A Comprehensive Review and Open Challenges of Stream Big Data

Bharat Tidke and Rupa Mehta

Abstract Research in big data becomes pioneer in the field of information system. Data stream is well-studied problem in traditional data mining environment, but still needs exploration while dealing with big data. This paper mainly reviewed different research activities, scientific practice, and methods which have been developed for streaming big data. In addition, examine well-known real-time platforms which are evolving to handle streaming problem and having existing similarity in terms of usage of main memory and distributed computing technologies for non-real-time data. Finally, summarize open issues and challenges faced by current technologies while acquisition and processing of big data in real time.

Keywords Big data · Stream data · Distributed mining

1 Introduction

Due to rapid growth of smart phones, access to Internet becomes quite easy, results in large amounts of unstructured data, which has been collected and stored from different areas of society [1]. In addition, real-time systems such as sensor-based technologies normally generate streams of data, which require quick storing as well as processing of incoming data. However, such data streams pose new features as compares to traditional stream data. Many applications such as traffic management, log data from Web search engines, Twitter, electronic mail also generates high volumes of stream data with velocity, which is difficult to handle with existing data streaming techniques [2, 3]. In past decade, big data comes into picture and can be

B. Tidke (✉) · R. Mehta
Department of Computer Engineering, SVNIT, Surat, India
e-mail: batidke@gmail.com1

R. Mehta
e-mail: rgm@coed.svnit.ac.in

M. Pant et al. (eds.), *Soft Computing: Theories and Applications*,
Advances in Intelligent Systems and Computing 584,
https://doi.org/10.1007/978-981-10-5699-4_10

defined in terms of its characteristics volume, velocity, variety, value, and veracity [3–5]. Many researchers proposed different techniques, tools, and complex processes for getting insight into different characteristics of big data. Exploring stream data with velocity is key challenge in big data research, which has been focused by many researchers but still has potential to explore for many applications and domains. Also, processing of stream data in real time differs from non-real-time data processing, since data has to be analyzed based on historic stored data, before itself gets stored for further analysis and prediction of upcoming streams.

Motivation
Data has been generated and acquired at rapid speed which involves volume with it ultimately create challenge to develop methods which must be automated and can respond quickly to make decision in specified time. Since size of data is too big, such data needs to be moved and stored in distributed environment for further computation as traditional data warehouses are ill suited. Further analysis of such data using classic OLAP cube also does not work and replaced by distributed storage environment such as Hadoop which uses master–slave architecture for storing data, also map-reduced technique for processing data in batches and NoSQL databases which uses different storage techniques having columns, graphs, documents, key-value stores which can work on top of Hadoop to make it suited for streaming big data. Some assumption that has been followed by traditional system while dealing with stream data mining can be solved using distributed data processing frameworks or tools for handling the problem of big data.

- Possible to collect and store whole data stream, not the sample or summaries of data.
- Integration and indexing of data in real time irrespective of the format in which they came.
- Velocity with which data come can be processed using distributed streaming algorithm in real time and stored in distributed fashion for improving existing model for further analysis.
- Analyzing existing or past data is crucial while making targeted future prediction, but decision based on operational or transactional data needs real-time analysis has to be processed in parallel with low-latency time.

2 Related Work

Data stream can be conceived as a continuous and changing sequence of data that continuously arriving at a system to store and process. Stream data processing deals with some or all data input as one or more continuous data stream.

2.1 Data Stream Mining

They explained different factors which are necessary to mine information from streaming data including time window which further can be divided into landmark window basically takes whole new data stream as a window instead of sample and considered them equally important which may cause problem to build model with limited memory. Another one is sliding window, one of the most used windowing technique in the field of stream mining that only takes recent data stream and discarded the old ones, and also, it is flexible depending upon the accuracy needed by the model makes it popular. Next one is fading window usually assigned weight to the data according to its arrival time newer one having higher as compare to older one and finally Tilted time window lies between sliding and fading window in terms of its variance.

2.2 Big data Stream Mining

In past few years, mostly research is based on collecting and storing data due to growth of Web 2.0 technologies and increase in bandwidth for data transfer. Many sources of data have been evolved and are mostly need real-time analysis of such stream data and also information extracting algorithms to analyze it. Even Hadoop has been used by Yahoo in its earlier days to collect, store, and analyze large volume of click stream data, which later used by ecommerce enterprises to solve the problems of customers, while choosing product and other followed product normally user tend to purchase. In addition, recommending similar products they want to purchase in future is based on their purchased experience. Many data mining algorithms have been proposed to overcome the challenges of stream data as seen in above section, but to overcome volume, velocity, and volatility challenges [3], a standard framework based on Lambada [6] architecture having different phases ranging from stream data collection to data visualization has been overview.

Nowadays, concept of "**Smart Cities**" are on its evolving stages, and data gathered using different technologies such as sensors from different aspects such as users location, social gathering information, ITS, temperature changes produce data. Similarly social media sites such as Twitter Facebook, Linkedin also generates large amount of data are some of main sources of big stream real-world data [7]. After acquisition of data from different sources the most important phase in mining any unstructured data is preprocessing unfortunately it has not been yet explored fully in terms of big data. Since data in real world is dirty as well as noisy, same is applied for real streaming data which makes it worse to analyze data which has not been preprocessed.

Beginning of any streaming processing paradigm was based on hidden information that comes with incoming data that further can be used to obtain useful results to do analysis. In this process, since data is arriving continuously and in

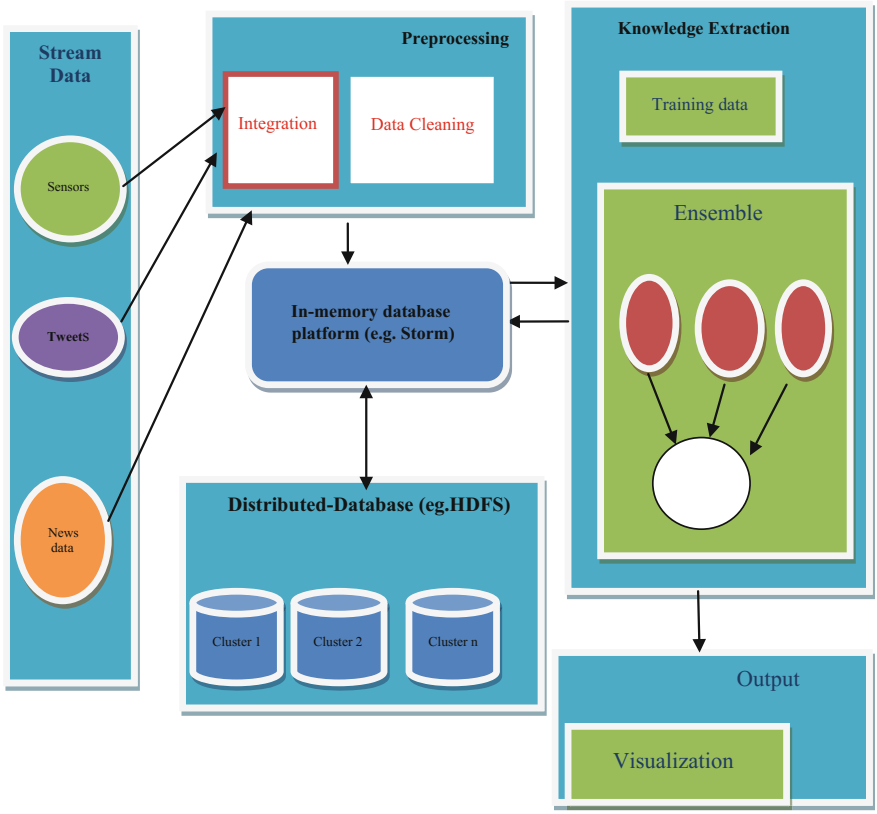

Fig. 1 Architecture for handling stream big data

huge amount, only a small fraction of stream data is stored in limited memory databases and process using stream processing system such as storm or kafka as shown in Fig. 1. Further, it can be stored in large and distributed databases such as HDFS for future use. Many machine learning algorithms have been used to extract hidden information from stream data. Different approaches present by authors has been discussed below and summarized in Table 1.

Rutkowski et al. [8] in one of his paper suggested that algorithms based on Hoeffding's bound which considered as one of the most used decision-tree technique in mining data stream needs to be revised and proposed a method using Mcdiarmid's inequality to split node in a tree by picking correct attribute, and they performed several experiments and evaluated their result using splitting measures Gini index and information gain.

Limitation of this method is that to split node among given n node, it needs to scan huge number of data elements before selecting right attribute, later this limitation has been overcome in [9] in that they used statistical method for selecting attribute to split node among given n node based on Taylor's theorem and

Table 1 Summary of various approaches proposed on stream big data

Authors	Handles big data	Platform	Technique	Model or algorithm	Evaluation measure	Dataset	Environment
Vu et al. [13]	Yes	SAMOA	Regression	VAMR (vertical adaptive model rules) and HAMR (hybrid adaptive model rules)	Mean absolute error (MAE) and root-mean-square error (RMSE)	UCI machine learning repository	Distributed
Bifet et al. [19]	Yes	SAMOA	Classification	Randomized ensembles	McNemar's test, sign test and Wilcoxon's signed-rank test. Statistic, kappa-temporal statistic ADWIN	UCI machine learning repository	Centralized
Fegaras et al. [14]	Yes	Hadoop and spark	Apache MRQL	Incremental	Group by, join-group by, and *k*-means clustering	Synthetic data	Distributed
Marron et al. [15]	No	GPU	Ensemble random forest	GVFDT (very fast decision trees on GPUs)	Classification accuracy and time	Synthetic data	Centralized
Khalilian et al. [27]	Yes	Vector space	Clustering	DCSTREAM	Mean difference, std. error, precision, recall, *F*-measure	KDDCUP 99 and synthetic datasets	Centralized
Fong et al. [28]	Yes	MOA	Accelerated particle swarm optimization (APSO)	Swarm search-feature selection (SS-FS)	Accuracy, kappa (kappa statistics), TP-FP rate, precision, recall, *F*-measure	Sensor data	Centralized

(continued)

Table 1 (continued)

Authors	Handles big data	Platform	Technique	Model or algorithm	Evaluation measure	Dataset	Environment
Yun et al. [16]	No	Sliding window	Frequent pattern mining	Weighted erasable pattern mining algorithm suitable for sliding window-based data stream (WEPS)	Runtime, memory usage, and scalability	FIMI repository, synthetic datasets r	Centralized
Agerri et al. [12]	Yes	STORM	Linguistic processors on virtual machines	NLP tools	Performance gain and time	Car dataset and wiki news dataset	Distributed
Duarte et al. [29]	No	MOA	Regression	Adaptive model rules	MAE and root-mean-squared error (RMSE)	UCI repository	Centralized

properties of the normal distribution to test evaluation of splitting criteria and proposed Gaussian decision-tree algorithm to improve the performance of mining streaming data. Again in [10, 11], they proposed firstly (mDT) algorithm based on splitting criteria called misclassification error combined with Gini index for creating tree node, which also decides the accurate attribute for existing and incoming stream data and secondly Decision Trees Based on the Hybrid Split Measure (hDT) which they tested on UCI repository dataset. Agerri et al. [12] presented new distributed and highly scalable architecture for analysis of stream textual news data using natural language processing (NLP). They performed experiment using different distributed pipeline modules on virtual machines and evaluated performance of the system using original incoming streaming news in which documents has been taken in reproducible manner. In addition, some limitation with proposed system still exists, and they suggested which can be solved by using distributed NoSQL databases like MongoDB.

Vu et al. [13] propose streaming algorithm based on AM rules Decision in distributed environment. This is first kind of experiment on adaptive rules in distributed platform for which they used SAMOA open-source software which basically built to deal large-scale data stream, and their main focus is to understand different decision rules in terms of regression. Fegaras [14] proposed framework based on incremental approach for distributed stream data, mainly focuses to improve traditional batch processing as used by map-reduced function in Hadoop, making it iterative incremental batch processing, enables it to store processing data in memory and tested their framework on dataset consist of complex arbitrary values, and also evaluated that their result are accurate instead of approximate. Marron [15] presented use of traditional classification algorithms such as random forest and VFDT for mining large amount of stream data using GPU, making these algorithm to run in parallel to deal with the volume of big data. They compare performance of Very Fast Decision Tree on GPU (GVFDT) and Random Forest algorithms with similar platform such as MOA and VFML having both algorithm, and they found that their results are better in terms of speed as well as accuracy. Yun et al. [16] worked on frequent pattern mining using sliding window technique and proposed algorithm WEPS (Weighted Erasable Pattern mining algorithm on sliding window-based data Streams) in that they assigned weight ti nodes of tree for creation and pruning purpose. The proposed architecture has been divided into two parts; first phase mainly concentrating on sliding window in that tree creation and recreation have been performed, and in second part, they prune the pattern based on weight assigned to it. Zliobaite and Gabrys [17] Proposed automated preprocessing technique based on adaptive technique for three different cases. In each one, different adaptive model has been used for preprocessing data as well for prediction using different techniques such as incremental approach ensemble classifier (Table 2).

Table 2 Summary of various processing models for streaming big data

System/tools	Processing model	Stream type	Operating system	Open source	Built-in language	Supportive languages	Current release/version	Developed at	Available on	Function	Features
Kafka	Batch and real time	Tuples	OS independent	Yes	Clojure	Any language	0.9.6	Back type	Apache software foundation	Distributed real-time computation	1. Secure 2. Multi-tenant deployment
Flink	Batch and real time	Strings	OS independent	Yes	Java and scala	Java, Scala, and Python	0.10.2	Data artisans	Apache software foundation	Distributed real-time computation	1. Low-latency stream processor 2. Flexible operator state and streaming windows
Spark	Batch and streaming	Discretized stream	Windows and Linux	Yes	Scala	Scala, Java, Python, R	1.6.0	UC, Berkeley	Apache software foundation	Large-scale data processing	1. Decentralized hides all cluster management tasks 2. Checkpointing and recovery minimize state loss
S4	Real time	Event	Windows and Linux	Yes	Java	Any language	0.6.0	Yahoo	Apache software foundation	Processing continuous Stream	Flexible deployment
Samza	Batch and real time	Messages	OS independent	Yes	Scala, Java	CQL, Pig	0.10.0	LinkedIn	Apache software foundation	Processing continuous stream	1. Simple API 2. Managed state 3. Fault tolerance 4. Durability 5. Pluggable processor isolation

3 Challenges in Various Domains

There are various challenges have been focused by different authors [18–21] and some of them are.

3.1 GIS

Existing technology available in geographic information system (GIS) is mainly concentrating on conventional databases, dealing normally with static data which limits it when it comes to analyzing big data. A new tools and techniques for GIS in terms of big data is require to meet new changing environment of spatial databases. Liu et al. [22] big data show that how big data can revolutionized the world of GIS system

3.2 Human Mobility Patterns

Due to large usage of smart phones, sensor-based mobiles are in the pockets of millions, so data of each individual and their traveling habits can be explore, but such data comes not only volume based, but with velocity and because of its spatial nature, in terms of variety as well. Gonzalez et al. [23] shows that individual human have high degree of temporal and spatial pattern regularity which can be used in epidemic prevention, emergency response urban planning as well as agent-based modeling [24].

3.3 Space Technology

The Sloan Digital Sky Survey has collected data which compromises of around 500 million photometric observations of objects from the sky which makes job of space scientist easy to get data without sending astronomer's into the sky. But still extracting knowledge from such vast collection of big data is tedious job. Business enterprises are also using big space data to carry their operation in different remote parts of the world [25, 26].

4 Conclusion and Unsolved Issues

Big Data means opportunities for different section of society to grow virtually, and most of the data are continuous in nature which creates research challenges to extract information from such huge volume of stream data. In this paper, we have presented concept of stream big data and highlighted the general architecture which can be for stream big data, also provided a literature survey on numerous techniques and mechanisms for getting information from stream big data. Still there are many issues need to be focused for usage of big data.

- ***Decision science***

Building system for real-time analytics which can transfer data science into decision science becomes vital to cope up with enormous need of today's information system.

- ***Distributed algorithms***

Many frameworks have been developed for distributed computing, but for analyzing and predicting accurate information for such application, there is a need to have distributed data mining algorithms.

References

1. Lohr, S.: The age of big data. New York Times 11 (2012)
2. Fan, W., Bifet, A.: Mining big data: current status, and forecast to the future. ACM SIGKDD Explor. Newsl. **14**(2), 1–5 (2013)
3. Labrinidis, Alexandros, Jagadish, H.V.: Challenges and opportunities with big data. Proc. VLDB Endow. **5**(12), 2032–2033 (2012)
4. Chen, C.P., Zhang, C.-Y.: Data-intensive applications, challenges, techniques and technologies: a survey on Big Data. Inf. Sci. **275**, 314–347 (2014)
5. Chen, M., Mao, S., Liu, Y.: Big data: a survey. Mob. Netw. Appl. **19**(2), 171–209 (2014)
6. Aggarwal, C.: Data streams: models and algorithms. Springer, Berlin (2007)
7. Nguyen, H.-L., Woon, Y.-K., Ng, W.-K.: A survey on data stream clustering and classification. Knowl. Inf. Syst. **45**(3), 535–569 (2015)
8. Rutkowski, L., Pietruczuk, L., Duda, P., Jaworski, M.: Decision trees for mining data streams based on the McDiarmid's bound. IEEE Trans. Knowl. Data Eng. **25**(6), 1272–1279 (2013)
9. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: Decision Trees for mining data streams based on the Gaussian approximation. IEEE Trans. Knowl. Data Eng. **26**(1), 108–119 (2014)
10. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: The CART decision tree for mining data streams. Inf. Sci. **266**, 1–15 (2014)
11. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: A new method for data stream mining based on the misclassification error. IEEE Trans. Neural Netw. Learn. Syst. **26**(5), 1048–1059 (2015)
12. Agerri, R., Artola, X., Beloki, Z., Rigau, G., Soroa, A.: Big data for natural language processing: a streaming approach. Knowl. Syst. **79**, 36–42 (2015)
13. Vu, A.T., De Francisci Morales, G., Gama, J., Bifet, A.: Distributed adaptive model rules for mining big data streams. In: IEEE International Conference on Big Data, pp. 345–353 (2014)

14. Fegaras, L.: Incremental query processing on big data streams. IEEE Trans. Knowl. Data Eng. **28**(11), 2998–3012 (2016). doi:10.1109/TKDE.2016.2601103
15. Marron, D., Bifet, A., De Francisci Morales, G.: Random forests of very fast decision trees on GPU for mining evolving big data streams. ECAI **14** (2014)
16. Yun, U., Lee, G.: Sliding window based weighted erasable stream pattern mining for stream data applications. Future Gener. Comput. Syst. (2016)
17. Zliobaite, I., Gabrys, B.: Adaptive preprocessing for streaming data. IEEE Trans. Knowl. Data Eng. **26**(2), 309–321 (2014)
18. Domingos, P., Hulten, G.: Mining high-speed data streams. In: Proceedings of Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 71–80 (2000)
19. Bifet, A., de Francisci Morales, G., Read, J., Holmes, G., Pfahringer, B.: Efficient online evaluation of big data stream classifiers. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 59–68 (2015)
20. Krempl, G., Žliobaite, I., Brzeziński, D., Hüllermeier, E., Last, M., Lemaire, V., Noack, T., Shaker, A., Sievi, S., Spiliopoulou, M., Stefanowski, J.: Open challenges for data stream mining research. ACM SIGKDD Explor. Newsl. **16**(1), 1–10 (2014)
21. Gaber, M., Zaslavsky, A., Krishnaswamy, S.: Mining data streams: a review. SIGMOD Rec. **34**(2), 18–26 (2005)
22. Liu, J., Li, J., Li, W., Wu, J.: Rethinking big data: a review on the data quality and usage issues. ISPRS J. Photogramm. Remote Sens. (2015)
23. Yue, P., Jiang, L.: BigGIS: how big data can shape next-generation GIS. In: Third International Conference on Agro-geoinformatics (Agro-geoinformatics 2014), IEEE (2014)
24. Gonzalez, M.C., Hidalgo, C.A., Barabasi, A.-L.: Understanding individual human mobility patterns. Nature **453**(7196), 779–782 (2008)
25. SDSS-III: Massive Spectroscopic Surveys of the Distant Universe, the Milky Way Galaxy, and Extra-Solar Planetary Systems, Jan 2008, http://www.sdss3.org/collaboration/description.pdf/
26. Jagadish, H.V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J.M., Ramakrishnan, R., Shahabi, C.: Big data and its technical challenges. Commun. ACM **57**(7), 86–94 (2014)
27. Madjid, K., Mustapha, N., Sulaiman, N.: Data stream clustering by divide and conquer approach based on vector model. J. Big Data **3**(1) (2016)
28. Fong, S., Wong, R., Vasilakos, A.V.: Accelerated PSO swarm search feature selection for data stream mining big data. IEEE Trans. Serv. Comput. **9**(1), 33–45 (2016)
29. Joao, D., Gama, J., Bifet, A.: Adaptive model rules from high-speed data streams. ACM Trans. Knowl. Discov. Data (TKDD) **10**(3), 30 (2016)
30. Beyer, M.A., Laney, D.: The importance of "Big Data: a definition. Gartner, Stamford (2012)