Research on Self-Adaptive Stream Data Mining

Fang Xiao\textsuperscript{1,2}

\textsuperscript{1}National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan, Hubei, China
\textsuperscript{2}Huazhong University of Science and Technology Library, Wuhan, Hubei, China

\textbf{ABSTRACT:} This paper presents a self-adaptive stream data mining algorithm. The modification of the currently existing outlier stream data mining algorithm can make the stream data mining algorithm have adaptability, thus adjusting to the changes in the nature of the stream data caused by the environmental changes. The research starts from two aspects to enhance the adaptability of the stream data mining algorithm: In terms of the adaptability of system resources, the algorithm implemented in this paper can dynamically adjust the system resources used by the algorithm according to the load of the host, thus giving play to the initiative of algorithm to a maximum extent; in terms of the adaptability of stream data content, this paper researches the characteristics of the stream data, and modifies the existing data mining algorithms, so the stream data mining algorithm can make a dynamic adjustment according to the characteristics of the stream data, thus enhancing the efficiency of mining. Finally, this paper carries out comparison with the FODFP-Stream and LOF algorithm, and compares the time consumption with space consumption through the experimental data.

\textbf{Keywords:} stream data mining; self-adaptive; outlier mining; system resources

1 BACKGROUND INTRODUCTION

With the continuous development of network and communication technology, the database technology is gradually mature, especially in the aspect of database modeling, and its application technology also becomes more and more mature. Meanwhile, with the continuous renewal of technology, a large number of data generates increasingly. In real life, the data generates at a rapid and exponential speed, for example, the data resources in the aspects of network monitoring, flow analysis, intrusion detection, network application and network click-stream, intelligence analysis, satellite remote sensing, meteorological and environmental monitoring. Most portal sites require online real-time monitoring, and also require personalized services. All these needs can generate huge amounts of data. Similarly, in the stock market, the information changes rapidly. For the need of real-time monitoring of financial data, there is a need of online analysis and dynamic tracking of the stock price fluctuations. How to analyze and manage these large and ever-increasing data is extremely urgent.

2 CORRELATION RESEARCH

In order to find appropriate number of information from the stream data, the researchers use a variety of data mining technology, such as clustering mining technology \cite{2, 3, 5}, frequent item set mining technology \cite{6-8} and outlier detection technology \cite{9-11}. Meanwhile, the research in the literatures \cite{12-15} points out that, by the use of these technologies for the static data set, the stream data mining brings a series of problems. Meanwhile, the data type of a variety of stream data is different, so the nature of the stream data is also very different. Therefore, in the process of data mining, there are many variable factors. In the literature \cite{15}, a series of analyses points out that, with the difference of the environment, the same stream data also has different patterns of manifestation. Meanwhile, the literature points out that, the same stream data applies for the same mining technology due to the difference of the environment, but with different performance. For example, when the operating system occupies a lot of CPU time and memory, the system resources allocated to other tasks (data mining) will reduce. However, when the system occupies less CPU time and
memory, the system resources allocated to other tasks (data mining) will be more. Therefore, with the difference of the environment and the pattern of manifestation of the stream data, the data mining requires necessary adjustments, thus adapting to different environments and patterns of manifestation of the stream data. Therefore, with the difference of the pattern of manifestation of the stream data, the data mining also requires adjustment. Therefore, there is a need of a self-adaptive data mining method to make an adaptive adjustment based on different stream data.

In the specific algorithm implementation, the researchers conduct a lot of research work, and achieve a variety of different mining algorithms. Gaber, et al. carry out an in-depth research of the stream data mining algorithm, structure and framework of data mining system [16]. Meanwhile, Jan, et al. used the association rules for data mining, which is also a new attempt [17]. In addition, in Data Streams: Models and Algorithms, Charu carries out an in-depth research of the current stream data mining, and makes fair assessment of the current major stream data mining method [1].

In addition, with the constant deepening of the research, the researchers can meet part of needs through improvement of the existing data mining algorithms. In order to make the data mining algorithm process data quickly, the researchers try to use the finite subsets of the stream data for mining, such as k-means algorithm used by CluStream [1] and fp-growth algorithm used by FP-Stream [18]. In addition, the outlier detection is also a very good mining algorithm in processing one-dimensional stream data [20, 21].

However, most of the stream data mining algorithms are static [1, 2], without self-adaptive adjustment. Therefore, only a certain type of stream data can be mined, but those dynamic stream data mining algorithms are very stiff, and the stream data mining algorithm can be only adjusted by the parameters defined by the users, so it can be adjusted by the given type of stream data. The literature [4] points out that, the stream data is ranked according to the order, so the stream data mining work is generally based on a single linear scanning algorithm. This way consumes huge time and space, and it simply stores the existing data in the stream data management system, and ignores most of the stream data to present multi-layer and multi-dimensional characteristics. Therefore, the data mining requires multi-layer and multi-dimensional processing.

3 SELF-ADAPTIVE STREAM DATA MINING MODEL

If the data mining algorithm requires adaptive operating environment and data environment, there is a need to adjust the algorithm when the environment changes, so as to adapt to the data mining environment. Therefore, there is a need of some mechanisms to detect the environmental changes, so as to achieve real-time measurement of the operating environment and data environment of the model, report back the detection results to the model, so the model can timely adapt to environmental changes. This section further modifies the model through analysis of the common data mining model and proposes a resource-driven model based on the time window, so the environment can affect the operation of the mining model.

3.1 Traditional stream data mining model

Most of the stream data mining algorithms adopt three layers of mining model. These three layers are respectively online mining layer, data summary layer and offline summary layer. Its logical structure is shown in Figure 1 [1, 22].

3.2 Environment-driven improvement of stream data mining model

The model shown in Figure 1 indicates that, the entire mining model does not take into account the allocating resources owned by the host of the mining algorithms, nor take into account the changes of the data features in the stream data. However, the self-adaptive mining model requires to consider the above two points, thus greatly improving the adaptability of the traditional models.

For the modification of the traditional models, the mining model can be modified from two parts: allocating resources of the host and the stream data characteristics. The adaptability of the model is ultimately the adjustment of the operating parameters of the model, such as the size of the time window, sampling frequency and so on. Therefore, the improvement of the model should start from detecting the environment,
and first detect the changes of the operating environment of the model, such as the available memory and CPU cycles of the host. When the environment of the model detected changes, the model can adapt to the current environment through modification of the operating parameters of the model. For example, in the quality-driven model proposed in the literature [22], the entire structure is divided into the stream data filtering, offline and online mining and resource detection modules.

In the environment-driven model, there is a need to monitor two kinds of environments—allocating resources of the host and the stream data characteristics. The detection of allocating resources of the host is the work related to the operating system, while the detection of the stream data characteristics is the work related to the stream data summary layer, so these two types of detection shall be done separately rather than combined as one module. Therefore, the detection model of the resources of the host and the detection model with the stream data characteristics that are included in the environment-driven model will automatically modify the parameters at the online and offline mining layers after the detection of the environmental changes, so the entire stream data mining model can adapt to the environmental changes. And Figure 2 shows the logical structure of the environment-driven model.

In Figure 2, the resource detection module and the data detection module can affect the parameters of the mining model. The parameters of the environment-driven model are window size and sampling frequency. These two parameters are used at the online mining layer. Therefore, the affected part of the resource detection module and data detection model is the online mining layer.

3.3 Model adaptability

This section will further discuss the environment-driven adaptive model. And the discussion is divided into two aspects. The one is analysis of the parameters of the mining model, and the other is analysis of mathematical modeling of the model. Through analysis of the mathematical model, the basis of modification of the parameters by the detection module can be further determined.

**Environmental adaptability is an optimization problem**

Through the environment-driven model described in the Section 3.2, the key part of the model is to adjust the parameters of the model based on the situations of the resources and stream data. There are two operating parameters in the environment-driven model: time window and sampling frequency. The situation of resources is the available memory and CPU cycles of the host, while the situation of the stream data is the number of stream data objects in the time window and the data summary of the stream data objects.

The literature [4] proposes that the environmental adaptability is an optimization problem. This paper points out that, when the available memory and other resources of the host are greater, the online mining memory and CPU cycles will be more, so the speed of processing the stream data objects at the online mining layer increases, that is, the throughput at the online mining layer increases. Since the throughput at the online mining layer can be increased, the time window size can be appropriately adjusted, or the sampling frequency can be increased. However, the increase of the time window or sampling frequency may bring two additional problems. First, due to the increase of the time window, the stream data objects at the online mining layer will also increase, thus increasing the data size to be detected by the data detection module. Meanwhile, the increase of the time window or sampling frequency means that, at the online mining layer, more processing results will be transmitted to the stream data summary layer, and the data processed by the stream data summary layer will be more, and the stream data summary layer will consume more host resources [4].

However, the literature [4] does not give a specific analytical method, nor propose how to obtain the op-
timal solution to the optimization problem. Specific to the environment-driven model designed in this paper, there are two main contradictions: The one is to allocate the limited host resources so that the consumption of the processing speed at the online mining layer and the consumption at the offline mining layer and data summary layer achieve an optimal value; in addition, the contradiction between the time window size and the number of outliers in the resource-driven model. The greater the time window is, the more the outliers is, so the similarity in the time window is low, and the detection module deliberately reduces the time window. Therefore, in order to ensure that the similarity in the time window is relatively stable, there is a need to take the optimum position of the time window.

**Dynamic programming problem in the environment-driven model**

Simply speaking, the first problem is to allocate the limited memory and computing resources with the increase of the time window or sampling rate, so the stream data objects to be processed also increase, thus affecting four parts in the environment drive—online mining layer, stream data summary layer, offline mining layer and data detection module. In the WEB server traffic, when a large number of memory and CPU time at the environment detection layer is not used, the throughput of the entire model can be increased through increasing the time window size and sampling frequency. However, after increasing the throughput of the stream data objects, the above four parts may have problems in the resource consumption.

Assuming that a stream data object is processed, the host resource consumed at the online mining layer is $R_1$, at the stream data summary layer is $R_2$, at the offline mining layer is $R_3$, and by the data detection module is $R_4$. In addition, the online data mining and data detection module belong to real-time operation, so the consumption of the host resources generates simultaneously; the offline mining layer and stream data summary layer belongs to the offline module, so its consumption is not greatly related to the online data mining layer. Therefore, the total resource consumption is divided into three parts: the consumption $R_1+R_4$ generated by the online stream data mining and detection; the consumption $R_2$ generated by the stream data summary layer; the consumption $R_3$ generated by the offline mining layer. In addition, assuming that the total amount of available resources is $R$, and more than one stream data objects are processed, the mining efficiency improved by the online mining layer, stream data summary layer, offline mining layer and data resource detection module is respectively $V_1$, $V_2$, $V_3$ and $V_4$. The resource allocation in the entire environment-driven model is an optimization problem. Precisely speaking, it is a dynamic programming problem of the resource allocation \([26, 27]\). First, the modeling is as follows:

Assuming that $M$ ($M$ is the number of processing stream data object, which is a positive integer) unit of the host resources requires to distribute to $n$ modules, and after the known $k$-th module obtaining $m(k)$ unit of resources, the mining efficiency $V(k,m(k))$ can be improved, which is an increasing function of $m(k)$. Therefore, in the part of resource allocation, the entire environment-driven model is to discuss how to distribute memory resources of $M$ unit, so the overall mining efficiency is the highest. The overall mining efficiency is the sum of $V(k, m(k))$:

$$V = \sum_{j=1}^{n} V(k, m(k)) \tag{1}$$

Therefore, according to the dynamic programming solving method in the optimization problem, the allocation process of the host resources can be divided into $n$ stages, of which the $k$-th stage is to allocate the host resources to the $k$-th module in the model. $l(k+1)$ represents the number of the remaining host resources after allocation of the $k$-th stage, and $p(l(k+1),k+1)$ represents the optimal value function of the $(k+1)$-th stage. That is, $p(l(k+1),k+1)$ is to allocate the remaining host resources $l(k+1)$ to the $(k+1, k+2, ..., n)$-th modules, so as to obtain the maximum mining efficiency. In accordance with the dynamic programming theory, first, the basic equation can be determined:

$$p(k,l(k)) = \max_{m(k)\in [1,M]}\{V(k,m(k))+p(l(k+1),k+1)\}, k=1,...,n$$

$$p(n,l(n)) = V(k,m(n)) \tag{2}$$

Second, the state transition equation in the model can be determined:

$$l(k+1) = l(k) - m(k) \tag{3}$$

So, the optimization modeling of the environment-driven model is completed. When modeling is completed, the optimal decision sequence $(m'(1), m'(2), ..., m'(n))$ can be obtained by the reverse recurrence, and the maximum value of the mining efficiency $(V=V(l(1)))$ can also be obtained. Therefore, the model can be solved by matlab tools. Specific to the environment-driven model described in this paper, there are four resource consumption modules: environment detection module, stream data filtering module, offline mining module and offline mining module. Therefore, in the model, $n=4$, and the total number of resources ($M$) is the current total resources of the host.

However, there is a need to note that, the total number of resources ($M$) is the number of dynamic change. The normal operation of other application programs and operating systems of the host requires consumption of a certain amount of resources, so the total number of resources may obtain different data after completion of environment detection, and there is a need to repeatedly apply the above models for real-time adjustment of the environment-driven model. Meanwhile, not all remaining resources of the host can
be allocated to the environment-driven model, thus causing too large load of the host and other processes dead. Therefore, it is recommended that the remaining resources obtained every time should be allocated to the environment-driven model based on a certain percentage which is about 30%.

**Maximum-minimum value problems of the environment-driven model**

For the other problem, with the increase of the stream data objects processed at the online mining layer in unit time, more summaries of the stream data objects will generate, and the similarity of the summary of the stream data objects within the same time window is low, thus causing that the data detection module requires to reduce the time window or sampling rate. Simply speaking, the allocation of the remaining resources and the number of processing objects in unit time constitute a contradiction, which not only prompts to increase the time window, but also promotes to reduce the time window.

For the environment-driven model, the time window is equivalent to the window of stream data object, and a window contains a plurality of stream data objects. Therefore, in order to facilitate the discussion, the number of the stream data objects used here is acted as a measurement of the time window size. So, for the outliers, the greater the time window is, the more the outliers is, so the similarity in the time window is low. Therefore, in order to ensure that the similarity in the time window is relatively stable, there is a need to obtain the optimum position of the time window so that too high similarity here will not affect the mining effect, or too low similarity will not affect the mining efficiency. In general, in order to obtain a single time window, the value of outliers of the stream data objects is smaller.

The literature [28] discusses the probability of occurrence of the outliers in detail, of which the probability of occurrence of the outliers and the ratio of the outliers in the training set present a similar linear relationship. Similarly, with the increase of the window, the window monitoring points increase, and there are more outliers. Therefore, there is the simplest assumption, assuming that there is the simplest linear function relationship between the probability of occurrence of the outliers (f) and the window size (W):

\[ f = aW + b, (a, b \in R) \]  

(4)

First, considering the impact of the window size on the stream data objects, the greater the window is, the more the outliers in the stream data is. Assuming that the probability of occurrence of the outliers in the unit discharge is a, the probability of occurrence of the outliers in the window size (W) is aW. Considering that it is free of any network communication, the host requires sending some messages that are required by the system (such as ARP inquiry, DHCP communication and so on). That is, in the absence of any stream data transmission, in fact, the system also occupies part of consumption for the network resources, so there is a need to use the parameter b to correct the data accuracy.

Then, the number of outlier in the window (P) is the product of the window size and outlier probability, and the outlier will not disappear due to the disappearance of the window. Therefore, there is a need to take the corrected parameter c to ensure the data accuracy. Thus, there are two corrected parameters: b corrects the outlier probability, and c corrects the number of outliers.

\[ P = W \cdot f = (aW + b)W + c \]

(5)

\[ P = aW^2 + bW + c \]

\[ P = aW^2 + bW + c \]

Therefore, in order to ensure that the numerical value of the outlier of the stream data objects in the unit window is smaller, that is, ensure that P/W data is smaller:

\[ \frac{P}{W} = aW + b + \frac{c}{W} \]

(6)

Therefore, in order to ensure the effectiveness and efficiency of the stream data mining, there is a need to obtain the minimum value of P/W. Through the above analysis, as described in the Formula 6, the distribution probability of the outlier is a composite function. According to Cauchy inequality, we can obtain the minimum value of P/W if and only if \( W = \sqrt{\frac{1}{c^2}} \). And the value of P/W is \( b + 2\sqrt{ac} \).

### 4 EXPERIMENT AND OBSERVATION

This chapter carries out a series of experiments for the environment-driven model and outlier mining algorithms, and carries out experiments by the use of the existing data sets, and then compares the advantages and disadvantages of the outlier mining algorithm of the environment-driven model ED-STREAM with other algorithms.

In order to make the mining algorithms referential, the author selects FODFP-Stream\(^{[23]}\) and LOF\(^{[23, 24]}\) algorithm for comparison. As described in Chapter II, FODFP-Stream is a frequent item mining algorithm of the stream data, and LOF is a data mining algorithm for local data. In addition, KDD-CUP-99 and T10-I5-D1000K data sets are selected as the experimental data for comparison in terms of the time and space complexity of the algorithm.

Figure 3 demonstrates the comparison of three algorithms in terms of the time consumption. As can be seen from Figure 3, ED-Stream is superior to FODFP-Stream algorithm and LOF algorithm in terms of the execution time. LOF requires incremental updating and near data outlier related to the detection point, so the time consumption of LOF algorithm is
much larger than that of ED-Stream and FODFP-Stream algorithm, and the extra consumption is related to the size of close region determined in LOF algorithm. In addition, compared with ED-Stream, FODFP-Stream requires weighted frequent calculation for the stream data, so its time consumption is higher than that of ED-Stream algorithm.

In terms of the space consumption of the algorithm, it mainly considers the memory consumption. By the use of KDD-CUP-99 data sets and T10-I5-D1000K data sets, when running the algorithm, the memory consumption of three algorithms in running two data sets can be obtained by the memory monitoring program. Therefore, the comparison with the memory consumption of three algorithms can be obtained. And the specific data and comparable graphics are shown in Figure 4.

In Figure 4, ED-Stream algorithm and FODFP-Stream algorithm are more similar in terms of memory consumption. However, ED-Stream algorithm is an adaptive algorithm, which can dynamically adjust the window size and sampling frequency, so its memory consumption is less than FODFP-Stream. In addition, LOF algorithm requires incremental update operation, that is, LOF algorithm requires to save the stream data information in close region, so as to facilitate updating. The greater the close region is, the more the memory consumption is. Such consumption and the size of close region present a quadratic relationship, so LOF is a kind of algorithm with a higher degree of space complexity.

5 CONCLUSION

This paper analyzes and researches the outlier detection of the stream data, and gives out an improved mining model—environment-driven model through analyzing the adaptability of the analysis and mining algorithm for the host resources and adaptability of the mining efficiency. This model has more advantages than that of the FODFP-Stream and LOF algorithms in terms of the time complexity and space complexity.

REFERENCES

[1] Charu C. Aggarwal, Jiawei Han, Jianyong Wang & Philip S. Yu. 2003. A framework for clustering evolving data streams. In 29th International Conference on Very Large Data Bases (VLDB), 3: 81-92.


