Received January 15, 2019, accepted January 20, 2019, date of publication February 1, 2019, date of current version February 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2896887

Forecasting Emergency Calls With a Poisson Neural Network-Based Assemble Model

HONGYUN HUANG[1], MINGYUE JIANG[2], ZUOHUA DING[2], (Member, IEEE), AND MENGCHU ZHOU[3], (Fellow, IEEE)

[1]Center of Multi-Media Big Data of Library, Zhejiang Sci-Tech University, Hangzhou 310018, China
[2]School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China
[3]Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA

Corresponding author: Zuohua Ding (zouhuading@hotmail.com)

This work was supported in part by the grants from the National Natural Science Foundation of China under Grant 61751210 and Grant 61572441, and in part by the Research Foundation of Zhejiang Sci-Tech University under Grant 17032184-Y.

ABSTRACT Forecasting emergency calls are of great importance in practice. By forecasting the occurrence of unfortunate events, we can learn from these events and further prevent their occurrence in the future. However, because of the uncertainty of event occurrences, it is hard to guarantee their prediction accuracy. In this paper, a combined model, which consists of two parts, is proposed. The first part is a Poisson neural network model (PNN). It is responsible for basic forecasting, and its initial weights and thresholds are trained by applying a genetic algorithm. The second part consists of multiple linear regression (MLR), autoregressive integrated moving average (ARIMA), and multivariate gray (GM), which are responsible for estimating residual errors. The basic prediction result adjusted by the residual error is used as the final forecasting result. The proposed model fully takes the advantages of PNN, MLR, ARIMA, and GM, and thus improves forecasting performance. Our method has been applied to the emergency calls of Ningbo, China. The experimental results show that the proposed model has advantages over some existing forecasting models, such as a neural network model, Poisson regression, and stochastic configuration networks in terms of mean absolute percentage error.

INDEX TERMS Accident forecasting, combined model, Poisson distribution, Poisson neural network, residual model.

I. INTRODUCTION

Emergency forecasting is of great significance in real-life applications, especially in an emergency medical service. Accurate forecasting of emergency accidents can save time and resources, and reduce property damage and personal casualties. Besides, it is also helpful to optimize the limited resource allocation.

There are several different approaches to model emergent or non-controllable accidents, such as scenario analysis [19], regression prediction [10], time prediction [2], [3], Markov Chain [20], [44], grey model [24] and nonlinear prediction [23]. Among these methods, scenario analysis is the most common one. It combines some other quantitative methods, such as regression prediction methods, to optimize the situation according to scenario analysis results, which can make the forecasting result more reasonable. it can be applied to a wide range of fields without any assumptions. But it is limited to a certain extent by people's knowledge, experience and ability. The common techniques of scenario analysis includes event tree [1], fault tree [36] and Petri net [39]. Nivolianitou *et al.* [29] compare these three techniques from the aspects of event sequencing, event factors, event dependency, modeling time and error recovery ability. Their conclusion shows that Petri nets provide the best time description for the accidents, while the event tree focuses on the factors to analyze the event, and the fault tree combs the main events affecting the accidents. These models are only a rough analysis of accident situations, which is unable to make an accurate prediction of specific events. Furthermore, the data used by these models is full of uncertainty [30]. Thus it is hard to repeat the experiment or use the model in other areas.

With the goal of providing high forecasting accuracy, in this paper, we make improvements from both model and data aspects, for forecasting emergency calls. In terms of data,

The associate editor coordinating the review of this manuscript and approving it for publication was Long Wang.

we not only collect the weather and time information, but also take the correlation between adjacent time into account. Then we can analyze the effect of time and weather on emergency calls. On the model side, Poisson neural network (PNN) based on the Poisson distribution is established, which is more suitable for modeling the occurrence time of random events of unit time than the traditional neural network. Moreover, a combined model, which is composed of multiple linear regression (MLR), multivariate gray (GM) and autoregressive integrated moving average (ARIMA) models, is established to correct the PNN model's error. We combine the output of PNN and the residual errors to compute the ultimate forecasting result. By combining different models, the resulting model can increase the chance to capture different patterns in the data and thus improves forecasting performance. Generally, PNN can be adequate in modeling the data with nonlinear relationships. MLR is suitable for modeling the linear relationships among variables, but MLR cannot deal with nonlinear relationships. ARIMA can capture the linear relationships for time series data. GM is a supplement to unknown information. By combining these models, complex structures in the data can be modeled in a more accurate way and the forecasting accuracy can often be improved as compared with the individual model. In addition, the combined model is more robust with regard to the possible structure change in data. We conduct experiments by using the dataset of emergency calls of Ningbo (a 8 million population city of China). Our experimental results demonstrate remarkable improvements on forecasting performance in comparison with the results obtained by such existing methods as BPNN, LR and Poisson Regression models [8].

The paper is organized as follows: Section II discusses the related work. Section II describes the framework of this paper. Section IV describes the PNN model and the combined model. Section V gives the experimental results by using Ningbo dataset. Finally, Section VI concludes the paper.

II. RELATED WORK

Building realistic forecasting models is not simple, because the arrival rates of emergency call are not only stochastic and time-dependent, but also often affected by external events. Much related work has been done.

Mielczarek [27] has studied the influence of changes in population size and structure on the volume of emergency service needs exhibited by patients arriving at hospital emergency departments in the area. He uses a Monte Carlo simulation model to examine demographic trends at the regional level, formulates forecasts for population changes, and extrapolated the simulated patterns of the demand for acute services.

Taylor [38] evaluated univariate time series methods for forecasting intraday arrivals for lead time from one half-hour ahead to two weeks ahead. They include seasonal ARIMA modeling, periodic AR modeling, Holt-Winters exponential smoothing, robust exponential smoothing based on exponentially weighted least absolute deviations regression, and dynamic harmonic regression.

Channouf *et al.* [7] developed and evaluate time-series models of call volume to the emergency medical service of a major Canadian city. They use three basic approaches for daily call volumes: standard regression, regression models with correlated residuals, and doubly-seasonal ARIMA. The first deterministic part captures the seasonal and non-seasonal components, and the second stochastic part (errors) captures the effect of omitted or nonobservable effects such as serial correlation.

Ibrahim *et al.* [16] reviewed and discussed the key issues in building statistical models for the call arrival process in telephone call centers. Then they survey and compared various types of models proposed so far. These models are used both for simulating and forecasting incoming call volumes to make staffing decisions and build (or update) work schedules for agents who answer those calls. Ibrahim *et al.* [15] review the existing literature on modeling and forecasting call arrivals, and discuss the key issues for building good statistical arrival models. In addition, the forecasting accuracy of selected models is evaluated in an empirical study with real-life call center data.

Jiang *et al.* [17] proposed a hybrid approach for forecasting the demand of the outpatient department in hospitals. In this approach, a feedforward deep neural network is applied to do the forecasting task, and a modified genetic algorithm is used for feature selection.

Matteson *et al.* [28] introduced a new method for forecasting emergency call arrival rates, which combines integer-valued time series models with a dynamic latent factor structure. Covariate information is captured via simple constraints on factor loadings. They consider all emergency priority calls received by Toronto EMS between January 2007 and December 2008. To quantify the impact of reduced forecast errors, they design a queueing model simulation that approximates the dynamics of an ambulance system.

In this paper, we focus on a Neural Network model, Poisson Regression model and their combined model.

A. NEURAL NETWORK

Neural network (NN) [12] is an information processing paradigm that is inspired by biological systems. The basic unit of NN is a neuron. It receives one or more inputs and then uses an activation function and the sum of its inputs to produce an output. Generally, NN consists of input layer, middle layer(s) or hidden layer(s) and output layer, each of which is made up of a number of interconnected neurons. The number of neurons in the input layer (namely, the input neurons) is determined by the dimension of an input vector, and the neurons in the output layer (namely, the output neurons) represent the result delivered by NN. The neurons in hidden layer(s) are not part of input and output, but they connect input neurons and also link to output neurons.

NN can be trained by different learning algorithms. In this study, we focus on the backpropgation algorithm [12], i.e., a backpropagation neural network (BPNN). In a BPNN, neurons of the hidden and output layers are designed to perform the following two kinds of calculation tasks: (1) calculating the function signal that appears at the output of a neuron, which appears as a continuous nonlinear function of the input signal and the synaptic weights associated with the neuron; and (2) estimating the gradient vector, which needs to go back through the network.

B. POISSON REGRESSION MODEL

Poisson distribution is a discrete probability distribution, which can be used to express the probability of a given number of events occurring in a fixed interval. Let λ denote the event rate, i.e., the average number of events occurring in the given time interval. An event Y is said to have a Poisson distribution with parameter λ if the probability of observing k occurrences of Y in an interval can be described by the following distribution function:

$$P(Y = k) = \frac{\lambda^k}{k!}e^{-\lambda}, \quad (k = 0, 1, 2, \ldots). \tag{1}$$

In other words, the mean and variance of this distribution can be described as

$$E(Y) = \lambda. \tag{2}$$

Poisson regression model [8] is a regression analysis model used for modeling technical data and association tables. Poisson regression assumes that the event rate λ is determined by a set of regressor variables, e.g., $a_1, a_2, \ldots,$ and a_n $(n \geq 1)$. The relationship between them can be expressed as

$$\ln(\lambda) = \theta_0 + \theta_1 a_1 + \cdots + \theta_n a_n. \tag{3}$$

Therefore, a Poisson regression model is also called a log-linear model. Let $\Theta = (\theta_0, \theta_1, \theta_2 \ldots, \theta_n)$ and $\mathbf{A} = (1, a_1, a_2, \ldots, a_n)$, Eq. (3) can be written as

$$\ln(\lambda) = \Theta\mathbf{A}^T. \tag{4}$$

C. COMBINATIONED MODEL

Recently, a common practice to improve the forecasting accuracy is to use hybrid models or combine several models. Each kind of prediction models has its unique characteristics, which can be useful for model analysis of a certain category of data. Therefore, different models are good at analysing different data sets. As a result, we are able to obtain more accurate forecasting results by properly combining different models. Many researchers have observed this point. For example, the experimental results [14] show that the combined model is often of better performance than the best single one. Yu and Kim have proved that a combined model can significantly improve the accuracy of the prediction result of each individual model [43]. The combinational model is also used to predict the linear or non-linear combination of the results from different prediction models [41].

Obviously, a combined model stands for an important category of forecasting models. Currently, there are several successful combined models, such as the well known boosting, bagging, random forest algorithm, time series neural network model [45], and gray neural network model [6], [42]. Their advantages can be summarized as follows:

i They can effectively make improvements on forecasting results, because they fully take advantage of the information expressed by different individual models.
ii It can leverage the randomness of errors and can thus reduce the error rate of forecasting results.

Currently, the combinational forecasting technologies can be implemented in the following ways: 1) Training several prediction models, from which the model with the smallest error is chosen as the predictive model; 2) Using the weighted average of the prediction results from multiple predictive models; 3) Combining two or more models to produce a new model or an enhanced model with complementary advantages. In this paper, we follow the second way to build prediction models for forecasting emergence calls.

III. PROPOSED FRAMEWORK

In this section, we present the framework of our method for forecasting emergency calls. Basically, our method makes use of NN and a combined residual error forecasting model.

Because of its characteristics, Poisson regression [35] can be used to model the number of emergency calls per day. However, its model can only fit the linear relationship but ignores the non-linear relationship among variables. Yet, this can be complemented by NN, because NN is suitable for fitting any nonlinear relation. Therefore, we propose a Poisson neural network (PNN) model based on Poisson distribution by combining a Poisson Regression model with an NN model. In this study, the PNN is first trained by some existing emergency call data, and then the established model is used for forecasting.

On the other hand, in order to improve the accuracy of predication results, the results from PNN should be corrected by a combined model [41]. In this paper, the latter is composed of three models, that is, multiple linear regression (MLR) [32], autoregressive integrated moving average(ARIMA) [5] and multi-gray model (GM) [37]. MLR is a regression model that can adjust the relationship between the dependent variables and the linear combination of independent variables. ARIMA is a time series forecasting model, which is suitable for analysing time series data, and is helpful for better understanding the data or predicting future points in the series forecasting. ARIMA can be expressed as ARIMA(p, d, q), where parameters p, d, and q are non-negative integers. p denotes the order (number of time lags) of the autoregressive model, d represents the degree of differencing (the number of times the data have had past values subtracted), and q is the order of the moving-average model [25]. Therefore, the ARIMA model is applied in some cases where data show evidence of non-stationarity. GM is a forecasting method for systems with uncertain factors. This combined model is used to conduct prediction on the residual error of the result from PNN, which can take advantage of the strength of MLR, GM and ARIMA.

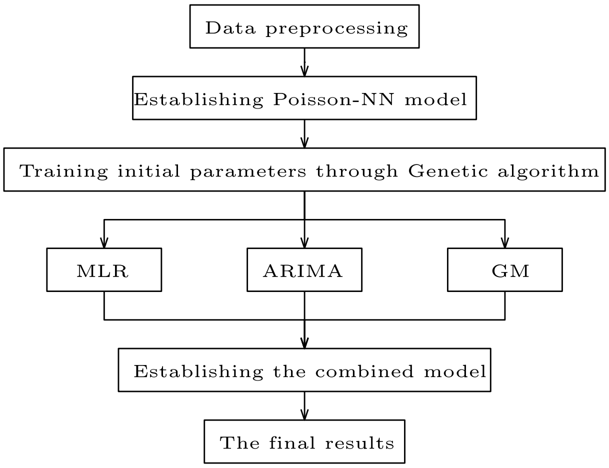

FIGURE 1. The workflow of the proposed prediction method.

The workflow of our forecasting method is presented in Fig. 1. At first, the original data is preprocessed, including the detection of anomaly values, normalization, and filling all missing values. The processed data is then fed into the proposed PNN model to produce the prediction value, the residual error of which is further corrected by a combinational model, which consists of MLR, ARIMA and GM. Finally, the sum of the results of PNN and the residual result predicted by the combined model is taken as the final prediction result.

IV. POISSON NEURAL NETWORK AND THE COMBINED MODEL

In this section, we present a PNN model, which is built upon the traditional NN and Poisson Regression model. Then we introduce some basic information about the NN and regression model, and show how to design a PNN model.

A. POISSON NEURAL NETWORK

We combine a Poisson regression model and NN model to produce a PNN model. PNN assumes that the predicted value follows the Poisson distribution, which is consistent with the nature of emergency calls. To this end, it uses the same structure as the ordinary NN, but adopts the exponential activation function $f(x) = e^x$, where x represents the input. This type of activation function has been widely used in probabilistic neural networks [33]. And the output for the layer can be computed by $out = f(w(x) + b)$,where x,w and b represent the output of the previous layer, weight and bias, respectively. Moreover, in this study, we apply a backpropagation neural network.

We first deduce the objective function of PNN by applying the maximum likelihood analysis. Suppose that there are m sample data, the i^{th} $(1 \leq i \leq m)$ sample can be expressed as (x^i, y^i), where x^i is the input data with n $(n \geq 1)$ dimensions, and y^i is the actual output value. The probability density function of one sample can be obtained via Poisson distribution, as follows:

$$p(y^i|x^i) = \frac{f_i^{y^i}}{y^i!}e^{-f_i}. \tag{5}$$

Then the probability density function of the whole sample set (namely, the likelihood function), can be obtained by multiplying the probability of each sample as

$$P = \prod_{i=1}^{m} p(y^i|x^i) = \prod_{i=1}^{m} \frac{f_i^{y^i}}{y^i!}e^{-f_i}. \tag{6}$$

Based on Eq. (6), the Logarithmic likelihood function can be expressed as follows:

$$\begin{aligned} \ln(P) &= \ln \prod_{i=1}^{m} \frac{f_i^{y^i}}{y^i!}e^{-f_i} = \sum_{i=1}^{m} \ln \frac{f_i^{y^i}}{y^i!}e^{-f_i} \\ &= \sum_{i=1}^{m} (\ln f_i^{y^i} + \ln(e^{-f_i}) - \ln(y^i!)) \\ &= \sum_{i=1}^{m} (y^i \ln f_i - f_i - ln(y^i!)). \end{aligned} \tag{7}$$

The goal is to maximize the probability of the sample set, or equivalently to find the minimal value of $-\ln(P)$. For a given training sample set, y^i is already known. Thus, the final objective function is

$$\min_{w,b} : L = -\ln(P) = \sum_{i=1}^{m} (f_i - y^i \ln f_i). \tag{8}$$

The parameter values of Eq. (8) can be obtained by using a gradient descent algorithm.

B. INITIALIZATION OF POISSON NEURAL NETWORK

The gradient descent algorithm is an iterative algorithm that must start with an initial value. The initial value has a great impact on the convergence of the network, and a poor initial value may make the network fall into local minimum. The problem is more serious when out activation function is applied to the output layer. To alleviate this problem, in this study, we apply a the Genetic algorithm (GA). GA has the ability to perform both global and local search, which is particularly suitable for dealing with complex and nonlinear problems. GA can often be used to optimize a neural network [22], [26]. So in this paper, We apply GA [9] to encode the parameters of a neural network, in order to find relatively optimal initial parameters. The detailed process is shown in Fig. 2.

In this paper, all the weights and thresholds of our PNN are encoded as a chromosome, that is, a chromosome can determine a specific network. The fitness function can be written as follow:

$$F = \frac{1}{\sum_{i=1}^{m} (y^i \ln f_i - f_i)}. \tag{9}$$

The population size is set to 20. Then the first generation population can be obtained by random initialization. 20 different PNN weights and thresholds are obtained, and the

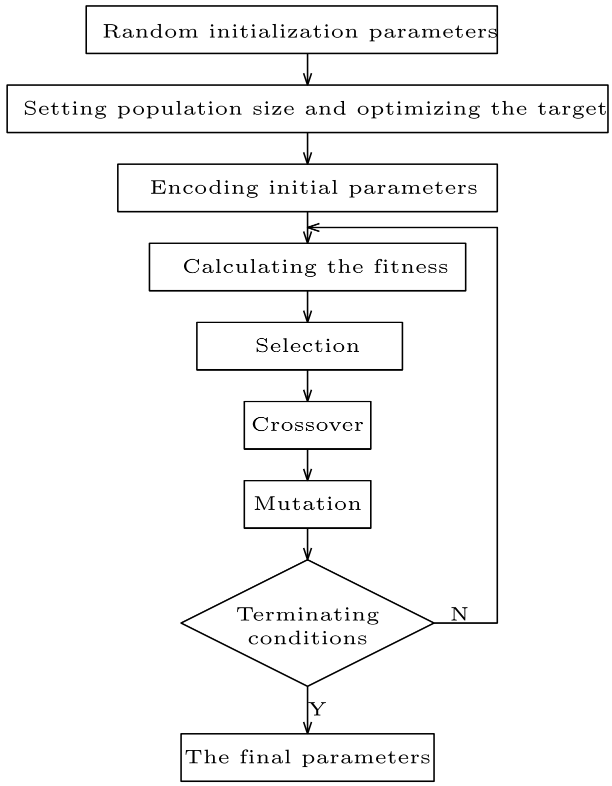

FIGURE 2. The process of genetic algorithm.

respective fitness can be calculated. Some individuals with large fitness values can be selected directly into the next generation. Then the next generation population can be produced by using crossover, mutation and other genetic operators to the current generation population. Repeat those steps until the target error or maximum iteration count is reached. A set of optimized weights can be obtained. The highest initial value is chosen as the final initial parameter of PNN. These initial parameters are then used to train the network.

C. COMBINED RESIDUAL MODEL

With the goal to improve the accuracy of the result produced by PNN and motivated by [41], we further propose the combined model. Our combined model consists of three different models, i.e., MLR, ARIMA and GM, which are used together to predict the residual error of the PNN. The residual error of the i^{th} sample can be written as:

$$\xi(i) = O(i) - P(i). \tag{10}$$

O(i) represents the true value of the original data and *P(i)* is the value predicted by PNN.

We apply each model to produce the relevant residual error, and finally the weighted average of residual errors from the three models are calculated. For example, suppose that the i^{th} sample predicted by MLR, ARIMA, and GM are $V_1(i)$, $V_2(i)$, and $V_3(i)$. Then, the final residual error is

$$\xi'(i) = \sum_{j=1}^{3} \omega_j V_j(i). \tag{11}$$

In Eq. (11), $\omega_1 - \omega_3$ are the weights corresponding to these three models, and $\omega_1, \omega_2, \omega_3$ satisfying:

$$\begin{cases} \omega_1 + \omega_2 + \omega_3 = 1 \\ \omega_1 \geq 0, \omega_2 \geq 0, \omega_3 \geq 0. \end{cases} \tag{12}$$

The final objective function of the residual prediction combined model can be written as:

$$min : L' = \frac{1}{2} \sum_{i=1}^{m} (\xi(i) - \xi'(i))^2. \tag{13}$$

The final predicted result is the sum of the results predicted by PNN and the result of the combined model.

V. EXPERIMENTAL STUDY

A. DATA PREPROCESSING

In the experiments, we use the data describing the emergency calls of Ningbo in 2011. The data set contains 365 samples, each of which records the information of emergency calls occurring on one day in 2011. The sample data includes input variables and output value. The input variables contain two kinds of variables respectively describing *Time* and *Weather*. Specifically, the *Time* information includes date, month and week information, while *Weather* includes the minimum, maximum temperature, wind power and the weather conditions. The output value represents the number of emergency calls requested at the relevant time point. Additionally, in order to explore the inner relationship among the adjacent days. we also choose the data of its first 6 days as input variables for every day.

For example, the first training sample is (x^1, y^1), where x^1 is the input data and y^1 is the output data, and:

$$x^1 = (7, 1, 5, 7, 1, 36, 6, 51, 57, 58, 55, 49, 71, 51).$$
$$y^1 = 51.$$

This sample data indicates that, in the first training sample, the date is 7, month is January, week is Friday, the maximum and minimum temperature is 7 degrees and 1 degree, the wind power is 36 km/h, the weather conditions is sleet and the emergency calls of the immediately past 6 days are 51, 57, 58, 55, 49 and 71. The last number 51 is the number of emergency calls on January 7.

The data distribution of emergency calls is shown in Fig. 3.

In order to eliminate the influence of data dimension and speed up the calculation, it is necessary to standardize the data. The standardization and de-standardization methods adopted in this paper are as follows:

$$l' = \frac{l - min(l)}{max(l) - min(l)}. \tag{14}$$

$$l = min(l) + l' \cdot (max(l) - min(l)). \tag{15}$$

l denotes the original column vector, l' is the normalized vector of l, $min(l)$ is the minimum value of the column vector, and $max(l)$ is the maximum value of the column vector.

The combined model contains ARIMA that is a time series model. Thus, we divide the samples into two sets in an

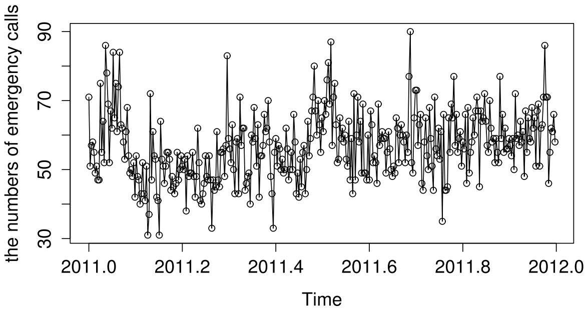

FIGURE 3. The data distribution of emergency calls, and the number of emergency calls per day is recorded.

ascending order according to the time: the former two-thirds of samples are used as training data for building the PNN model and the combined model, and the latter one-third of samples are used as the validation data for evaluating the performance of the proposed model.

B. EVALUATION METHOD

In order to properly evaluate the performance of the proposed model, we adopt Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). In this paper, RMSE used to measure the difference between predicted and true values. MAPE is the ratio of the absolute error to the true value.

For the i^{th} sample, let T_i and P_i respectively denote its actual value and the predicted value from the model. They can be calculated as

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m}(T_i - P_i)^2}{m}}. \tag{16}$$

$$MAPE = \frac{100}{m}\sum_{i=1}^{m}|\frac{T_i - P_i}{T_i}|. \tag{17}$$

C. CONFIGURATION OF NEURAL NETWORK

As mention in Section II-A, a neural network includes input layer, output layer and hidden layer(s), where for each layer, the number of neurons are crucial to the NN performance. According to our data presented in Section V-A, there are 13 input variables and one ouput variable. Hence, our NN is configured to contain 13 input neurons and one output neuron. The structure of our NN is shown in Fig. 4, where $x_j(j = 1, 2, \cdots, 13)$ denotes the j^{th} feature of the training sample x, y represents the output value of the NN.

It has been proved that a multi-layer feedforward network containing only a hidden layer of enough neurons can be approximated to any complex continuous function with arbitrary precision [13]. In order to avoid over-fitting, we choose the NN with a single hidden layer. The problem of setting the number of hidden neurons has not yet been solved. Many researchers have studied how to choose the number of hidden layer neurons. One method to develop constructive approaches for building neural networks, starting with a small

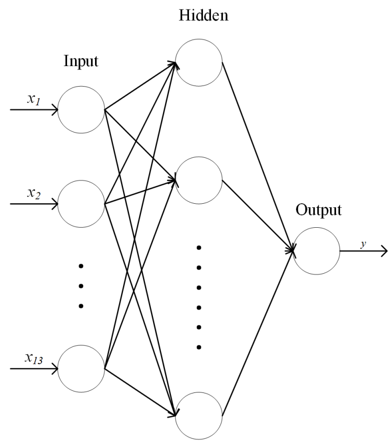

FIGURE 4. The structure of neural network.

size network, followed by incrementally generating hidden nodes and output weights until a pre-defined termination criterion is met. Barron proposes a greedy learning framework based on the work in [4], and establishes some significant results on the convergence rate. Kwok and Yeung [18] present a method to construct neural networks through optimizing some objective functions. Additionally, randomized approaches [11], [21], [31], [34] have been proposed to solve the problem, which randomly assign the input weights and biases and evaluate the output weights by some methods. Wang and Li [40] propose Stochastic Configuration Networks (SCNs). They randomly assign the input weights and biases of the hidden nodes in the light of a supervisory mechanism, and the output weights are analytically evaluated in either constructive or selective manner. In this paper, we use a trial and error method to determine the number of hidden layer neurons. Finally, the number of neurons in the hidden layer is set to 20.

D. EXPERIMENTAL RESULTS AND ANALYSIS

We can get the initial parameters of our PNN from GA, and then the gradient descent algorithm is applied to train the network in the training data set. The trained PNN is then used for prediction. For each predication, the residual error can be calculated via Eq. (10). Then, a new data set is created by combining the residuals and variables of the original training data set, which is used to train the combined model. Then the whole model is evaluated on the test set.

The trained PNN and the combined model are finally used together to conduct the predication task. That is, the sum of the result predicted by PNN and that of the combined model is taken as the final prediction result.

In this study, we set the learning rate as 0.01. The number of hidden neurons is set to 20, and the PNN has only one hidden layer. Then our model is applied to the validation data, and the final prediction results are shown in Fig. 5. Fig. 6 shows the relative error and Fig. 7 shows the absolute error.

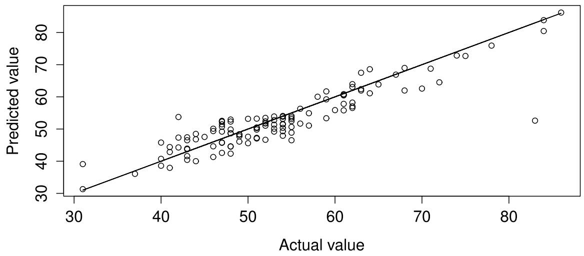

FIGURE 5. The prediction results.

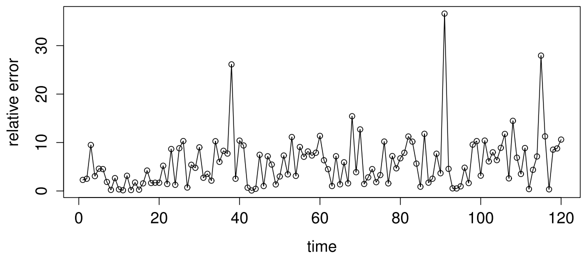

FIGURE 6. The relative error.

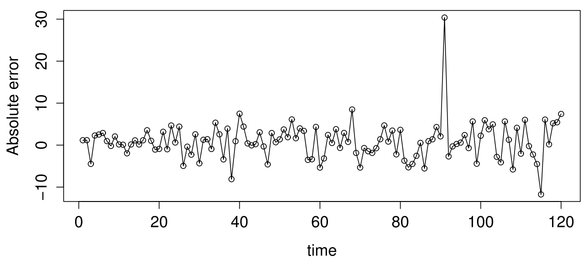

FIGURE 7. The absolute error.

TABLE 1. The MAPE and RMSE of different models.

model	MAPE %	RMSE %
BPNN	10.30	8.11
PR	10.19	7.78
ARIMA	12.02	9.09
MLR	14.5	9.64
GM	11.2	8.23
SC-I	9.73	7.45
PNN combined model	**7.44**	**6.46**

After obtaining the prediction results, the relevant RMSE and MAPE can be calculated via Eqs. (16)- (17). In order to evaluate the performance of the proposed model, we respectively apply BPNN, PR, GM, ARIMA, MLR and the proposed model on the same data set, and then compare their prediction results. Especially, we also compare our model with SC-I proposed by Wang and Li [40].

Table 1 presents the forecasting performance of each model in the terms of MAPE and RMSE. The parameters of ARIMA are set as ARIMA(0,1,1). In SC-I, the maximum number of times of random configuration T_{max} is set as 200, the maximum number of hidden nodes L_{max} is set as 50. The structure of BPNN is set as 13 × 20 × 1. Table 1 shows that SC-I gives slightly better forecasts than BPNN, GM, MLR, ARIMA and PR models. Our PNN combined model has an 25.54% (27.39%) decrease in RMSE over BPNN (GM) and an 49.23% (40.71%) decrease over MLR (ARIMA). With MAPE, the improvement of the PNN combined model over the BPNN, PR, ARIMA, MLR, GM and SC-I are 38.44%, 36.96%, 61.56%, 94.89%, 50.53% and 30.77%, respectively. Additionally, we find that SC-I can achieve a relatively high accuracy by determining the number of hidden neurons automatically. The results show that among the selected models, our model is superior to others. This suggests that our combined model can fully take advantage of the strength of its sub-model and thus improves the forecasting performance.

VI. CONCLUSION AND FUTURE WORK

In this paper, a new Poisson neural network model is proposed to forecast the number of emergency calls in the city of Ningbo. In order to improve the forecasting accuracy, a combined model consisting of a multiple linear regression model, autoregressive integrated moving average model and multiple gray model is proposed to forecast the residual error. From the experimental results, we see that the proposed model can effectively improve the prediction results.

Due to the lack of more accurate address information, this work can only predict the number of emergency calls for the whole Ningbo city without knowing where these calls belong to. In the future, by obtaining detailed call address information and specific time information, we can partition Ningbo city into a number of smaller districts and establish specific prediction model for each district. Then the numbers of calls from each district at a specific time period can be predicted. Moreover, the internal links among the partitions can be used to improve the accuracy of the model forecasting.

REFERENCES

[1] J. D. Andrews and S. J. Dunnett, "Event-tree analysis using binary decision diagrams," *IEEE Trans. Rel.*, vol. 49, no. 2, pp. 230–238, Jun. 2000.

[2] J. S. Armstrong, F. Collopy, and J. T. Yokum, "Decomposition by causal forces: A procedure for forecasting complex time series," *Int. J. Forecasting*, vol. 21, no. 1, pp. 25–36, 2005.

[3] M. Abdel-Aty and H. Abdelwahab, "Analysis and prediction of traffic fatalities resulting from angle collisions including the effect of vehicles' configuration and compatibility," *Accident Anal. Prevention*, vol. 36, no. 3, pp. 457–469, 2004.

[4] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, May 1993.

[5] L. Carnis and E. Blais, "An assessment of the safety effects of the French speed camera program," *Accident Anal. Prevention*, vol. 51, no. 1, pp. 301–309, 2013.

[6] P.-C. Chang, Y.-W. Wang, and C.-Y. Tsai, "Evolving neural network for printed circuit board sales forecasting," *Expert Syst. Appl.*, vol. 29, no. 1, pp. 83–92, 2005.

[7] N. Channouf, P. L'Ecuyer, A. Ingolfsson, and A. N. Avramidis, "The application of forecasting techniques to modeling emergency medical system calls in Calgary, Alberta," *Health Care Manage. Sci.*, vol. 10, no. 1, pp. 25–45, 2007.

[8] M. Cupal, O. Deev, and D. Linnertova, "The Poisson regression analysis for occurrence of floods," in *Proc. 2nd Global Conf. Bus., Econ., Manage. Tourism*, Prague, Czech Republic, Oct. 2014.

[9] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA, USA: Addison-Wesley, 1989.

[10] M. A. Gebers and R. C. Peck, "Using traffic conviction correlates to identify high accident-risk drivers," *Accident Anal. Prevention*, vol. 35, no. 6, pp. 903–912, 2003.

[11] A. N. Gorban, I. Y. Tyukin, D. V. Prokhorov, and K. I. Sofeikov, "Approximation with random bases: Pro et Contra," *Inf. Sci.*, vol. 364, pp. 129–145, Oct. 2016.

[12] R. Hecht-Nielsen, "Theory of the backpropagation neural network," in *Neural Networks for Perception*. Amsterdam, The Netherlands: Elsevier, 1992, pp. 65–93.

[13] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989.

[14] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 10, pp. 993–1001, Oct. 1990.

[15] R. Ibrahim, H. Yeb, P. L'Ecuyer, and H. Shen, "Modeling and forecasting call center arrivals: A literature survey and a case study," *Int. J. Forecasting*, vol. 32, no. 3, pp. 865–874, 2016.

[16] R. Ibrahim, N. Regnard, P. L'Ecuyer, and H. Shen, "On the modeling and forecasting of call center arrivals," in *Proc. Winter Simulation Conf. (WSC)*, Dec. 2012, pp. 1–12.

[17] S. Jiang, K.-S. Chin, L. Wang, G. Qu, and K. L. Tsui, "Modified genetic algorithm-based feature selection combined with pre-trained deep neural network for demand forecasting in outpatient department," *Expert Syst. Appl.*, vol. 82, pp. 216–230, Oct. 2017.

[18] T.-Y. Kwok and D.-Y. Yeung, "Objective functions for training new hidden units in constructive neural networks," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 1131–1148, Sep. 1997.

[19] F. I. Khan and S. A. Abbasi, "An assessment of the likelihood of occurrence, and the damage potential of domino effect (chain of accidents) in a typical cluster of industries" *J. Loss Prevention Process Ind.*, vol. 14, no. 4, pp. 283–306, 2001.

[20] H. Kantz, D. Holstein, M. Ragwitz, and N. K. Vitanov, "Markov chain model for turbulent wind speed data," *Phys. A, Stat. Mech. Appl.*, vol. 341, nos. 1–2, pp. 315–321, 2004.

[21] D. S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks," *Complex Syst.*, vol. 2, no. 3, pp. 321–355, 1988.

[22] P. D. Loeb and W. A. Clarke, "The determinants of truck accidents," *Transp. Res. E, Logistics Transp. Rev.*, vol. 43, no. 4, pp. 442–452, 2007.

[23] C.-J. Lee and K. J. Lee, "Application of Bayesian network to the probabilistic risk assessment of nuclear waste disposal," *Rel. Eng. Syst. Saf.*, vol. 91, no. 5, pp. 515–5321, 2006.

[24] M. Z. Mao and E. C. Chirwa, "Application of grey model GM(1, 1) to vehicle fatality risk estimation," *Technol. Forecasting Social Change*, vol. 73, no. 5, pp. 588–605, 2006.

[25] T. C. Mills, *Time Series Techniques for Economists*. Cambridge, U.K.: Cambridge Univ. Press, 1991.

[26] D. M. Miller and D. Williams, "Shrinkage estimators of time series seasonal factors and their effect on forecasting accuracy," *Int. J. Forecasting*, vol. 19, no. 4, pp. 669–684, 2003.

[27] B. Mielczarek, "Estimating future demand for hospital emergency services at the regional level," in *Proc. Winter Simulation Conf.*, Dec. 2013, pp. 2386–2397.

[28] D. S. Matteson, M. W. McLean, D. B. Woodard, and S. G. Henderson, "Forecasting emergency medical service call arrival rates," *Ann. Appl. Statist.*, vol. 5, no. 2B, pp. 1379–1406, 2011.

[29] Z. S. Nivolianitou, V. N. Leopoulos, and M. Konstantinidou, "Comparison of techniques for accident scenario analysis in hazardous systems," *J. Loss Prevention Process Ind.*, vol. 17, no. 6, pp. 467–475, 2004.

[30] V. Nicoletta, E. Lanzarone, A. Guglielmi, V. Bélanger, and A. Ruiz, "A Bayesian model for describing and predicting the stochastic demand of emergency calls," in *Proc. Int. Conf. Bayesian Statist. Action*, in Proceedings in Mathematics & Statistics, vol. 194. Springer, 2016, pp. 203–212.

[31] Y.-H. Pao, G.-H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector functional-link net," *Neurocomputing*, vol. 6, no. 2, pp. 163–180, 1994.

[32] K. J. Preacher, P. J. Curran, and D. J. Bauer, "Computational tools for probing interactions in multiple linear regression, multilevel modeling, and latent curve analysis," *J. Educ. Behav. Statist.*, vol. 31, no. 4, pp. 437–448, 2006.

[33] D. F. Specht, "Probabilistic neural networks," *Neural Netw.*, vol. 3, no. 1, pp. 109–118, 1990.

[34] W. F. Schmidt, M. A. Kraaijveld, and R. P. W. Duin, "Feedforward neural networks with random weights," in *Proc. Int. Conf. Pattern Recognit. Methodol. Syst.*, Aug./Sep. 1992, pp. 1–4.

[35] Z. Sawalha and T. Sayed, "Traffic accident modeling: Some statistical issues," *Can. J. Civil Eng.*, vol. 33, no. 9, pp. 1115–1124, 2006.

[36] M.-H. Shu, C.-H. Cheng, and J.-R. Chang, "Using intuitionistic fuzzy sets for fault-tree analysis on printed circuit board assembly," *Microelectron. Rel.*, vol. 46, no. 12, pp. 2139–2148, 2006.

[37] F.-M. Tseng, H.-C. Yu, and G.-H. Tzeng, "Applied hybrid grey model to forecast seasonal time series," *Technol. Forecasting Social Change*, vol. 67, nos. 2–3, pp. 291–302, 2001.

[38] J. W. Taylor, "A comparison of univariate time series methods for forecasting intraday arrivals at a call center," *Manage. Sci.*, vol. 54, no. 2, pp. 253–265, 2008.

[39] D. Vernez, D. Buchs, and G. Pierrehumbert, "Perspectives in the use of coloured Petri nets for risk analysis and accident modelling," *Saf. Sci.*, vol. 41, no. 5, pp. 445–463, 2003.

[40] D. Wang and M. Li, "Stochastic configuration networks: Fundamentals and algorithms," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3466–3479, Oct. 2017.

[41] Y. Yao, Z. Lian, Z. Hou, and W. Liu, "An innovative air-conditioning load forecasting model based on RBF neural network and combined residual error correction," *Int. J. Refrig.*, vol. 29, no. 4, pp. 528–538, 2006.

[42] X. G. Yin and W. D. Yu, "The virtual manufacturing model of the worsted yarn based on artificial neural networks and grey theory," *Appl. Math. Comput.*, vol. 185, no. 1, pp. 322–332, 2007.

[43] S. Yu, J. Kim, H. Min, and S. Yoon, "Ensemble learning can significantly improve human microRNA target prediction," *Methods*, vol. 69, no. 3, pp. 220–229, 2014.

[44] X. L. Zhang, Z. Y. Sun, J. H. Wang, and S. X. Lu, "Applying Markov chain to forecast the scale of cities fire," *Fire Saf. Sci.*, vol. 15, no. 3, pp. 168–171, Jul. 2006.

[45] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003.

HONGYUN HUANG received the M.S degree in software engineering from Zhejiang Sci-Tech University, where she is currently a Data Analyzer with the Center of Multi-Media Big Data of Library. Her current research interests include data analysis, system modeling, and AI computing. She has authored and co-authored over six papers on the above areas.

MINGYUE JIANG received the B.Sc. degree in computer science and technology from Yunnan Normal University, the M.Eng. degree in computer applied technology from Zhejiang Sci-Tech University, and the Ph.D. degree from the Swinburne University of Technology, Australia. Her current research interests include software testing and automated program repair.

ZUOHUA DING (M'11) received the M.S. degree in computer science and the Ph.D. degree in mathematics from the University of South Florida, Tampa, FL, USA, in 1996 and 1998, respectively.

He is currently a Professor and the Director of the Laboratory of Intelligent Computing and Software Engineering, Zhejiang Sci-Tech University, Hangzhou, China. He has authored and co-authored over 70 papers. His current research interests include system modeling, software reliability prediction, intelligent software systems, and service robots.

MENGCHU ZHOU (S'88–M'90–SM'93–F'03) received the B.S. degree in control engineering from the Nanjing University of Science and Technology, Nanjing, China, in 1983, the M.S. degree in automatic control from the Beijing Institute of Technology, Beijing, China, in 1986, and the Ph.D. degree in computer and systems engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990.

He joined the New Jersey Institute of Technology (NJIT), Newark, NJ, USA, in 1990, where he is currently a Distinguished Professor of electrical and computer engineering. He has over 680 publications, including 12 books, over 360 journal papers (over 260 in the IEEE TRANSACTIONS), and 28 book chapters. His current research interests include petri nets, sensor networks, web services, big data, semiconductor manufacturing, transportation, and energy systems.

Dr. Zhou is a Life Member of the *Chinese Association for Science and Technology*, USA, and served as its President, in 1999. He is a fellow of IFAC and AAAS. He was a recipient of the Perlis Research Award and the Fenster Innovation in Engineering Education Award from NJIT, the Humboldt Research Award for U.S. Senior Scientists, the Leadership Award, the Academic Achievement Award from the Chinese Association for Science and Technology, USA, the Outstanding Contributions Award, the Distinguished Lecturership, the Franklin V. Taylor Memorial Award, the Norbert Wiener Award of the IEEE SMC Society, and the Distinguished Service Award from the IEEE Robotics and Automation Society. He is the Founding Editor of the IEEE Press Book Series on Systems Science and Engineering.