

# Flexible Adaptive Control of Snake-Like Robot Based on LSTM and Gait

Sheng Ouyang<sup>1,\*</sup> and Wu Wei<sup>1</sup>

<sup>1</sup> Department of Automatic Control and Engineering, South China university of technology, Guangzhou 510640, China

\*shengouy13@foxmail.com

**Abstract.** This paper presents a flexible control method for snake-like robots to adapt to the environment autonomously. This method enables the snake-like robot to sense the environment and adjust autonomously according to the changes of the environment. For example, snake-like robots climb pipes with varying diameters. In order to achieve efficient and flexible motion, we adopt closed-loop gait control. This control method uses a parameterized sine wave (gait function) and long short-term memory network (LSTM) model. Because of the structure of LSTM is suitable for prediction of time series data, we use LSTM to predict the changes of joint angles that can best represent the shape change of snake-like robot, and integrate the predicted joint angle values with the gait values to realize the control of robot. Because this control method will eventually allow the snake-like robot to move autonomously in the changing environment, we can achieve the flexible adaptive behavior of the robot.

## 1. Introduction

Snake-like robot is a kind of robot with high redundancy consisting of constrained links chained together in series. We can use parameterized sin waves [1] to control the periodic motion of the robot. This parameterized sine wave is called gait. By changing some of the parameters of this gait, the snake-like robot can be controlled to make different motions in different environments. A survey on the model, control and various gait of snake-like robots is presented in [2].

The gait method based on parameterized sin wave form can carry out efficient open-loop control. Each module in the robot contains a low-level controller that drives its joint angle to the commanded angle, and feedback is provided on the module's actual joint angle [3]. However, for complex terrain and environment, this method cannot make snake-like robot adjust its motion posture to adapt to the changing environment. Therefore, we need to find an appropriate way to integrate feedback joint angle information into gait control to perceive the external environment and timely adjust the robot's posture to adapt to this change. In this method, the real joint Angle information of snake-like robot is processed and added to the gait control, which actually completes a closed-loop feedback control. We use the LSTM model to process the feedback joint angles and combine the predicted values obtained from the model with the gait parameters to achieve real-time closed-loop gait control.

The advantage of using LSTM is that it can memorize and train long-term time-series information, and the LSTM model can effectively predict the state change of time-series information at the next moment. For this paper, LSTM can well process the real-time feedback of robot joint Angle information. By combining the predicted Angle with the gait to achieve the self-adaptive adjustment of snake-like robot.



In this paper, the effectiveness of this flexible control method is verified by designing a snake-like robot to climb along an external pipe whose diameter gradually decreases. Through experiments, we proved that this control method can perform better and safer climbing on pipes with different diameters than the control method based solely on parameterized sin wave gait. The closed-loop control realized in this paper can adjust the joint Angle rapidly with the change of pipe diameter, and this autonomous behavior is flexible enough. This method can achieve more complex movements than adjusting gait parameters based on manual control. For a pipe with a gradually changing diameter, the snake-like robot can adjust its gait autonomously rather than manually to achieve a safe climbing movement.

## 2. Theories

### 2.1. The gait of a snake-like robot

Orthogonal joints are used to connect adjacent joints of the robot, and the rotation axis of each joint of the robot is perpendicular to each other. The rotation axis of the latter joint is obtained by rotating the rotation axis of the former joint 90 degrees around the central axis of the snake body and then translating the length of the joint backward. Considering the orthogonal structure of snake-like robot, in order to realize the simple and efficient movement, we adopted the extended form of parameterized sin control function based on Hirose's serpenoid curve [4], and its 3D extensions [5]. for the motion control method of snake-like robot. The control equation of each joint angle is:

$$\theta_i = A_i \sin(\omega t + ki) + \gamma_i \quad (1)$$

In the above equation,  $i$  is the number of joints,  $\theta_i$  represents the rotation angle of joints,  $A_i$  represents the amplitude,  $\omega$  describes the spatial frequency of the macro shape of robot relative to the joint number  $i$ ,  $t$  represents the time variable,  $k$  represents the motion control parameter, and  $\gamma_i$  represents the Angle offset.

In this paper, we adopt the spiral climbing gait, in which the snake-like robot presents a twisted spiral posture. We wrap the robot around the tube and the robot rolls up and squeezes the outside of the tube. The control mode is consistent with the control mode in equation (1) except that some parameters are constants.

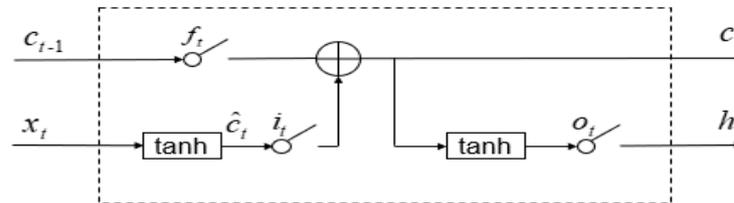
$$\theta_i = A \sin(\omega t + ki) \quad (2)$$

In the formula,  $k$  determines the winding number of the snake-like robot. In order for the robot to climb the diameter tube (4.5-7cm) we set in the experiment,  $A$  was set to 2.  $k$  is set to 1.55. The frequency  $\omega$  is set to 0.8. The angle offset constant  $\gamma_i$  is set to 0.

### 2.2. The structure of Long Short - Term Memory (LSTM) Network

LSTM is a special type of recurrent neural network (RNN), first proposed by Sepp Hochreiter and Jurgen Schmidhuber in 1997 [6]. Since then, there have been many minor changes to the original version [7,8,9,10]. In fact, training and prediction using RNN is not very effective. The main reason is that the problem of gradient disappearance and gradient explosion will occur in the process of Back Propagation Trough Time (BPTT) of RNN [11,12]. LSTM solves these problems very well. Therefore, this special deep learning network is widely used for text analysis and time-series data prediction [7,13]. The core of LSTM is that it constructs three gate structures to reduce and increase the information of storage units, which are input gate, forgetting gate and output gate. The input gate determines the information that needs to be updated. The forgetting gate determines the information that needs to be deleted. The output gate determines the information to be output. With this special gate structure, LSTM can not only effectively solve the problem of gradient disappearance, but also save important information for a long time. It has the ability of long-term memory and can effectively

deal with long sequence problems. In addition, LSTM improves the accuracy of the previous RNN network. Figure 1 shows the cell structure of the LSTM.



**Figure 1.** Memory cell structure of LSTM hidden layer.

The general principle of LSTM can be expressed as equation (3) - (8):

$$i_t = s(W_i x_t + H_i h_{t-1} + b_i) \quad (3)$$

$$f_t = s(W_f x_t + H_f h_{t-1} + b_f) \quad (4)$$

$$o_t = s(W_o x_t + H_o h_{t-1} + b_o) \quad (5)$$

$$\hat{c}_t = \tanh(W_c x_t + H_c h_{t-1} + b_c) \quad (6)$$

$$c_t = f_t * c_{t-1} + i_t * \hat{c}_t \quad (7)$$

$$h_t = o_t * \tanh(c_t) \quad (8)$$

Where  $f_t$ ,  $i_t$  and  $o_t$  respectively represent forgetting gate, input gate and output gate;  $s$  represents the sigmoid activation function with output value between 0 and 1;  $\tanh$  represents the hyperbolic tangent activation function;  $\hat{c}_t$  is the candidate value of memory cell at time  $t$ ;  $c_t$  is the state of the current memory cell at time  $t$ ;  $W(W_i, W_f, W_o, W_c)$  and  $H(H_i, H_f, H_o, H_c)$  are weight matrices;  $b(b_i, b_f, b_o, b_c)$  is the offset vector [14];  $h_t$  is the output value after output gate filtering.

### 3. Methods

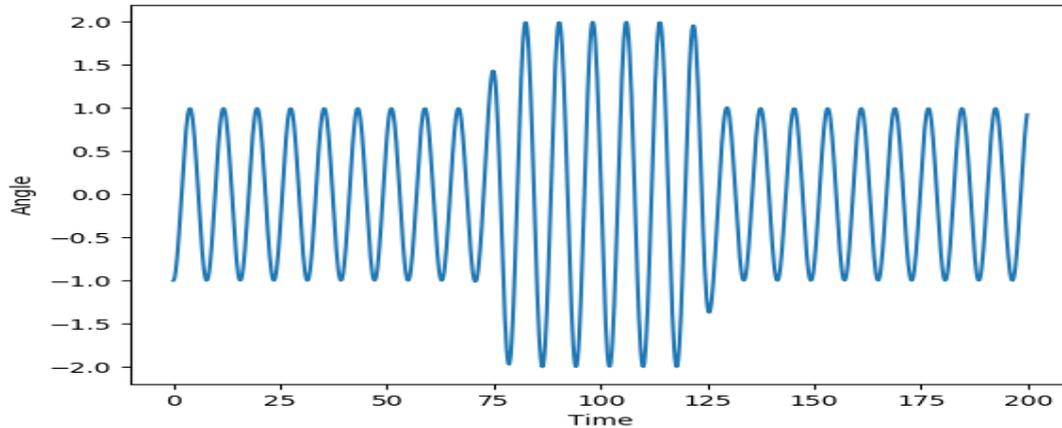
#### 3.1. Training and experimental data

In this paper, parameterized sin wave function is used to control the motion of the robot, so each joint of the robot rotates periodically in the form of sinusoidal wave. However, the sin wave control method can't make the snake-like robot to climb well when the pipe diameter changes. LSTM model needs to learn important feature of real-time adjustment of joint angle according to the change of pipe diameter, so we construct an improved sine wave as training data: parameterized sine wave function based on sigmoid function, as shown in equation (9) - (10). The characteristic of this function is that with the increase of time, the amplitude of the function gradually increases from the lower constant value to the higher constant value and then decreases to the lower constant value. We sample the function at an interval of 0.2 to construct a total of 1,000 sampling points, take the value of function at 40 adjacent sampling points as the sequence input value of LSTM model, and take the value of the 41st sampling point as the corresponding mark value. By training the model to learn the increasing characteristics of flexibility, it can learn the changing rules of snake-like robot climbing on the pipe. In the experiment, the real feedback joint angle of the snake-like robot is used as the model input, and then the model predicts the change of the joint angle at the next moment.

$$a = \frac{1}{1 + e^{-t+75}} - \frac{1}{1 + e^{-t+125}} + 1 \quad (9)$$

$$\theta_i = a * \sin(\omega t + k_i) \quad (10)$$

$a$  represents the amplitude change in the form of sigmoid, and the other variables have the same meaning as the gait control function variable of the snake-like robot described in 3.1. The corresponding shape of this function is shown in figure 2.



**Figure 2.** Shape of LSTM model training function.

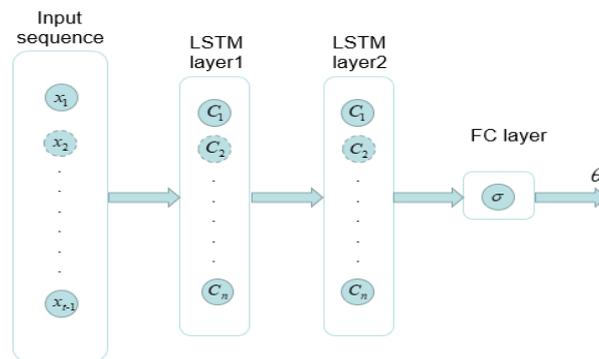
### 3.2. LSTM model framework

In this paper, an LSTM model framework with dropout is designed to predict the joint angle of the snake-like robot, and the predicted results will directly participate in the motion control of the robot. The model consists of three parts: input layer, hidden layer and output layer. The input layer is composed of the joint angle value of snake-like robot with time series. The hidden layer adopts the two-layer LSTM structure. Because model overfitting is a widespread problem in deep learning [15]. Therefore, Dropout [16] is used here to prevent overfitting.

The structure of LSTM model is shown in figure 3. The number of input sequences of the entire model is 40, and the number of units in each LSTM layer is 32. The unit marked as dotted line represents drop units (along with their connections) from the neural network randomly during training to prevent units from co-adapting too much [16].  $\sigma$  represents the  $\tanh$  activation function. Joint angle value  $\theta_t$  at time  $t$  is predicted by the fully connected layer of only one neuron, whose activation function is  $\tanh$ . The function of  $\tanh$  activation function is expressed as equation (11), whose value range is -1 to 1.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (11)$$

Compared with the traditional sigmoid activation function,  $\tanh$  function has the advantage of improving training efficiency. The output layer is composed of a neuron, which indicates that the predicted value of the joint angle at the next moment can be obtained through the LSTM hidden layer. Mean Square Error [17] (MSE) was used as the loss function, and Adam was used as the optimal weight method of the model.



**Figure 3.** The structure LSTM model with dropout for joint angle prediction of snake robot.

### 3.3. Performance criteria

For the LSTM model obtained by training, MSE is adopted as the Performance criteria of trained model prediction accuracy. MSE is expressed by equation (12). The smaller the value of MSE, the higher the accuracy and better performance of the trained model.

$$MSE(y, y') = \frac{\sum_{i=1}^n (y_i - y_i')^2}{n} \quad (12)$$

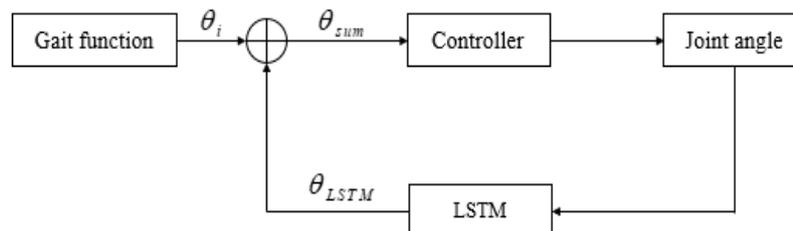
Where  $y_i$  is the correct value of the  $i$ th sample and  $y_i'$  is the predicted value.  $n$  is the number of samples.

### 3.4. Control strategy

The adaptive climbing control strategy of snake-like robot combines sin wave gait and joint rotation angle predicted by LSTM model to form an adaptive closed-loop control of snake-like robot, which is expressed by equation (14).

$$\theta_{sum} = c_1 \times \theta + c_2 \times \theta_{LSTM} \quad (13)$$

In the equation,  $c_1$  and  $c_2$  correspond to scaling factors of climbing gait function angle and LSTM prediction angle of snake robot. In this experiment, 0.4 and 0.8 were adopted respectively to achieve the best effect.  $\theta$  is the angle value of gait function,  $\theta_{LSTM}$  is the prediction angle of LSTM model, and  $\theta_{sum}$  is the control value of robot joint angle. The closed-loop control strategy in the climbing process of snake-like robot is shown in figure 4.



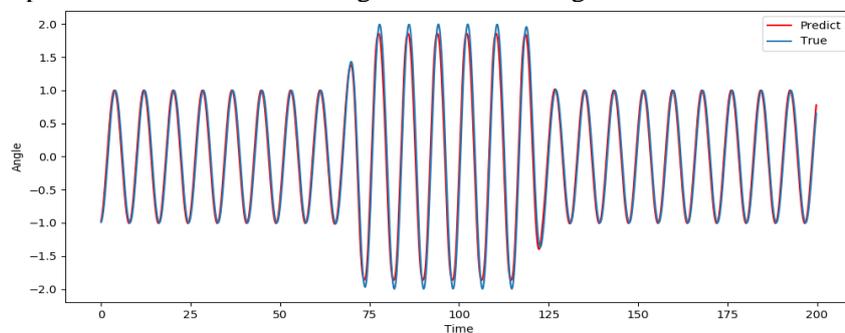
**Figure 4.** The closed-loop feedback control process in the climbing process of snake robot.

The snake-like robot shows the form of closed-loop feedback control in the process of tube climbing. For example, when the pipe diameter decreases, the LSTM model predicts the joint angle of the snake-like robot at the next moment based on the feedback, showing an increasing trend. Then the model outputs the joint angle value which slowly increases and adds it to the angle calculated by the gait function to obtain the control angle value of the snake robot

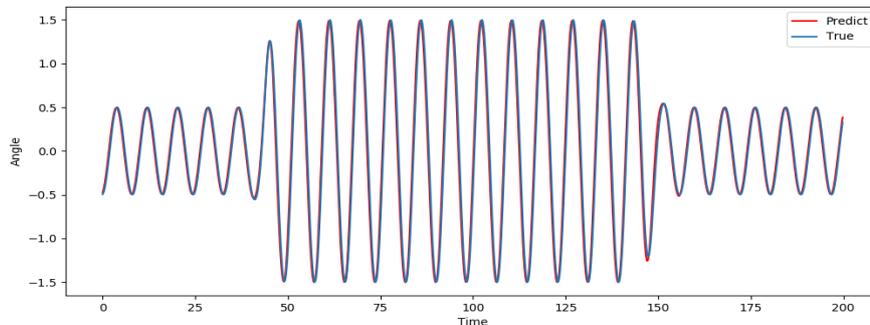
#### 4. Simulation and experiment

The experiment was divided into two parts. In the first part, the LSTM model structure is tested and proved to be able to predict the change of joint angle in the climbing process of snake-like robot. In the second part, the simulation of snake-like robot tube climbing is carried out to verify the effectiveness and performance of flexible adaptive control.

After training, the LSTM model can predict the training data accurately, as shown in figure 5. We found that the model after training has excellent stretching performance in time domain and space domain. The trained model can accurately predict the sin wave input with amplitude change at any different time period, which proves that LSTM model can effectively solve the time uncertainty of the pipe diameter change process. The meaning of time uncertainty is that we do not know in which period of time the robot will climb at the position of the pipe diameter change. The other conclusion is that the training data given to us during training is a sin wave that goes from low amplitude to high amplitude to low amplitude. In the prediction, we can accurately predict the sine wave of amplitude between the low amplitude and the high amplitude, which is a good extensibility of the model in space. Figure 6 shows prediction of data with changes in time and angle values.

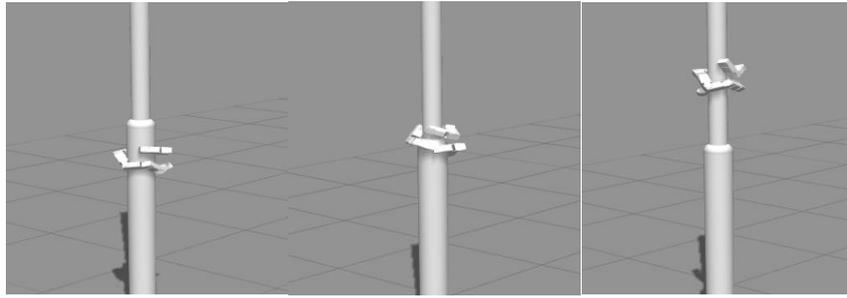


**Figure 5.** Prediction of training data by LSTM model. The blue curve is the training data, and the red curve represents the predicted value of LSTM model.



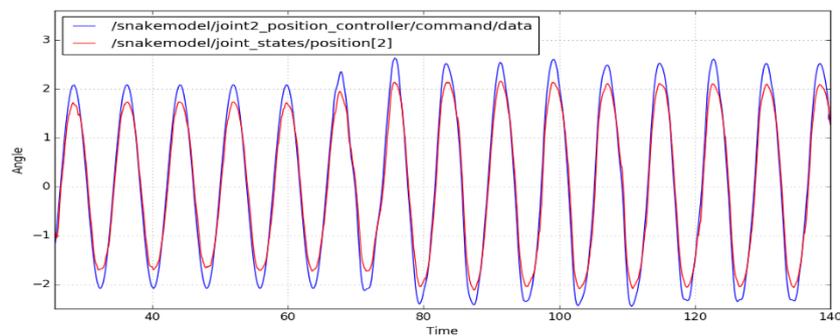
**Figure 6.** Prediction of data with changes in time and angle values.

In the simulation process of pipeline climbing of snake-like robot, the joint angle values at 40 moments are collected and feedback to the LSTM model. The model predicts the joint angle at the next moment based on the first 40 historical samples values. And adds the angle values to the gait control function to achieve adaptive flexible control. Figure 7 shows the change of the movement posture of the snake-like robot during climbing. Figure 8 shows the control angle value and feedback joint angle value of the second joint of the snake robot. Note that there is a constant deviation between the two angles because the control function needs to add a constant to overcome the gravity of the snake-like robot and keep it from skidding as it climbs. The rest of the joint changes are basically the same.



**Figure 7.** Simulation experiment of snake-like robot climbing with different pipe diameter.

The figure on the left shows the climbing movement of the snake robot before the pipe diameter changes. In the middle is the moment when the diameter changes. On the right is the climbing movement of the snake-like robot after the pipe diameter is narrowed.



**Figure 8.** Angle control value and feedback value of the second joint of snake robot.

The blue curve is the control angle value based on the addition of the predicted value of LSTM and the gait value, and the red curve is the feedback angle value.

In the climbing process of the snake-like robot, when the pipe diameter changes, the LSTM model will timely predict the angle at the next moment based on the feedback angle value sampled at the first 40 moments. At the same time, it can accurately predict the angle value when the pipe diameter gradually shrinks. The angle adaptive adjustment is carried out in a flexible way to achieve the joint angle value suitable for small diameter pipe climbing.

## 5. Conclusion

In this paper, a flexible adaptive control strategy of snake-like robot is proposed, which combines LSTM model with gait method. This control method allows the snake-like robot to autonomously change its posture to adapt to the complex environment. In the control method, LSTM model is used to predict the joint angle of snake-like robot, so as to judge whether the movement of snake-like robot will change with the change of external environment. For example, with the decrease of pipe diameter, the maximum value of the joint angle with periodic change increases, and the corresponding angle amplitude increases. By adding the predicted value and the gait function value in a certain proportion, the control strategy can adapt to the pipe diameter change.

In the simulation experiment, there are two reasons why the snake robot can be flexible. The first is that the training data constructed by us are parameterized sin waves based on sigmoid curve, whose amplitude changes slowly and flexibly. Therefore, LSTM model learns such flexible changes. In the experiment, the large diameter tube and the small diameter tube are connected by a curved surface, instead of the diameter suddenly changing from large to small, which is exactly suitable for the training data of LSTM model. Dropout added in the LSTM model is of great significance. It can really prevent overfitting and correctly predict the angle value of joints in the experiment.

At present, we are considering applying this control method to the crawling of snake-like robots in

pipelines and the adaptive movement of rugged ground. In future work, we hope to apply this control strategy to more complex environments, such as urban underground pipeline systems and urban ruins after disasters. However, in these complex terrains, we consider that more feature information needs to be included to realize LSTM model training and prediction and realize the adaptive movement of snake-like robots. We hope that researchers in the field of robotics can see the wide application value of LSTM model and apply it to the motion control of robots.

### Acknowledgments

The authors would like to thank 220 Laboratory of Control and Engineering Department in South China university of technology for their help.

### References

- [1] Tesch M, Lipkin K, Brown I, Hatton R, Peck A, Rembisz J, and Choset H 2009 Parameterized and Scripted Gaits for Modular Snake Robots *Adv. Robotics* **23** 1131–58
- [2] Transteth AA, Pettersen KY, and Liljebäck P 2009 A survey on snake robot modeling and locomotion *Robotica* **27** 999-1015
- [3] Wright C, Buchan A, Brown B, Geist J, Schwerin M, Rollinson D, Tesch M and Choset H 2012 Design and Architecture of the Unified Modular Snake Robot *IEEE Int. Conf. on Robotics and Automation* (Saint Paul: Minnesota) pp 4347–4354
- [4] Hirose S and Morishima A 1990 Design and control of a mobile robot with an articulated body *Int. J. Robot. Res.* Vol. 9 pp 99-114
- [5] Ohno H and Hirose S 2001 Design of slim slime robot and its gait of locomotion *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (MAUI) pp 707–715
- [6] Hochreiter S and Schmidhuber J 1997 Long Short-Term Memory *Neural Comput.* **9** 1735–80
- [7] Graves A and Schmidhuber J 2005 Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures *Neural Netw.* **18** 602–610
- [8] Gers FA, Schmidhuber J 2000 Recurrent Nets that Time and Count *IEEE/INNS/ENNS Int. Joint Conf. on Neural Netw.* (Como: Italy) PP 189-194
- [9] Greff K, Srivastava RK, Koutnik J, Steunebrink BR and Schmidhuber J 2017 LSTM: A Search Space Odyssey *IEEE Trans. Neural Net. and Learn. Sys.* **28** 2222-32
- [10] Wu YT, Yuan M, Dong SP, Lin L and Liu YQ 2018 Remaining useful life estimation of engineered systems using vanilla LSTM neural networks *Neurocomputing* **275** 167-79
- [11] Razvan P Mikolov T and Bengio Y 2013 On the Difficulty of Training Recurrent Neural Networks *In Proceedings of the 30th Int. Conf. on Machine Learn.* (Atlanta: Georgia) Vol 28 pp 1310–1318
- [12] Bengio Y, Simard P and Frasconi P 1994 Learning Long-Term Dependencies with Gradient Descent Is Difficult *IEEE Trans. Neural Netw.* **5** 157–66
- [13] Graves A Mohamed A.R and Hinton G 2013 Speech Recognition with Deep Recurrent Neural Networks *In Proceedings of the 2013 IEEE Int. Conf. on Acoustics, Speech and Signal Processing* (Vancouver: Canada) pp. 6645–6649
- [14] Qian F, Chen L, Li J, Ding C, Chen XF and Wang J 2019 Direct Prediction of the Toxic Gas Diffusion Rule in a Real Environment Based on LSTM *Int. Jour. of Env. Res. and Pub. Health* **16** DOI 10.3390/ijerph16122133
- [15] Hawkins DM 2004 The Problem of Overfitting *J. Chem. Inf. Comput. Sci.* **44** 1–12
- [16] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 Dropout: A Simple Way to Prevent Neural Networks from Overfitting *J. Mach. Learn. Res.* **15** 1929–58
- [17] Wang Z and Bovik AC 2009 Mean Squared Error: Love It or Leave It? A new look at signal fidelity measures *IEEE Signal Processing Magazine* **26** 98-117