International Journal of Applied Science and Mathematics
Volume 8, Issue 3, ISSN (Online): 2394-2894

A Method of Cellular Base Station Traffic Prediction

Zhengyan Xie [1*] and Zhongqing Zhao [2]
[1] Department of Communication Engineering, Faculty of Engineering, Yanbian University, Yanji, China.
[2] Department of Communication Engineering, Faculty of Engineering, Yanbian University, Yanji, China.
*Corresponding author email id: xiezy2000@126.com

Date of publication (dd/mm/yyyy): 09/06/2021

***Abstract* – In this paper, the base station uplink and downlink traffic load is modeled. Based on the traffic data in several cells for a period of time, short-term and long-term traffic predictions are made utilizing gray prediction, LSTM model, and ARIMA model, respectively. The number of carrier frequencies needed by base stations in each time can be selected according to the prediction, and capacity of certain base station can be optimized in advance for the mobile communication operator. This enables mobile operators to rationalize the planning and design of base station capacity in advance.**

***Keywords* – Data Segmentation Statistics, Base Station Traffic, Gray Prediction, LSTM Model.**

I. Introduction

With the rapid development of cellular network, the traffic load problem for the base stations becomes increasingly serious. Due to the base station tidal phenomeno, the number of users rises or falls drastically at certain times of the day, and the number of carrier frequencies required to be allocated to base stations changes dramatically. With the large number of base stations, and high cost of physical expansion, it is important to predict the traffic load in advance, such that planning and layout design can be done properly.

In this paper, we first selected uplink and downlink traffic for a certain period of time in several cells as reference data and preprocessed them. Then we used an improved and optimized gray prediction model, and trained the model with the processed data. Utilizing the trained mode, we achieved dynamic prediction of hourly traffic changes. Moreover, we built an LSTM model for long-term traffic prediction, and by solving this model, we obtained the prediction of total daily upstream and downstream traffic for a certain time period in the future. The use of these two rationalized forecasts will provide an effective reference for mobile operators to rationalize the planning and design of base station capacity in advance.

Table 1. Symbol description table.

Symbolic Representation	Meaning Statement
$\lambda(k)$	Sequence Level Ratio
C	Residual variance ratio
p	Small error probability
$x^{(0)}$	Original data sequence
$x^{(1)}$	Generate sequences in one accumulation
x_t	Current input
h_t	Current Output

Symbolic Representation	Meaning Statement
C_t	Current cell status

II. Data Preprocessing

2.1. *Data Processing and Integration*

The first step of data pre-processing is to classify the data, which is achieved by calling the pandas library through Python, using different cells as the basis for classification, and filtering the statistics of the upstream and downstream traffic data of each cell. Due to a large amount of data, the following are the results for some cells from 0:00 to 6:00.

Table 2. Data results from 0:00 to 6:00.

Subdivision Number	0:00	1:00	2:00	3:00	4:00	5:00	6:00
uploadGB-1	0.3247	0.0705	0.0651	0.0326	0.0306	0.0284	0.0517
downloadGB-1	3.5607	1.1540	0.5718	0.4352	0.1000	0.0974	0.6969
uploadGB-2	0.1372	0.0355	0.0365	0.0229	0.0395	0.0177	0.0674
downloadGB-2	0.5368	0.8086	0.6957	0.6295	0.2144	0.1305	0.4768
uploadGB-3	0.0825	0.0699	0.0509	0.0406	0.0224	0.0347	0.0333
downloadGB-3	1.5374	0.4558	0.1574	0.1742	0.3417	0.1584	0.2185
uploadGB-4	0.0104	0.0030	0.0016	0.0064	0.0025	0.0013	0.0105
downloadGB-4	0.2812	0.0319	0.0037	0.0273	0.0072	0.0145	0.0259
uploadGB-5	0.0156	0.0049	0.0032	0.0029	0.0035	0.0040	0.0106
downloadGB-5	0.5642	0.0407	0.0086	0.0072	0.0202	0.0168	0.1333
uploadGB-6	0.0258	0.0070	0.0058	0.0065	0.0040	0.0081	0.0095
downloadGB-6	0.0777	0.0457	0.0400	0.0246	0.0434	0.0327	0.1093
uploadGB-7	0.0156	0.0070	0.0086	0.0015	0.0017	0.0011	0.0068
downloadGB-7	0.0722	0.0202	0.0357	0.0041	0.0056	0.0042	0.0607
uploadGB-8	0.0027	0.0004	0.0004	0.0004	0.0004	0.0005	0.0037
downloadGB-8	0.0545	0.0057	0.0005	0.0006	0.0006	0.0006	0.0375
uploadGB-9	0.0102	0.0030	0.0013	0.0014	0.0008	0.0027	0.0051
downloadGB-9	0.0989	0.0345	0.0022	0.0058	0.0012	0.0097	0.0616

2.2. *Visualization and Analysis of Data*

In the following, we further analyze the correlation between the upstream and downstream traffic and the total traffic using the Pearson correlation analysis [1]. A preliminary visualization is shown for the correlation between the upstream and downstream traffic in the cell. In Figure 1, a simple correlation matrix is plotted for the upstream traffic, the downstream traffic, and the total traffic.

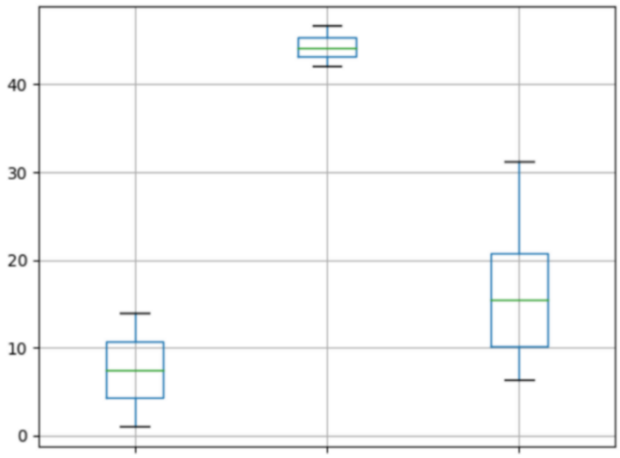

Fig. 1. Correlation matrix.

According to the correlation matrix, the upstream and downstream is highly correlated, and they both are correlated with the total traffic. In the following part, we analyze the correlation in detail.

First, we performed correlation tests based on the correlation coefficients to determine the strength of the correlation between the three factors and the regression variables.

The correlation coefficient, i.e. the degree of correlation between two things (in the data we call them variables). If there are two variables X and Y, the meaning of the final calculated correlation coefficient can be understood as follows.

- When the correlation coefficient is 0, the two variables X and Y are not related.
- When the value of X increases (decreases) and the value of Y increases (decreases), the two variables are positively correlated and the correlation coefficient is between 0.00 and 1.00.
- When the value of X increases (decreases) and the value of Y decreases (increases), the two variables are negatively correlated, with a correlation coefficient between -1.00 and 0.00.
- The larger the absolute value of the correlation coefficient, the stronger the correlation, the closer the correlation coefficient is to 1 or -1, the stronger the correlation, and the closer the correlation coefficient is to 0, the weaker the correlation. The correspondence table between the range of values of correlation coefficient and the strength of correlation of judged variables is as follows.

Table 3. Relationship between correlation coefficient and correlation strength.

Correlation Coefficient	**0.0-0.2**	**0.2-0.4**	**0.4-0.6**	**0.6-0.8**	**0.8-1.0**
Related Strength	Very weak or no correlation	Weak correlation	Moderate correlation	Strong Related	Extremely strong correlation

We used Pearson correlation analysis to calculate the correlation coefficients between the factors and the target variables [2]. The mathematical model is simply described as follows.

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E((X-\mu_X)(Y-\mu_Y))}{\sigma_X \sigma_Y} = \frac{E(XY)-E(X)E(Y)}{\sqrt{E(X^2)-E(X^2)}\sqrt{E(Y^2)-E(Y^2)}} \quad (1)$$

$$\rho_{X,Y} = \frac{N\sum XY-\sum X\sum Y}{\sqrt{N\sum X^2-(\sum X)^2}\sqrt{N\sum Y^2-(\sum Y)^2}} \quad (2)$$

$$\rho_{X,Y} = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{\left(\sum X^2 - \frac{(\sum X)^2}{N}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{N}\right)}} \tag{3}$$

Next, the correlations were further tested and analyzed using SPSS, and the results are shown in the following table.

Table 4. SPSS test and analysis of correlation.

Factors to be Judged	Correlation Coefficient	Related Types
Uplink traffic	0.436	Negative correlation
Downstream traffic	0.692	Positive correlation
Product Features	0.602	Positive correlation

The results of the analysis show that the correlation coefficient for the factor of upstream flow is 0.436, the correlation coefficient for downstream flow is 0.692, and the correlation coefficient for total flow is 0.602. Comparing the results of the above analysis with Table 2, we can conclude that there is a certain or even a strong, correlation between the three factors we analyzed.

Therefore, the correlation weights between flows need to be calculated in the modeling and used as a parameter in the model.

III. Short-Term Prediction

3.1. *Modeling*

3.1.1. *Basic Model*

The methods used for prediction include the interpolation and fitting, regression analysis, gray prediction algorithm, neural network, and time series analysis. The gray prediction method has the advantages of easy operation and high modeling accuracy, but the accuracy of the common gray prediction method is usually low for large sample data. Therefore, we choose to use the improved and optimized $GM(2)$ method as the solution for short-term forecasting.

Before proceeding with the modeling, we need to perform the necessary tests on the known data series in order to ensure the feasibility of the model.

Let the reference data be $x^{(0)} = \{x^{(0)}(1), x^{(0)}(2), x^{(0)}(3), \cdots, x^{(0)}(n)\}, i = 1,2,\cdots,n$, and calculate the level ratio of the sequence.

$$\lambda(k) = \frac{x^{(0)}(k-1)}{x^{(0)}(k)}, k = 2,3,\cdots,n \tag{4}$$

If all the ratios $\boldsymbol{\lambda(k)}$ fall in the tolerable coverage $\boldsymbol{\theta} = (\boldsymbol{e}^{-\frac{2}{n+1}}, e^{\frac{2}{n+2}})$, the series $\boldsymbol{x}^{(0)}$ can be used as data for the model for gray prediction. Otherwise, the sequences need to be transformed as necessary to make them fall within the tolerable coverage. This is done by taking the appropriate constants and making a translational transformation.

$\boldsymbol{y}^{(0)}(\boldsymbol{k}) = \boldsymbol{x}^{(0)}(\boldsymbol{k}) + \boldsymbol{c}, \boldsymbol{k} = \boldsymbol{1}, \boldsymbol{2}, \cdots, \boldsymbol{n}$ such that the sequence $\boldsymbol{y}^{(0)} = \{\boldsymbol{y}^{(0)}(\boldsymbol{1}), \boldsymbol{y}^{(0)}(\boldsymbol{2}), \boldsymbol{y}^{(0)}(\boldsymbol{3}), \cdots, \boldsymbol{y}^{(0)}(\boldsymbol{n})\}$

with step ratios $\boldsymbol{\lambda_y(k)} = \frac{y^{(0)}(k-1)}{y^{(0)}(k)} \in \boldsymbol{\theta}, \boldsymbol{k} = \mathbf{2}, \mathbf{3}, \cdots, \boldsymbol{n}$ fall in the tolerable coverage.

The gray prediction model $\boldsymbol{GM}(\mathbf{1}, \mathbf{1})$ reflects a first-order differential function of a variable with respect to time, and its corresponding differential equation (i.e., the whitening equation) is:

$$\frac{dx^{(1)}}{dt} + ax^{(1)} = u \tag{5}$$

where $x^{(1)}$ is the sequence generated after one accumulation;t is the time; a,u is the parameter to be estimated and is called the developmental gray number and the endogenous control gray number, respectively. The gray modeling is performed as follows.

(1) Create a cumulative generation sequence. Denote the original sequence as: $x^{(0)} = \{x^{(0)}(1), x^{(0)}(2), x^{(0)}(3), \cdots, x^{(0)}(n)\}, i = 1, 2, \cdots, n$

A cumulative generation process is performed as follows to obtain a cumulative generation sequence: (n is the sample space).

$$x^{(1)}(i) = \sum_{m=1}^{i} x^{(0)}\,(m) i = 1, 2, \cdots, n, \tag{6}$$

i.e. $x^{(1)} = \{x^{(1)}(1), x^{(1)}(2), x^{(1)}(3), \cdots, x^{(1)}(n)\}, i = 1, 2, \cdots, n$

(2) Use the least-squares method to find the parameters a and u. Let $D = \begin{bmatrix} -[x^{(1)}(2) + x^{(1)}(1)] & 2 \\ -\frac{3}{2}[x^{(1)}(3) + x^{(1)}(2)] & 3 \\ \vdots & \vdots \\ -\frac{n}{2}[x^{(1)}(n-1) + x^{(1)}(n)] & n \end{bmatrix}$

$\hat{a} = (a, b)^T GM(2), , ,$

$y_n = [x^{(0)}(2), x^{(0)}(3), \cdots, x^{(0)}(n)]^T$, The parametric vector is derived as $\hat{a} = \begin{bmatrix} a \\ u \end{bmatrix} = (B^T B)^{-1} B^T y_n$

(3) Modeling $GM(1, 1)$

$$\hat{x}^{(1)}(i+1) = (x^{(1)}(1) - \frac{u}{a})e^{-ai} + \frac{u}{a}, , , \begin{cases} \hat{x}^{(0)}(1) = \hat{x}^{(1)}(1) \\ \hat{x}^{(0)}(i) = \hat{x}^{(1)}(i) - \hat{x}^{(1)}(i-1), i = 2, 3, \cdots n \end{cases}$$

(4) Test the model accuracy. The test methods are residual test, correlation test, and posterior difference test, and we adopt the posterior difference test here.

First, calculate the $S_0 = \sqrt{\frac{S_0^2}{n-1}}$, $S_0^2 = \sum_{i=1}^{n}\left[x^{(0)}(i) - \bar{x}^{(0)}\right]^2$, and $\bar{x}^{(0)} = \frac{1}{n}\sum_{i=1}^{n} x^{(0)}(i)$

Then calculate the mean squared deviation of the residual series$\varepsilon^{(0)}(i) = x^{(0)}(i) - \hat{x}^{(0)}(i)$ S_1. which is defined as $S_1 = \sqrt{\frac{S_1^2}{n-1}}$, $S_1^2 = \sum_{i=1}^{n}\left[\varepsilon^{(0)}(i) - \bar{\varepsilon}^{(0)}\right]^2$, and $\bar{\varepsilon}^{(0)} = \frac{1}{n}\sum_{i=1}^{n} \varepsilon^{(0)}(i)$

Based on this, the variance ratio is further calculated as : $C = \frac{S_2}{S_1}$ and the probability of small errors should be $p = \left\{\varepsilon^{(0)}(i) - \bar{\varepsilon}^{(0)}\right| < 0.6745 \cdot S_0\right\}$.

Finally, according to the prediction accuracy equivalence division table (see Table 2), the prediction accuracy of the model was decided.

Table 5. Classification of prediction accuracy level.

Small Error Probability p-value	C-value of Variance Ratio	Prediction Accuracy Level
> 0.95	< 0.35	Good
> 0.80	< 0.5	Qualified
> 0.70	< 0.65	Barely pass
≤ 0.70	≥ 0.65	Failure

(5) If the test passes, the model can be used to make predictions. That is, using $\hat{\boldsymbol{x}}^{(0)}(\boldsymbol{n}+\mathbf{1}) = \hat{\boldsymbol{x}}^{(1)}(\boldsymbol{n}+\mathbf{1}) - \hat{\boldsymbol{x}}^{(1)}(\boldsymbol{n})$, $\hat{\boldsymbol{x}}^{(0)}(\boldsymbol{n}+\mathbf{2}) = \hat{\boldsymbol{x}}^{(1)}(\boldsymbol{n}+\mathbf{2}) - \hat{\boldsymbol{x}}^{(1)}(\boldsymbol{n}+\mathbf{1})$, and $\cdots\cdots$ as $\boldsymbol{x}^{(0)}(\boldsymbol{n}+\mathbf{1})$ the predicted values of $\boldsymbol{x}^{(0)}(\boldsymbol{n}+\mathbf{1})$, $\boldsymbol{x}^{(0)}(\boldsymbol{n}+\mathbf{2})$, … and so on.

At this point, we have developed a gray prediction model, namely $\boldsymbol{GM}(\mathbf{1},\mathbf{1})$.

3.1.2. *Model Improvement and Optimization - Model*

For the $\boldsymbol{GM}(\mathbf{1},\mathbf{1})$ model, improvements and optimizations are made, and the resulting $\boldsymbol{GM}(\mathbf{2})$ model is built as follows.

Let$\boldsymbol{x}^{(0)} = \{\boldsymbol{x}^{(0)}(\mathbf{1}), \boldsymbol{x}^{(0)}(\mathbf{2}), \boldsymbol{x}^{(0)}(\mathbf{3}), \cdots, \boldsymbol{x}^{(0)}(\boldsymbol{n})\}$ be a non-negative sequence and $\boldsymbol{x}^{(0)}$ be the sequence generated by one accumulation of $\boldsymbol{x}^{(1)} = \{\boldsymbol{x}^{(1)}(\mathbf{1}), \boldsymbol{x}^{(1)}(\mathbf{2}), \boldsymbol{x}^{(1)}(\mathbf{3}), \cdots, \boldsymbol{x}^{(1)}(\boldsymbol{n})\}, \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \cdots, \boldsymbol{n}$

For $\boldsymbol{x}^{(1)}$ establish the following differential equation $\frac{d\boldsymbol{x}^{(1)}(\boldsymbol{t})}{d\boldsymbol{t}} + \boldsymbol{a}\boldsymbol{x}^{(1)}(\boldsymbol{t})\boldsymbol{t} = \boldsymbol{b}\boldsymbol{t}$

The model determined by this equation is called the gray square model, denoted as $\boldsymbol{GM}(\mathbf{2})$, because the exponent in its time response equation is $\boldsymbol{t}^{\mathbf{2}}$ of the form where $\boldsymbol{a}, \boldsymbol{b}$ is the parameter, denote $\hat{\boldsymbol{a}} = (\boldsymbol{a}, \boldsymbol{b})^{\boldsymbol{T}}$

$$D = \begin{bmatrix} -[x^{(1)}(2) + x^{(1)}(1)] & 2 \\ -\frac{3}{2}[x^{(1)}(3) + x^{(1)}(2)] & 3 \\ \vdots & \vdots \\ -\frac{n}{2}[x^{(1)}(n-1) + x^{(1)}(n)] & n \end{bmatrix}$$

$$\boldsymbol{Y}_{\boldsymbol{n}} = (\boldsymbol{x}^{(0)}(\mathbf{1}), \boldsymbol{x}^{(0)}(\mathbf{2}), \boldsymbol{x}^{(0)}(\mathbf{3}), \cdots, \boldsymbol{x}^{(0)}(\boldsymbol{n}))^{\boldsymbol{T}}, \tag{7}$$

and

$$\hat{\boldsymbol{a}} = (\boldsymbol{D}^{\boldsymbol{T}}\boldsymbol{D})^{-1}\boldsymbol{D}^{\boldsymbol{T}}\boldsymbol{Y}_{\boldsymbol{n}} \tag{8}$$

The solution of the equation is

$$\hat{\boldsymbol{x}}^{(1)}(\boldsymbol{t}) = \boldsymbol{c}\boldsymbol{e}^{-\boldsymbol{a}\boldsymbol{t}^{2}/2} + \frac{\boldsymbol{b}}{\boldsymbol{a}} \tag{9}$$

where c is $\widehat{()}$ determined by the initial condition $\hat{x}^{(1)}(1) = \hat{x}^{(0)}(1)$, i.e.

$$\hat{x}^{(0)}(t) = \left(\hat{x}^{(0)}(1) - \frac{b}{a}\right)e^{-at^{2}/2} + \frac{b}{a} \tag{10}$$

The predicted values of the model are obtained by doing the cumulative subtraction operation $\hat{x}^{(1)}(t)$ [3-5].

At this point, the improved $GM(2)$ gray prediction model has been established, and its solution process is shown in Figure 2.

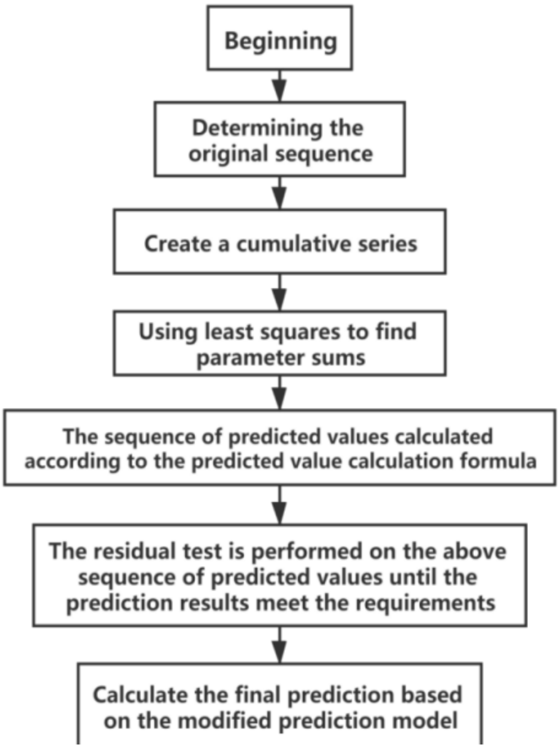

Fig. 2. Flow chart of gray prediction model solution.

3.2. *Solution of the Model*

The results of the cascade $\lambda(k)$ test on the "training data" are shown in the following figure.

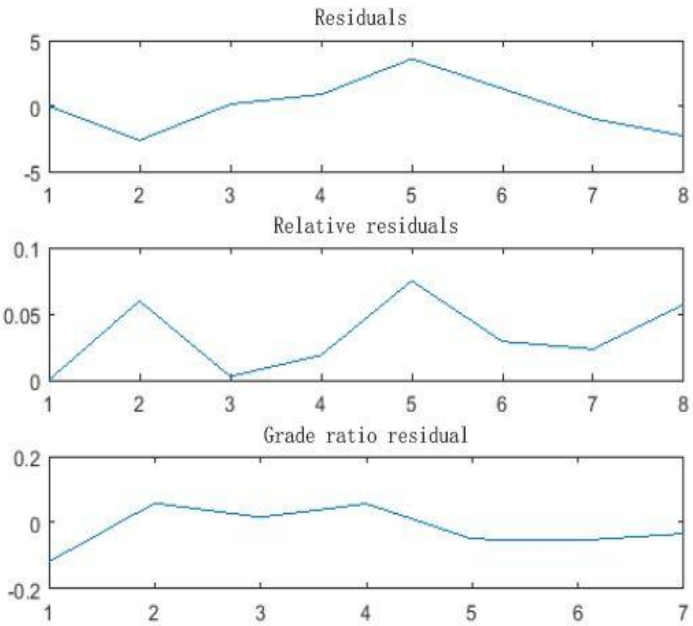

Fig. 3. Results of the cascade test on the data.

From the Figure 3, we can find that the original sequence does not satisfy the level ratio requirement, so we choose $x^{(0)} = (338,322,545,783,439)$ after several tests, as the original sequence, and the results of its rank-ratio test are shown in the Figure 4.

International Journal of Applied Science and Mathematics
Volume 8, Issue 3, ISSN (Online): 2394-2894

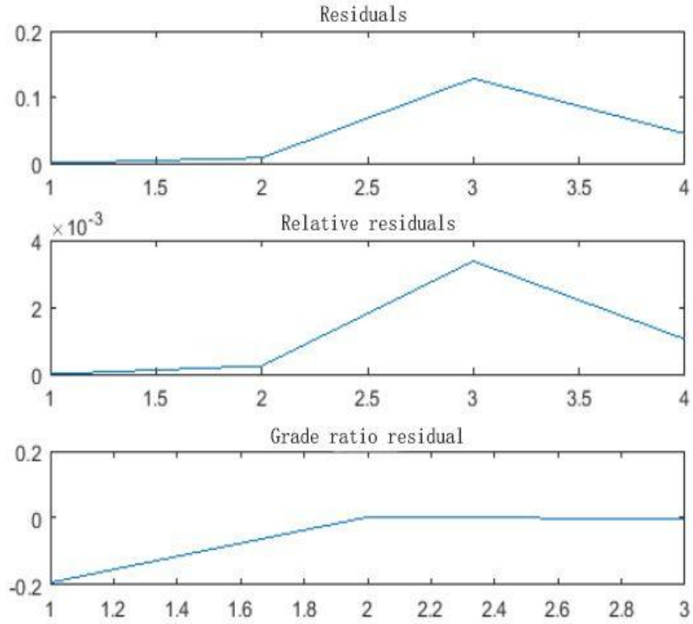

Fig. 4. Correction results of the cascade test on the data.

With proper sequence, the level ratio and residual meet the requirements, indicating that the sequence can be used for the solution of the model. The original sequence list is as follows.

k	1	2	3	4	5
$x^{(0)}(k)$	338	322	545	783	439

From equation (6), we find the cumulative generation sequence after one time $x^{(1)}(k)$, as follows.

k	1	2	3	4	5
$x^{(1)}(k)$	338	660	1205	1988	2427

Next, the model derived from Eqs [6] is trained using the "training data", and the resulting matrix is as follows.

$$B=\begin{bmatrix} -\frac{1}{2}[x^{(1)}(1)+x^{(1)}(2)] & 1 \\ -\frac{1}{2}[x^{(1)}(2)+x^{(1)}(3)] & 1 \\ -\frac{1}{2}[x^{(1)}(3)+x^{(1)}(4)] & 1 \\ -\frac{1}{2}[x^{(1)}(4)+x^{(1)}(5)] & 1 \end{bmatrix}=\begin{bmatrix} -499 & 1 \\ -932.5 & 1 \\ -1596.5 & 1 \\ -2207.5 & 1 \end{bmatrix}$$

$$y_n=[660,1205,1988,2427]^T$$

Based on the trained model, the hourly uplink and downlink traffic of the cell can be predicted. Due to the overwhelming amount of data, we only show the results for a single cell over 48 hours as follows.

Table 6. 48 hours Data results.

Date	**Time**	**UploadGB**	**DownloadGB**	**Date**	**Time**	**UploadGB**	**DownloadGB**
4/20	0:00:00	0.0087	0.0468	4/21	0:00:00	0.0010	0.0107
4/20	1:00:00	0.0046	0.0187	4/21	1:00:00	0.0170	0.0213
4/20	2:00:00	0.0013	0.0035	4/21	2:00:00	0.0013	0.0020

Date	Time	UploadGB	DownloadGB	Date	Time	UploadGB	DownloadGB
4/20	3:00:00	0.0002	0.0004	4/21	3:00:00	0.0028	0.0063
4/20	4:00:00	0.0004	0.0004	4/21	4:00:00	0.0007	0.0013
4/20	5:00:00	0.0032	0.0117	4/21	5:00:00	0.0000	0.0000
4/20	6:00:00	0.0014	0.0039	4/21	6:00:00	0.0002	0.0023
4/20	7:00:00	0.0005	0.0011	4/21	7:00:00	0.0004	0.0009
4/20	8:00:00	0.0052	0.0499	4/21	8:00:00	0.0016	0.0099
4/20	9:00:00	0.0015	0.0039	4/21	9:00:00	0.0008	0.0009
4/20	10:00:00	0.0142	0.0151	4/21	10:00:00	0.0023	0.0387
4/20	11:00:00	0.0086	0.3461	4/21	11:00:00	0.0042	0.0247
4/20	12:00:00	0.0066	0.2493	4/21	12:00:00	0.0024	0.0278
4/20	13:00:00	0.0043	0.1131	4/21	13:00:00	0.0101	0.1737
4/20	14:00:00	0.0002	0.0006	4/21	14:00:00	0.0193	0.1726
4/20	15:00:00	0.0000	0.0000	4/21	15:00:00	0.0008	0.0067
4/20	16:00:00	0.0013	0.0007	4/21	16:00:00	0.0004	0.0031
4/20	17:00:00	0.0027	0.0112	4/21	17:00:00	0.0008	0.0019
4/20	18:00:00	0.0024	0.1694	4/21	18:00:00	0.0003	0.0036
4/20	19:00:00	0.0107	0.3081	4/21	19:00:00	0.0024	0.0358
4/20	20:00:00	0.0077	0.0975	4/21	20:00:00	0.0006	0.0008
4/20	21:00:00	0.0036	0.0088	4/21	21:00:00	0.0003	0.0011
4/20	22:00:00	0.0107	0.1830	4/21	22:00:00	0.0006	0.0012
4/20	23:00:00	0.0016	0.0022	4/21	23:00:00	0.0004	0.0020

IV. Long-Term Prediction

4.1. *Selection of Prediction Methods*

None of the common forecasting models are able to predict long-term forecasts with high accuracy and efficiency. The LSTM model has the inherent ability to quickly adapt to sharp changes in the trend, which allows it to work well in fluctuating time series, and after model testing and parameter optimization, the corresponding model can be used to achieve the prediction.

4.2. *Model Building and Solving*

4.2.1. *Recurrent Neural Network (RNN)*

International Journal of Applied Science and Mathematics
Volume 8, Issue 3, ISSN (Online): 2394-2894

Before building the LSTM model, we first build the RNN model. In traditional neural networks, it is difficult to use the previous scenes to intervene in the later predictions, but RNNs solve this problem by persisting the information, as illustrated below.

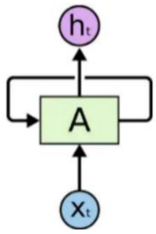

Fig. 5. Recurrent neural network with recurrence.

In the figure above, a set of neural networks A receives certain inputs x_t, and outputs a value h_t. The loop allows information to be passed from one step of the network to the next.

A RNN can be thought of as multiple copies of the same module, each passing a message to the next time step. Expanding the loop yields:

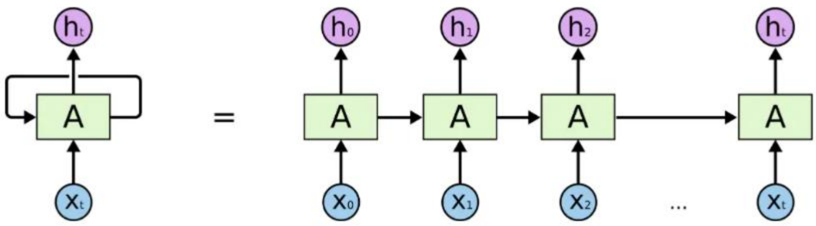

Fig. 6. The expanded form of recurrent neural network.

This chain-like feature reveals that RNNs are closely related to sequences and lists. They are natural neural network structures used for such data.

RNNs can learn to make use of historical information when the relevant information and the need for that information are close to each other.

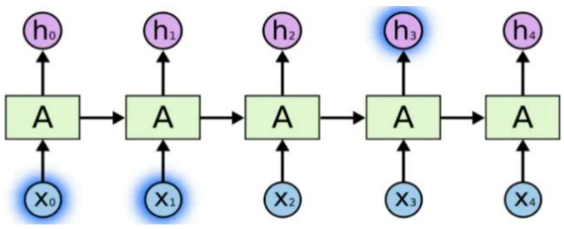

Fig. 7. RNN at short distances using historical information.

However, as the distance increases, RNNs cannot effectively use historical information.

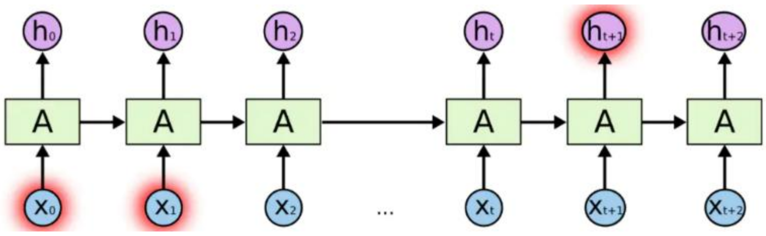

Fig. 8. RNN at long distances using historical information.

Hence we need to introduce the LSTM model to solve the problem with long-distance, i.e., for the long-term prediction problem.

4.2.2. *LSTM Model*

LSTMs are carefully designed to avoid the long dependency problem. Remembering longer history information is actually their default behavior, not something they struggle to learn.

All RNNs have the form of a chain of repeating modules of the neural network. In a standard RNN, this repetition module will have a very simple structure, such as a single tanh layer.

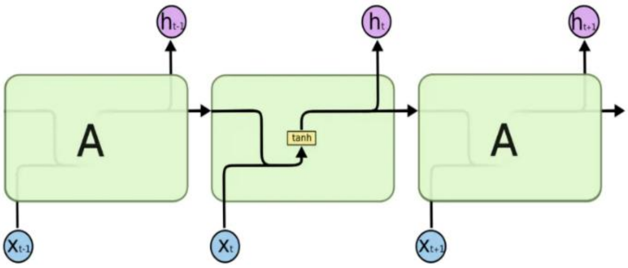

Fig. 9. Single-layer neural network with repetition module in standard RNN.

The LSTM also has this chain structure, but the repetition module has a different form. In contrast to the simple layer of a neural network, the LSTM has four layers, which interact in a special way.

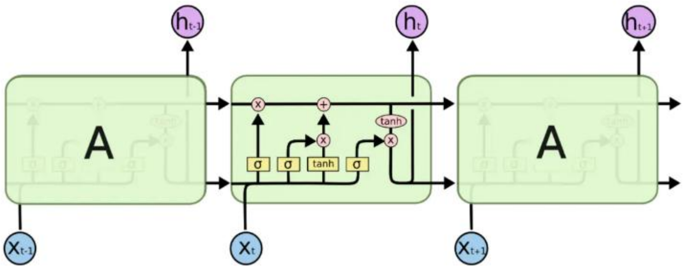

Fig. 10. The four interacting neural network layers contained in the repetition module in LSTM.

The first step in LSTM is to decide what information we want to discard from the cell state. This decision is implemented by a Sigmoid layer called the "forget gate". It looks at h_{t-1} the (previous output) and x_t (current input) and outputs a number between 0 and 1 for each number in the cell state C_{t-1} (previous state). 1 means complete retention, while 0 means complete deletion.

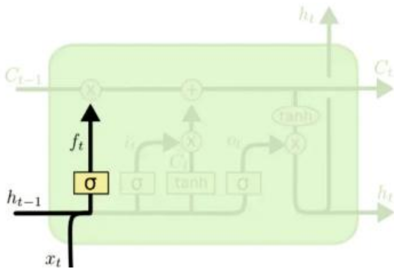

Fig. 11. Schematic diagram of the first step of LSTM.

This process can be expressed by Equations (11).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{11}$$

The next step is to decide what information we want to store in the cell state. This part is divided into two steps. First, a Sigmoid layer called the "input gate layer" determines which values we will update [7]. The next tanh layer creates a candidate vector C_t that will be added to the cell state. In the next step, we will combine these two vectors to create the updated values.

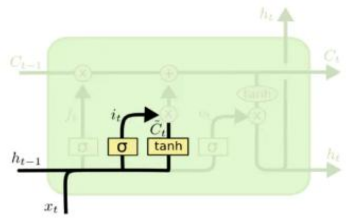

Fig. 12. Schematic diagram of the second step of LSTM.

The following equation can be established.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{12}$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{13}$$

Now it's time to update the last state value C_{t-1} to C_t. The previous steps have already determined what should be done, we just need to actually execute it.

We multiply the last state value by f_t, which expresses the part that we expect to forget. After that, we add the obtained value to $i_t * \tilde{C}_t$. This obtains the new candidate values, measured by how much we decide to update each state value.

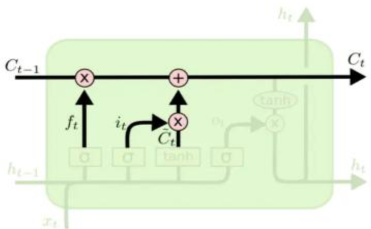

Fig. 13. Schematic diagram of the third step of LSTM.

The updating procedure is defined as

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{14}$$

Finally, we need to decide what we want to output. This output will be based on our cell state but will be a filtered version. First, we run a Sigmoid layer, which determines which parts of the cell state we want to output [8]. Then we pass the cell state through the tanh (normalizing the value to between -1 and 1) and multiply it by the output of the Sigmoid gate, up to this point we have only output those parts we decided on.

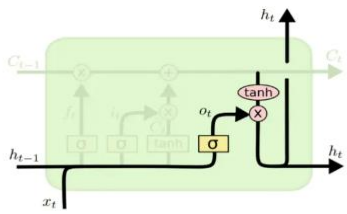

Fig. 14. Schematic diagram of the fourth step of LSTM.

The output is generated as [9-11]

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{15}$$

$$h_t = o_t * tanh(C_t) \tag{16}$$

4.3. *Solving Model*

After the above modeling, and also combined with the practical situation, we achieved the prediction through coding.

Table 7. Long-term forecast results.

Date	Subdivision Number	UploadGB	DownloadGB	Date	Subdivision Number	UploadGB	DownloadGB
11/1	67	0.7807	9.1968	11/1	69	1.1172	9.3550
11/2	67	1.2752	13.9063	11/2	69	1.4672	16.4684
11/3	67	0.4013	5.8873	11/3	69	0.0450	0.3961
11/4	67	0.4430	6.3068	11/4	69	1.3477	13.4807
11/5	67	0.8824	5.5209	11/5	69	0.8244	9.0728
11/6	67	0.3513	4.3096	11/6	69	0.7853	11.1429
11/7	67	0.9487	9.6476	11/7	69	0.6020	6.3042
11/8	67	0.5117	6.7757	11/8	69	0.6051	5.5211
11/9	67	0.5156	4.3647	11/9	69	0.2203	1.6776
11/10	67	0.1540	1.5230	11/10	69	0.2158	2.8706
11/11	67	0.2556	3.6125	11/11	69	0.2452	3.7940
11/12	67	0.2529	2.4432	11/12	69	0.0871	0.7445
11/13	67	0.3697	4.1763	11/13	69	0.5217	2.3769
11/14	67	0.4485	3.8711	11/14	69	0.1885	2.2613
11/15	67	0.4624	5.0965	11/15	69	0.2671	2.5425
11/16	67	0.7289	9.0732	11/16	69	0.1055	1.4448
11/17	67	0.6393	10.3238	11/17	69	0.1283	1.1747
11/18	67	1.0634	12.1426	11/18	69	0.5673	6.5271
11/19	67	1.2650	13.2121	11/19	69	0.5285	9.2793
11/20	67	1.0438	8.3035	11/20	69	2.2328	28.0077
11/21	67	0.6454	4.4410	11/21	69	0.4587	3.2840
11/22	67	0.9744	12.0267	11/22	69	0.5351	6.5420

Date	Subdivision Number	UploadGB	DownloadGB	Date	Subdivision Number	UploadGB	DownloadGB
11/23	67	0.4081	4.2948	11/23	69	0.7684	6.6403
11/24	67	1.4529	14.4800	11/24	69	0.9416	12.3891
11/25	67	0.6700	10.2469	11/25	69	0.7798	11.0753
.....	......	......	......	.....	......	......	......
11/1	133362	0.3564	3.2954	11/1	134060	0.2458	2.5000
11/2	133362	0.2668	2.0166	11/2	134060	0.3074	2.3563
11/3	133362	0.1538	1.3897	11/3	134060	0.0923	0.9863
11/4	133362	0.2827	2.9789	11/4	134060	0.2955	2.4358
11/5	133362	0.2919	2.6610	11/5	134060	0.1000	0.7382
11/6	133362	0.4822	6.1582	11/6	134060	0.3864	3.3056
11/7	133362	0.4032	4.7694	11/7	134060	0.2823	2.7515
11/8	133362	0.2145	1.8837	11/8	134060	0.1625	1.4763
11/9	133362	0.5130	4.6865	11/9	134060	0.0471	0.5471
11/10	133362	0.4736	3.9931	11/10	134060	0.0921	0.8275
11/11	133362	0.3477	3.1980	11/11	134060	0.0639	1.2212
11/12	133362	0.4000	3.7367	11/12	134060	0.1550	0.3393
11/13	133362	0.4698	2.9853	11/13	134060	0.1628	1.8784
11/14	133362	0.3786	1.7978	11/14	134060	2.0709	2.6591
11/15	133362	0.1937	1.4999	11/15	134060	0.1420	1.2881
11/16	133362	0.3306	4.4954	11/16	134060	0.3201	4.2930
11/17	133362	0.2572	2.8767	11/17	134060	0.5058	6.0343
11/18	133362	0.2756	3.0342	11/18	134060	0.3802	3.5881
11/19	133362	0.3435	2.5464	11/19	134060	0.5806	4.8270
11/20	133362	0.2730	2.5247	11/20	134060	0.8302	7.5538
11/21	133362	0.2941	1.7085	11/21	134060	0.6206	4.3348
11/22	133362	0.1350	0.7901	11/22	134060	0.2336	2.0523
11/23	133362	0.2228	1.8544	11/23	134060	1.0690	9.8248
11/24	133362	0.2406	3.8427	11/24	134060	0.3989	5.2127
11/25	133362	0.2353	1.9734	11/25	134060	0.5824	4.2192

V. Conclusion

In this paper, from the perspective of mobile communication operators, the traffic variation of base station traffic at different times is used as a background for pre-processing the collected large-scale data, and then the hourly traffic variation of a specified cell is predicted by an improved GM(2) gray prediction model; subsequently, the total daily uplink and downlink traffic for a certain period of time is predicted by using an LSTM model. This prediction method is designed for mobile communication operators to solve the problem of rationalizing the number of carrier frequencies arranged in base stations during peak traffic periods.

References

[1] Liu J, Chen H-B. Forecasting the scale of private investment in Jiangsu Province based on CM(1,1) model[J]. Statistics and Decision Making, 2005, 000(007):47-48.
[2] Mao Xiuzhen, Zhang Lujie, Mao Tianxiang, et al. A study of fund impact based on Pearson correlation analysis and regression analysis methods [J]. Journal of Zhejiang University of Technology: Social Science Edition, 2017(4).
[3] Zhang Qimin. Gray prediction model [J]. Journal of Ningxia University (Natural Science Edition), 2002(02):56-58.
[4] Xie Nai-Ming, Liu S-F. Modeling mechanisms of discrete and gray prediction models[J]. Systems Engineering Theory and Practice, 2005, 25(1):93-99.
[5] Yang, Guan. Analysis of achievement prediction based on gray system theory [J]. Mathematics Learning and Research: Teaching and Research Edition, 2018.
[6] Wang Shuxun, Xiong Qicai, Liu Feng. The establishment and practical application of gray model [J]. Journal of Shaanxi University of Technology, 2000(03):93-96.
[7] Li Zhaoyu,Ai Qian,Zhang Yufan,et al. LSTM neural network based on attention mechanism Ultra-short-term load forecasting method[J]. Electricity supply and use, 2019, (1) 17-22.
[8] Yao Z., Xin P., Shi B. J., et al. Short-term electric load forecasting based on LSTM time-recursive neural network [J]. Yunnan Hydropower Development, 2019, (3) 163-165.
[9] Anonymous, How to understand LSTM simply - it's not that complicated https: //www.jianshu.com/p/4b4701beba92 2021.1.16
[10] Ding Y, Sang N, Li XY, Wu Fei Zhou. A recurrent neural network-based approach to capacity data prediction in the telecommunication industry [J/OL]. Computer Applications: 1-12 [2021-01-18]. http://kns.cnki.net/kcms/detail/51.1307.TP.20210114.0948.040.html.
[11] Cheng Chen, LSTM for Everyone https://zhuanlan.zhihu.com/p/32085405

Author's Profile

First Author
Zheng Yan Xie, was born in Hebei, China on April 15, 2000. Study in the department of communication engineering of Yanbian University. email id: xiezy2000@126.com

Second Author
Zhong Qing Zhao, 2000, Study in the department of communication engineering of Yanbian University (Mei Hekou, China). email id: 1670428990@qq.com