A Topic-based Dynamic Clustering Algorithm for Text Stream

Y. Rao, X.J. Li
Lab of Social Intelligent and Complex Data Processing, College of Software, Xi'an Jiaotong University
Xi'an, China

***Abstract*—In order to provide real-time early warning from the public sentiment information in social network for making decision, a topic-based dynamic clustering for text stream (TBDC4TS) algorithm is proposed to cluster the text stream, which is formed by web crawler to continuously grab the web pages. The sliding time window (SWt) can be used to split the text stream into continuous segmentation, which includes a set of web news' pages related to the velocity of stream and the size of sliding window. Furthermore, a multi-phase cluster method in TBDC4TS is used to merge the micro-cluster in each sliding window and Macro-cluster in single-pass engine together. The results of experiments, used 2650 web news pages to form a simulate text stream by web crawler, show that the TBDC4TS algorithm has 22.8 times executing efficiency and the higher clustering qualify, such as precision and recall rate, than Single-pass.**

Keywords-sliding window; data stream; text mining; multi-phases clustering

I. INTRODUCTION

With the development of mobile cloud computing and social network application, the value of data is changing the users' consume and decision habit in the big data era. Especially, the rapidly increasing data with high velocity to form a data stream, which is continuous, unlimited and dynamic variable data set [1]. In order to help web users making decision in real-time, how to achieve and extract the valuable information from the data stream and to form different topic clusters in time are new challenges, especially, how to extract text feature vector with multi-dimension and to design a cluster algorithm for text stream with lower time and space complexity. Therefore, the evaluation about the quality of cluster algorithm, such as precision rate, recall rate, efficiency and robustness, are becoming the key points, when a single pass scanning method [2] is used in this process. To solve above problems, this paper proposes a Topic-Based Dynamic Clustering Algorithm for Text Stream (TBDC4TS), which uses a sliding time window to split the text stream into continuous segments and to transform the text stream cluster from flow data to continuous batch data processing.

II. RELATED WORKS

There are three kinds of models to process the data stream, such as the time-limited model, the sliding window model and the snapshot model [3, 4]. The data scale of all these three models depend on the selection of time interval, which are defined by the time interval from an initial time to current time, a certain time widow size and a certain time interval between each snapshot operation, respectively. Moreover, some researchers focus on the structure of data stream clustering, which includes some algorithms based on the Single-pass and Clu-stream algorithm[5]. Single-Pass is a classic incremental clustering algorithm with single scanning the whole data set. The upcoming data in stream, which is captured by system, should be compared to existed clusters one by one, if there is a cluster which has the highest similarity degree with the new data and larger than the threshold, then merge the new data into this cluster and recalculate the new average feature of cluster, else a new cluster can be created by this new data point. This algorithm is suitable to the large data with certain number of clusters, but not suitable to the situation that the number of cluster is varying when the data volume is constant increasing with data flow. Based on the Single-Pass strategy, Zhu[6] analyze the influence factors on the efficiency and quality of clustering by feature's weighted coefficient for the dimension of a feature vector. Yi[7] proposed a method of periodicity incremental clustering to obtain a new centre point of cluster. Yin[8] also put forward a method to split the data stream into a serials of chunk of data to optimize and decrease the effect of the sequence of data stream.

Clu-Stream algorithm is a clustering framework for data stream with two phases, including a real-time online clustering and off-line clustering. In the process of online clustering, the micro-cluster can cluster the data sets in different segments of data stream. Then, in the process of offline cluster, the macro-cluster can put these new clusters created in online into the whole cluster sets and merge them into existed clusters by similarity measuring. In addition, a Pyramid model of time framework is designed to store the clustering results in different granularities and phases. Li[9] proposed a sliding time window, based on the Clu-Stream algorithm, to increase the efficiency of mirco-cluster in online. However, because the Clu-Stream adopts the hierarchical clustering method with BIRCH algorithm, which just is suitable for the data set with same number of feature's dimension, and not suitable to cluster the text data sets with variable dimensions of feature vector. But, all these methods have lower efficiency with larger computation to index the high frequency words in text.

A. *The Definition of Concepts*

Definition 1: Basic Time Window (BW_t): Let t and p represent the time and time interval, respectively. The document set, i.e. $\{Doc_i, 0 < i < n\}$, is obtained by system as a time series in the time interval of (t, $t+p$]. Then, the Basic Time Window (BW_t) is a document set in the certain time interval, i.e.,

$$BW_t = \{Doc_{i,j}, 0 < i \le n, t \le j \le (t+p)\} \quad (1)$$

Where, the length of BW_t is the time interval p.

Definition 2: Sliding Time Window(SW_t) is a special BW_t that can slide forward and operate in p cycle and form a control flow for a series of BW_t, i.e.,

$$SW_t = \{(BW_t)_k, 0 < k < m\} \quad (2)$$

Where, if k means the t moment, then $k+1$ means the $t + p$ moment. The size of SW_t is the volume of data, which is captured in time interval of p, the length of BW_t, and included a set of documents. Furthermore, when $m \to \infty$, the process of SW_t can become a continuous and infinite window stream.

Definition3: the Size of Basic Time Window (SBW_t): Let v and p mean the velocity of data stream and the length of Basic Time Window, respectively. The SBW_t means that the total amount of data in one BW_t, which can be calculated as follow:

$$SBW_t = p * v \quad (3)$$

According to the above definition, let $p=1$(*unit time*) and $v=1$ (*doc per unit time*), then $SBW_t = 1$, which means that just one document flowed into the window in unit time interval. This process is become a real time continuous process, namely, the classical Single-Pass process. Moreover, if p is fixed as a constant, the SW_t also is a stable Sliding Time Window, but if p is variable, the SW_t is a dynamic window with more complexity and flexibility.

B. *The Framework about Topic-based Dynamic Clustering for Text Data Stream*

Based on the definition above mentioned, the Single-Pass clustering algorithm has lower efficiency and robustness, because it not only must calculate the similarity by distance between each new data and existed cluster in time serial processing, but would be affected by the sequence of data input time. Therefore, a Sliding Time Window can be used in the paper to remedy the defects from Single-pass and provided a batch processing in continuous window, which can greatly improve the clustering efficiency and reduce the influence of the data sequence added into the stream. The framework about topic-based dynamic clustering for text stream is illustrated in figure 1. Where, there are four core modules in the whole framework, such as Web Crawler, Dynamic Clustering Engine, the Text Topic Processing Toolsets and the Outputting Results Layout and Analysis. But, the mechanism about Dynamic Clustering Engine is core modules, which also includes three sub-modules to process the dynamic clustering for text stream, such as the Sliding Window Scheduler, Text Clustering Engine and Single-Pass Clustering Engine.

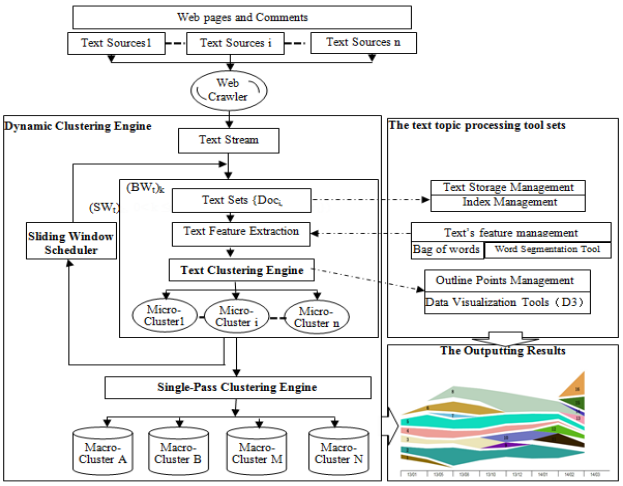


FIGURE I. THE FRAMEWORK ABOUT TOPIC-BASED DYNAMIC CLUSTERING FOR TEXT STREAM.

The module of Web Crawler can capture the web pages as text data source to form a continuously text stream. Then, in order to promote the efficiency about the key words extraction, it needs a pre-processing for text filtering, such as reduplication URL address, embed image or video, advertisement and others, to avoid producing the negative influence for text feature extracting.

In Dynamic Clustering Engine, the sub-module of Sliding Window Scheduler (SWS) is utilized to control the Basic Time Window $(BW_t)_k$ in a special procedure and change the data stream into the batch of data stream. Text feature in document sets in $(BW_t)_k$ can be extracted and indexed by the segment word tool and the bag of words in Chinese. The vector of text feature, i.e. $docFeature_i$, can be represented by four-tuple as<*Person, Address, Time, FrequentTerms*>.

Where,

-*Person* means that a sets of the person name emerged in the document, i.e., $\{person_1, person_2 \ldots, person_n\}$;

-*Address* means that a sets of the geographical place feature emerged in the document, i.e., $\{addr_1, addr_2 \ldots, addr_m\}$;

-*Time* means that a sets of the time related feature information emerged in the document, i.e., $\{time_1, time_2 \ldots, time_k\}$;

-*FrequentTerms* means that a sets of high-frequency words in the document by TF-IDF algorithm filtering, i.e.,$\{term_1, term_2 \ldots, term_j\}$;

After the text feature extracted and a feature vector of the document formed, the Text Clustering Engine is triggered to cluster by the density distribute clustering algorithm, such as DBSCAN, among the different feature vectors in *k-th* BW_t, then a sets of clusters can be output, including {$Cluster_{k1}$, $Cluster_{k2}$…, $Cluster_{kn}$}, into the Single-pass Engine as a data set, in which each cluster, same as a data point, also has its own feature that is an average of features of documents in the cluster. Therefore, each Micro-cluster in one BW_t should be calculated the distance between each of existed Macro-clusters, and then merged it into the Marco-cluster which has the shortest distance and less than the *threshold*. In addition, the process and result of clustering can be visualized by D3 software suit to provide for the decision support.

Therefore, the framework of TBDC4TS bridge the sliding time window and clustering process together to provide a multi-phase strategy for text stream. The concrete algorithm of TBDC4TS with three logic phases is realized as follow:

Input: *Text Stream {Doc_{ik}, $0<i\leq n$, $0<k\leq m$};*

-p: the length of Basic Time Window;

-Eps: the threshold of radius;

-MinPts: the threshold of density;

-STH：the similarity threshold for Single-Pass;

Output: the Sets of Cluster by Topic： {$Cluster_{T1}$, $Cluster_{T2}$…, $Cluster_{Tg}$}

Processing:

// phase1: to control and schedule about SW_t

Step1.1 *Startup the Web Crawler to form a text stream;*

Step1.2 *To slide the k-th ($0<k\leq m$) time window $(SW_t)_k$ with the length p by the module of SWS;*

For ($k=1$, $k\leq m$, k++)

{

***//phase2: the text feature processing and* Micro-clustering in $(BW_t)_k$**

Input: { Doc_{ik}, $0<i\leq n$ **}**;

Output: *the sets of Micro-clustering in $(BW_t)_k$;*

Processing:

Step:2.1 FOR ($i=1$, $i\leq n$, i++)

{

$docFeature_{ik}$(*Person*, *Address*, *Time*, *FT*);

//Mining *i-th* document and extracted the feature of text, such as person's name,

address, time and high-frequency words;

buildIndexByFeature*(Person, Address, Time)*;

//to build the index of text

DBSCAN({*$docFeature_{ik}$****})****;*

Return*({$Mirco\text{-}clusters_{fk}$, $0<f\leq n$})*

}

//phase3: Macro-clustering in Single-Pass

Input: {$Mirco\text{-}clusters_{fk}$, $0<f\leq n$}

Output: {$Cluster_{Tn}$, $0<Tg<m*n$}

Processing:

Step 3.1 FOR ($f=1$, $f\leq n$, f ++)

//ergodic the set of *Mirco-cluster*

{

Single-pass ({$Mirco\text{-}clusters_{fk}$});

Return({$Cluster_{Tn}$, $0<Tg\leq k*n$});

Update the database with new cluster's feature;

}

} // Repeat Next $(SW_t)_{k+1}$

III. EXPERIMENT RESULTS AND ANALYSIS

In order to test the efficiency and precision of TBSC4TS algorithm, the web news pages can be grabbed from the website of http://news.sina.com.cn by web crawler to simulate the continuous text stream, which includes 7 topics and 2650 web pages as testing documents with a certain velocity. In the initialization phase, the setting parameters of system shows as follow: the length, velocity and size of SW_t are *p, v* and SBW_t, respectively. The threshold in DBSCAN is *Eps=5, MinPts=3* and *STH=0.2*, respectively.

Test 1: the Analysis of Execution Time in TBDC4TS

TBDC4TS and Single-pass algorithms are utilized to compare the execution time with the variation of SBW_t from 30 to 450 by a stable interval of 30. The results of the experiment are illustrated in figure 2. The TBDC4TS algorithm has 22.8 times executing efficiency than Single-pass, especially, when the text data reaches to 450 documents, the consume time about two algorithm is 3.92s and 89.4s, respectively.

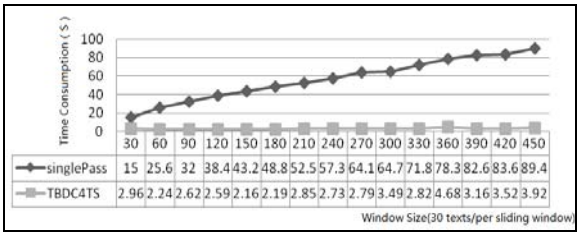
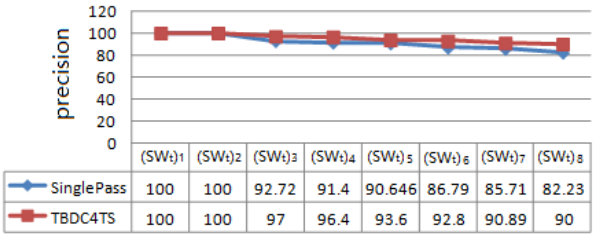


FIGURE II. THE COMPARISON FOR EXECUTION TIME BETWEEN TBDC4TS AND SINGLE-PASS WITH THE INCREASING OF DATA VOLUME.

Test 2: the Comparison for Cluster Qualify between TBDC4TS and Single-pass

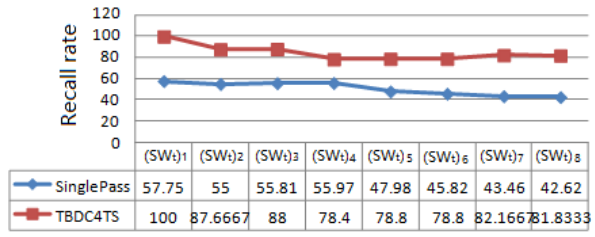
The results of the experiment for the precision rate between TBDC4TS and Single-pass algorithm are illustrated in figure 3. With the sliding of the Time Window, the precision rate of both of two algorithms, which all adopt four-tuples text feature vectors to describe the document, also decrease in a certain degree, but the precision of TBDC4TS algorithm still is higher than Single-pass. Especially, owning to batch processing in each window and multi-phase clustering, the influence by the sequence of text stream is avoided in TBDC4TS algorithm, but is still disturbed in Single-pass algorithm.



	$(SW_t)_1$	$(SW_t)_2$	$(SW_t)_3$	$(SW_t)_4$	$(SW_t)_5$	$(SW_t)_6$	$(SW_t)_7$	$(SW_t)_8$
SinglePass	100	100	92.72	91.4	90.646	86.79	85.71	82.23
TBDC4TS	100	100	97	96.4	93.6	92.8	90.89	90

FIGURE III. THE COMPARISON FOR PRECISION RATE BETWEEN TBDC4TS AND SINGLE-PASS.

Results of recall rate in figure 4 show that the recall rate of TBDC4TS has about two times higher than Single-pass. With the increasing of text documents, the new clusters in each of sliding time window (SW_t) should be merged into existed clusters. If the density of existed clusters is sparser, which means the topics of text is lower interrelated and scattered distribution, and then the recall rate in clustering also will be decreased. The experiments results in above mentioned shown that the TBDC4TS algorithm has higher efficiency and performance than Single-pass with multi-phase clustering method.



	$(SW_t)_1$	$(SW_t)_2$	$(SW_t)_3$	$(SW_t)_4$	$(SW_t)_5$	$(SW_t)_6$	$(SW_t)_7$	$(SW_t)_8$
SinglePass	57.75	55	55.81	55.97	47.98	45.82	43.46	42.62
TBDC4TS	100	87.6667	88	78.4	78.8	78.8	82.1667	81.8333

FIGURE IV. THE COMPARISON FOR RECALL RATE BETWEEN TBDC4TS AND SINGLE-PASS.

IV. CONCLUSIONS

In this paper, a TBDC4TS algorithm is proposed to modify the processing of text stream by sliding time window. When the web news pages are captured continuously by web crawler and formed a text stream. The TBDC4TS algorithm utilizes the dynamic sliding window to split the text stream into continuous segmentation, which merged the topic- based text clustering and the Single-Pass clustering together. In each of sliding time window (SW_t), there are a set of web news pages related to the velocity of stream and the size of sliding window. Furthermore, framework and core modules of TBDC4TS algorithm are designed to adopt a multi-phase clustering method with micro-clustering in sliding time window and macro-clustering in single-pass engine. The results of experiments, used 2650 web news pages in the website of http://news.sina.com.cn to form a simulate text stream by web crawler, illustrated that the TBDC4TS algorithm has 22.8 times executing efficiency than Single-pass, and has higher clustering qualify, including the precision and recall rate, than Single-pass.

However, with the increasing volume of text stream, the next research works of the paper is focusing on how to modify the TBDC4TS algorithm to be perfect for the situation about reality of social public sentiment analysis and providing a real-time early warning for making decision. Maybe, the optimization of efficiency and performance about this algorithm exist a lot of challenges for future work.

ACKNOWLEDGEMENTS

This paper is joint supported by “2012 National Torch Plan Project in China (2012GH571817)”, “2013 Shanxi Province Key Scientific and Technological Project (2013KRZ10 and 2013K06-20)” and Yulin City Scientific project (2012cxy3-2).

REFERENCE

[1] Huang Lei, Mining Stream Data: A Survey [J], Journal of Software (in Chinese), Vol. 15(1): p. 1~7, 2004;

[2] A. Forestiero, C. Pizzuti, G. Spezzano, A single pass algorithm for clustering evolving data streams based on swarm intelligence[J], Data Mining Knowledge Discovery, Vol. 26:p.1~26, 2013;

[3] A. Arasu and G. Manku, Approximate counts and quartiles over sliding windows [C], the Processing of the 2004 ACM Symp. Principles of Database Systems, p.286-296, June 2004;

[4] M. Oyamada, H. Kawashima, H. Kitagawa, Data Stream Processing with Concurrency Control[J], SIGAPP Applied Computing Review, Vol.13(2): p.54~64, June 2013;

[5] C. Junghans, M. Karnstedt, M. Gertz, Quality-driven Resource-adaptive Data Stream Mining?[J], SIGKDD Explorations Newsletter,Vol.13 (1): P.72~82, 2011;

[6] Zhu Hengmin, Zhu Weiwei, Study on Web Topic Online Clustering Approach Based on Single-pass Algorithm[J], New Technology of Library and Information Service(in Chinese), Vol.12, p.52-57, 2011;

[7] Shui Yidong, Qu Youli, Huang Houkuan, A New Topic Detection and Tracking Approach Combining Perodic Classification and Single-pass Clustering[J], Journal of Beijing Jiaotong University (in Chinese), Vol.33 (5): p.85-89, 2009;

[8] Yin Fengjing, Xiao Weidong, Ge Bin, etc., Incremental Algorithm for Clustering Texts in Internet-oriented Topic Detection[J], Application Research of Computers, Vol. 28(1): 249-252, 2011;

[9] Li Na, Xing Changzhen, Density-based Data Stream Clustering Algorithm over Time-based Sliding Window[J], Journal of Computer Applications, Vol. 31(5): 1363-1366, 2011.