Draft Review State of The Art

May 16, 2017

Abstract

1 Introduction Draft

The infiltration of Internet into day to day applications led to the explosion of data within different acitivty sectors. Therefore the need to group these data and extract pertinant information became necessary. Recent studies have shown that the big four (Google, Amazon, Microsoft and Facebook) alone hold more than 1200 petabytes of data, 1 petabyte is equal to 1000 terabytes and each terbayte to 1000 gegabytes, excluding other big data holders as DropBox...
The increase in both volumr and variety of data requires advances in methodology to automatically understand, process and summarize the data[1].This growth of the amount of information being circulated between users all over the globe made the need for grouping this information really urgent. This kind of grouping is called data analysis which can be concerned with predictive modeling: Being given some training data we want to predict the behavior of the unseen test data. This task is defined as learning, often there is a clear distinction between learning problems that are defined into two categories

- Supervised learning also known as classification, in the context of artificial intelligence (AI) and machine learning, is a type of system in which both input and desired output data are provided. Input and output data are labelled for classification to provide a learning basis for future data processing.
- Unsupervised learning also known as clustering, which is a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labeled responses.

In other words, the difference between the methods presented before is that for classification the class labels are known by the user. On the other hand clustering is more challenging than classification for the simple reason that the class labels ar not predefined and they will be known over time. Semi-supervised

learning which is a hybrid model[2] consisting on a combination of classification and clustering: only a small set of the training data is labeled.

Every second, approximately 6,000 tweets are tweeted; more than 40,000 Google queries are searched; and more than 2 million emails are sent, according to Internet Live Stats, a website of the international Real Time Statistics Project. This information leads us to the conclusion that Big Data is in constant for mining and therefore clustering and classification. The challenge in the process of learning Big Data lies on the fact that most of the applications are online therefore the stream of data is represented by a contious flow. This led to both classification and clustering algorithms that function in an online and incremental way in order to be able to handle continuous data streams.

The aim of these learning techniques is to extract potentially useful knowledge from data streams which is a big challenge since most of the data mining techniques suppose that there is a finite amount of data that can be physically stored and analyzed. For data stream mining, however, the successful development of online incremental clustering and/or classification methodologies has to take into account the foollowing restrictions[3]:

- Data objects arrive contiously;
- There is no specific order of arrival of the data;
- The size of the data stream cannot be predefined;
- Important data can be discarded over time;
- The distribution may change over time;

The rest of this paper will be organised in the follow way: in section 2 we will include the extended definition of clustering, a historical overview of the clustering algorithms and there development till today, and presentation of the latest incremental online clustering technologies. Section 3 will include an extended definition of data classsification, a historic overview of classification methods and the lastest online incremental classification methods along with there necessity.

2 Static Data Analysis

The rest of the paper will be organised as follows: Detailed presentation of static data, extended definition of both clustering and classification of data.

2.1 Static Data Definition

Data can be divided into two major types: Static data which can be defined by the fact that these data occupy a limited space memory wise and this type of data doesn't vary with time. On the other hand there is dynamic data streams which will be presented in detail throughout this paper.

2.2 Clustering definition

Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. A loose definition of clustering could be the process of organizing objects into groups whose members are similar in some way. A cluster is therefore a collection of objects which are similar between them and are dissimilar to the objects belonging to other clusters.

An alternative definition of clustering can be defined by the following: givern a representation of n objects, find K groups based on a measure of similarity such that the measure of similarity between the data that belong to a same cluster presents a high score while the same parameter should show a low value when the case is represented by data in different groups.

In the current digital era, according to (as far) massive progress and development of the internet and online world technologies such as big and powerful data servers, we face a huge volume of information and data day by day from many different resources and services which were not available to humankind just a few decades ago. Services over the world wide web are providing big masses of data: such as Facebook, Twitter, Youtube... that should be handled, therefore the need of clustering algorithms that can group these data and help make an extraction of useful information.[4]

A search via google Scholar[5] for the sequence data clustering has gotten 3,1 million scientific documents attached to this domain. This vaste litterature shows the importance of data clustering and taking into consideration from 2013 509 thousands results which corresponds to 16 percent of the total documents.

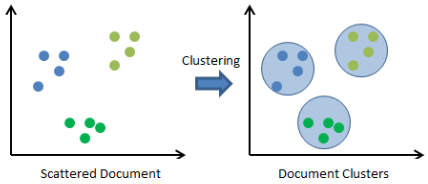

Figure 1: Clustering: The process of regrouping showing high similarities into separate coherent groups

From basic infromation retrieval applications to resolving cases in forensics data clustering can be considered vital for many reasons:

- Permits the extraction of the underlying structure of the data being processed: gaining insight into the data, generating hypotheses, detecting anomalies...
- Another reason relies in determining the level of similarity among the studied data.

- Summarizing the data therefore being able to compress it in order to get the general idea with a considerable reduction within space and time.

2.2.1 From then till now

The term clustering first was used in 1954 in the anthropological domain. Several nominations existed throughout the time for the term clustering depending on the domain in which it was used[6]. Multiple books have treated the subject of clustering in detail and were cited enormously by researchers for their authenticity and contributions. Those books gave a deeper image of what clustering is and how it can be used in the aspects of daily life as well as in research:Hartigan [7] defined in detail what clustering is in his book and discussed multiple application domains for its application. Since then the link between clustering and data mining was made and became a subject of research and study and was best elaborated in Han and Kamber 2000 [8]. Next to data mining machine learning, which can be presented as a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of computer programs that can change when exposed to new data, also used the presence of data clustering in order to help develop artificial intelligence algorithms aiming to achieve the automation and independance of computers especially in pattern recognition which is well described in [9].

2.2.2 K-means

One of the most influencial clustering algorithms is K-means. This algorithm involves randomly selecting K initial centroids where K is a user defined number of desired clusters. Each point is then assigned to a closest centroid and the collection of points close to a centroid form a cluster. This methodology was first discovered in the year of 1956[10]. This methodology has been studied throughout history within multiple articles [11], [12]...

Even though K-means has been originally presented over 50 years ago it is still widely used and was included in new ways of developing clustering methods. In the following, there will be a presentation of how does K-means work.

K-Means Algorithm The K-Means algorithm in detail can be defined as follows:

- Definition of the number of clusters. This number in the original form of K-means have to be chosen by a human factor from outside.
- Initialization of the centroids that define the K-means clustering algorithm.
- Calculate the distance between each centroid and the data that is used in order to see for each data point which cluster is the most appropriate.
- Relate the items to the closest centroid.

2.3 Data classification definition

Data classification is the process of sorting and categorizing data into various types, forms or any other distinct class. Data classification enables the separation and classification of data according to data set requirements for various business or personal objectives. It is mainly a data management process.

The differences between data classification and data clustering relies on the fact that in the process of data classification labels of the potential regroupments of data are already known and defined by a certain human intervention while in data clustering data are grouped into clusters based on their similarities[13].

2.3.1 Historical development of data classification

Classification is both an ancient discipline (Aristote's classification of animals, plants and other objects) and a modern one[14]. The idea of classification as presented before started in ancient times in order to have a certain order within the normal life. With time this functionality evolved in an asymptotic manner in a way that data classification is now used within multiple life aspects. Within the medical domain multiple data mining techniques are used in order to advance in the classification of tumerous patients images in order to help prescribe the right treatment.

2.3.2 SVM

Compared to K-means in relation with data clustering, SVM (Support Vector Machines) are considered one of the most important data classification techniques. A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples.

SVM is considered as one of the most powerful method for classification and performs well in lots of various applications[15].Support Vector Machines (SVMs), introduced by Vapnik and his co-workers[16][17]. SVMs are efficient to handle large-scale classification problems and achieve great success in many applications, such as handwritten digit recognition, text categorization or face detection[18].

Originally, SVM is designed for binary classification by finding an hyperplane with a maximum margin between two classes (cf. fig2). Then, binary SVM has been extended to solve multi-category problems. A brief review of binary SVM and several methods of multi-category SVM will be presented in the next sections.

Binary SVM Binary SVM is initially designed as a function classifying two sets of linearly distinguishable data.

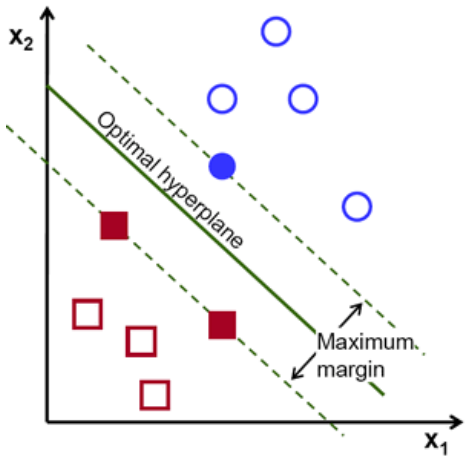

Figure 2: Clustering: The process of regrouping showing high similarities into separate coherent groups

Multi-class SVM There are two main categories of multi-category SVM. One is constructed by combining several binary classifiers, e.g., one-against-one SVM and one-against rest SVM. The other has only one classifier, in which all data are treated by one optimization formulation, as all-against-all method[19].

One againt-rest SVM One-against-rest method is probably the earliest approach implemented for multi- class classification. Figure 3 shows a simple example of one-against-rest SVM applied to three class recognition. There are three classifiers: Class1 vs Class(2,3); Class2 vs Class(1,3) and Class3 vs Class(1,2). Such kind of multi-class SVM is easy to understand and to compute. However, it exists large areas of difficult decision (overlapping of all areas), which are colored on Figure 2.1. In these case, the usual is to consider the classifier that has the largest margin as the "safer" decision.

One-against-one SVM The one-against-one method is introduced in[20] , which constructs binary SVM classifiers for all pairs of classes, such as C1 against C2, C2 against C3, etc. For K class problem, the total number of binary SVMs is K(K1)/2.

All-against-all SVM Similar to binary SVM, all-against-all method solves a K class problem by addressing a single quadratic optimization problem of size (K-1)/n. That is to say,it needs to find an optimal hyperplane for separating all classes[19].

The mathematical development of the SVM all-against-all formulas is available and well described in[21].

Three kinds of multi-class SVM approaches have been presented. As far as

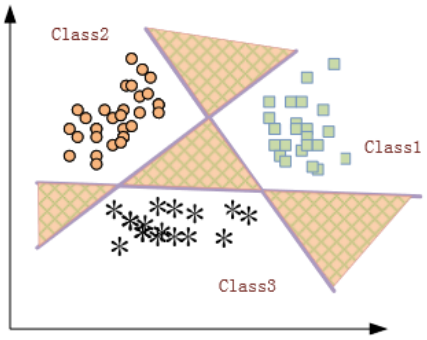

Figure 3: Diagram of one-against-rest SVM applied to a three-class classification problem

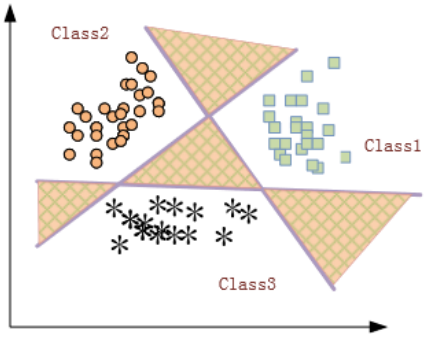

Figure 4: Diagram of one-against-rest SVM applied to a three-class classification problem

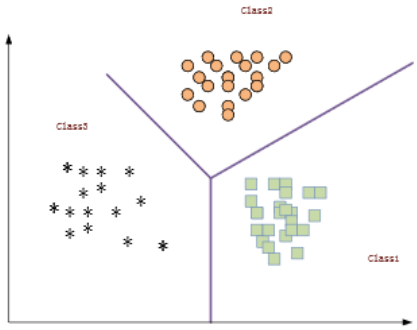

Figure 5: Diagram of one-against-rest SVM applied to a three-class classification problem

the training cost is concerned, one-against-rest SVM is preferable, because it needs only K binary SVMs. However, it has a largest areas of difficult decision, as shown by the comparison of Figures 2.1, 2.2 and 2.3. Compared to one-against-all SVM, one-against-one SVM algorithm is more time-consuming. However, it generally performs better and is more suitable for practical use. Nevertheless, all these methods are usually applied in offline applications, in which test step can begin only when training step has finished and all parameters of SVMs have been chosen[22].

2.4 Application on Static Data

Multiple examples can be given on how clustering can be applied in static data domains:

- One major clustering client is image processing. Having a finite number of images, multiple needs can exist weither it's histogramm clustering in order to recognise the most used colors or even to categorize the objects presented in the images.[23].
- The medical world presents itself as an essential candidate[24]. Application vary from normal data extraction in order to improve patient's conditions to studying genome data in genetical engineering[25].
- In the world of commerce clustering is used to regroup different types of customers in order to target them with marketing strategies built within their preferences.

Other aspects of daily life can also be included in this list of activities that benefit of clustering.

3 Onine data streaming study

Data stream mining is an active research area that has recently emerged to discover knowledge from large amounts of continuously generated data. For the last decade, we have seen an increasing interest in managing these massive, unbounded sequences of data objects that are continuously generated at rapid rates, the so-called data streams[26].

Applications of data streams include mining data generated by sensor networks, meteorological analysis, stock market analysis, and computer network traffic monitoring, just to name a few. These applications involve data sets that are far too large to

t in main memory and are typically stored in a secondary storage device.

3.1 Online incremental data mining needs

For data stream mining, however, the successful development of algorithms has to take into account the following restrictions:

- Data objects arrive continuously.
- There is no control over the order in which the data objects should be processed.
- The size of a stream is (potentially) unbounded.
- Data objects are discarded after they have been processed. In practice, one can store part of the data for a given period of time, using a forgetting mechanism to discard them later.
- The unknown data generation process is possibly non-stationary, i.e., its probability distribution may change over time.

In conclusion of what is presented before, we can conclude that a need is presented in the form of data mining techniques that can at the same time perform in an incremental way and without going offline. These two factors have been the major study element of multiple techniques divided between supervised (data classification) and unsupervised (data clustering) technologies.

The rest of this paper will be presented as follows: Presentation of unsupervised learning techniques (data clustering) which function in an online incremental way, then the presentation of supervised learning techniques (data classification) which function in the same way followed by an analytical study of what's presented in this section.

3.2 Data stream clustering techniques

In this section multiple data clustering techniques will be presented and what these methodologies have in commun is the fact that the authors of these algorithms aimed to achieve what is close to optimal online incremental data clustering.

3.2.1 DBSCAN

One of the pioneer methodologies in data clustering technologies is DBSCAN. DBSCAN was presented the first time in 1996 [27] where this algorithm presented the solution to multiple needs as dicovering clusters with arbitrary shape having small apriori knowledge of the arriving data.

This algorithm is essential in order to understand how will the data stream clustering algorithms function, since the majority of the newly discovered algorithms are related to DBSCAN in one way or another.

3.2.2 Data Stream algorithms

Data stream mining is an active research area that has recently emerged to discover knowledge from large amounts of continuously generated data. In this context, several data stream clustering algorithms have been proposed to perform unsupervised learning.

DenStream Multiple data stream clustering algorithms have been presented recently in order to achieve what is close to optimal online incremental data clustering. DenStream[28] was first presented in 2009 in order to fulfill multiple needs in data mining:

- No assumption on the number of clusters. The number of clusters is often unknown in advance. Furthermore, in an evolving data stream, the number of natural clusters is often changing.
- Discovery of clusters with arbitrary shape. This is very important for many data stream applications. For example, in network monitoring, the distribution of connections is usually irregular. In environmen observation, the layout of an area with similar environment conditions could be any shape.
- Ability to handle outliers. In the data stream scenario, due to the influence of various factors, such as electromagnetic interference, temporary failure of sensors, weak battery of sensors, etc., some random noise appears occasionally.

Different approaches were presented trying to handle data streams[29][30][31] but these methods were limited to knowing the amount of data that will be present as the entry of the system. Other methodologies didn't take into consideration the clusters of arbitrary shape either if they were one-pass methods[32][33][33] or even evolving methods[34]...

DenStream consists of two major steps: The first is merging (cf. Algorithm 1)

Algorithm 1 Merging (p)

```
Try to merge p into its nearest p-micro-cluster c_p;
if r_p (the new radius of c_p) ≤ ε then
    Merge p into c_p;
else
    Try to merge p into its nearest o-micro-cluster c_o;
    if r_o (the new radius of c_o) ≤ ε then
        Merge p into c_o;
        if w (the new weight of c_o) > βμ then
            Remove c_o from outlier-buffer and create a
            new p-micro-cluster by c_o;
        end if
    else
        Create a new o-micro-cluster by p and insert it
        into the outlier-buffer;
    end if
end if
```

Figure 6: Clustering: The process of regrouping showing high similarities into separate coherent groups

Zfter merging is done DenStream is applied over a certain period of time in order to have a check on the data periodically(cf. Algorithm 2).

Algorithm 2 DenStream (DS, ϵ, β, μ, λ)

$T_p = \lceil \frac{1}{\lambda} \log(\frac{\beta\mu}{\beta\mu - 1}) \rceil$;
Get the next point p at current time t from data stream DS;
Merging(p);
if (t mod T_p)=0 **then**
  for each p-micro-cluster c_p **do**
    if w_p(the weight of c_p)< $\beta\mu$ **then**
      Delete c_p;
    end if
  end for
  for each o-micro-cluster c_o **do**
    $\xi = \frac{2^{-\lambda(t - t_o + T_p)} - 1}{2^{-\lambda T_p} - 1}$;
    if w_o(the weight of c_o)< ξ **then**
      Delete c_o;
    end if
  end for
end if
if a clustering request arrives **then**
  Generating clusters;
end if

Figure 7: Clustering: The process of regrouping showing high similarities into separate coherent groups

Clustream The CluStream method is a method of clustering data streams, based on the concept of microclusters. Microclusters are data structures which summarize a set of instances from the stream, and is composed of a set of statistics which are easily updated and allow fast analysis.

CluStream has two phases. In the online phase, a set of microclusters are kept in main memory; each instance coming from the input stream can then be either appended to an existing microcluster or created as a new microcluster. Space for the new microcluster is created either by deleting a microcluster (by analyzing its expiration timestamp) or by merging the two closest microclusters. The offline phase will apply a weighted k-means algorithm on the microclusters, to obtain the final clusters from the stream[**?**].

References

[1] Anil K Jain. Data clustering: 50 years beyond k-means. *ECML/PKDD (1)*, 5211:3–4, 2008.

[2] Olivier Chapelle, Mingmin Chi, and Alexander Zien. A continuation method for semi-supervised svms. In *Proceedings of the 23rd international conference on Machine learning*, pages 185–192. ACM, 2006.

[3] Jonathan A Silva, Elaine R Faria, Rodrigo C Barros, Eduardo R Hruschka, André CPLF de Carvalho, and João Gama. Data stream clustering: A survey. *ACM Computing Surveys (CSUR)*, 46(1):13, 2013.

[4] Adil Fahad, Najlaa Alshatri, Zahir Tari, Abdullah Alamri, Ibrahim Khalil, Albert Y Zomaya, Sebti Foufou, and Abdelaziz Bouras. A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE transactions on emerging topics in computing*, 2(3):267–279, 2014.

[5] data clustering. https://scholar.google.fr/. Accessed: 2017-01-26.

[6] Anil K Jain and Richard C Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.

[7] John A Hartigan and JA Hartigan. *Clustering algorithms*, volume 209. Wiley New York, 1975.

[8] Jiawei Han and Micheline Kamber. Data mining: concepts and techniques (the morgan kaufmann series in data management systems). 2000.

[9] Christopher M Bishop. Pattern recognition. *Machine Learning*, 128:1–58, 2006.

[10] Hugo Steinhaus. Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, 1(804):801, 1956.

[11] Geoffrey H Ball and David J Hall. Isodata, a novel method of data analysis and pattern classification. Technical report, DTIC Document, 1965.

[12] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[13] data classification definition. https://www.techopedia.com/definition/13779/data-classification/. Accessed: 2017-04-16.

[14] Phipps Arabie, Lawrence J Hubert, and Geert De Soete. *Clustering and classification*. World Scientific, 1996.

[15] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[16] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[17] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[18] Alex M Andrew. An introduction to support vector machines and other kernel-based learning methods by nello christianini and john shawe-taylor, cambridge university press, cambridge, 2000, xiii+ 189 pp., isbn 0-521-78019-5 (hbk,£ 27.50)., 2000.

[19] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, 13(2):415–425, 2002.

[20] Stefan Knerr, Léon Personnaz, and Gérard Dreyfus. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In *Neurocomputing*, pages 41–50. Springer, 1990.

[21] Erin J Bredensteiner and Kristin P Bennett. Multicategory classification by support vector machines. In *Computational Optimization*, pages 53–79. Springer, 1999.

[22] Yanyun Lu. *Online classification and clustering of persons using appearance-based features from video images: application to person discovery and re-identification in multicamera environments*. PhD thesis, Lille 1, 2014.

[23] Zhenyu Wu and Richard Leahy. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 15(11):1101–1113, 1993.

[24] Krzysztof J Cios and G William Moore. Uniqueness of medical data mining. *Artificial intelligence in medicine*, 26(1):1–24, 2002.

[25] Pierre Baldi and G Wesley Hatfield. *DNA microarrays and gene expression: from experiments to data analysis and modeling*. Cambridge university press, 2002.

[26] Varun Chandola, Olufemi A Omitaomu, Auroop R Ganguly, Ranga R Vatsavai, Nitesh V Chawla, Joao Gama, and Mohamed M Gaber. Knowledge discovery from sensor data (sensorkdd). *ACM SIGKDD Explorations Newsletter*, 12(2):50–53, 2011.

[27] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[28] Feng Cao, Martin Estert, Weining Qian, and Aoying Zhou. Density-based clustering over an evolving data stream with noise. In *Proceedings of the 2006 SIAM international conference on data mining*, pages 328–339. SIAM, 2006.

[29] Alexander Hinneburg, Daniel A Keim, et al. An efficient approach to clustering in large multimedia databases with noise. In *KDD*, volume 98, pages 58–65, 1998.

[30] Wei Wang, Jiong Yang, Richard Muntz, et al. Sting: A statistical information grid approach to spatial data mining. In *VLDB*, volume 97, pages 186–195, 1997.

[31] Charu C Aggarwal, Jiawei Han, Jianyong Wang, and Philip S Yu. A framework for projected clustering of high dimensional data streams. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 852–863. VLDB Endowment, 2004.

[32] Moses Charikar, Liadan O'Callaghan, and Rina Panigrahy. Better streaming algorithms for clustering problems. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 30–39. ACM, 2003.

[33] Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. Clustering data streams: Theory and practice. *IEEE transactions on knowledge and data engineering*, 15(3):515–528, 2003.

[34] Charu C Aggarwal, Jiawei Han, Jianyong Wang, and Philip S Yu. A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*, pages 81–92. VLDB Endowment, 2003.

[35] Definition of clustrem algorithm. http://huawei-noah.github.io/streamDM/docs/CluStream.html. Accessed: 2017-05-16.