Harnessing Big Data: a Technique and Tool for User Directed Selective Data Filtering

Shikharesh Majumdar

Chancellor's Professor
Director Real Time and Distributed Systems Research Centre
Dept. Of Systems & Computer Engineering,
Carleton University
Ottawa, CANADA
majumdar@sce.carleton.ca

Acknowledgments

Systems & Computer Engineering, Carleton University:

Bannya Chanda

Glenn Davidson

Amarjit Dhillon

Marc St-Hilaire

TELUS:

Ali El-Haraki

- **Financial Support**:
  - Natural Sciences and Engineering Research Council of Canada (NSERC)
  - TELUS

Outline

- Introduction
- Big Data Platforms
- Data Filtering Technique
  - Stored Data
  - Streaming Data
- Summary and Conclusions

Application Area Examples

Big Data Sources

- Sensor Based Cyber Physical Systems:
  - Generates Data that needs to be analyzed
- Enterprises
- Scientific research
- Social Networks

Example Use Cases

- Filtering Technical Articles
- Smart Bridges/Smart Machinery
- Twitter/Social Networks
- Managing Meeting Minutes
- Remote Patient Monitoring

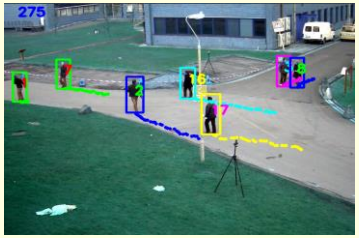
Remote Health Care
From https://demigos.com/blog-post/remote-patient-monitoring-software/

From www.libelium.com

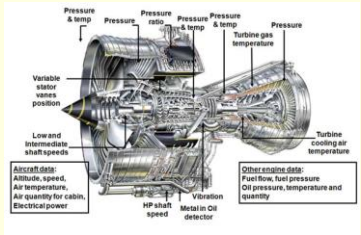
Bridge Management

Twitter

From:http://india.smartcitiescouncil.com/sites/default/files/india/images/Smart-Buildings-Key-to-smart-cities.jpg



Aerospace "Pratt & Whitney's Geared Turbo Fan (GTF) Engine"



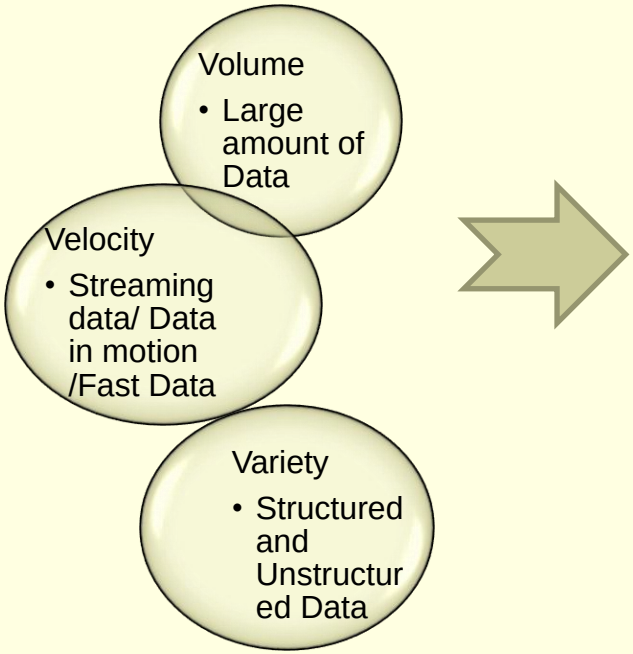
From: www.rolls-royce.com

From http://www.venturemagazine.me/wp-content/uploads/2015/04/222-770x375.jpg

Object Identification and Tracking

Harnessing Big Data

Challenges

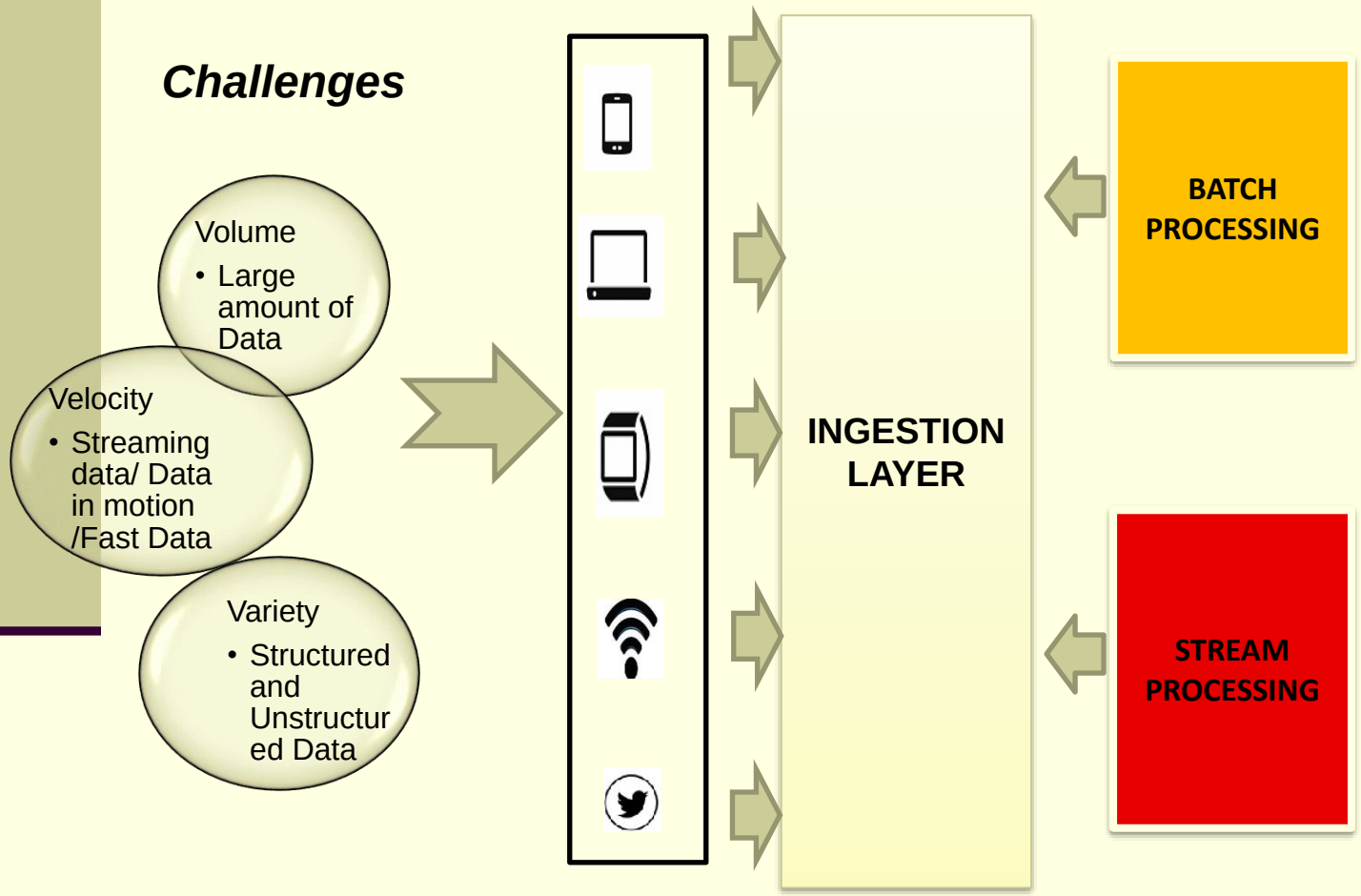


Two Approaches

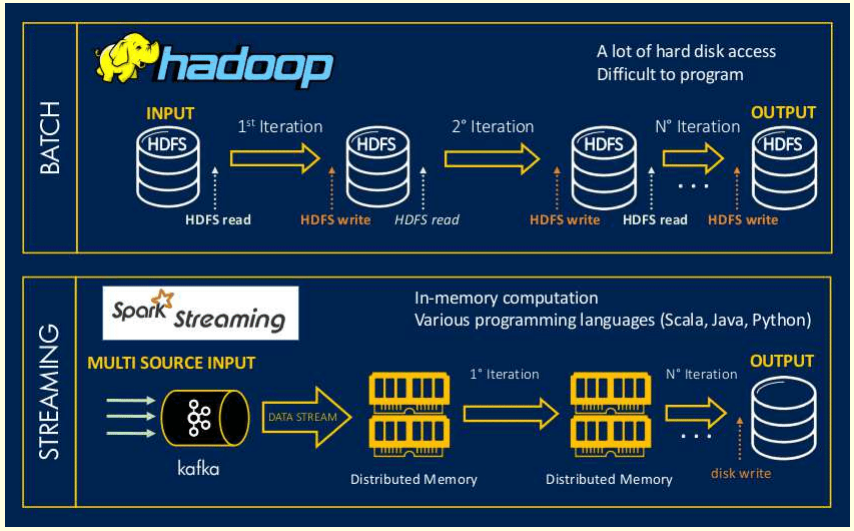
•Speeding Up Processing

•Reducing Data Volume/ Indexing

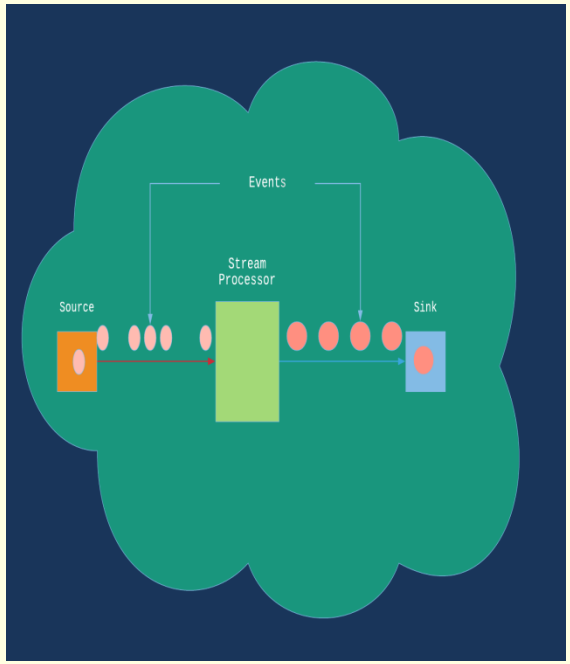
Types of Big Data Analytics



Platforms for Batch and Streaming Analytics



From: https://www.researchgate.net/figure/Hadoop-vs-Spark-Example-of-Big-Data-Analytics-platforms-for-batch-and-streaming_fig1_315095908



Stream Processing with Siddhi
From: https://codeburst.io/stream-processing-with-siddhi-af4c55d11166

Challenege: Performing Analaytics in a Timely Manner

The Big Data Problem: Volume

- Large volumes of Data are produced on a daily basis
- A specific user may be interested in only a selected set of documents containing some specific content
  - **How to Store and Locate Required Data Efficiently?**
- Example Use Cases:
  - *The research papers use case*
    - *Interested in papers focusing on specific topics*
  - *The meeting minutes use case*
    - *e.g.. Interested in meetings that discussed specific products*
  - *The journalist use case:*
    - *Interested in tweets focusing on specific events or persons*
  - *The medical practitioner use case*
    - *Stored Data*
    - *Streaming Data*

From: https://www.istockphoto.com/search/2/image?phrase=needle+in+haystack

From: https://ellipse.prbb.org/the-art-of-publishing-a-scientific-article/

From: https://www.healthcareitnews.com/news/emea/patient-records-electronic-or-electric-you-decide

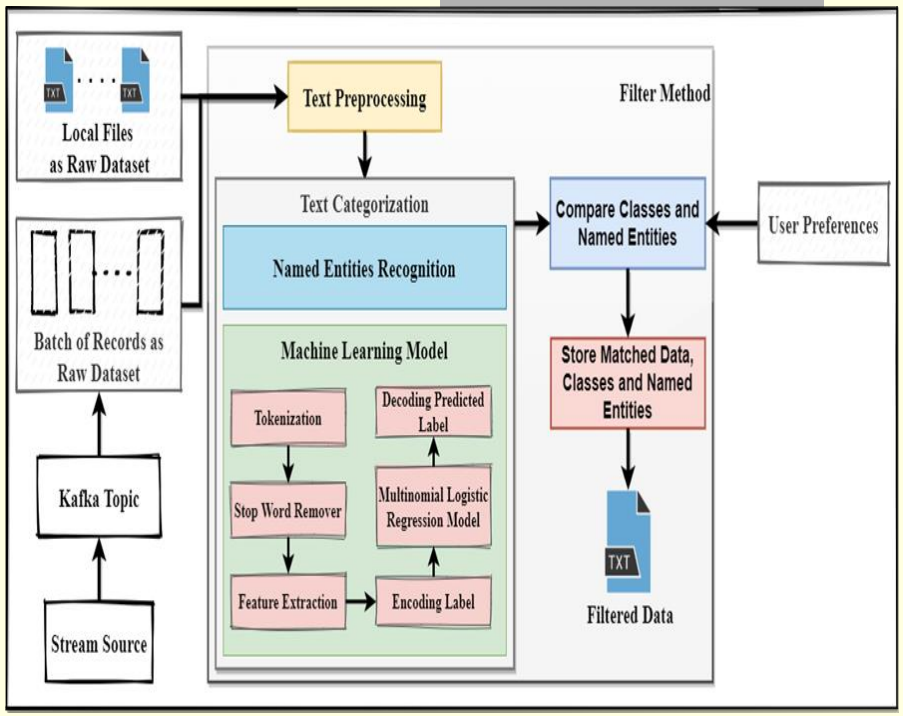
From: https://planetinnovation.com/perspectives/how-to-build-a-regulated-remote-patient-monitoring-product-that-clinicians-and-patients-will actually-use/

Data Filtering

- Need to extract selected data items (user preferences) from large data sets
- Challenge: Searching the required information from the data set.
  - More difficult for large data sets
  - More difficult when there are multiple data sets.
  - Becomes more time consuming to search.
  - Takes a large space to store
- Objective: Devise an effective filtering technique
  - Extract user “preferred” data as filtered data in a timely manner.
  - Store only the filtered data
  - Decreases the search latency
  - Decreases the size of the storage required.
- Challenge: Filtering is processing intensive:
  - extract the selected information from the large raw data sets.
- Solution: Parallel Processing.
  - Apache Spark

The Filtering Technique for Streaming Data

- Apache Spark based
- Categorize dataset
  - using a machine learning model
  - and named entity recognition.
- Filter method filters data
  - matches with user preferences
  - stores the filtered data as a comma separated values file.
- Filtered data contains
  - filtered text,
  - entity classes (recognized by the named entity recognition function)
  - extracted name entities
  - class predicted by the machine learning model
    - Multinomial Logistic Regression

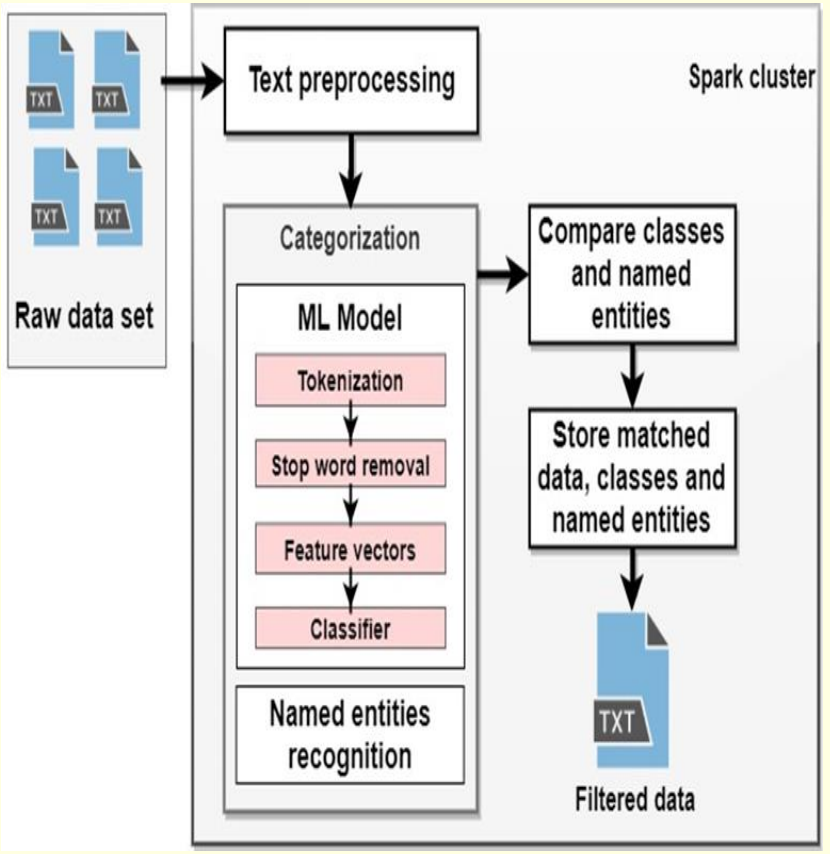


- Adapted to stored text data files
  - Conversion from PDF to text needs to be performed by *pdfminer*

Chanda, B., and Majumdar, S., "A Technique for Extracting User Specified Information from Streaming Data", Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), Online, July 2021.

The Filtering Technique for Stored Data

- Categorize dataset
  - using a machine learning model
  - and named entity recognition.
- Filter method filters data
  - matches with user preferences
  - stores the filtered data as a comma separated values file.
- Filtered data contains
  - filtered text,
  - entity classes (recognized by the named entity recognition function)
  - extracted name entities
  - class predicted by the machine learning model
    - Multinomial Logistic Regression



Chanda, B., Majumdar, S. “A Parallel Processing Technique for Extracting and Storing User Specified Data”, Proc. IEEE 8th International Conference on Future Internet of Things and Cloud (FiCloud), Rome, August 2021.

Performance Analysis

- Experiments performed in an Apache Spark cluster.
  - set up on an Amazon EC2 cloud infrastructure.

- Analysis of impact of the workload and system parameters on performance

- Raw Dataset: Synthetic
  - Stored data.
  - Stored as multiple files in a local directory.
  - (*Also experimented with wikipedia files*)

- Workload and System Parameters:
  - number of executor cores (N),
  - number of worker nodes (N_W),
  - raw dataset sizes (S_R)
  - Data partitioning strategies (DP)
    - Centralized (C_{DP})
    - Equal Distribution (E_{DP}).

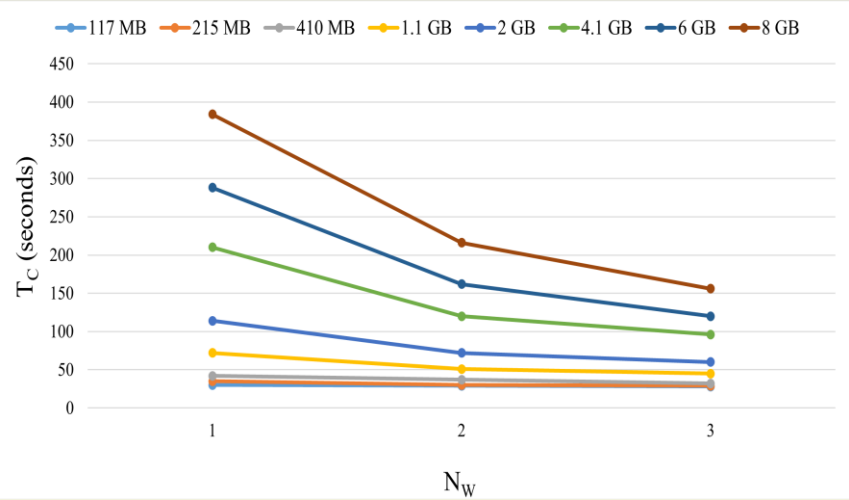
- Performance Metrics:
  - computation time (T_C),
  - speedup (S(N))
  - efficiency (E(N)).

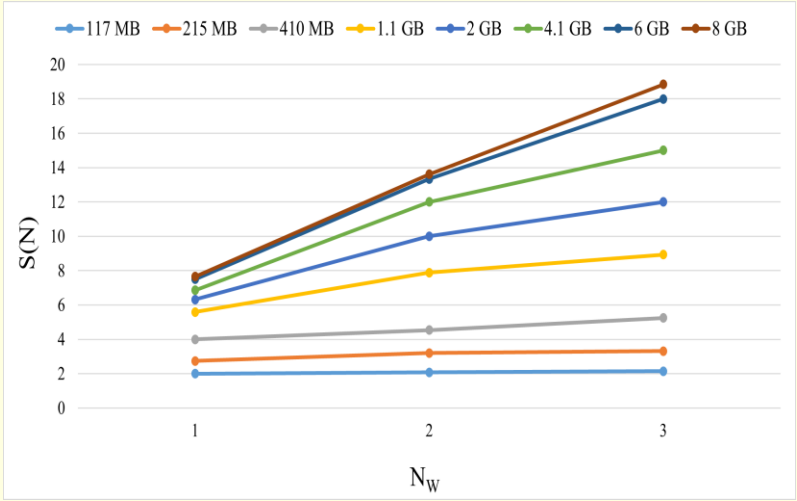
- The experiments are performed by following a factor-at-a-time approach
  - one of the parameters is changed while others are held at their default values.

Impact of Number of Worker Nodes on Performance

- Every worker node comprises 8 cores
- Increase in number of worker nodes increase in parallelism
  - Smaller T_c
  - Higher S(N)
- For smaller raw datasets (117 MB–410 MB)

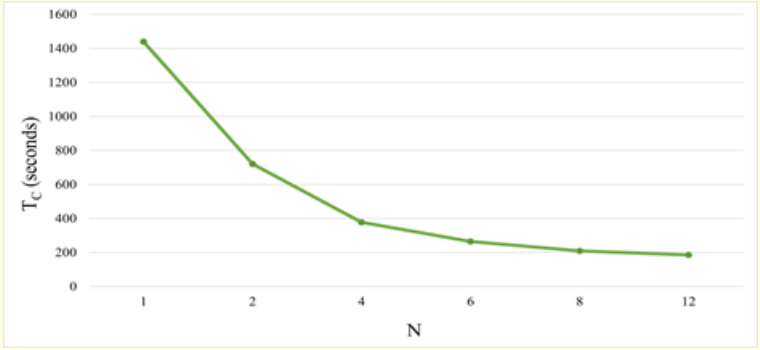
  Increase in N does not improve speedup significantly





Executor core Parallelism vs Node Parallelism

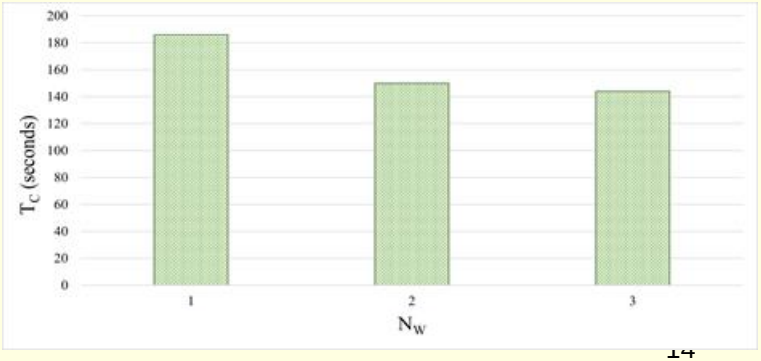
- First Figure: Increase in the no. of cores
  - Improves performance
- Second Figure:
  - Total number of cores = 12
  - No. of cores / node = $12/N_w$
  - A smaller number of executor cores within worker nodes seems to give rise to superior performance



Filtering Efficiency:

E_F= (search time achieved with non-filtered data)/(search time achieved with filtered data)

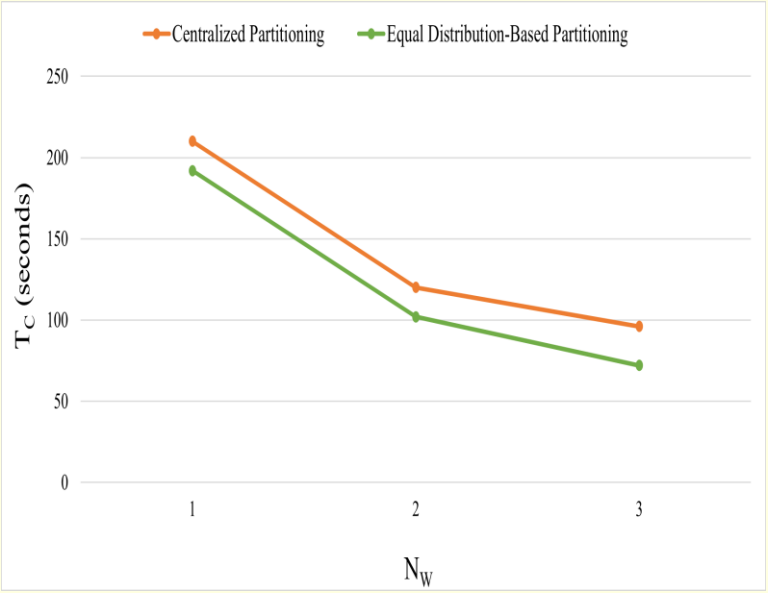
Search method	Search by	E_F
Sequential	Keywords	105
Sequential	Sentences	57.6
Parallel	Keywords	63.6
Parallel	Sentences	29



Effect of Data Partitioning Strategy

- N_w = 3 - T_C decreases from 90 (C_{DP}) seconds to 66 seconds (E_{DP}) when the data partitioning strategy changed from C_{DP} (C_{DP}) to E_{DP}.

- The speedup increases from 16 (C_{DP}) to 17.143 (E_{DP})
- *A priori* balancing of the load Results in higher performance



Boolean Logical Operator Based Filtering

- User preferences can have multiple keywords/terms connected by the Boolean OR, AND and NOT operators
- **Example:** *User Specification*:
- ‘Canada’ OR (‘prime minister’ AND ‘country’)
- *Raw Data:*
- 1. Trudeau is the current prime minster of the country
- 2. The general elections for the country is typically held once every four years.
- 3. July 1 is celebrated as Canada Day in all parts of the country

Boolean Logical Operator Based Filtering (Sample Results)

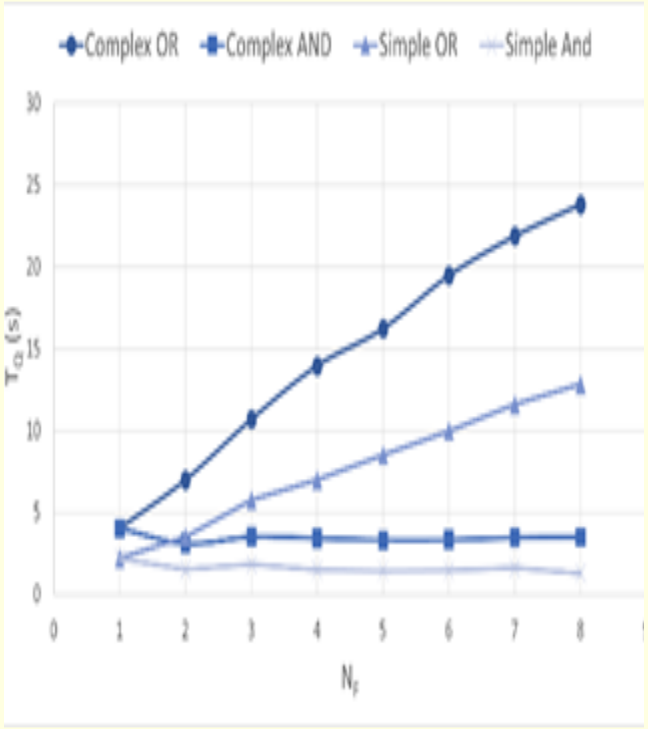
N_F: No. of filter terms

Simple: N_F terms connected by an OR/AND operators

Complex: Each complex term comprises three simple terms

Simple OR, Complex OR: Filtering time increases with N_F

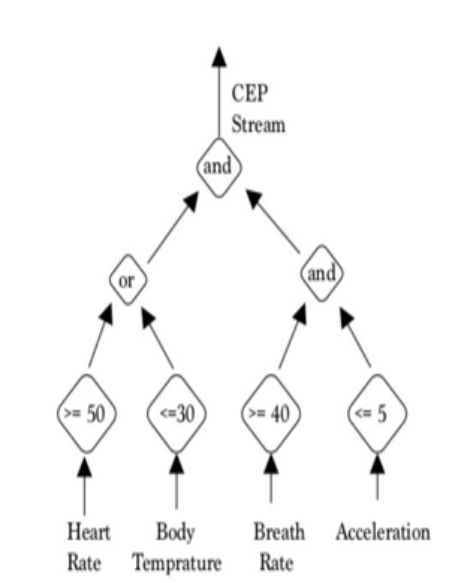
Simple AND, Complex AND: Filtering time seems to be insensitive to the value of N_F



No. of Worker Nodes = 3

Patient Data Filtering System

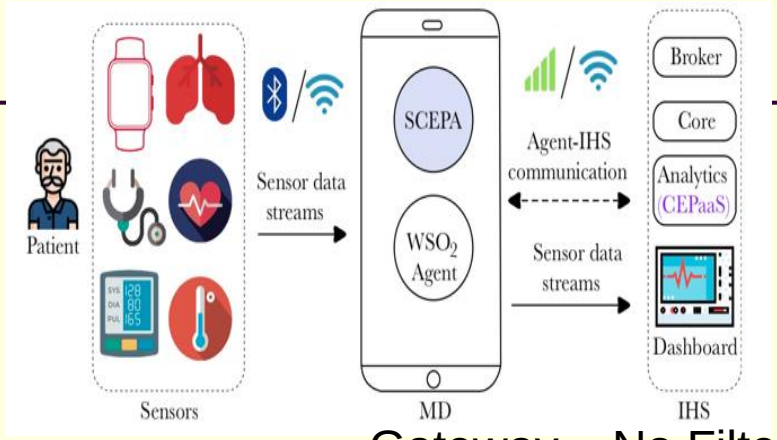
- Complex Event Processing (CEP) is a technology which can consume sensor data from one or multiple sensors and analyze them in real time using CQL.
  - Deployed on Apache Siddhi running on a smart phone
- Goal of CEP based data filtering system is to find complex events which can lead to alerts/notifications.
  - Defined by user
- Data Filtering in Remote Patient Monitoring (RPM) System.
  - Uses wearable health sensors connected to a mobile device.
- Current (central server based) methodologies collects the real time health sensor strea forward them to a remote (Centralized) server for detecting complex events.
- Such a technique has many limitations some of them are:
  - It necessitates the mobile device to remain connected to the hospital server at all times.
  - Increase in user cost as sensor data streams have to be forwarded to remote server.
  - It can lead to queuing delays at server side.
  - It can lead to out-of-order delivery of various sensor streams with respect to one another.



Complex Event Processing (CEP) Based Filtering

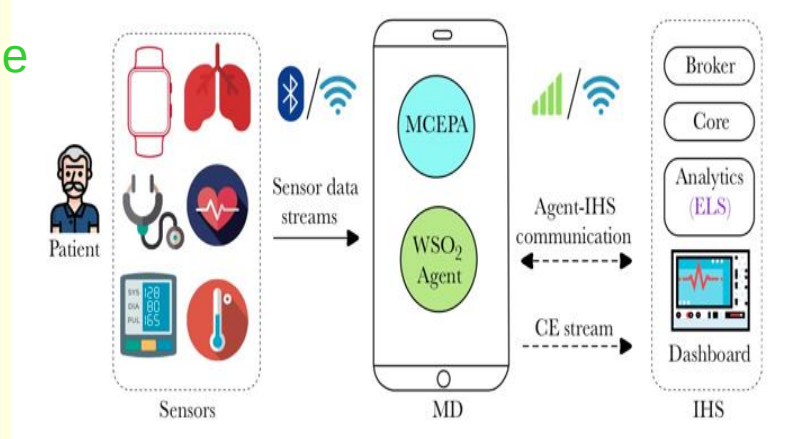
Edge

Backend



Gateway – No Filtering

Edge



Backend

Filtering performed at the Edge

CEP – Based Filtering

* Dhillon, A.S., Majumdar, S., St-Hilaire, M., El-Haraki, "MCEP: a Mobile Device Based Complex Event Processing System for Remote Healthcare", *Proc. The 11th IEEE International Conference on Internet of Things (IThings)*, Halifax, Canada, August 2018.

Summary and Conclusions

- **Challenges for Big Data Processing**
  - Volume
  - Velocity
  - Leads to data and compute intensive systems
- **Varied Application Areas:** enterprises, scientific research to medical applications
- **Two approaches to addressing challenges**
  - Speeding up Computation
  - Reducing Data Volume
- **Two Types of Data**
  - Data at Rest (Batch Analytics)
  - Data in Motion (Streaming Analytics)
- **Reducing Data Volume**
  - Parallel Data Filtering
  - Mobile Edge Computing

Backup Slides

Stream Processing (Data in Motion)

Also Referred to as Data Plumbing

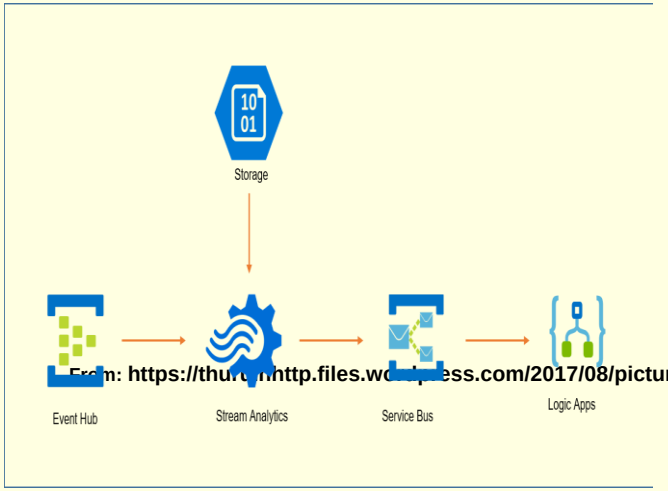
Definition - What does *Plumbing* mean?

"Plumbing is a term used to describe the technology and connections between systems in a cloud computing model. It includes the systems, storage, network and the interconnection components that form the cloud environment. The term is an analogy to the plumbing of water systems."

From: https://www.techopedia.com/definition/31509/plumbing

From: https://ca.news.yahoo.com/plumbing-codes-based-100-old-192700125.html

"Twenty years ago, when I was at Cisco in the early days, I learned how sexy plumbing could be in a digital world" Mike Volpi, Index Ventures



From: https://thurrott.files.wordpress.com/2017/08/picture13.png/

Need – Platforms & Effective Resource Management Algorithms for Streaming Data Analytics