	67.75875595777089
Stacked-CNN	0,96435	20,25682541
Double Stacked-DNN	**0,9997**	**2,4736**
Double Stacked-LSTM	0,63371	146,7686548
Double Stacked-CNN	0,98766	12,16694009

7. Conclusion And Learnt Lessons

In this study, we performed DNN, LSTM and CNN learners and stacked meta-learners with two cases studies (Tunisia and China) to forecast the daily COVID-19 confirmed cases number. We also performed learners from China case study and reused them for Tunisia case study. Our findings demonstrated that the stacked deep ensemble learning models perform better than individ- ual deep learning models and they contributed to improving the predictive accuracy. More specifically, we found that the stacked meta-learner DNN re- sulted in the best RMSE as well as accuracy. Our findings indicated also that it is possible to reuse china learners and trained models to make prediction of the epidemic trend in Tunisia and then across the world when preventive and control measures are comparable. To the best of our knowledge, this is the first time stacking and ensemble deep learning has been used to predict COVID-19 outbreaks. Our learnt lessons confirmed that deep ensemble learn- ing may be used as an accurate decision support tool for improving COVID-19 surveillance, controlling infection, and managing epidemic forecasts.

Declarations

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Bandyopadhyay, K., Dutta, S.: Machine Learning Approach for Confirmation of COVID-19 Cases: Positive, Negative, Death and Release. medRxiv. 2020.03.25.20043505 (2020)
2. Barbero Jim´enez, A´., L´opez L´azaro, , Dorronsoro, J.R.: Finding optimal model pa- rameters by deterministic and annealed focused grid search. Neurocomputing. 72 (13), 2824–2832 (2009)
3. Basheer, I.A., Hajmeer, M.: Artificial neural networks: fundamentals, computing, design, and Journal of Microbiological Methods. 43 (1), 3–31 (2000)
4. Chae, , Kwon, S., Lee, D.: Predicting Infectious Disease Using Deep Learning and Big Data. International Journal of Environmental Research and Public Health. 15 (8), 1596 (2018)
5. Deng, , Platt, J.C.: Ensemble Deep Learning for Speech Recognition. 5
6. Frank, J., Davey, N., Hunt, S.P.: Time Series Prediction and Neural Networks. 13
7. Graczyk, , Lasota, T., Trawinski, B., Trawinski, K.: Comparison of Bagging, Boosting and Stacking Ensembles Applied to Real Estate Appraisal. In: Nguyen, N.T., Le, M.T., and Swiatek, J. (eds.) Intelligent Information and Database Systems. pp. 340–350. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
8. Gu, J., Wang, , Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, L., Wang, G., Cai, J., Chen, T.: Recent Advances in Convolutional Neural Networks. arXiv:1512.07108 [cs]. (2017)
9. Hamzah, A.B., Lau, C.H., Nazri, H., Ligot, D.V., Lee, G., Tan, C.L., Shaib, M.K.B.M., Zaidon, U.H.B., Abdullah, A.B., Chung, M.H., Ong, C.H., Chew, P.Y.: CoronaTracker: World-wide COVID-19 Outbreak Data Analysis and Prediction. nCoV (2020)
10. Hinton, E.: Reducing the Dimensionality of Data with Neural Networks. Science. 313 (5786), 504–507 (2006)
11. Hu, Z., Ge, Q., Li, S., Jin, L., Xiong, M.: Artificial Intelligence Forecasting of Covid-19 in China. arXiv:2002.07112 [q-bio]. (2020)
12. Huang, C.-J., Chen, Y.-H., Ma, Y., Kuo, -H.: Multiple-Input Deep Convolutional Neural Network Model for COVID-19 Forecasting in China. medRxiv. 2020.03.23.20041608 (2020)

13. Islam, M., Hasan, M., Wang, , Germack, H., Noor-E-Alam, M.: A Systematic Re- view on Healthcare Analytics: Application and Theoretical Perspective of Data Mining. Healthcare. 6 (2), 54 (2018)
14. Li, M., Zhang, Z., Jiang, S., Liu, Q., Chen, C., Zhang, Y., Wang, : Predicting the epidemic trend of COVID-19 in China and across the world using the machine learning approach. medRxiv. 2020.03.18.20038117 (2020)
15. Miotto, R., Wang, , Wang, S., Jiang, X., Dudley, J.T.: Deep learning for healthcare: review, opportunities and challenges. Briefings in Bioinformatics. 19 (6), 1236–1246 (2018)
16. Nguyen, T. T: Artificial intelligence in the battle against coronavirus (COVID-19): a survey and future research directions. (2020)
17. Pascanu, R., Gulcehre, C., Cho, K., Bengio, Y.: How to Construct Deep Recurrent Neural Networks. arXiv:1312.6026 [cs, stat]. (2014)
18. Pontes, F.J., Amorim, G.F., Balestrassi, P., Paiva, A.P., Ferreira, J.R.: Design of experiments and focused grid search for neural network parameter optimization. Neuro- computing. 186 22–34 (2016)
19. Ravi, D., Wong, , Deligianni, F., Berthelot, M., Andreu-Perez, J., Lo, B., Yang, G.-Z.: Deep Learning for Health Informatics. IEEE Journal of Biomedical and Health Informatics. 21 (1), 4–21 (2017)
20. Runeson, , Host, M.: Guidelines for conducting and reporting case study research in software engineering. Empirical Software Engineering. 14 (2), 131–164 (2009)
21. Sainath, T.N., Vinyals, O., Senior, A., Sak, H.: Convolutional, Long Short-Term Mem- ory, fully connected Deep Neural Networks. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 4580–4584. IEEE, South Brisbane, Queensland, Australia (2015)
22. Venna, R., Tavanaei, A., Gottumukkala, R.N., Raghavan, V.V., Maida, A.S., Nichols, S.: A Novel Data-Driven Model for Real-Time Influenza Forecasting. IEEE Access. 7 7691–7701 (2019)
23. Yang, -T., Chen, Y.-A., Chan, Y.-W., Lee, C.-L., Tsan, Y.-T., Chan, W.-C., Liu, P.- Y.: Influenza-like illness prediction using a long short-term memory deep learning model with multiple open data sources. The Journal of Supercomputing. (2020)
24. GitHub - CSSEGISandData/COVID-19: Novel Coronavirus (COVID-19) Cases, pro- vided by JHU CSSE, https://github.com/CSSEGISandData/COVID-19, Accessed: April 13, 2020
25. WHO — Novel Coronavirus – China, WHO, http://www.who.int/csr/don/12-january- 2020-novel-coronavirus-china/en/, Accessed: April 13, 2020

Figures

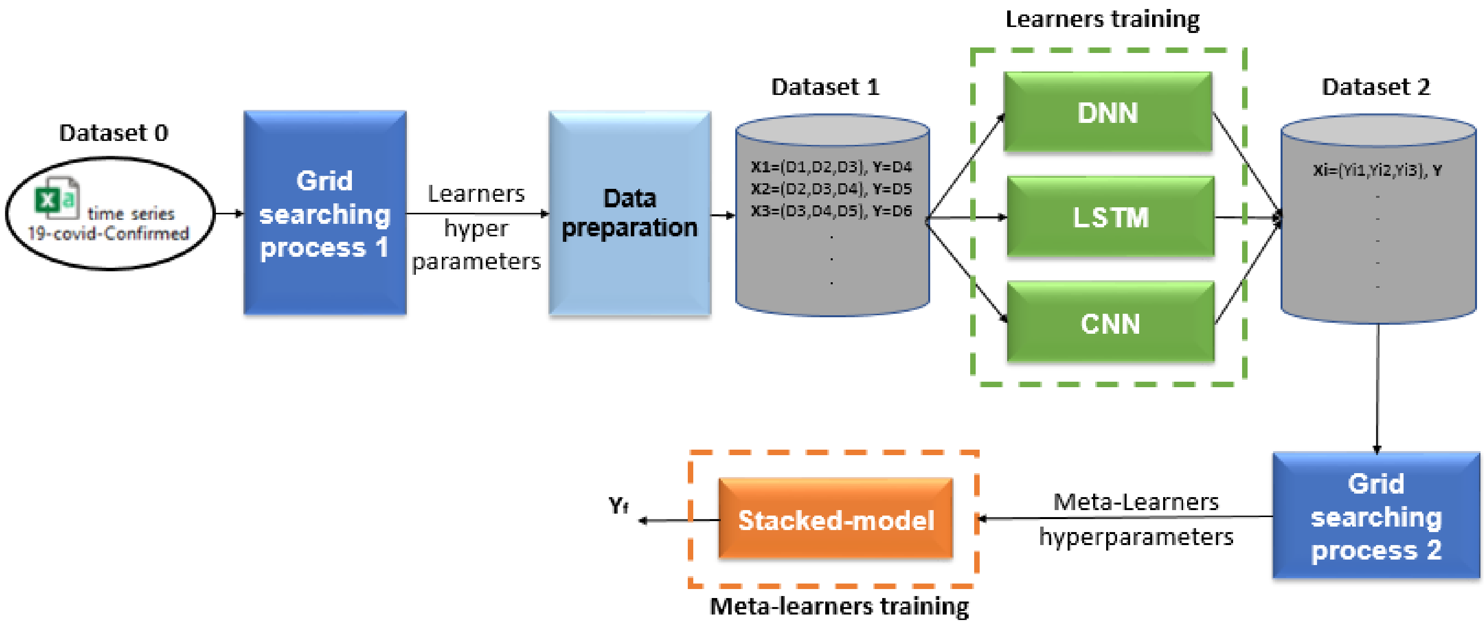


Figure 1

Architecture of the proposed method

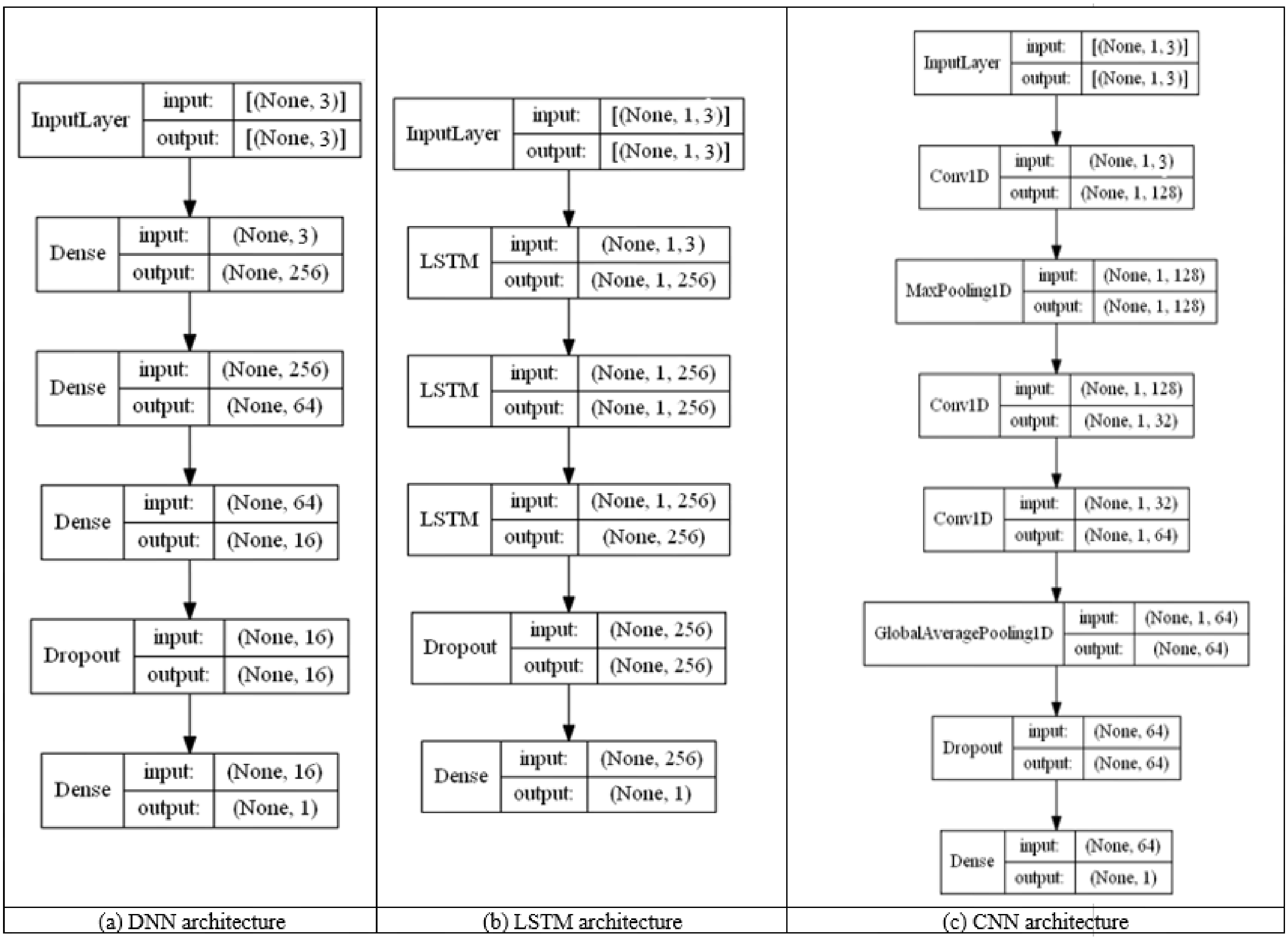
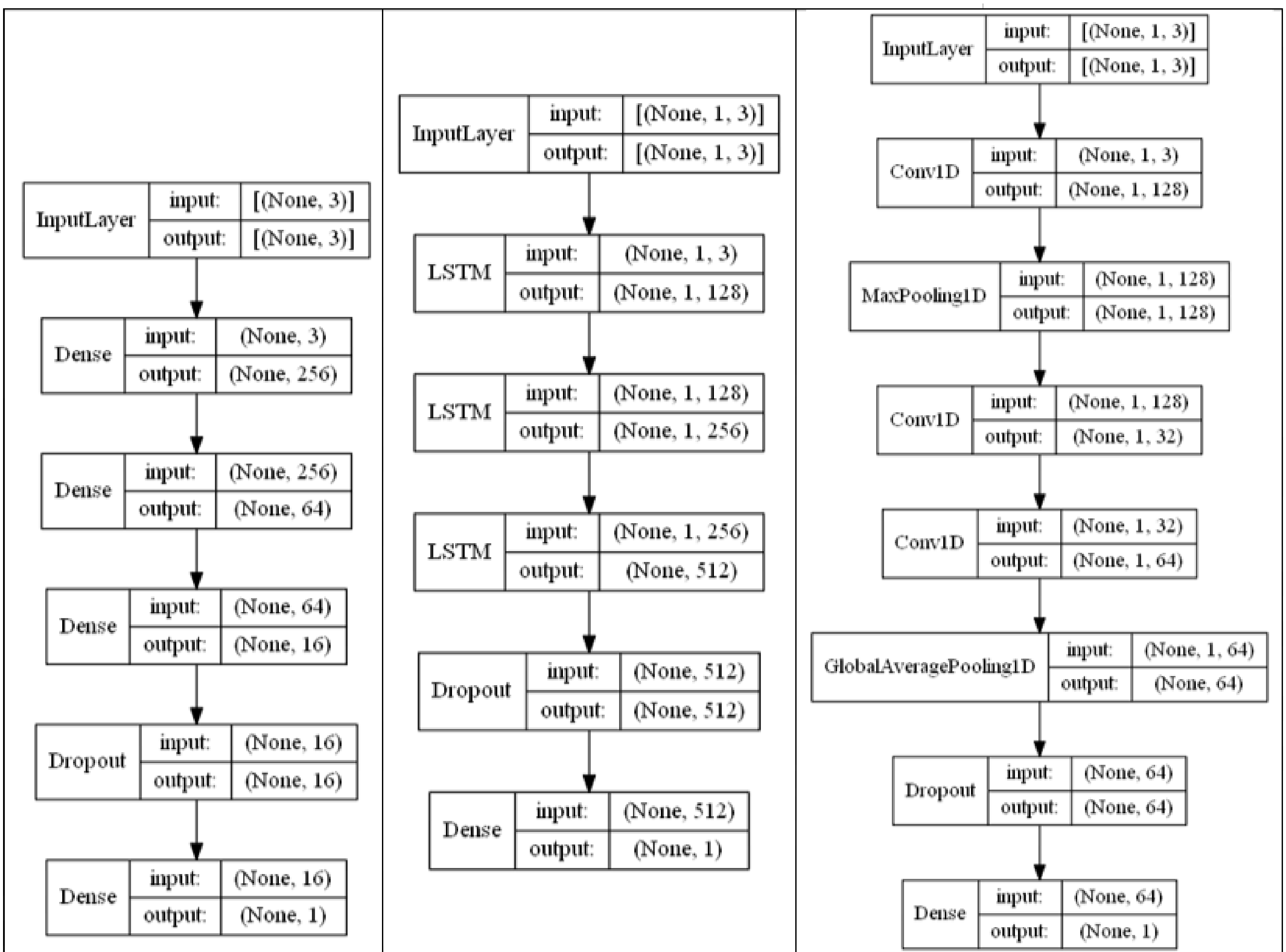


Figure 2

Learners architectures for Tunisia case study



(a) Stacked-DNN architecture | (b) Stacked-LSTM architecture | (c) Stacked-CNN architecture

Figure 3

Meta-learners architectures for Tunisia case study

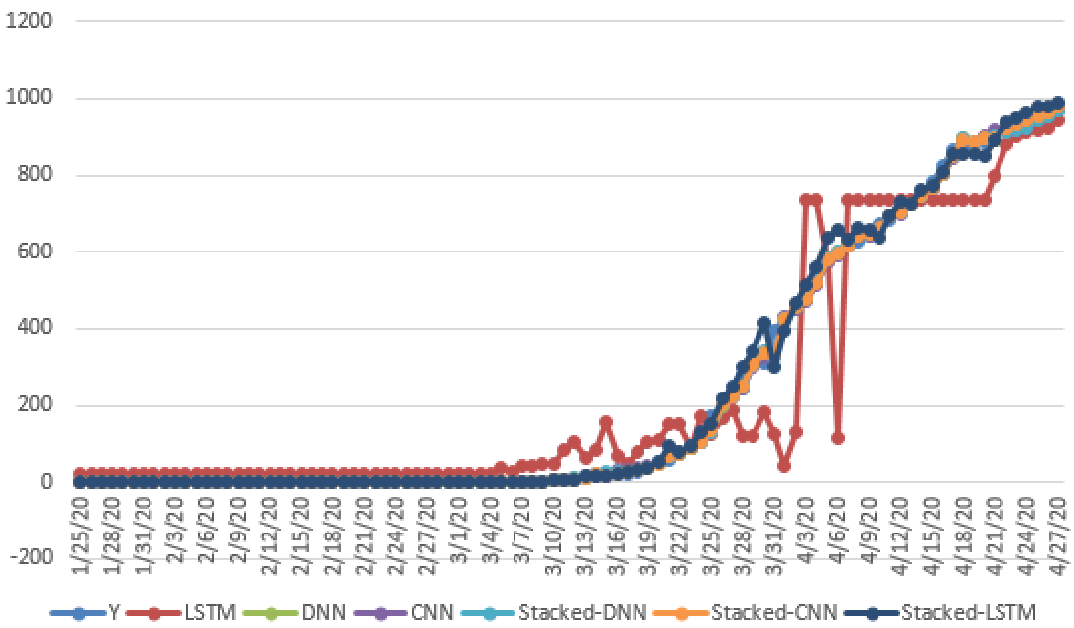


Figure 4

Forecasted values of learners and meta-learners vs real values

City model	LSTM		CNN		DNN	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
Anhui	0,9184	72,1931	0,9989	7,5059	0,9952	23,599
Beijing	0,5064	119,4997	0,9976	8,4122	0,9892	17,6659
Chongqing	0,1059	160,017	0,992	15,1354	0,9791	24,4514
Fujian	0,9036	29,6797	0,9966	5,585	0,9918	8,6536
Gansu	0,9066	13,4591	0,995	3,1269	0,9931	3,6538
Guangdong	-0,0003	464,2315	0,9996	9,6581	0,9981	20,1638
Guangxi	0,7153	40,6114	0,9972	4,001	0,9932	6,2729
Guizhou	0,8375	21,9397	0,995	3,8637	0,9939	4,2531
Hainan	0,986	6,186	0,9888	5,5543	0,9972	2,7571
Hebei	0,9598	22,1855	0,9953	7,581	0,9989	3,7281
Heilongjiang	0,6812	140,2313	0,9993	6,63	0,9988	8,607
Henan	0,9471	97,7368	0,9969	23,6378	0,9961	26,3993
Hong Kong	0,9817	47,1392	0,9935	28,1294	0,9914	32,3991
Hubei	0,9978	19,8231	0,9945	14,1854	0,9832	24,7863
Hunan	-6,2203	877,4929	0,9981	14,2241	0,999	10,3383
InnerMongolia	0,9025	15,7109	0,9988	1,7425	0,9978	2,3829
Jiangsu	-0,0041	217,4812	0,9997	3,5273	0,9993	5,8369
Jiangxi	0,581	211,1184	0,9962	20,1472	0,9966	19,0677
Jilin	0,9782	5,1739	0,9985	1,3542	0,9979	1,5919
Liaoning	0,7524	18,9975	0,996	2,4279	0,9877	4,2403
Macau	0,9725	2,3157	0,986	1,6523	0,9889	1,4737
Ningxia	0,9356	6,3217	0,9968	1,4154	0,9967	1,4253
Qinghai	0,9296	1,3052	0,9666	0,8985	0,9691	0,8642
Shaanxi	0,9478	17,8166	0,9981	3,4232	0,9954	5,2769
Shandong	-0,0054	260,4031	0,996	16,4365	0,997	14,1877
Shanghai	-0,0125	164,5664	0,9976	8,0113	0,996	10,3275
Shanxi	0,8939	16,8745	0,9848	6,3803	0,9872	5,8556
Sichuan	0,5224	119,243	0,9829	22,5875	0,9864	20,1479
Tianjin	0,8208	22,7279	0,9974	2,7584	0,9854	6,4816
Tibet	0,7171	0,1672	0,6357	0,1897	0,6438	0,1876
Xinjiang	0,9453	6,2169	0,9976	1,3035	0,9983	1,0963
Yunnan	0,5395	34,1546	0,9954	3,4182	0,9555	10,6175
Zhejiang	0,8163	159,397	0,9687	65,8145	0,9907	35,8373

Figure 5

Performances of learners for China case study

City model	Stacked-LSTM		Stacked-CNN		Stacked-DNN	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
Anhui	0,53741635	196,514963	0,9989	6,9576	0,99952426	4,58918391
Beijing	0,726	65,2408	0,9966	6,4701	0,9971	5,9561
Chongqing	0,8434	42,2	0,9968	5,621	0,9979	4,5751
Fujian	0,8785	21,5137	0,9966	4,5027	0,9927	3,0636
Gansu	0,9165	9,177	0,9897	2,9363	0,9896	2,9504
Guangdong	0,3197	247,1109	0,9984	10,5856	0,9993	6,8291
Guangxi	0,8518	19,6562	0,9913	4,288	0,9918	4,1495
Guizhou	0,9271	10,3127	0,9946	2,6324	0,9984	1,4459
Hainan	0,9657	6,6076	0,9932	2,7397	0,9949	2,3846
Hebei	0,9231	21,4222	0,9988	2,4478	0,9987	2,4736
Heilongjiang	0,861	56,6498	0,9992	3,4347	0,9992	3,3724
Henan	0,9144	82,7012	0,999	8,1733	0,999	8,1149
Hong Kong	0,8242	148,0537	0,9968	8,6824	0,997	8,413
Hubei	0,99617685	26,2827444	0,98728752	17,398025	0,99589598	9,88529184
Hunan	0,957	42,806	0,9995	4,4288	0,9995	4,431
InnerMongolia	0,5408	21,4703	0,9469	3,8261	0,9574	3,4264
Jiangsu	0,8294	59,9495	0,9984	5,2881	0,9993	3,5346
Jiangxi	0,716	116,9418	0,9982	9,2998	0,9979	8,6106
Jilin	0,9847	2,9833	0,9973	1,1279	0,9982	0,9037
Liaoning	0,8575	9,2185	0,9954	1,4317	0,9977	1,0143
Macau	0,9976	0,6819	0,9835	1,0652	0,9847	1,0249
Ningxia	0,9775	2,5693	0,9862	1,9198	0,9847	1,8179
Qinghai	0,958	0,583	0,9969	0,135	0,9974	0,1251
Shaanxi	0,9447	11,7065	0,9989	1,5863	0,9989	1,5276
Shandong	0,8275	81,1387	0,9407	44,2062	0,9428	43,406
Shanghai	0,554	76,5518	0,9798	11,1253	0,9824	10,3698
Shanxi	0,7911	14,3299	0,9972	1,3047	0,9983	1,0195
Sichuan	0,9646	21,9261	0,9981	4,6034	0,9987	3,7713
Tianjin	0,9342	10,1255	0,9882	3,2117	0,9891	3,0826
Tibet	0,9363	0,0504	0	0,2222	0	0,2084
Xinjiang	0,9819	2,5447	0,9983	0,7267	0,9987	0,628
Yunnan	0,9785	4,3513	0,9912	2,4513	0,9947	1,8991
Zhejiang	0,9134	67,3583	0,9848	24,7524	0,989	21,0532

Figure 6

Performances of meta-learners for China case study

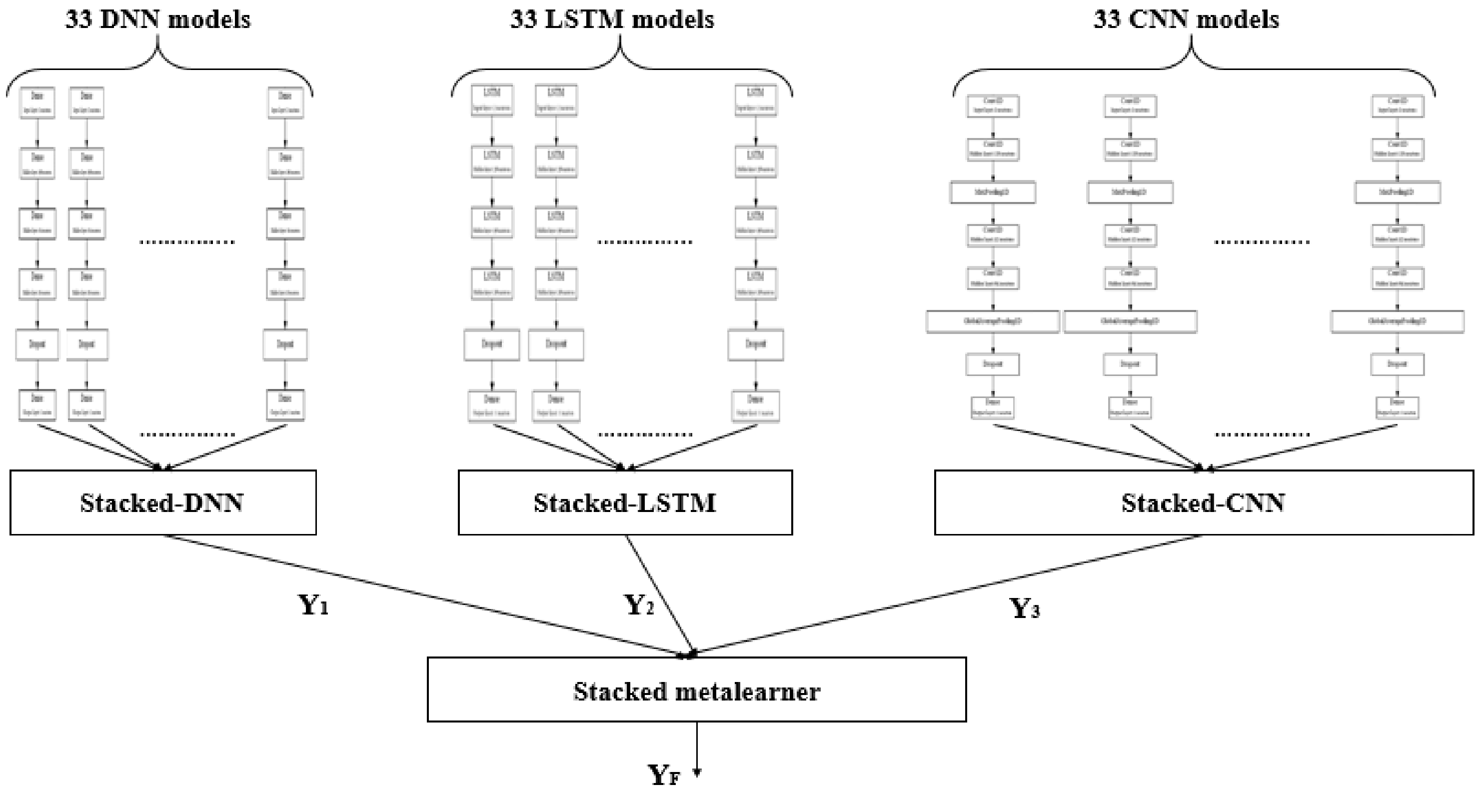


Figure 7

Augmented prediction process