# Streaming Data Analysis: Research and models

**Shrihari Kulkarni**

*Department of Information Science and Engineering, R V College of Engineering, Bengaluru-560059*

---

**Abstract**

Streaming data refers to the kind of data sent continuously and within a small frame of time in chunks. Since the size of data is large and the speed at which it arrives is very frequent, analysis of streaming data. This paper analyses the variety of methods of analysis of streaming data and also With an increase in the number of informational events that organizations generate, it becomes challenging for administrators to manage these events and convert them into actionable insights.

**Literature Survey**

[1]This paper introduced a Markov chain prediction based node scheduling model to analyse large streaming data in live by following three steps: (i) Using markov chain to construct graph of data state transition and predict the varying trend of large streaming data; (ii) The process of selecting appropriate cloud nodes for processing large streaming data based on the result in step(i).(iii)Ensure load balancing to allocate the data to these nodes to ensure synchronously execute the task. The paper also analyses the way in which data arrives and also explores an algorithm to modify it in order to achieve better results.

[2]brings out an algorithm to mine data streams using a clustering algorithm. It initially proposes K means and algorithm and then goes on to suggest an improvised version of the algorithm. The technique distinguishes information focuses that can be successfully packed, information focuses that must be kept up in memory, and information focuses that can be disposed of. The calculation works inside the bounds of a constrained memory cradle. But, the pressure plans utilized by K-Means can present critical overhead. The crucial difference between the two methods is that convergence criteria the number of exchanges is taken to be one and the Incremental K-implies doesn't iterate until the next step is done. Another significant contrast in Incremental K-Means is that introduction is done utilizing worldwide measurements of the informational index as opposed to utilizing a test of k exchanges.

[3] is a case study highlighting the importance of analysis of streaming data in healthcare industry and how it can be used to improve the diabetes management system. It lists out architecture involving the steps of induction, collection, transformation, analytics and presentation phases and brings out a detailed analysis on the characteristics and techniques.

**Streaming Data Use cases**

Streaming data can be used heavily in the manufacturing sector where timely and live data needs to be sent. There is a huge database of things which needs to be maintained and analysed at real time and any anomaly or fault should be detected immediately and corrected.

Streaming data is heavily used in cyber security. If we have a situation of a Denial of Service attack, we might have to tens of millions of data within an instant with no means of getting that back.

Streaming data is also very useful in the healthcare sector. We need patient's real time data to be updated into the system's database at every moment. This includes things like heart rate, blood pressure. Hence different kinds of algorithms need to be used to deal with these.

Weather is another very good example for the usefulness of streaming data. Weather is something which changes multiple times in a day and things like temperature, humidity and other conditions need to be updated in real time to any

weathering system database. This will also help the meteorologists predict any unforeseen circumstances like heavy rain, cyclone etc. and give an idea on the expected climatic condition in the upcoming days.

**Current Approaches to dealing with streaming data and algorithms**

1. K-medoid algorithm: The strategy used here is that the point represented by the cluster is a representative of the entire cluster

i) Initialize: select any random k points of the available n data points.

2. By use of any common distance metric method associate each of the data points to the nearest medoid.

2. BJKST algorithm

The BJKST calculation utilizes a set to keep the examined things. By running Θ (log (1/δ)) free duplicates in equal and returning

The mechanism of these yields, the BJKST calculation (ε, δ) - approximates the F0-standard of the multiset S.

The essential thought behind the inspecting plan of the BJKST calculation is as per the following:

I. Leave B alone a set that is utilized to hold tested things, and B = ∅ at first. The size of B is

O (1/ε2) and just relies upon the estimation parameter ε. 3.3. Tallying Distinct Elements 7

ii. The underlying inspecting likelihood is 1, for example the calculation keeps all things seen so far in B.

iii. At the point when the set B turns out to be full, contract B by expelling about half things and from that point on the example likelihood decreases.

3. K implies calculation.

This is a basic yet incredible calculation. This strategy segments the information accessible into different bunches and the allotments are picked to such an extent that in order to limit the blunders inside the groups.. The calculation functions as follows.

i) Decide on an estimation of k which speaks to the quantity of groups the information is to be isolate into.

ii) Assign all the focuses to one of the group dependent on the least separation measure.

iii) The centroids are refreshed dependent on the current new focuses.

iv) If the bunch to which an article has been allocated has changed as of late at that point proceed with step ii) and iii) else stop.

4. Count min-sketch:

Check Min Sketch is a calculation, where the goal is to know the repeat of standard things, yet we need more space for a full table of counters for each and every possible thing. The central idea is we can hash each moving toward thing a couple of unmistakable ways, and expansion incorporate for that thing in an assortment of spots, one spot for every one of a kind hash. Undeniably, considering the way that each bunch that we use is significantly more diminutive than the amount of stand-out things that we see, it will be essential for more than one thing to have to a particular territory. The trick is that for the any of most typical things, more likely than not, in any occasion one of the hashed regions for that thing will simply have impacts with less fundamental things. That suggests that remember for that territory will be generally controlled by that thing. The issue is the way to find the cell that simply has impacts with less standard things. Furthermore, henceforth to manage this issue, the accompanying calculation was presented.

5. Probe Using min of Counts

Right when we endeavour to find the mean for the mainstream thing, we look in all the spots it hashed to and take the base incorporate that we find in any of them, believing that no progressively realized thing collided with it in those regions. In case at any rate one of those has recently less standard things, we'll get a very better than average count. Since we take the base of the sum of the watches that we find, we understand that we have found the least influenced count. That

is the Count Min Sketch which is a kind of a sketch that is very useful for checking number of occasions of the most standard things we have seen.

6._T_-digest

The thought behind this calculation here is that on the off chance that one needs to process the quantiles of an enormous number of tests, one might need to keep a huge number of these quantiles for various types of subsets of everything So It's an OLAP 3D square sort of a game plan, however for all the circulations. So one can hope to get a guess of the whole one dimensional dissemination of the information.

7. STREAM

Stream is an improvement over the k medoid algorithm It achieves a constant factor approximation in a single pass and a small amount of space. The algorithm is as follows

Algorithm Small-Space(S)

1. Dividing S in disjoint pieces X1.. X.
2. For each of the I , find k centres in Xi. Thereby assigning each point in given Xi to its closest centre.
3. Cluster X' to find k centres.

**Conclusion**

A considerable lot of the calculations examined here can be applied in an assortment of cases relying upon the database and the utilization case we need to tackle. Henceforth having a comprehension of the current database assumes an essential job in picking the calculation we need to utilize. Any alterations, if necessary ought to be done to the information without it losing any of its worth so the full intensity of the calculation can be used. The calculations can be extremely ground-breaking in their spaces and can be utilized to take care of numerous ongoing issues as talked about in the underlying piece of the paper.

**REFERENCES**

1. Yi-Hong Lu and Huang, "Mining data streams using clustering," 2005 International Conference on Machine Learning and Cybernetics, Guangzhou, China, 2005, pp. 2079-2083 Vol. 4, doi: 10.1109/ICMLC.2005.1527288.
2. A. Ara and A. Ara, "Case study: Integrating IoT, streaming analytics and machine learning to improve intelligent diabetes management system," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, 2017, pp. 3179-3182, doi: 10.1109/ICECDS.2017.8390043.
3. M. Pechenizkiy, "Predictive analytics on evolving data streams anticipating and adapting to changes in known and unknown contexts," 2015 International Conference on High Performance Computing & Simulation (HPCS), Amsterdam, 2015, pp. 658-659, doi: 10.1109/HPCSim.2015.7237112.
4. H. Isah, T. Abughofa, S. Mahfuz, D. Ajerla, F. Zulkernine and S. Khan, "A Survey of Distributed Data Stream Processing Frameworks," in IEEE Access, vol. 7, pp. 154300-154316, 2019, doi: 10.1109/ACCESS.2019.2946884.