Does the Zero Carry Essential Information for Artificial Neural Network learning to simulate the contaminant transport in Urban Areas?

M. Berendt-Marchel¹, A. Wawrzynczak¹,²

¹Siedlce University, Faculty of Exact and Natural Sciences, Institute of Computer Sciences, Poland
²National Centre for Nuclear Research, Swierk-Otwock, Poland

E-mail: monika.berendt-marchel@uph.edu.pl

Abstract. The release of hazardous materials in urbanized areas is a considerable threat to human health and the environment. Therefore, it is vital to detect the contamination source quickly to limit the damage. In systems localizing the contamination source based on the measured concentrations, the dispersion models are used to compare the simulated and registered point concentrations. These models are run tens of thousands of times to find their parameters, giving the model output’s best fit to the registration. Artificial Neural Networks (ANN) can replace in localization systems the dispersion models, but first, they need to be trained on a large, diverse set of data. However, providing an ANN with a fully informative training data set leads to some computational challenges. For example, a single simulation of airborne toxin dispersion in an urban area might contain over 90% of zero concentration in the positions of the sensors. This leads to the situation when the ANN target includes a few percent positive values and many zeros. As a result, the neural network focuses on the more significant part of the set - zeros, leading to the non-adaptation of the neural network to the studied problem. Furthermore, considering the zero value of concentration in the training data set, we have to face many questions: how to include zero, scale a given interval to hide the zero in the set, and include zero values at all; or limit their number? This paper will try to answer the above questions and investigate to what extend zero carries essential information for the ANN in the contamination dispersion simulation in urban areas. For this purpose, as a testing domain, the center of London is used as in the DAPPLE experiment. Training data is generated by the Quick Urban & Industrial Complex (QUIC) Dispersion Modeling System.

1. Introduction

The release of hazardous substances to the atmosphere due to accidents during the transport and storage of toxic materials is a serious threat to human health and the environment. When such an event occurs, the most important thing is to quickly locate the source of the leak in order to minimize the damage. If the source of the contamination is known, the emergency responders can take appropriate action quickly. The most problematic situation is when the network of sensors distributed over the city shows non-zero values of a hazardous substance, which source is unknown. In such cases, it is vital to have a system that can respond in real-time to changing concentrations on the sensor grid and approximately indicate the location of the source of the release of the hazardous substance. Algorithms that can deal with this task...
Figure 1. The domain representing the area of central London used in DAPPLE experiment.

fall into two categories. The first are based on the backward approach and are applied to events in open areas or on a continental scale. The latter is based on the forward approach. In this case, the dispersion model parameters (including the source location) are sampled to find the smallest difference between the model result and the values measured on the sensors in the spatial domain under consideration.

For event reconstruction in urban areas, which is of interest in this paper, an advanced dispersion model is required to consider wind turbulence between buildings. However, to obtain the most likely location of the source of contamination, the dispersion model must be run tens of thousands of times, which is very time-consuming computationally. Therefore, utility dispersion models need to be fast to be used in a real-time working emergency system.

In [1] and [2] were first reported reconstructions in urban scales using building models. In [1], the authors used an adjoint representation of the source receptor relationship and applied a Bayesian inference methodology in conjunction with Markov Chain Monte Carlo (MCMC) sampling procedures. In [2] authors applied the methodology presented in [3] to reconstruct the flow around an isolated building and the flow during IOP3 and IOP9 of the Joint Urban 2003 Oklahoma City experiment. In this reconstruction, the FEM3MP [4] model was applied to predict the atmospheric dispersion of the released substance. In [5] authors applied the approximate Bayesian computation algorithm (ABC) to localize the source of contamination in the highly urbanized terrain of the center of London utilizing the actual field experiment data from DAPPLE experiment [6]. The Quick Urban Industrial Complex (QUIC) Dispersion Modeling System was applied [7]. Even though in [5] the fast convergence of the ABC algorithm was proven, the whole framework cannot be implemented in the real-time emergency system due to the long computational time required by the dispersion model in urbanized terrain. Therefore, one of the ideas was to train the artificial neural network (ANN) to simulate contamination transport in urban areas effectively. In [8], [9] and [10] it was confirmed that artificial neural networks could successfully replace dispersion models in localization systems. Nevertheless, first, training on a large, diverse dataset is needed. The training process requires the preparation of
Figure 2. The scatter plots representing results of training, testing, and validation process of the ANN for whole training data set. The dashed line represents the ideal fit.

Figure 3. The scatter plots representing results of training, testing, and validation process of the ANN for whole training data set, which is rescale to interval $<0,1>$. The dashed line represents the ideal fit.

many simulations as a training set for the neural network. Preparing a fully informative training data set causes many computational challenges. The first is the small amount of data from actual experiments. Therefore, in order to train the neural network, synthetic data obtained e.g., from the dispersion model, should be used. The preparation of a synthetic training set is computationally expensive. However, once trained, the network can be a high-speed tool for estimating the point-concentrations point for a given source of contamination.

The second challenge is that a single airborne toxin dispersion simulation in an urban area can contain over 90% of zero concentration in sensor positions. This leads to the situation when the ANN target includes a few percent positive values and many zeros. As a result, the neural network focuses on most of the set zeros and does not adapt to the studied problem. Thus, when considering a zero concentration value in the training set, we have to face many questions: how to include zero, scale a given interval to hide the zero in the set, including zero values at all; or limit their number? This paper will try to answer the above questions and investigate to what extent zero carries essential information for the ANN in the contamination dispersion simulation in urban areas.

2. Artificial Neural Network

Artificial neural networks (ANN) are computational models imitating natural neurons and synapses connecting them with their architecture. The basic element of a neural network is a single neuron [11]. Neurons are linked together to form a structure capable of learning from sets of training samples without knowing any laws or equations. When considering a neural network as a contaminant dispersion model, the information comes from $n$ separate parameters that make up input $X_i (i = 1,\ldots,n)$. These parameters may include source parameters, sensor location, and meteorological conditions. The answer to such a system is the specific output - in
Figure 4. The scatter plots representing results of training, testing, and validation process of the ANN for whole training data set, which is rescale to interval $<10^{-5}, 1>$ and logarithmized. The dashed line represents the ideal fit.

Figure 5. The scatter plots representing results of training, testing, and validation process of the ANN for whole training data set, which is rescaled to interval $<10^{-15}, 1>$ and logarithmized. The dashed line represents the ideal fit.

our case, the concentration value. A single independent neuron represents each input parameter. Each neuron computes a linear combination between weights $\omega_{ij}$ and inputs including bias $b_i$, then each layer is summed up according to the formula $C_j = \sum_i \omega_{ij}X_i + b_i$. Formula $C_j$ is transformed by some activation function $f$, which may be linear or non-linear, e.g., log-sigmoid or hyperbolic tangent. Bias allows the activation function to be offset from zero. Both the initial weight and bias values may be given or selected randomly. The output of a single layer of neurons is an input $X_j = f(C_j)$ for the next layer called the hidden layer. A neural network can contain many hidden layers with varying numbers of neurons in it. However, too many neurons can lead to over-fitting. Having such a framework of input variables and sets of functions, the ANN has to be trained in order to obtain the best estimate for each weight $\omega_{ij}$. The weight values in the next iterations are determined by an optimization procedure, the so-called learning algorithm [12]. The result of the neural network is compared with the target in order to calculate a predefined value of the error function. The error value is sent over the network, and the algorithm adjusts the weights of each connection between neurons in different layers, respectively, to reduce the value of the error function. This repeated process corresponds to the number of training iterations that cause the network result to coincide with a state where the error between the output and the target is minimal.

2.1. Selected ANN Topology

There are many types of artificial neural networks. In this work, we use the feed-forward neural network, which is one of the simplest and most widely used. This network has a simple structure. The signal only flows in one direction from the input neurons to the output neurons. A feed-forward neural network is used, i.a., in image and speech recognition as well as in classification
Figure 6. The scatter plots representing results of training, testing, and validation process of the ANN for training data set including the zero concentrations, which is rescaled to interval $<1\times10^{-5}, 1>$ and logarithmized. The dashed line represents the ideal fit.

Figure 7. The scatter plots representing results of training, testing, and validation process of the ANN for training data set excluding the zero concentrations, which is rescaled to interval $<1\times10^{-5}, 1>$ and logarithmized. The dashed line represents the ideal fit.

It was also used successfully in the problems of predicting the transport of pollutants, e.g., [13], [14] and [15]. The neural networks used in this article contain 5 hidden layers with 42, 35, 26, 14, and 5 neurons in each layer, respectively. The Levenberg-Marquardt algorithm was chosen as the training algorithm. The input data set was divided into training data set - 70%, validation and testing data sets - 15% each.

In order to obtain a diverse set of training data for a neural network, sufficiently large synthetic data should be created. For this purpose, we use the Quick Urban and Industrial Complex Dispersion Modeling System (QUIC). The test domain is the center of London used in the DAPPLE experiment [6]. The domain was $752 \times 652 \times 80$ m in which we have placed representations of actual buildings. The average height of the buildings is 21.6 m (range 10 to 64 m). The entire considered domain is presented in Fig. 1.

3. Results

Fig. 2 shows scatter plots representing the training, test, and validation process of the neural network. The dashed line represents a perfect linear fit, where the neural network’s output coincides with the target. A too high number of zero concentrations in the training set’s target causes the neural network to not adapt to the problem under consideration. Instead, the neural network adjusts to the prevailing number of values in the set - zeros. One way to include zero values in the training set is to scale the set to interval $<0, 1>$. However, if the target in the training set contains an extensive range of data and the difference between the minimum and maximum values is very large, for example, 0 to 20000, rescaling data to interval $<0, 1>$ and leaving the interval closed at zero will not improve the result which is presented on Fig. 3.
neural network will react in the same way as for unscaled data because zero concentration values are still too large. In that case, the good idea is to logarithmize the target in the training set to reduce the data space. However, we cannot logarithmize the zero value. So in the first step, we should scale the set to such an interval in which we replace the value of zero with another positive value like $1e^{-05}$ (Fig. 4) or $1e^{-15}$ (Fig. 5). The selected value should depend on the range of values in the training set. As it is presented in Fig. 4 and Fig. 5 result will depend, i.a., on how the values will be distributed in the new data space and how ample the data space will be. We also pose a question: does in the case of simulation of airborne toxin dispersion in an urban area, we should include zero concentration in the positions of the sensors at all? Fig. 6 presents the scatter plots representing the training, testing, and validation process of the ANN for the training set, including the zero concentrations, and Fig. 7 presents the same for the set excluding the zero concentrations. The results are very similar. The value of the regression coefficient $R^2$ of all subsets in both trained neural networks is approximately 0.92. This may lead to the conclusion that we do not have to take zero values of concentration in the training set. Moreover, it would result in a significant acceleration of the computation time because the neural network training would take place on a set, truncated by even 90% of samples. However, in the problem of contaminant transport, the most important thing is to train the neural network to simulate the physical stages of contaminant dispersion appropriately.

Fig. 8 illustrates sample distribution of gas concentration during thirty minutes after the 800 s release of 10 $Mg$ of gas from the source located at $x = 212$ m, $y = 158$ m within the DAPPLE domain as simulated by the QUIC model. Comparing the contamination dispersions simulated by ANN (Fig. 9), trained on the set that excludes zero concentrations, under the same condition as in the example simulation made by the QUIC model, it can be noticed that the neural network predicts a similar gas distribution very close to the position of the hazardous gas source. Unfortunately, the neural network does not reflect the gas transport correctly following the wind direction. On the other hand, the simulation of contaminant dispersion by a neural network including zero concentration values in the training set, presented in Fig. 10
is very similar to the QUIC simulation (Fig. 8). The neural network correctly predicts the gas distribution close to the source of the contamination. Moreover, the gas spreads correctly with the wind direction, and the gas concentration slowly decreases in the next time steps.

**Figure 9.** The dispersion of the contaminant simulated by the ANN which exclude the zero concentrations during consecutive thirty minutes after the 800 seconds release of 10 Mg of gas from the source located at $x = 212 \text{ m}$, $y = 158 \text{ m}$ within the DAPPLE domain.

**Figure 10.** The dispersion of the contaminant simulated by the ANN which include the zero concentrations during consecutive thirty minutes after the 800 seconds release of 10 Mg of gas from the source located at $x = 212 \text{ m}$, $y = 158 \text{ m}$ within the DAPPLE domain.
4. Conclusions
Artificial neural networks are a suitable replacement for computationally expensive dispersion models. They can simulate the transport of hazardous substances in urban areas. However, first, they need to be trained on a large, diverse dataset. Due to the small amount of data from actual experiments, synthetic data have to be used. These data may contain many zero concentration values, which can cause numerous computational problems and lead to the neural network’s failure to adapt to the problem under study. In this paper, we tried to answer how to include zero and how to scale a given interval to hide the zero in the set. We presented how the ANN training set is influenced by the rescaling and the logarithmization of the target, which aims to reduce the dimensionality of the training set so that the neural network can better adapt to the problem under study. We also answered the following question: does in the case of simulation of airborne toxin dispersion in an urban area, we should include zero concentration in the positions of the sensors at all? The obtained results showed that in the case of a neural network trained on training datasets including and excluding zero concentration values, the $R^2$ coefficient for the training, test and validation set is approximately 92. However, in the case of contamination transport simulation in urban areas, the most important thing is that the neural network is able to simulate the physics of contaminant dispersion correctly. Considering that, a neural network trained on a set that includes zero concentration values is better in predicting the direction of contamination spread and the behavior of the gas cloud in subsequent time steps, compared to a neural network trained on a set that excludes zero values.

We can conclude that in training an artificial neural network that simulates airborne contaminant dispersion, the inclusion in training set the data reflecting zero concentrations on the individual sensor carries essential information and allows the neural network to learn the physics of contaminant transport in urban areas properly. Moreover, the obtained results are better than in the case of zero concentrations not being taken into account.

Acknowledgments
We thank Michael Brown and Los Alamos National Laboratory for the possibility to use the Quick Urban & Industrial Complex Dispersion Modeling System.

References
[1] Keats A, Yee E, Lien F S 2007 Atmospheric environment 41(3) 465–479
[2] Chow F K, Kosovia B, Chan S 2008 Journal of applied meteorology and climatology 47(6) 1553–1572
[3] Johannesson G, Hanley B, Nitao J 2004 Dynamic bayesian models via monte carlo-an introduction with examples. No. UCRL-TR-207173 (Lawrence Livermore National Lab.)
[4] Chan S T, Leach M J 2007 Journal of applied meteorology and climatology 46(12) 2127–2146
[5] Kopka P, Wawrzynczak A 2018 Atmospheric environment 195 63–77
[6] Wood C R et al. 2009 Bulletin of the American Meteorological Society 90(7) 955–970
[7] Williams M D et al. 2004 QUIC-PLUME theory guide. (Los Alamos National Laboratory) 43
[8] Wawrzynczak A, Berendt-Marchel M 2021 Studies in Computational Intelligence 902
[9] Wawrzynczak A, Berendt-Marchel M 2020 Lecture Notes in Computer Science 12138
[10] Wawrzynczak A, Berendt-Marchel M 2019 J. Phys.: Conf. Ser. 1391
[11] Bishop C M 1995 Neural networks for pattern recognition (Oxford university press)
[12] Haykin S 1994 Neural networks: a comprehensive foundation (Prentice Hall PTR)
[13] AllDabbous A N, Kumar P, Khan A R 2017 Atmospheric Pollution Research 8(3) 446-454
[14] Sham S R et al. 2021 Urban Climate 37 100837
[15] Ma D, Zhang Z 2016 Journal of hazardous materials 311 237–245