ECMM409: Nature Inspired Computation
CA 2 Team Report

Eta Team
Candidate Numbers: 077832,019740,046781,047467,036414

January 11, 2021

1 Abstract

We explored multiple approaches in the field of nature inspired algorithms to tackle with water quality abnormal detection. We proposed a method that combines both artificial immune system (AIS) and probabilistic models. The AIS method we proposed simulates a group of cells and each one of them responses differently to the input data based on the threshold values assign to them. Then in the training stage, the threshold values will be adjusted accordingly to fit the joint distribution of the data. If there are preset number of features in one sample exceed threshold value of some certain cells, then these cells are dead. If the ratio between dead cells and total number of cells surpass a pre-defined extinction rate, then this sample will be classified as abnormal. Finally, in the predict stage, these threshold values are fixed, then we calculate and compare aforementioned values and output the result. The experiment results showed a great performance on the data and it outperformed previous results in the GECCO 2017 water quality challenge.

2 Research

We first examine the training dataset to understand and try to grasp some feature of the data. Given that what we have is a time series data, we draw the abnormal events across time and we found that there seemed to be some connection in time. The abnormal events plot in temporal are illustrated in Fig.1.

Then we plot the data distribution of each feature and Fig.2 presents the normal vs. abnormal data distribution of the temperature feature. As we can learn from Fig.2, the distribution of abnormal samples tends to have a spike at around 4.6 to 4.8, whereas the normal samples show a bimodal distribution and happened to have relatively low probability between 4.6 and 4.8.

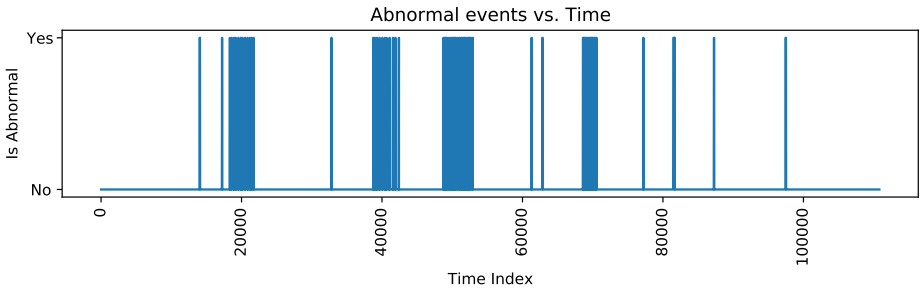

Figure 1: The abnormal events plot in temporal.

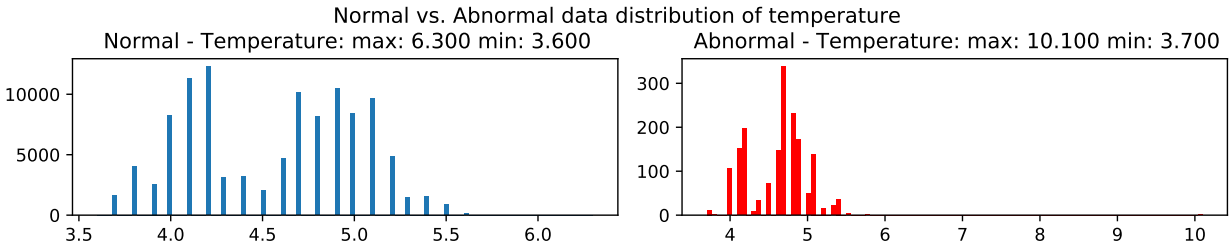

Figure 2: The temperature data distribution.

3 LSTM Approach

For simple comparison, we explored the approach of LSTM (long short-term memory) as well and built a LSTM network with the following architecture as shown in Fig.3. As the training proceeding, we can see from Fig.4 that the loss reduced fast at the first a few epochs and then barely dropped. We agree that the quick decrease presumably caused by the highly skewed data and the flatten area are due to lacking of enough abnormal samples.

As for the result of LSTM network, it successfully identified 182 out of 2768 abnormal events, which was equivalent to 0.5582 macro f1-score. The confusion matrix is shown in Fig.5.

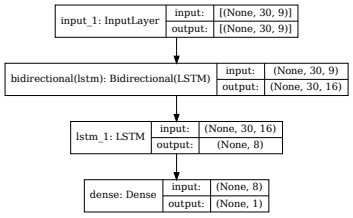

Figure 3: LSTM network architecture.

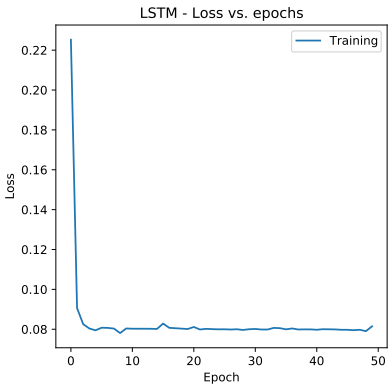

Figure 4: LSTM network training loss vs. epochs.

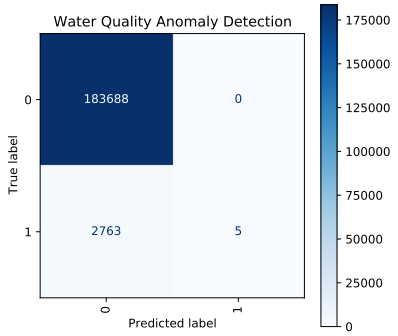

Figure 5: Confusion matrix when use the LSTM network to predict on test data.

4 The Proposed AIS Method

With our observation of the data, we proposed the following technique. We define one data sample with n features as $S = (f_1, f_2, \cdots, f_n)$ and our data set with m samples as $D = (S_1, S_2, \cdots, S_m)$. Also, we generate a group of cells with size c and death rate d.

Without loss of generality, suppose the range of first feature f_1 is $[u_1, l_1]$. Firstly, we can linearly discrete this range into k subranges. Then we assign different threshold value t to each of these ranges for each cell, i.e., every cell has different sensitivity to the environment. For example, t_i^j is the i-th cell's sensitivity to feature f_j and s_p^j is the feature value f_j of the p-th sample.

The next step is the training process. For each training sample, we will first find the corresponding subrange of s_p^j and then get threshold values for all cells in this subrange in terms of feature j. Afterwards, if the water quality is abnormal, we find out all indices of the cells which will die, i.e., their threshold values are more/less than the preset death rate d. If a feature is negatively correlated to the event then the cells with probability less than the death rate of the feature are chosen to be dead. Similarly, if a feature is positively correlated to the event, then the cells with probability greater than death rate are considered dead.

The next step is to tune the probabilities of the cells depending upon the event of the sample. In an abnormal case, randomly select a small group of dead cells to increase their immune ability. Also, we assume that this can also affect neighbours. Thus we increase their immune ability by 1% regarding to this feature. Meanwhile, for the each feature, we will append the dead cells' indices to a variable. Along with this small fraction of cells gaining immunity of 1%, the remaining cells lose confidence, say 25% due to the abnormality. In normal case, 25% of the cells lose immunity by 1% and the remaining cells gain confidence of 25%.

Finally, we calculate the number of dead cells that caused by multiple features. If the ratio between the number of dead cells and total size of cells exceeds a pre-defined extinction rate, then this sample will be classified as abnormal.

As for the prediction stage, these threshold values are fixed, then we calculate and compare aforementioned values and return the classification result.

4.1 Divide Into Small Tasks

We have divided the problem into 4 tasks, they are (i) load and preprocessing data, (ii) design how the cells would response to abnormal and normal samples, (iii) design which cells should be considered as dead, (iv) find a proper way for initialising these cells' thresholds, i.e., how to generate random solutions.

5 Experiment

5.1 Experiment Settings

In our experiment, we set the number of cells to 1000 and death rate of each feature is proportionate to its correlation with the event. A cell is considered as dead if there are 4

name	min	max	#subranges	name	min	max	#subranges
Temperature	-10	100	320	Trueb	0	1	100
Cl	0	1	200	Cl_2	0	1	100
pH	0	14	400	Flow rate 1	0	5000	5000
Redox	0	2000	500	Flow rate 2	0	5000	5000
Trueb	0	5000	5000				

Table 1: The minimum, maximum and the number of subranges of each feature.

Method	Dataset	TP	FP	TN	FN	F1-score	Macro F1
LSTM	Test	74	**0**	**241870**	2694	0.05207600	0.52327
Proposed Method	Test	**1075**	572	241328	1693	**0.48697622**	**0.74042**
Proposed Method	Training	**820**	123	120471	920	**0.61125606**	**0.80347**
FITORE	Training	472	636	**120670**	556	0.44194757	0.71852
RISHAB and et al.	Training	487	9707	111599	**541**	0.08679380	0.52145
LAM (Crazy)	Training	242	24392	96914	786	0.01886057	0.45195
ROMAN and et al.	Training	234	32662	88644	794	0.01379554	0.42752
LAM (Basic)	Training	297	45920	75386	731	0.01257276	0.38814

Table 2: Performance comparisons between different methods.

feature values exceed its threshold and the group extinction rate is set to 0.75. As for the features, there are 9 features in each sample. The minimum, maximum and the number of subranges of each feature are listed in Table.1.

5.2 Experiment Results

The confusion matrixes (CM) for the training data and test data are shown in Fig.6 and Fig.7 respectively. The macro f1-score of our method is 0.74042 and the standard f1-score is 0.48697622. The prediction results and comparisons are shown in Table.2. However, the official result page only gives the performance measurements of each submission on the training data. Therefore we also test the proposed method on the training data as well.

As we can see from Table.2, the propose method outperformed previous results in the GECCO 2017 water quality challenge in terms of the macro f1-score and the standard f1-score.

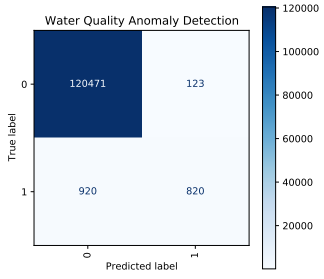

Figure 6: Training Data CM

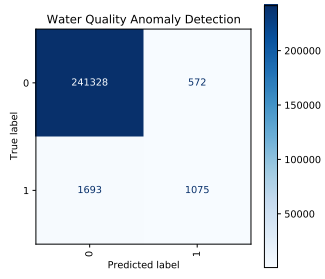

Figure 7: Test Data CM