San José State
UNIVERSITY

CMPE- 257 Machine Learning

Individual Research Project

Streamline Data Analysis with Machine Learning Algorithms

Mounika Batthina

010693566

Department of Computer Engineering

Fall 2017

Table of Contents

ABSTRACT

Complex and large measures of data are being created where conventional data processing applications are insufficient to manage them. However, there is a developing requirement for extracting data from operational information in real time, which is vital in quick evolving circumstance. The quicker one can process bits of knowledge from real time data, the more noteworthy the advantage in driving quality, taking, lessening costs, and expanding proficiency.

Streamline processing is to enhance the proficiency of a procedure, business or companies by reducing or eliminating unwanted steps, utilizing advanced technologies, or adopting different strategies. Normally a lot of data stays unused because maintaining analytics on data to make insights requires excess time. Streamline Analytics is an on-request, cloud-based analytics, which changes data into simple to-utilize, metrics driven analysis.

The predominant Technologies and tools that are being utilized as a part of Big Data and real-time analytics are Hadoop, as a base for information storage and processing, is the most vital technology being utilized as a part of this zone, but for Streamline analytics, Spark is a capable tool for in-memory processing and computing which we can analyze real time data. Additionally, for streamline data processing there will be a requirement for data ingestion technologies like Kafka, Storm, and Flume which are utilizing examples to import information in a way that is prepared to be investigated on group. This report will discuss about the need for streamline data analytics in today's world, different methodologies to implement it. This also includes certain applications which uses these machine learning technologies and are successful in business scenario.

1.0 Introduction

Applications which needs data processing of huge amounts of data streams are having limits on conventional data processing systems or infrastructures. These streamlines based approaches has many applications such as electronic trading, fraud detection, infrastructure monitoring, market feed processing etc. These days, many technologies have evolved involving off-the-shelf stream processing engines to solve the daily challenges of operating real time, huge volume data without even using of personalized code. In this paper, description, necessity, methods of stream analytics using machine learning algorithms are discussed in detail.

The progress and innovation are no longer hindered by the ability to collect data. But by the ability to manage, analyze, summarize, visualize and discover knowledge from the collected data in a timely manner and in a scalable fashion is the most important thing to consider. Streamline analytics is analysis of huge, data-in-motion which is called event streams. These includes events that happens as an outcome of single action or group of actions like machine failure, financial transaction or any other alarm/ trigger. These alarms can be unpolished like something happened within a system at a point in time, a tweet on measurable activity, sensor reading. Increasing number of gadgets, connected devices, IoT devices exponentially enlarge the number of events that include business activity. The more amount of data processors or business produces the more advantages from stream line analytics.

1.1 What is Stream Line Data Analysis?

Stream Analytics give companies, organizations, industries to have a real-time processing analytics which computes data on live sensor data, applications, social media, websites, devices etc. They provide immediate and accurate time relative operations and processing. Additionally, with integrating language and initiative specifications this approach has become more responsive. This uses SQL and decrease complication of streamline analytic systems.

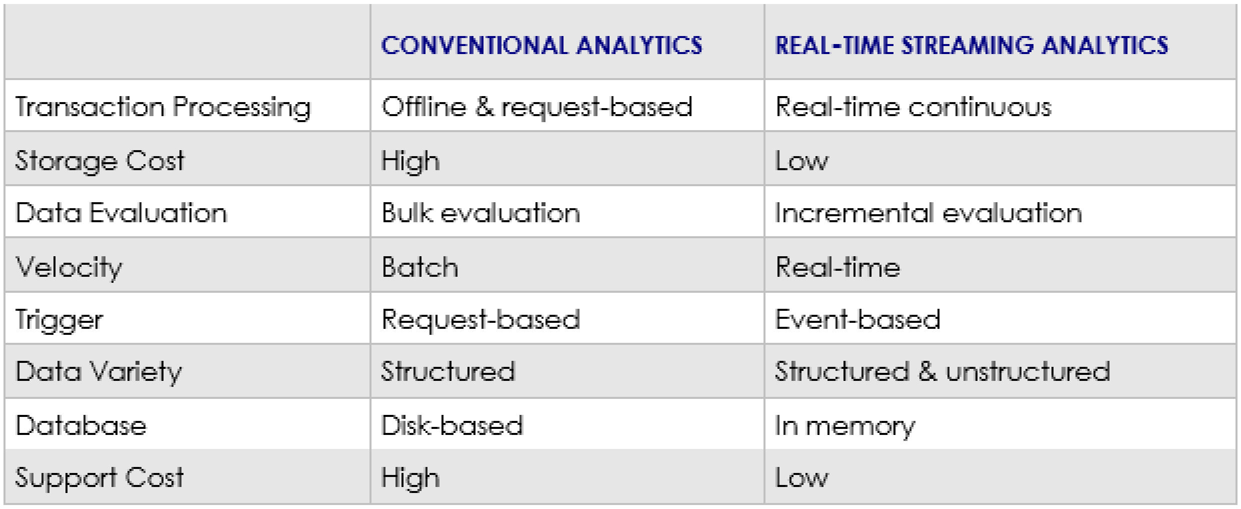
Some stream analytics are managed with event operation engine which is made up of real time analysis and operations on stream line data. This analysis could be in any form such as statistical analysis, mathematical calculations, packet inspection, image inspection etc. This facilitates analysis of “Data in Motion”.

1.2 Why Stream Line Analysis?

Streamline analytics replaces traditional style processing and analyzing data by introducing real time knowledge insights to our decision-making strategies. In other words, this gives businesses a better decision-making capability by keeping in mind about streaming and real-time data. Conventional methods depend on batch operations and processing, in which data is stored based on a time and schedule and operate upcoming data hourly bases, weekly or overnight. They are consistently reactive as they work on old data, information. This means the proposed system can give decisions or react only based on past data, conditions or events Conventional methods, architectures are restricted as they have face problems in giving efficient tracking and management of events

Additionally, Event Stream Line processing methods and architectures could capture actions, assess action data, generate decisions based on the data and publish or share the outcomes and everything inside a time gap or window. This makes a system to extremely proactive to responsive about the differencing events and conditions that improve processing and operations. This will all together improvise customer interactions and give event-driven streamline insights.

Streamline analytics is a crucial step in the transformation of information-driven industries. Initial step is data visualization after consolidation. Next is moving these into visualizations and making machines run business work processes, in which actual decisions are made. Doing this process in batches trouble human decision-making time and accuracy. Everything should be done in parallel so that single mistake is not done based on past data.



	CONVENTIONAL ANALYTICS	REAL-TIME STREAMING ANALYTICS
Transaction Processing	Offline & request-based	Real-time continuous
Storage Cost	High	Low
Data Evaluation	Bulk evaluation	Incremental evaluation
Velocity	Batch	Real-time
Trigger	Request-based	Event-based
Data Variety	Structured	Structured & unstructured
Database	Disk-based	In memory
Support Cost	High	Low

In this paper, all the streaming analytics techniques and benefits it can give to various sectors such as social media, healthcare, sales industry are discussed. Basic architectures and how can they be implemented is also given. Machine learning algorithms which are embedded in streaming analytics are described in detail. Many use cases are to be discussed and followed by my opinion and understanding about real time streaming analytics with machine learning algorithm is explained.

2.0 Background

The capability of efficiently take advantage of the knowledge present in time series data has been most challenging as the speeds and volume of data obtained has increased. However, opportunities are more than ever. Traditional Data processing has faced challenges such as, introduces too much "decision latency", responses are delivered after the impact, maximum value of the identified solution is lost, decision are made on old and stale data. Main problem with conventional processing technologies is "Data is at Rest"

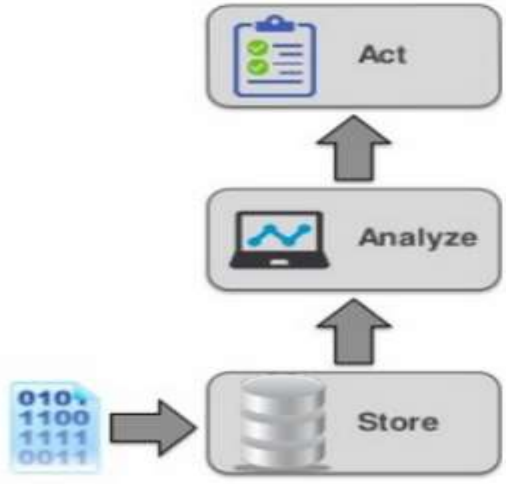


Fig 1 Traditional Data Processing

Streaming Data Analytics has following features which outburst traditional methods. Events are analyzed and processed in real-time as they arrive, decisions are timely, contextual and based on fresh data, decision latency is eliminated, "Data is in motion"

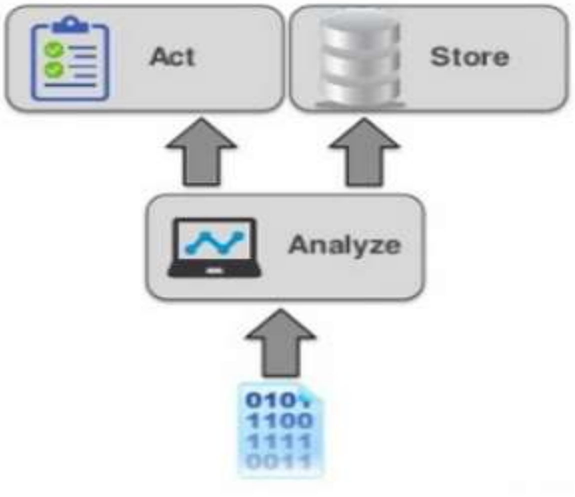


Fig 2 Streaming Data Processing

2.1 Current State of Art

Present real time streaming platforms are Hadoop, in spite of complex big data processing and following tools, this is actually simple. Hadoop can be used for data which can be batch processed, split into little processing jobs, which can spread over a cluster and recombined efforts. As stated in [9] Spark is an open sourced data processing framework, in memory on clusters, lightning fast performance, standalone, or on YARN, can handle streaming, machine learning and graph processing. According to [9] there are two technologies that support real time analytics layer. First is Stream analytics in which it acts as a linchpin, able to give temporal analytics on data in motion. Second is Machine learning where predictive analytics are able of consuming either multiple columns or single record via response / request API. This consumes a file for batch asynchronous scoring. Based on [6] increased relevant and stem value from a customer purchasing behavior knowledge leads to healthy and strong relations between customers and business sales executives. Recommendations are being used all over the industries and mostly at shopping websites. Another added advantage with this system is discover new and relevance offers depending on buying behavior of the consumer with promotions and coupons. According to [5] Streaming analytics could help in detecting customers who have higher likelihood to churn and if possible take them in a social network with the industry. Predictive analytics allows providers to take their analytics project on looking past data and compare it with current data in preventative and predictive manner. This Churn prediction and prevention helps in reducing churn around its capability to operate information on all the factors Which effect customer's experience i.e., bandwidth consumption, network coverage, billing information, everything in streaming data. Fast response to consumer problems could help in happiness of subscriber and churn prevention.

3.0 Technical Descriptions

Input for streaming analysis is a real-time or streaming data. This data could be send to any storing system like Azure from an IoT device or hub. To operate or examine this data, we should create a Job for stream Analytics which gives or specifies from which place this data is coming from. This also says about the modification and transformation, also how to look for information, relationships, or patterns. To do this action Streamline analytics uses a SQL based query language to sort, filter, join or aggregate real-time data during a period.

Lastly, this task or job gives an outcome for the changed/transformed data. To understand and control the response and to analyze the data we must

- Send an instruction to modify device settings,
- Forward information to a queue which is monitored for potential actions depending on outcomes.
- Forward data to a Business Intelligence dashboard.
- Forward data to securely store in Data storages like Azure Blob storage, SQL Database, Lake Store.

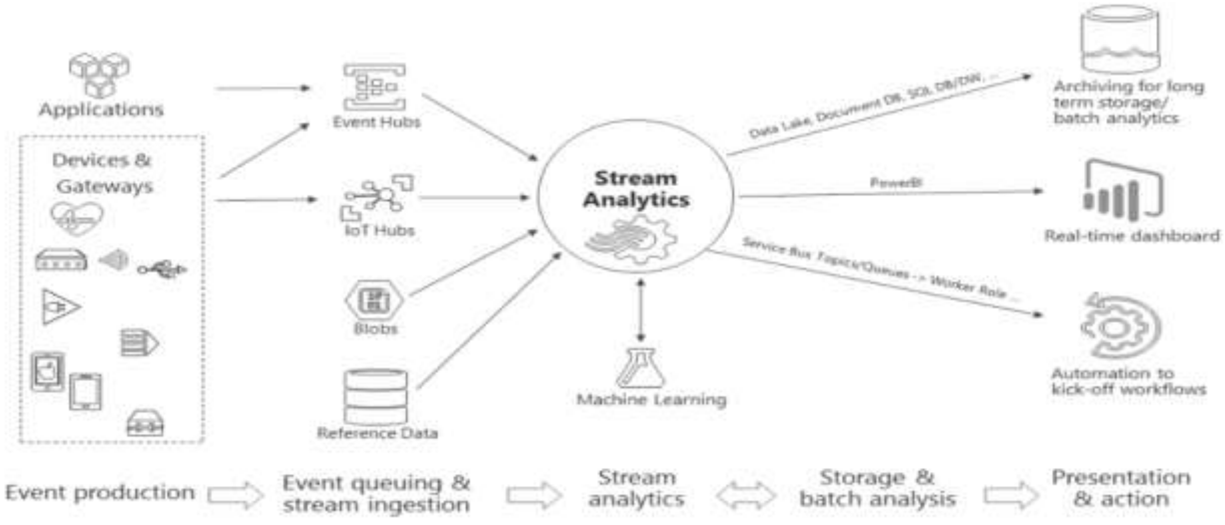


Fig 3 Working of Streaming Analytics

Following are the Eight Rules or Requirements for real-time streaming processing

1) In real-time data stream processing first rule is to keep data in motion and without the need for storing, operate "in stream" messages in random order or in a sequence order. This should utilize a non-polling active processing standard.

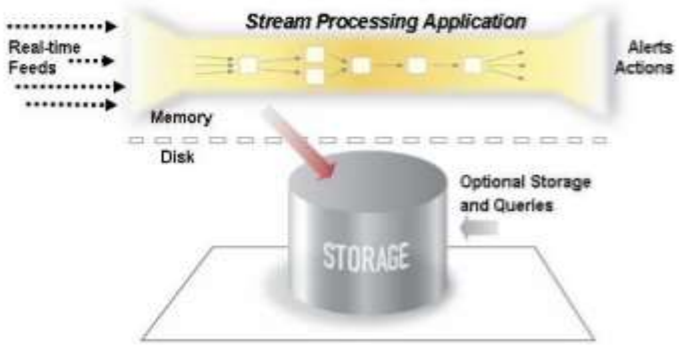


Fig 4 In-Stream Data Processing

2) Second rule is to use a "StreamSQL" high level language along with some built-in streaming extensible operators and primitives.
3) Third rule is to use built-in systems and methods which can give resiliency towards stream deformity which includes out-of-order and lacking data as they are usually in real-time stream data.
4) Fourth rule is that the streamline data processing system should always give same outcome for specified data and predictable results.
5) Fifth rule is able to effectively access, save, and transform data and mix it with real-time data. For integration, it should utilize common language while dealing with any of the available data.
6) Sixth rule is make sure if the systems or applications are always available and up. Also, integrity of data is always maintained inspite of failures.

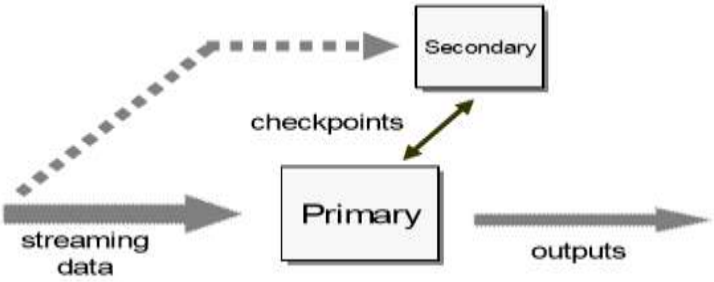


Fig 5 Availability of Systems

7) Seventh rule is able to distribute operations all over multiple processors and systems should have incremental scalability. Usually this distribution should be transparent.
8) Eighth rule is that this system should always have a minimal overhead and highly optimized engine to have highly responsive for huge-volume applications.

3.1 Basic Architectures

There are at three different software technology architectures which can be applied to determine low latency, huge volume real time problems.

3.1.1 Simple Event Processing (SEP)

These are specifically designed to work with streaming real-time data and is recently got into picture. The basic architecture is given in figure. SPEs operate on SYQ query based style processing on the input data and messages as they pass by, without actually necessary to store data. If storing data is essential, then traditional SQL databased systems would work efficiently. This uses a special constructs and primitives i.e., time windows to address streamline oriented processing algorithm.

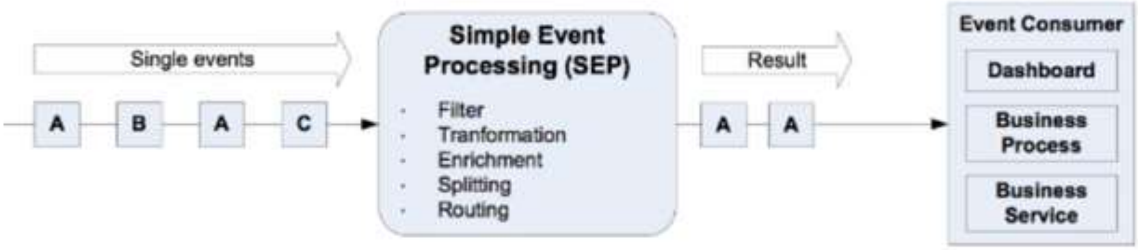


Fig 6 SPE Architecture

3.1.2 Event Stream Processing (ESP)

Event is any kind of action or occurrence that happened at any given particular time which is recorded in a group of fields. Stream is a continuous flow of events or data, or constant rush of data that moves in and out of a business or system from many connected devices and other sensors(IoTs). Processing is analyzing data. McNeil says that in ESP, processing can be done in three places,

1) At-Edge analytics
2) In-Stream analytics
3) At-rest analytics

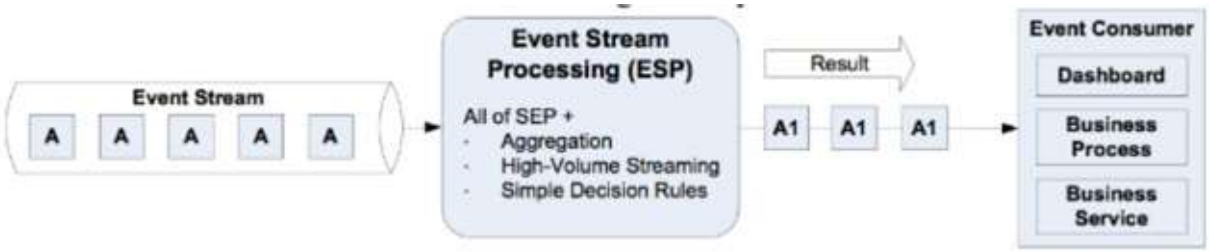


Fig 7 ESP Architecture

3.1.3 Complex Event Processing (CEP)

Complex event processing is the utilization of techniques to estimate high level actions or events which likely to be outcome for particular group of low level reasons or factors. It detects and analyzes cause and effect relationships between actions in real time and facilitates personnel to actively choose efficient measures with respect to particular situations. Important feature about this architecture is business activity monitoring. By using this we can actively define and analyze the highest priority risks and opportunities. This has applications in customer relationship management, security risk management, middleware and application servers. Following is the figure for CEP architecture.

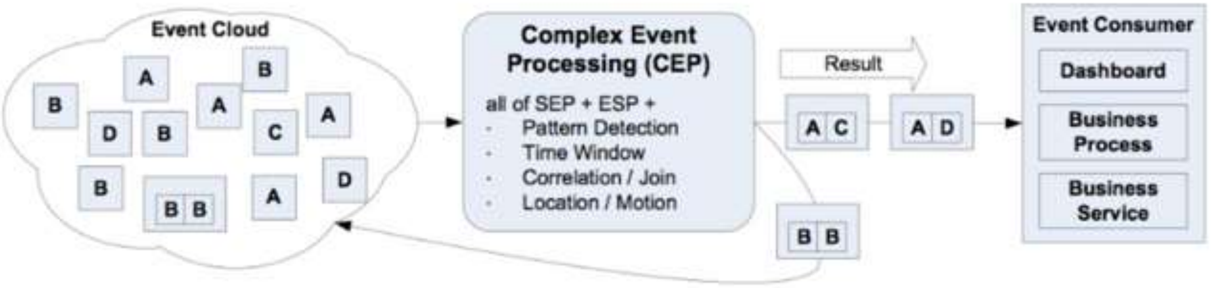


Fig 8 CEP Architecture

3.1.4 Native Streaming vs Micro-Batching Processing

In Native Streaming

- Events are processed as they arrive
- Low latency
- Less throughput
- Less fault tolerance and is expensive

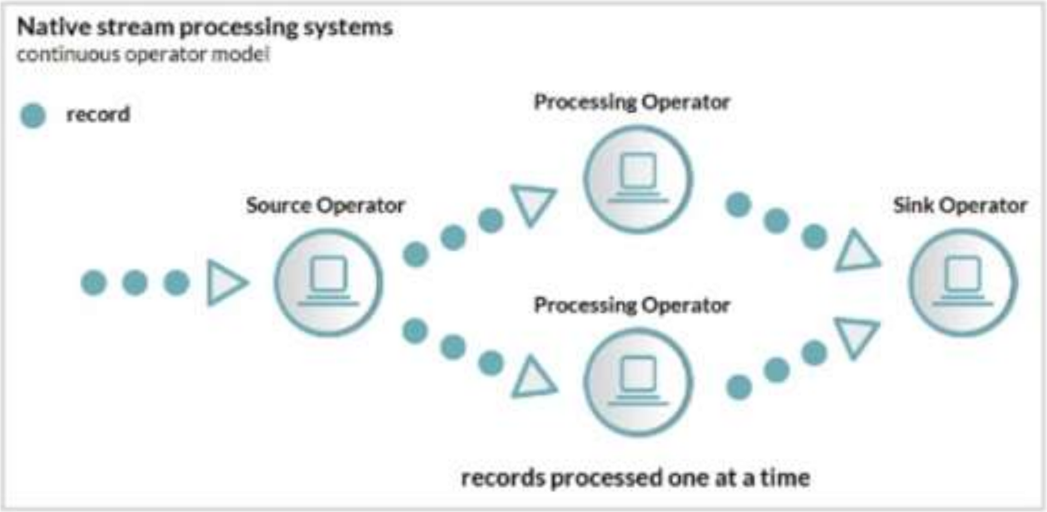

Native stream processing systems
continuous operator model
record
Source Operator
Processing Operator
Processing Operator
Sink Operator
records processed one at a time

Fig 9 Native Streaming

In Micro-Batch processing

- Splits incoming stream in small batches
- Higher throughput
- Easier fault tolerance
- Lower latency

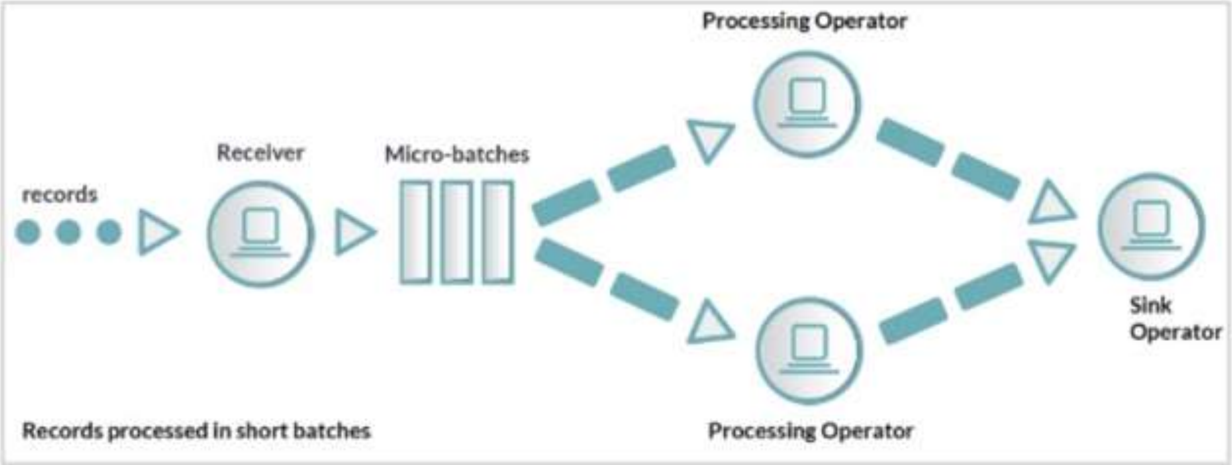


Fig Micro-Batch Streaming

3.2 Machine learning Algorithms

The aggregation of streamline analytics and machine learning gives us a rich ability for both producing prediction and on top of that how to act to predictions is given. Machine learning will allow software or let it think or figure out situations by itself. For instance, the Denstream Clustering machine learning algorithm helps us input stream data and figure out if there are any associated or relevant clusters available without the need to know it in beforehand. Most important feature is the it can identify outliers. In other words, this clustering algorithm find out the groups of casual behavior and those from the unusual or anomaly ones for us to react on. Another coolest feature about this is it adapts with time by using older data and provide updated or more weights to the new events. This identifies the new normal behavior even before a human could recognize the change. This facilitates automated adaptability and learning which open potential predictions which are not only based on old historic data but also on present right now data.

3.2.1 DenStream (Clustering) Algorithm

This algorithm is an incremental clustering. It utilizes the theory of micro-clustering to aggregate clusters of subjective shaped and a cut back technology to identify outliers. It is not sensitive to disturbances or noises. This unique pruning technology leads to improvised memory management of real time streaming data this is emerging. This is frequently utilized to identify unknown or hidden patterns in the specified data. Conventional batch clustering techniques usually taka all the accessible data as input which isn't suitable for real time streaming data as these streaming processes allows only single pass or input of data. Finest fit to solve those problems where past data is not an important factor which effects the outcome. Buying behavior of a customer using credit or debit card will usually show an exclusive pattern for a particular user. This algorithm finds clusters of casual purchasing pattern for particular user. Additional combining of this cluster behavior of huge groups can remove false alarms. DenStream can detect clusters of shapes which are arbitrary and can handle disturbance.

Procedure

- Creates Damped window, where weights of data objects tend to exponentially decrease with respect to time

$$f(\mathrm{t}) = 2^{-\alpha \mathrm{t}}, \alpha > 0$$

- Creates Micro clusters MC= (*WLS*, *WSS*, *w*, t_c), where

  $WLS = \sum\_(i = 1)\hat{}n\text{▒}\ 〖f(t - Ti).\ pi〗$ (Weighted Linear Sum)

  $WSS = \sum\_(i = 1)\hat{}n\text{▒}\ 〖f(t - Ti).\ pi〗\ * pi$ (Weighted Squared Sum)

  w (Weight of MC)

  t_c (Creation Time of MC)

- Center for the cluster can be derived from c = WLS / w
- Radius of the cluster can be derived from the following equation.

$$r = \sqrt{\frac{||WSS||_2}{w} - \left(\frac{||WLS||_2}{w}\right)^2} \le \varepsilon \text{ (Radius of MC)}$$

- Micro-clusters are classified depending on weight w, if w >= μ then that cluster is a core-micro-cluster. Potential core-micro-clusters (p-micro-clusters) are when w >= β*μ and Outlier micro-clusters (o-micro-clusters) are when w < β*μ

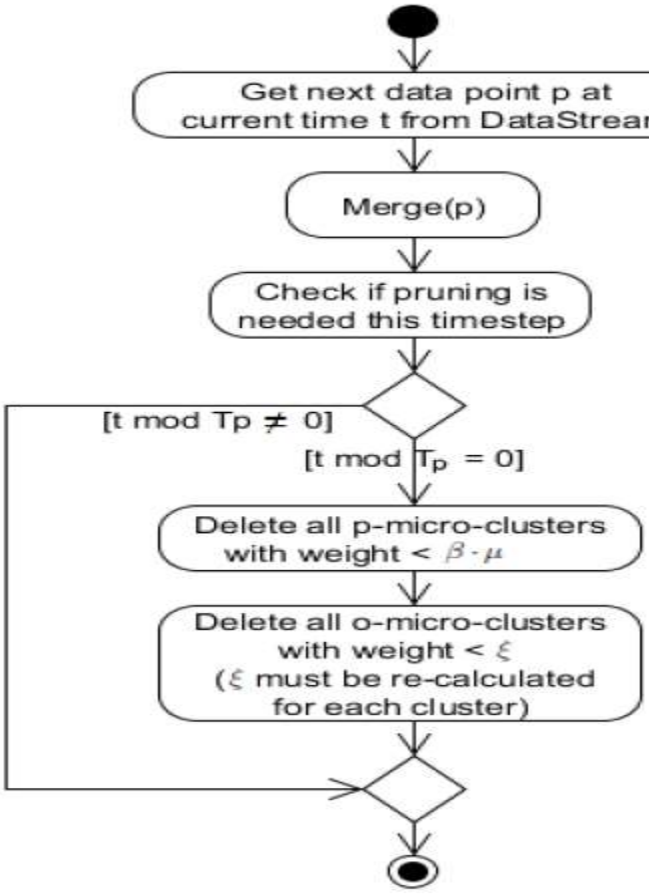

Get next data point p at current time t from DataStream
Merge(p)
Check if pruning is needed this timestep
[t mod Tp ≠ 0]
[t mod Tp = 0]
Delete all p-micro-clusters with weight < β · μ
Delete all o-micro-clusters with weight < ξ
(ξ must be re-calculated for each cluster)

Fig 11 DenStream Algorithm

3.2.2 Adaptive Hoeffding Tree Algorithm

Adaptive Hoeffding is an incremental decision tree algorithm utilizing restricted number of samples to select the finest or best tree node which is splitting the label or attribute. Based on past or old data a tree-like graph is learning a model or form decision rules and form a map of observations with its target information as estimation or prediction. Identify concept drift and modifies the tree model accordingly and automatically. This algorithm has advantages of low-overfitting as there is no need for pruning and samples are only used once. Low variance as it has statistical support along with stable decisions. Low resource usage as it is restricted hardware resources such as CPU, I/O and Memory.

This algorithm has no magic /in-built parameters, it self-adapts to changes unlike other decision trees. It gives better accuracy than CVFDT, which has a concept of drift, windows and several parameters. This uses less memory as no windows used. Moderate overhead which is less than

50%. Meticulous guarantees possible. This replaces counter nodes with ADWIN node and adds ADWIN node to see if there is an error of every subtree.

3.3 Use Cases

3.3.1 Sales Enrichment

Using real-time events to give a prediction of what a customer is interested in that specified period. Data is Transactions, current search keywords, Mobility or location, web pages visited, weather. Results or decisions based on this outcome are to deliver applicable coupons to customers before they move out of store, show relevant advertisement as the customer swipe a card at any departmental store or a petrol bunk. Send promotions to encourage change in spending of customer and behavior.

3.3.2 Security/Fraud

Using real-time data to deliver alerts if an activity is prone to fraud or a mishandled situation has occurred. Data is location, store browsing history patterns, Network or machine activity. Results are to determine if an online system or credit card is fraudulent even before the it goes out of pending transactions and purchase is submitted.   Find and stop DDoS (Distributed Denial of service) attacks even before it takes down the entire system.

3.3.3 Anomaly Prediction

Using real-time streaming data to estimate the anomalous activity or behavior even before it happens. Data is system metrics, server logs and sensors. This is to estimate a network switch failure to enable full capture of every network information before crash would happen to allow have root origin analysis. Forecast a brake failure or black ice event in a associated car. Detect Dysfunction of drilling in an Oil Rig to prohibit lost productivity or breakages.

Following are other streamline analytics applications

- Algorithm trading
- Online Fraud Detection
- Geo Fencing
- Proximity/Location tracking
- Intrusion detection systems

- Traffic Management
- Recommendations
- Churn Detection
- Internet of Things
- Social Media Data analytics
- Game Data Feed

4.0 Discussions

Sometimes even one minute slow is too late to give predictions. Every year amount of data generated is getting doubled. This is too much to store and even before its analyzed data is lost. There are many engines which take historic data and dashboards are overloaded with so much old data predictions. There is no much of future predictions to plan ahead and react. Most of the data existing are unstructured and analytics cannot work or operate with streaming data as in from sensors, acoustic etc., Moreover data has too much noise in it. These are the challenges with traditional processing. Streaming analytics could take very high volume of streaming scoring requests. It required low latency. It can also take data from variety of incoming sources.

Streaming analytics continuously integrates processes data in motion to send analytics and it has both fast runtime architecture and development environment. This makes industries to find insights, opportunities, risks in streaming data, where only detected in moments notice. One challenge for this kind of streaming data is navigating huge volume and high velocity data flow from multiple resources such as mobile devices, IoT devices, transactions, clickstreams etc. remain highly unnavigable. Most of the streaming computation solutions give a development platform which helps us to build many applications in best language we can work on. Platforms like Spark Streaming Scala, InfoSphere Streams are development IDE's which has integration with Business Intelligence, visualization tools and warehousing.

Event Streaming is to react on event driven data in real time, whenever an even occurs. Since the past data may be insignificant after a certain period and expected nature may drift or shift over time we go with Event stream processing (ESP). When streaming processes uses machine learning algorithms gives the capability of immediately incorporating present data into predictive algorithms than polling periodically a data from external source. Additionally, it has ability to progressively and instantly adapt to the transforming behaviors and conditions. Also, machine learning algorithm leverages that are modeled for analytics which are continuous with latency low.

5.0 Summary

Transforming the increasing deluge of information into streaming real-time information is an exhausting task that are provided must handle if they had to meet their industry objectives that focus on providing profits delivering best consumers and producing opportunities. Streaming real time analytics will play an important role in the success of actions as this ability will give business analyst with real-time intelligence and also help to maximize income from a small opportunity. There are many large-scale existing new upcoming technologies which needs highly sophisticated streamline processing of large volume of data. All these technologies are discussed, and many use cases are being provided. Machine learning algorithms combined with streaming analytics would change the worlds market drastically, these algorithms are portrayed in detail. In the analysis, found that DenStream clustering and Adaptive Hoeffding tree algorithm combination would best fit in the streaming analytics which can give actions to the predictions obtained.

References

[1] M. Stonebraker, U. Çetintemel and S. Zdonik, "The 8 requirements of real-time stream processing", *ACM SIGMOD Record*, vol. 34, no. 4, pp. 42-47, 2005.

[2] J. Riedy and D. Bader, "Massive streaming data analytics", *XRDS: Crossroads, The ACM Magazine for Students*, vol. 19, no. 3, p. 37, 2013.

[3] K. Madia, "Analyzing the evolution of streaming analytics architectures", *BigData and Analytics Hub*, 2015.

[4] "Introduction to Stream Analytics", *Docs.microsoft.com*, 2017. [Online]. Available: https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-introduction.
[Accessed: 31- Oct- 2017].

[5] A. Banerjee, *Real-Time Streaming Analytics for Telecom: The Essential Guide*, 1st ed. 2011

[6] P. Hall, W. Phan and K. Whitson, *The Evolution of Analytics Opportunities and Challenges for Machine Learning in Business*, 1st ed. Sebastopol: O'Reilly Media, Inc, 2016.

[7] R. Waywell, "Applying Machine Learning to Real- Time Streaming Analytics", Las Vegas, 2016.

[8] S. Hanna, "IBM Big Data Streaming Analytics", IBM Corporation, 2014.

[9] D. Jayanthi and G. Sumathi, "A Framework for Real-time Streaming Analytics using Machine Learning Approach", vol. 1, no. 2320-0790, 2016.

[10] B. Yadranjiaghdam, N. Pool and N. Tabrizi, "A Survey on Real-time Big Data Analytics: Applications and Tools", *International Conference on Computational Intelligence*, 2016