

Dynamically reconfigurable data pipelines in the edge network

M.G.I.M. Nuwanthilaka (219376N)

Supervised by Dr.Gayashan Amarasinghe

Outline

1. Introduction
2. Problem
3. Objectives
4. Literature review
5. Identified research gaps
6. Methodology
7. Challenges
8. Resources required

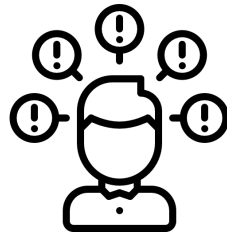


Introduction



- Data pipelines are playing a significant role in emerging data science landscape as modern organizations ingest streams and batches of data with high variety, value, veracity, velocity and volume, from different sources.
- Pipelines mainly focus on extraction, data preparation, cleaning , transforming, training models and visualization.
- There are many data pipeline requirements and challenges in the field such as data cleansing on large scale data, security implementation to pipelines, visualization on large data sets etc. Also data pipelines need to comply with the security standards.
- Architectural support to implement these requirements is a major requirement

Problem



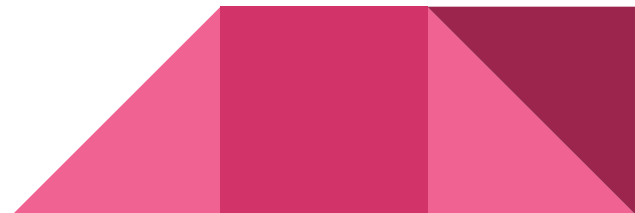
- Edge devices are resource rich compared to a few years back.
- Edge computing community grows faster with many toolings (e.g:Tensorflow Mobile SDK).
- But **most data pipelines are static with fixed configurations for all the models** deployed on the pipeline
- As a result, there exists an opportunity to optimise the configurations based on the constraints and requirements of the application.
- The research work of Akoglu and Vargas-Solar [1], Raj et al. [4], Helu et al. [2], Konstantinou and Paton [3] and Renart et al. [5] have proposed edge frameworks and reconfigurable cloud designs.
- **We investigate the possibilities of reconfigurable architectures on the edge for data pipelines**

Objectives



In this research, following three main objectives are addressed.

1. Develop a framework that supports dynamic reconfiguration of the pipeline architecture.
2. Optimal/near-optimal utilisation of edge resources to support proposed architecture.
3. Evaluation of algorithms that are benefited from this approach (performance evaluation).



Literature Review

The literature review for this research is presented under relevant categories based on the context of each research.

- 1.Data pipeline challenges and requirements
- 2.Pipeline design and architectures
- 3.Data and delivery pipeline security



Literature Review - Data pipeline challenges and requirements

1) Manufacturing process data analysis pipelines: a requirements analysis and survey [Ismail et al,2019]

Mainly, two questions are being surveyed in this paper.

1. Key requirements for a big data analysis pipeline.
2. Available tools and solutions for pipelines in academic literature.



To answer the first question, they have identified key phases and their requirements.

- a) Ingestion: This is the main component for entering data to the big data analysis platforms. This opens several other operations like identification, validation, filtering etc tasks
- b) Communication: Communication channels are a much-needed component in the data pipelines as it is the medium where data is routed among the different tools in the big data platforms.
- c) Storage: In the 1990s, the product processes stored only a few megabytes, but now the situation is much more changed and 10s of gigabytes are being generated daily in the manufacturing pipelines.



Table 4 The tools used in the respective data analysis pipelines of each paper

Paper	Ingestion	Communication	Storage	Analysis	Visualization
[44]	Custom	—	HDFS, HBase, MongoDB Infinispan,	Hadoop, Hive, Pig, Elasticsearch	Custom
[45]	Custom	—	HDFS, MySQL	Hadoop	— (∼)
[46]	WSO2 BAM	WSO2 ESB	HDFS, RDB (∼), Cassandra (∼)	Hadoop, WSO2 CEP	Custom (WSO2 UES)
[47]	—	Kafka	HDFS	Hadoop, Storm	—
[48]	—	—	HDFS, HBase, MongoDB Cassandra,	Hadoop, Hive	—
[49]	—	—	HDFS	Hadoop, Mahout, Jena Elephas	—
[50]	—	—	MySQL	Matlab, QuickCog	—
[51]	Custom	—	Microsoft SQL 2012	Custom	Custom
[52]	Custom	Kafka	HDFS, HBase	Hadoop, Storm, Hive, Radoop, Rapid-miner	— (∼)
[53]	Sqoop	—	HDFS, HBase	Hadoop, Hive, Impala	—
[54]	Sqoop	Flume	HDFS, HBase, MySQL	Hadoop, Hive	Custom
[55]	Custom	Custom	MongoDB	Custom	Custom
[56]	Custom	—	MongoDB, PostgreSQL	RStudio, Watson Analytics, QlikSense	Custom
[57]	Flume (∼), Sqoop (∼)	Custom	HDFS, HBase	Hadoop, Hive, Impala, Spark, Pig	Custom
[58]	Custom	Custom	Cassandra	Spark	Zeppelin (∼)
[59]	Kafka	Kafka	Cassandra, Onto-QUAD	Spark	Custom, Jupyter, Ontos Eiger
[60]	Custom	—	HDFS	Hadoop, Hive, Spark	Custom
[61]	Storm	Kafka	MongoDB	Storm	Custom

Literature Review - Data pipeline challenges and requirements

2) Examining the Challenges in Development Data Pipeline [Pervaiz et al,2019]

- Authors have conducted several interviews about the development
- Common challenges - correcting text fields, extracting textual data from different documents like PDF, merging hierarchical data etc.
- Have identified three main phases ; data collection, data cleansing and data analytics.

a) Data collection: In the data collection phase they have segmented it into two subtopics - data entry and tools.

- human errors
- extraction errors
- less skilled and lacking the training

b) Data cleansing: This section includes two subtopics: corrections and structuring

- replacing values or masking, so the privacy of the data is preserved
- In the structuring authors have mentioned merging data from different sources, restructuring




c) Data analytics: In this segment, two important subtopics are mentioned; calculations and visualizations.

- find the missing data
- identify the data outliers
- calculate accuracy for test data
- fine tune values to remove the biases in data sets and training



Literature Review - Data Pipeline Architectures

1) Putting Data Science Pipelines on the Edge [Akoglu et al, 2021]

- new architecture named as “Just in Time architecture for data science (JITA-4DS)”
 - Their main concern is that one-for-all architectures are not efficient enough to cater the requirements of different data science pipelines
 - Those one-fits-all architectures assumes,
 1. High reliability and the high availability of the network
 2. High energy and economically consuming resources
 - Their concern is about the configurations on the cloud infrastructure.
 - dynamic reconfiguration in cloud leaving opportunities on the edge network.
- 



Literature Review - Data Pipeline Architectures

2) Modelling Data Pipelines [Raj et al,2020]

- Propose a new conceptual model of data pipeline
- Designed based on interviews with professionals and organizations to identify challenges in data collection pipeline

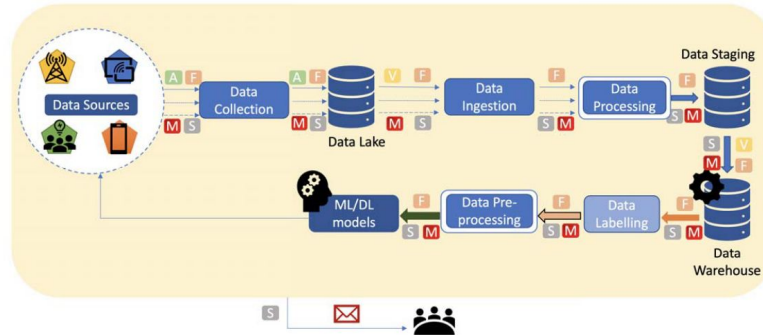


Fig. 6. Conceptual model of data pipeline

Literature Review - Data Pipeline Architectures

3) Data Pipeline Architecture for Serverless Platform [Chinmaya et al,2020]

- propose a novel data pipeline architecture to decompose monolithic data pipeline into independently deployable pipeline components
- Based on TOSCA language which is recently developed standard for cloud-based applications

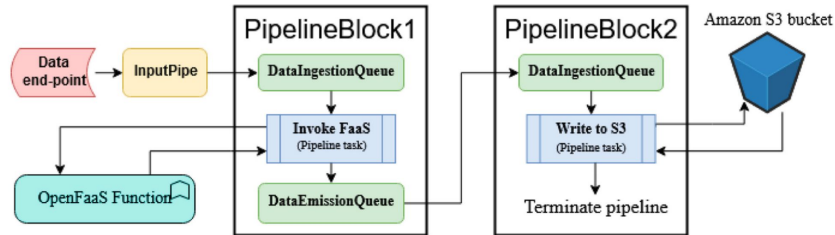
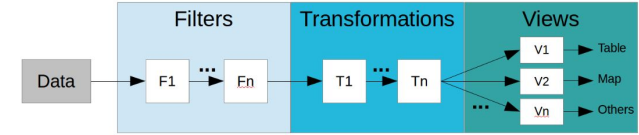


Fig. 2. An example of multiple interconnected PipelineBlock [2].

Literature Review - Data Pipeline Architectures

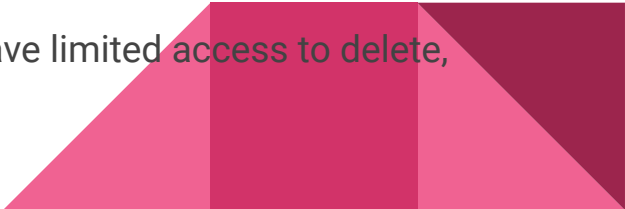


4) Pipeline architecture for mobile data analysis[Arlindo et al, 2014]

- presents a cloud architecture which provides support to collect and analyse mobile data
 - provides easy UI components so anyone coming to this system needs not to have programming knowledge
 - consists of two main components: mobile component and server component
- a) Mobile component is an Android application, and it is an engine of interpreter design pattern - it interprets the xml description
 - b) Server component consists of form editor, analytics editor and database etc.

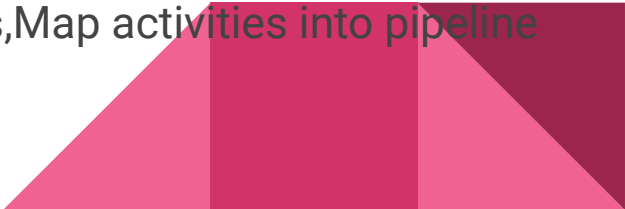
Literature review- Data Pipeline Security

2) Security Supporting Continuous Deployment Pipeline (CDP) [Ullah et al,2017]

- research on the security tactics that can be used in the CDP without affecting the existing behaviour
 - Proposes tactics
 1. Limiting the access of the repository branches so only selected groups can commit/merge code to the default build branch.
 2. Securing main server access with private key along with SSH. AWS supports this.
 3. Use role-based access control (RBAC) for the main server. AWS has an IAM ecosystem.
 4. Setting up clean VM environments for build phase in Jenkins CI server.
 5. Use Jenkins role plugin to control access so that the users can have limited access to delete, create, modify pipelines.
- 

Literature review- Data Pipeline Security

1) Integration Of Security Standards in DevOps Pipelines

- Explore the systematic approach to DevOps pipelines to comply with IEC 62443-4-1 standard
 - Describe the automation capabilities for 4-1 standards with the available tools in the industry
 - proposed eight practices: security management (SM), specification for security requirement (SR), secure design (SD), secure implementation (SI), security validation and verification testing (SVV), management of security related issues (DM), security update management (SUM), security guidelines (SG)
 - Authors suggest three main activities - Describe standard requirements as activities, Determine 4-1 standard automation capabilities, Map activities into pipeline stages
- 

Identified research gaps

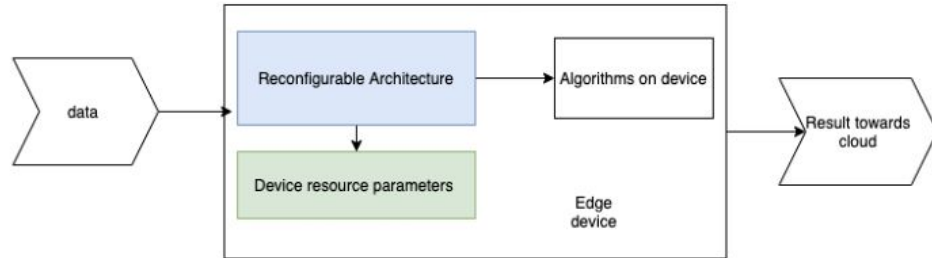
By looking at the literature, few key knowledge gaps are identified.

- One important area is data pipeline architecture which seems to be having less research done.
- Reconfiguration in the edge is not proposed by others.
- We are inspired by the research work of 'Putting Data Science Pipelines on the Edge - Akoglu et al '.
- Following them, we propose research regarding the possibilities of reconfigurable architectures on the edge for data pipelines.

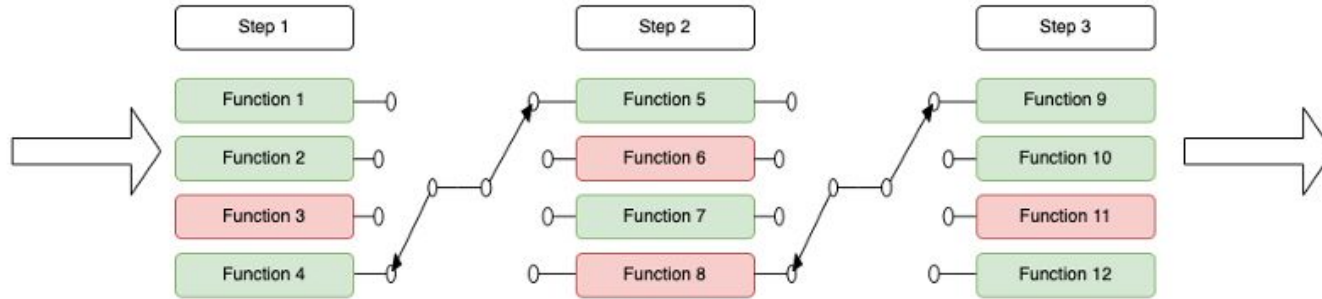


Methodology

1. Review resource availability of the selected devices - Raspberry Pi, Mobile Device etc - consider CPU, RAM, OS capabilities-file system/IO speed
2. Identify the set of features/computational tasks we can provide with these resources
3. Propose an architecture



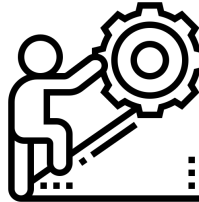
Methodology Cont...



4. Possible idea to develop the architecture as functions layers and connect them based on the resource availability
5. Evaluate the proposed architecture and identify set of algorithms which are benefited by our research.

Identified Challenges in the research

1. Hardware - Find what are the commonly used edge devices in data science domain (Raspberry Pi, Arduino boards, Mobile phones etc)
2. Conceptual design and do the implementation of the edge component and cloud backend within given time frame.
3. Benchmarking the proposed architecture and finding the benefitted category of algorithms.
4. Tune the architecture and map to real world product which will use our design and get improved the efficiency.



Resources required

1. Edge devices - Raspberry Pi Cluster at CSE
2. Cloud infrastructure -AWS
3. Data sets - E.g : <https://github.com/swinedge/eua-dataset> - need to find a better data set while doing the research.

