Task 1: Analysis and Discussion (Case Study)

Selected Article(s):

https://www.mdpi.com/1424-8220/21/7/2282

https://www.komprise.com/wp-content/uploads/2021_AWS_Partner_Komprise_Case_Study_FINAL.pdf

1.0 Introduction

This report will be focusing on Pfizer which collabed with Komprise (an AWS storage partner) to handle the vast and growing volumes of data generated throughout the COVID-19 pandemic. (*Pfizer's 75% storage cost reduction: Cloud Data Management* 2024).

However, the selected article lacks detailed information on the steps of Big Data, resulting in the need of additional resources on Big Data's impact on the pandemic. For instance, Alsunaidi et al. (2021) provides a comprehensive literature review on COVID-19-related Big Data analysis, including a taxonomy of applications used to manage and control the pandemic. The study also addresses the challenges faced in analyzing COVID-19 data and suggests valuable directions for future research and practical applications. *(Alsunaidi et al., 2021).*

Figure 1. Pfizer Logo (Pfizer Malaysia, 2024)

The World Health Organization on January 30, 2020, pronounced coronavirus COVID-19 as a public health emergency of international concern (PHEIC). As a result, governments were directed to act firmly and immediately to stop the spread of the virus *(Alsunaidi et al., 2021)*. As for the pandemic, Pfizer solutions provider engaged in vaccine development known as COVID-19 vaccine manufacturer worldwide faced a problem related to data management. Comprising stipulating cold data strategy, 5 petabytes of disorganized data (64 percent of those data were last used over two years ago) posed a problem in the direction of appropriate placement of the company's infinite, ever-increasing amounts of data. With the help of Komprise, Pfizer was able to select and effortlessly transmit cold files to AWS while minimizing expenditures and sequentially framing a geo-tagged data lake. *(Pfizer's 75% storage cost cut: Cloud Data Management 2024).*

In order to effectively manage the Covid-19 pandemic, it is important to examine and study its attributes, features and behavior through the analysis of large scale data sets. Furthermore, the utilization of advanced surveillance technologies for the monitoring and tracking of suspected COVID-19 patients generates huge datasets, therefore presenting a range of possibilities whereby big data analytics can be employed to improve the health care system. Tools such as Apache's cloud-optimized big data components aid cloud-based solution development for specific data analytics. The Six V's in Big Data — Value, Volume, Velocity, Variety, Veracity, and Variability which contain underlying features – have their usage well anchored on the healthcare sector, even if at first only Volume, Velocity and Variety were deemed seminal. *(Alsunaidi et al., 2021).*

There are various industries which previously believed that big data analytics only has theoretical concepts and yet, the healthcare industry especially during COVID-19 demonstrated otherwise especially for virus tracking, predicting demands and optimizing treatment. *(Alsunaidi et al., 2021).* This paper is organized in a manner to highlight its key achievements, reports challenges met in utilizing and interacting with COVID-19 data, and suggests possible future research avenues. It also focuses on data analytics made in Amazon S3 and Komprise concerning data collection, data integration, data storage and data processing.

2.0 Data Collection

In Alsunaidi et al. (2021) the authors utilized the survey method for data collection, gathering insights from various studies and real-world applications to analyze the role of big data in healthcare, particularly during the COVID-19 pandemic.

In the same way, Pfizer chooses to collect its user data by the same methods mentioned. gathering personal information that users choose to provide through registrations, applications, surveys, or inquiries. This may include details such as name, contact information, health status, insurance, or financial information, typically in connection with promotions, patient assistance or support programs, or clinical trials. Healthcare providers may also share information related to their specialties and professional affiliations. Additionally, Pfizer collects data automatically during interactions on the Site, including information like IP address and user navigation patterns. *(Privacy policy-en: Pfizer - working together for a healthier world, n.d).*

2.1 Data Collection Methods Analysis

In the study, Alsunaidi et al. (2021) illustrate the data collection techniques, which involve a number of sources and provide a detailed understanding of the health status of the patients. EHR includes important facts about the patient, such as their demographic information, health history and vital signs, and allows for a narrower encapsulation of the entire cycle of treatment EHR allows for. Physiological variables such as heart rate or oxygen saturation commonly, and physical activity to a certain extent are measured today via medical or body-worn devices like sensors or wearables that provide dynamic data and enable early detection of health status and its changes. *(Abdel-Basst, Mohamed, & Elhoseny, 2020)*. The former consists of headphones and a mobile phone to screen for potential breathing problems; the detected signals are then saved in audio format in the mobile application.*(Stojanovic, Skraba, & Lutovac, 2020)*. The nature of insurance claims allows obtaining information both about finances as well as treatment; hence this would assist in establishing health seeking behaviour in terms of cost and the nature of health services utilization.

Table 2 in the article describes the data analysis techniques, types, sources, and findings derived from existing studies within the healthcare sector. It provides an overview of prior research concerning data analysis methodologies and their corresponding outcomes. The figure below is a screenshot from Table 2. The authors Abdel-Basst, Mohamed, & Elhoseny have proposed a model to differentiate between COVID19 and four other viral chest diseases. Several detectors are incorporated in the model including temperature, blood & glucose and heart rate sensors, and respiratory monitors which allow to obtain data and supervise the health condition of the patient. *(Abdel-Basst, Mohamed, & Elhoseny, 2020)*

Area	Ref	Aim	Technique	Used Data Type	Data Source	Findings
	[39]	Develop a diagnosis model for COVID-19 detection and diagnosis of symptoms to define appropriate care measures	Best Worst Method (BWM)	Symptoms and CT scans	Body sensors	The model can differentiate COVID-19 from four other viral chest diseases with 98% accuracy

Figure 2. Table 2 of Applications of Big Data analytics to control Covid-19 (Alsunaidi et al., 2021)

Based on type and source, medical data can be categorized into six groups, as shown in Figure 3. We can predict future events, understand current situations, and make decisions with the help of this data analysis.

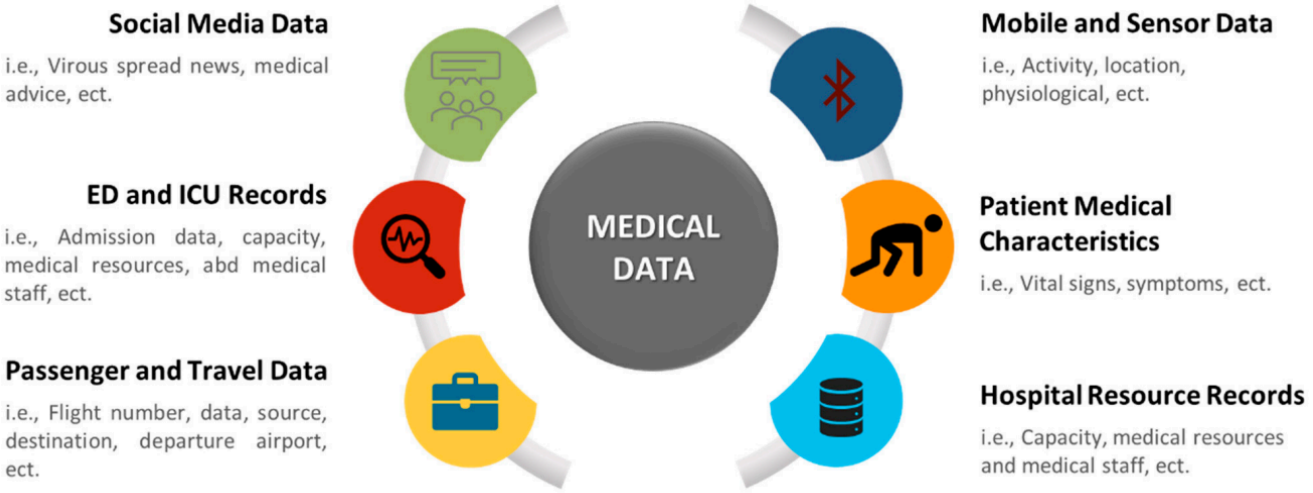

Figure 3. Type and Source of medical data (Alsunaidi et al., 2021)

2.2 Data Collection Challenges

The researchers mentioned in the paper, Abdel-Basst, Mohamed, and Elhoseny, (2020) when analyzing this perspective have come across a major drawback, and that is "It is unclear how the health information of the patient will be communicated to the hospital personnel," which bears on the problem of observing the confidentiality of the patient concerned. Moreover, security of healthcare data and the privacy of patients are sensitive issues for authorities and patients alike. Medical data is distributed in select circumstances to select professionals and or researchers for certain reasons. Hence, there is a need to come up with effective and firm frameworks, plans and policies that will regulate who access medical data and in what context without infringing privacy of patients nor allow the abuse of the data for unscrupulous reasons, in this case crisis situations and outbreaks of deadly disease epidemics like the recent COVID-19 pandemic. *(Abouelmehdi et al, 2017)*

About patient anonymity, the immigration from the Wuhan area owing to the COVID-19 virus raised several red flags owing to the virus's potential such that it posed threats to other States and the need for each State to have protective measures that will help deter the spread of the virus. This challenge can be overcome using Blockchain technology *(Ahir et al, 2020),* which allows patients to remain anonymous while letting realistic sharing of large volumes of information more securely.

These situations involve instances when wrong information is provided and included in cyberspace including misinformation about diseases, effects of vaccines and other health-related issues. Such misinformation may work against the goals and objectives of governmental and health organizations and slows down the efforts to fight the spread of the virus infection and protect the health concerns of the people. Furthermore, lack or incorrectness of data in observational studies may create further bias in the research findings. *(Richardson et al., 2020)*. However, artificial intelligence-enhanced services and tools, combined with big data analytics, can be used to search for and filter harmful information on the Internet, issue warnings, and remove it from the web. *(Alotaibi, Mehmood & Katib 2020)*

Finally, patients are the most essential people as they are the first to understand the characteristics of newly emerging epidemics and in that way there's an escalating request for sharing sustenance information like their health constitutional data with research organizations. Also, sharing more physiological data acquired from the use of wearable devices can also assist in the creation of such predictive systems. But then, a sizable proportion of people are reluctant in sharing their health and personal details including sex and place of origin. In the past, a poll carried out in January 2020 managed to establish that only about 37 percent of the 4600 individuals who had no chronic diseases were ready to submit this sort of information to organizations dealing with medical research. (Davis, 2020). It is therefore very important to educate people on the need for blind data sharing. For better confidence in data privacy, it is also necessary to define the legal entities collecting the data and the policies they follow.

2.3 Tools for Real-Time Data Collection

In the article, they gave a lot of options of tools used; the Internet of Things (IoT) refers to a network of interconnected devices that collect and share data, it refers to tools like wearable devices, smart watches or health-monitoring devices, which are equipped with various sensors, can collect continuous patient data. By automating data collection and analysis, IoT increases efficiency, lowers costs, and accelerates medical research. *(Pegg, 2024)*. In reference to some of the other tools, we have headsets and mobile phones, web Apps, and body sensors. All of these tools were mentioned in Table 2 of the article. *(Alsunaidi et al., 2021)*.

3.0 Data Integration

3.1 Purpose in COVID-19 Healthcare

Data integration is critical for COVID-19 healthcare because it allows healthcare organizations to combine data from multiple sources into a centralized repository. The COVID-19 pandemic generated large amounts of data from Electronic Health Records (EHRs), medical devices, insurance claims, testing data, and social media, among others. Integrating these diverse datasets is essential to create a holistic view of the pandemic's impact, predict trends, allocate resources efficiently, and make real-time, data-driven decisions. Without integration, data from different sources remain isolated, making it difficult to track the virus's spread, manage patient care, predict healthcare resource demands (e.g., ICU beds, ventilators), or develop effective public health interventions. *(CData Software, 2024).*

3.2 Data Integration Tools

As mentioned, Komprise (an AWS storage partner) provides organizations like Pfizer, one platform for unstructured data management & data mobility. Komprise's data tiering and integration with AWS enabled Pfizer to migrate data without disrupting user access, ensuring continuity in research and testing processes. Through the Komprise dashboard, Pfizer gained clear visibility into data status and age, facilitating more efficient data management. By automating processes and utilizing cost-effective AWS S3 storage, Pfizer successfully streamlined its data operations and significantly lowered overall costs. *(Pfizer's 75% storage cost reduction: Cloud Data Management 2024)*.

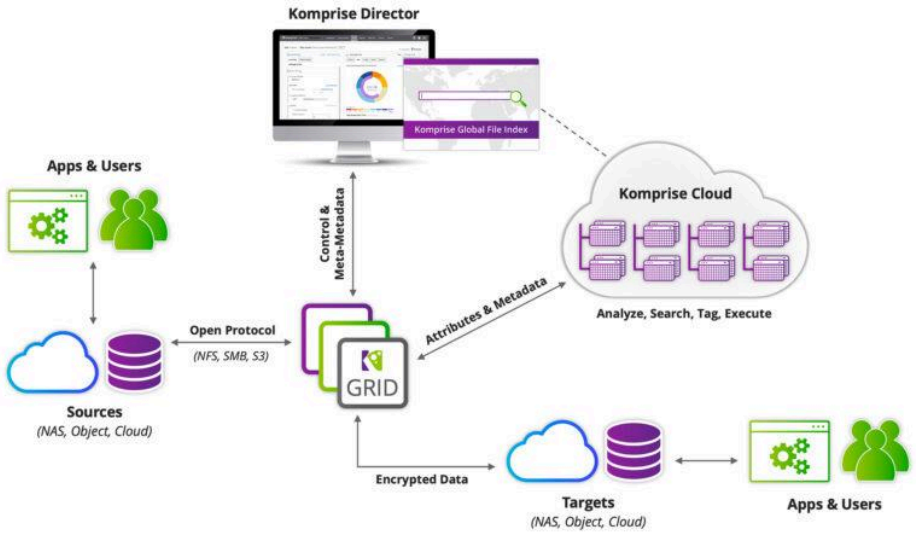

Figure 4. Architecture of Komprise (Elastic, scale-out architecture: Komprise Data Management 2024)

Figure 4 illustrates the architecture of Komprise, which is built on a distributed, scalable, and fault-tolerant framework. This architecture includes stateless Observers positioned near storage, optimizing data analysis and mobilization. Komprise provides a centralized Global File Index and management console, allowing unified visibility and control over data across multiple silos. The platform operates using standard protocols (NFS, SMB, S3), without relying on agents or stubs, and avoids the hot data path. Delivered as a cloud service, Komprise is designed to be both easy to set up and user-friendly. *(Elastic, scale-out architecture: Komprise Data Management 2024)*.

Through its partnership with Amazon S3 (an AWS storage tool), Komprise supports networking, storage, and security features. *(Benefits of using Amazon EMR - AWS documentation 2024)*.

4.0 Data Storage

4.1 Data Type and Volume

According to Pfizer's 75% storage cost reduction: Cloud Data Management 2024 (2024), there is also data known as unstructured data that the company has. This kind of information is contained in clinical notes, records of surgical procedures, hospital discharge summaries, radiological examinations, medical imaging, as well as pathology records mapped within the electronic health records (EHRs) *(Sun et al. 2018)*. There is a consensus that unstructured data is difficult to deal with but contains stenographic information which is important. For example, free-text consisting of care details of the patients in the course of the entire hospital stay is included in the discharge summary and other clinical documents but is hard to obtain because it exists in varied contexts and has a level of uncertainty which mostly characterizes medical reporting. Most clinical texts are often written in a language that is filled with a lot of complexities; grammatical/ spelling mistakes, use of ambiguous language, and use of abbreviations which makes the task of processing and analyzing this data ever challenging *(Tayefi et al, 2021)*.

Volume in big data is often misunderstood as its importance in providing insights in the organization but what it actually stands for is the amount of data created and stored. This data can range from tera-based up to five to six different scales of data (peta, exa). Such huge amounts of data (especially big data) have their own challenges in the areas of storage, processing and analysis as well. *(Framework 2024)*. Incase of Pfizer, it has been said before that the company has around five petabytes of unstructured data. *(Pfizer's 75% storage cost reduction: Cloud Data Management 2024)*.

To see the full scope of datasets collected in the COVID-19 pandemic Alsunaidi et al., 2021 summarizes it in the bar graphs below.. These datasets include demographic data, social data, activity

data, and travel data, etc. In order to visualize the amount of data, figure 3, 4, 5 displays the multiple data types found in healthcare and data distribution, citing from multiple sources done by the authors of the paper. *(Alsunaidi et al., 2021)*.

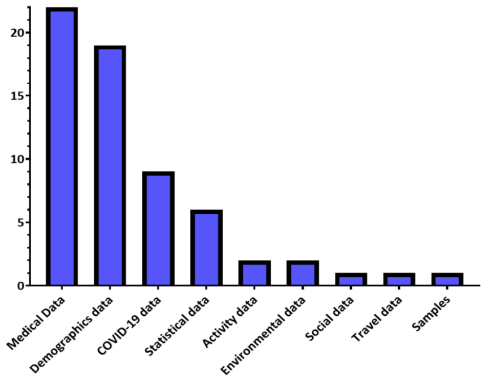

Figure 5. COVID-19 data distribution in the reviewed studies. (Alsunaidi et al., 2021)

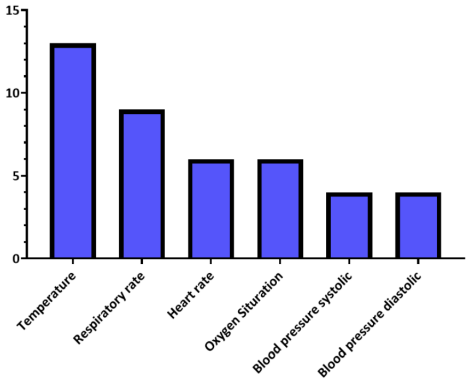

Figure 6. Vital signs' distribution in the reviewed studies. (Alsunaidi et al., 2021)

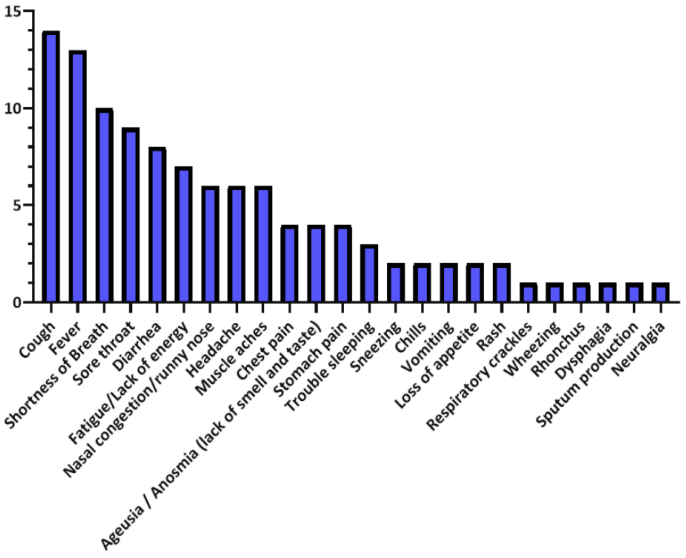

Figure 7. Symptoms' distribution in the reviewed studies. (Alsunaidi et al., 2021)

4.2 Data Storage Tools

Komprise, in partnership with Amazon S3, delivers data storage solutions adapted to Pfizer's needs for managing extensive unstructured data. As an AWS Migration and Modernization competency partner, Komprise provides an efficient, flexible path to cloud storage for both file and object data, integrating seamlessly with AWS services, including Amazon EFS, Amazon FSx, and Amazon S3. This integration supports advanced storage classes, such as S3 Glacier Flexible Retrieval and Glacier Instant Retrieval, to optimize Pfizer's data storage capabilities. With an analytics-driven platform, Komprise enables Pfizer to simplify unstructured data management, offering immediate implementation without agents, stubs, or disruptions to existing infrastructure. (*Seamlessly archive cold data to Amazon S3 Glacier using .., n.d.*)

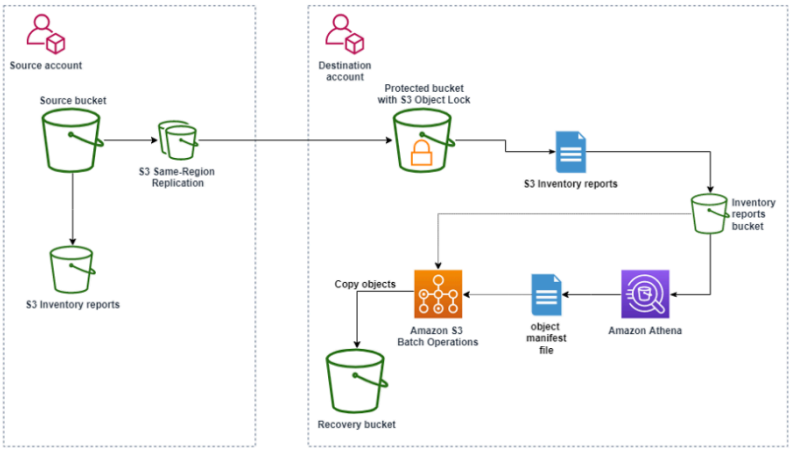

Figure 8. Architecture layout of Amazon S3 (Bhattacharya et al., 2023)

Figure 8 above displays the architecture layout of Amazon S3 and its components. Amazon S3 is an industry-leading object storage service recognized for its scalability, durability, and security. Designed for high reliability, it supports a wide range of use cases, including data lakes, machine learning, analytics, and cloud-native applications, and is built to achieve 99.999999999% (11 nines) durability. This architecture enables Pfizer to store and secure critical data on a highly resilient platform, optimized for performance and data protection. (*Amazon S3 - Cloud Object Storage - AWS, n.d*)

The key benefits of using Amazon S3 within Pfizer's data environment include scalability, where storage expands elastically to support exabytes of data without manual provisioning, and durability and availability, backed by industry-leading SLAs. Amazon S3's security and data protection features are also notable, offering encryption by default, stringent compliance capabilities, and advanced auditing to

monitor access to sensitive resources. Moreover, Amazon S3 offers multiple storage classes designed to optimize costs according to data access needs, enabling efficient storage of frequently accessed data as well as long-term archival data. (*Amazon S3 - Cloud Object Storage - AWS, n.d*)

4.3 Other Data Storage Tool recommendations

Other sources of recommended storage systems in the article are under Table 4 of the article - *(Alsunaidi et al., 2021)*. Figure 9 is a screenshot that shows us the table and its sources.

Data Storage	Description	Website
Cloudera	It extends the Hadoop with extra services	https://www.cloudera.com, Accessed on: 18 March 2021
Apache Cassandra	Distributed database management system, multiple servers	https://cassandra.apache.org/, Accessed on: 18 March 2021
Chukwa	Hadoop distributed file system (HDFS)	http://chukwa.apache.org/, Accessed on: 18 March 2021
Apache HBase	Hadoop distributed file system (HDFS)	http://hbase.apache.org/, Accessed on: 18 March 2021
MongoDB	Document-oriented database	https://www.mongodb.com/, Accessed on: 18 March 2021
Neo4j	java—graph database	https://neo4j.com/, Accessed on: 18 March 2021
CouchDB	Globally distributed server-clusters	https://couchdb.apache.org/, Accessed on: 18 March 2021
Terrastore	Distributed Database Management System (DBMS) that provides per-document consistency guarantees	https://code.google.com/archive/p/terrastore/, Accessed on: 18 March 2021
HibariDB	Hibari is a distributed, ordered key-value store	https://hibari.readthedocs.io/en/latest/index.html, Accessed on: 18 March 2021
Riak	NoSQL database, cloud storage	https://riak.com/, Accessed on: 18 March 2021

Figure 9. Data Storage recommendations. (Alsunaidi et al., 2021)

5.0 Data Processing

5.1 Real-Time and Batch Data Processing and their Examples

Batch Processing

Referencing a different article, it talks about Apache Hadoop for managing COVID-19 data, emphasizing its global reach and flexibility. The authors introduce a "Hadoop Fusion Approach" as an optimal solution for handling diverse data needs. They demonstrate how Hadoop delivers consistent results across different models, whether partial submodels or complete systems. Additionally, Hadoop's adaptability makes it well-suited to meet the varied requirements of COVID-19's heterogeneous data landscape. Finally, the authors propose a framework of strategic data processing conditions, identifying Hadoop as a key solution to overcome the challenges posed by the pandemic's massive data demands. *(Azeroual & Fabre, 2021)*.

In the context of COVID-19, healthcare data processing has become a staple challenge because of unprecedented surges of heterogeneous data collected from multiple sources. Apache Hadoop provides solutions for batch processing as well as real time processing of this data. The MapReduce is a common batch processing framework in Hadoop. It divides massive datasets into smaller blocks of data that are processed in parallel over a cluster of nodes. It enables healthcare organizations to manage Electronic Health Records (EHRs), COVID-19 test results, and patient monitoring device data quickly worldwide. *(Azeroual & Fabre, 2021)*. Figure 10 gives us the systematic view of MapReduce and how it processes data.

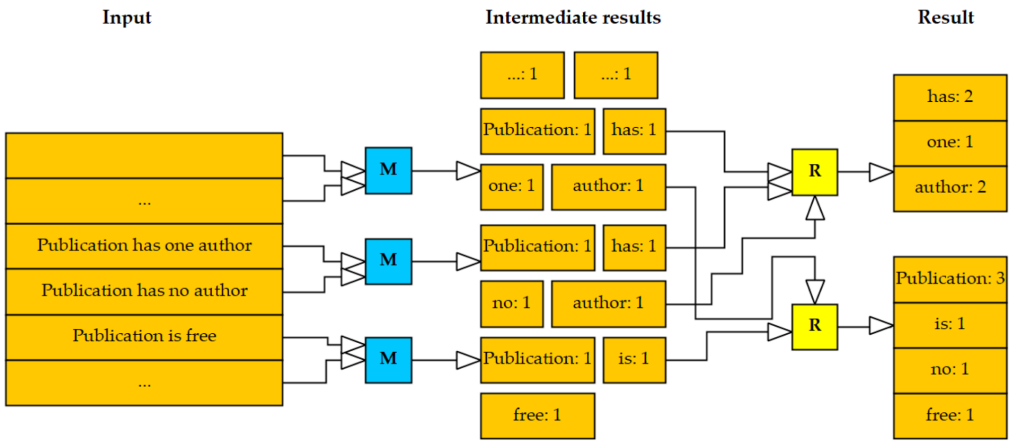

Figure 10. Map Reduce Systematic View. (Azeroual & Fabre, 2021)

Real-Time Processing

Hadoop is also capable of on-line processing but is not inherently suited to it, hence there can be integration with other tools such as Apache Spark which are pivotal for real-time data streams (e.g. Monitoring Patients Vitals in ICU). This combination ensures that patient health data is processed without delay so that providers can react as soon as a change in their patients' conditions occurs. To give an example, wearable sensors measuring patients' heart rates, oxygen levels and temperatures can continuously stream big data to Spark. Labored breathing or a sudden drop in oxygen levels can trigger the system with an alert, allowing doctors to react immediately instead of waiting weeks for test or scan results. *(Azeroual & Fabre, 2021).*

Spark became an open-source project in 2010. It was initially developed by Zaharia at UC Berkeley's AMPLab in 2009. Spark provides several benefits for developers working on big data applications. It introduced two key concepts: Resilient Distributed Datasets (RDD) and Directed Acyclic Graphs (DAG). These techniques work together to make Spark significantly faster than Hadoop in some cases—up to ten times faster—but typically it performs two to three times faster than MapReduce *.(Ahmed et al., 2020).*

Spark supports multiple data sources, includes fault tolerance mechanisms, allows caching, and enables parallel operations. It can also represent a single dataset in multiple partitions. When running on a Hadoop cluster, RDDs are created on HDFS in formats supported by Hadoop, such as text and sequence files. The DAG scheduler expresses the dependencies between RDDs. For every Spark job, a DAG is created, and the scheduler breaks it into stages of tasks, which are then sent to the cluster. The DAG includes both map and reduce stages to represent all dependencies. Figure 10 shows the iterative operation on RDDs. However, in theory, limited Spark memory can lead to slower performance. *(Ahmed et al., 2020).*

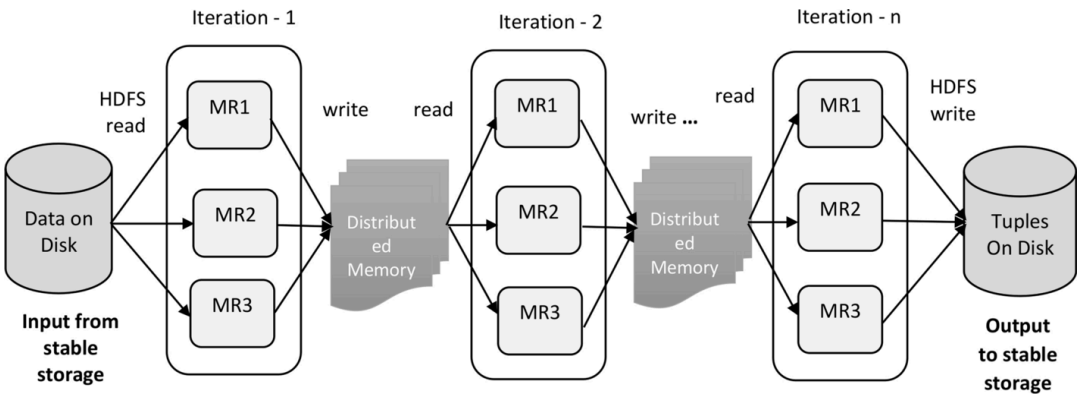

Figure 10 Spark Workflow. (Ahmed et al., 2020)

Task 2: Analysis and Evaluation of existing solutions

Selected Article: https://link.springer.com/article/10.1186/s40537-019-0271-7#citea

6.0 Analysis and Evaluation

a) Select and analyze an existing architecture diagram related to a healthcare big data solution, highlighting its different layers, main components, and their relationships.

The proposed architecture is a data processing and monitoring application based on Kafka and Spark streaming. This application is designed to collect real-time data from connected devices, process it, and store it for real-time analysis. Figure 11 shows the proposed architecture.

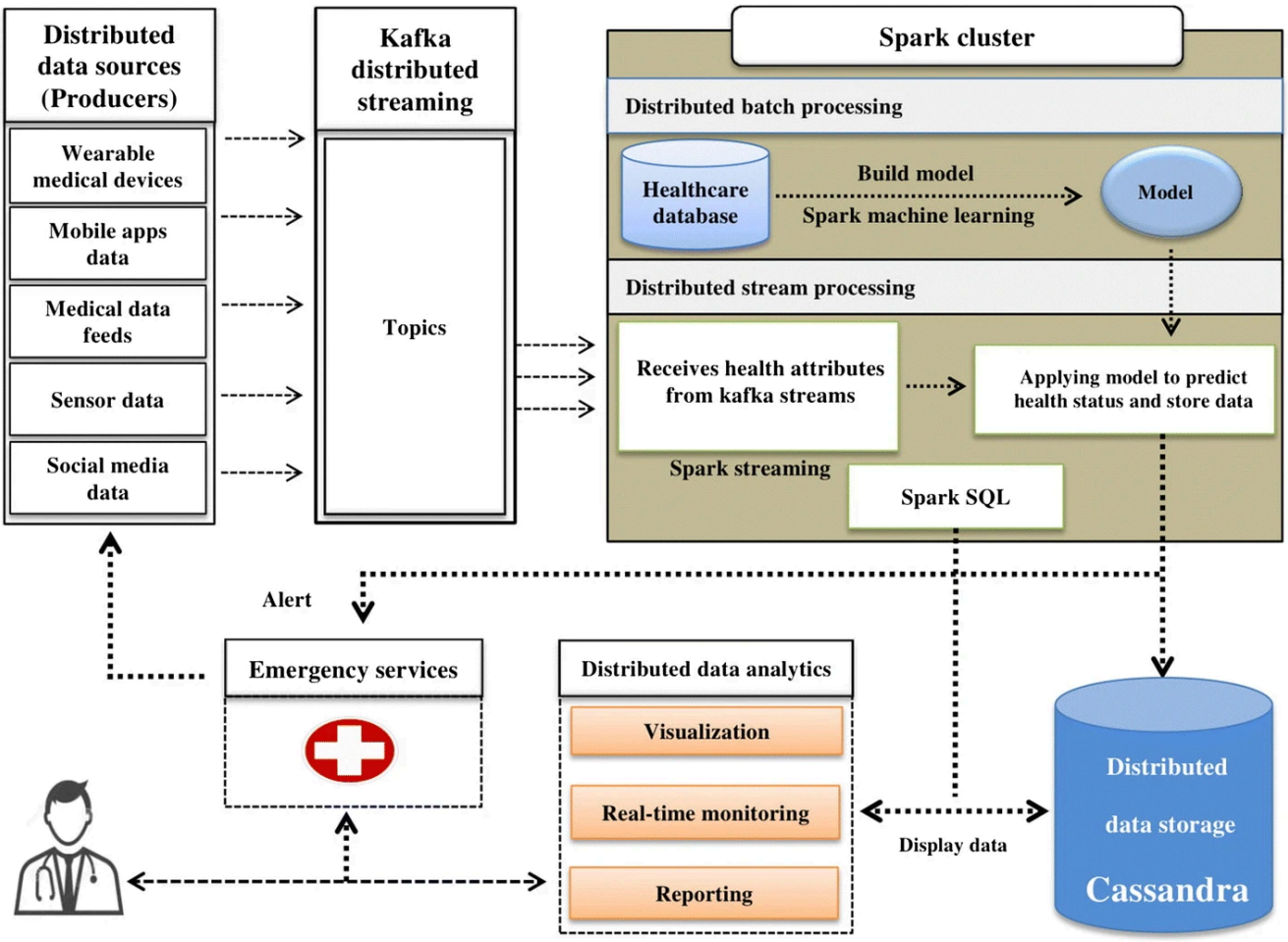

Figure 11. Architecture of real-time health status prediction and analytics system (Ed-daoudy & Maalmi, 2019)

1. **Data Source Layer (Distributed Data Sources)**
    - **Components:** Wearable medical devices, mobile apps, medical data feeds, sensor data, and social media data. *(Ed-daoudy & Maalmi, 2019)*
    - **Functionality:** This layer serves as the entry point for raw data collection. Various data sources continuously generate healthcare-related data, which is then forwarded to the next layer for processing. *(Ed-daoudy & Maalmi, 2019)*
    - **Relationship:** These data sources act as producers and feed data to the streaming platform (Kafka) for further processing. *(Ed-daoudy & Maalmi, 2019)*
2. **Data Streaming Layer (Kafka Distributed Streaming)**
    - **Components**: Kafka topics. *(Ed-daoudy & Maalmi, 2019)*
    - **Functionality:** This layer is in charge of receiving, buffering, and routing continuous streams of data from various sources. Apache Kafka is a distributed streaming platform that organizes high-volume, high-velocity healthcare data into topics. *(Ed Daoudy & Maalmi, 2019)*
    - **Relationship:** Kafka forwards the organized data streams to the Spark cluster, where they undergo further processing. It decouples data producers and consumers, ensuring data flows efficiently to the processing layer. *(Ed-daoudy & Maalmi, 2019)*
3. **Processing Layer (Spark Cluster)**
    - **Components:** Distributed batch processing, distributed stream processing, Spark machine learning, Spark streaming, Spark SQL, and the healthcare database. *(Ed-daoudy & Maalmi, 2019)*
    - **Functionality:** This layer processes and analyzes data using Apache Spark. It is divided into:
        - **Distributed Batch Processing:** Uses data from the healthcare database to build models via Spark's machine learning capabilities. *(Ed-daoudy & Maalmi, 2019)*
        - **Distributed Stream Processing:** Real-time data processing from Kafka, applying machine learning models to predict health status. *(Ed-daoudy & Maalmi, 2019)*

- **Relationship:** The batch processing segment uses historical data from the healthcare database to build predictive models. The stream processing segment receives real-time health attributes from Kafka, applies the models, and stores processed data in a distributed database (Cassandra). Spark SQL supports structured query processing and data manipulation within this layer. *(Ed-daoudy & Maalmi, 2019)*

4. **Data Storage Layer (Distributed Data Storage - Cassandra)**

   - **Components:** Cassandra (a NoSQL database). *(Ed-daoudy & Maalmi, 2019)*
   - **Functionality:** In this layer, it can store real-time data and process data for a more extended period to be stored in large storage. Ideal for big data, Cassandra is scalable and fault-tolerant. (Ed-daoudy & Maalmi, 2019)
   - **Relationship:** Finally, Cassandra is called to take over the processed data from the Spark cluster and it acts as a central storage point where data can be analyzed in real-time or accessed for an urgent alerts system. (Ed-daoudy & Maalmi, 2019)

5. **Analytics and Visualization Layer (Distributed Data Analytics)**

   - **Components:** Visualization, real-time monitoring, and reporting. *(Ed-daoudy & Maalmi, 2019)*
   - **Functionality:** This layer supports healthcare professionals in monitoring patient data in real-time, visualizing health metrics, and generating reports. This information is crucial for informed decision-making and continuous patient care. *(Ed-daoudy & Maalmi, 2019)*
   - **Relationship:** The analytics layer fetches data from Cassandra, displaying key insights and supporting the generation of alerts. *(Ed-daoudy & Maalmi, 2019)*

6. **Emergency Alert System (Emergency Services)**

   Components: Alert system and emergency response.*(Ed-daoudy & Maalmi, 2019)*

   Functionality: This component is designed to detect critical health incidents and trigger alerts to healthcare providers and emergency services, ensuring rapid response. *(Ed-daoudy & Maalmi, 2019)*

   Relationship: The emergency alert system is integrated with the analytics layer and receives alerts when specific health conditions are met. This system can then inform healthcare providers or emergency services directly for immediate intervention. *(Ed-daoudy & Maalmi, 2019)*

b) Discuss and evaluate the role of each component in the architecture, explaining how they interact to achieve the solution's objectives.

1. IoT Devices and Sensors - Data Collection from Distributed Sources

This architecture is built on a foundation of Internet of Things (IoT) devices — think wearable health monitors — that constantly create and send patient data. Fitbit and other similar devices that sense metrics of interest such as heart rate, glucose concentration, and activity patterns connect to Kafka in real-time. The IoT devices gather a lot of data which is essential in generating health profiles and warning signs that the individual may be deteriorating in their health. The workflow of the proposed system with different data sources included is shown in Figure 12. *(Ed-daoudy & Maalmi, 2019)*

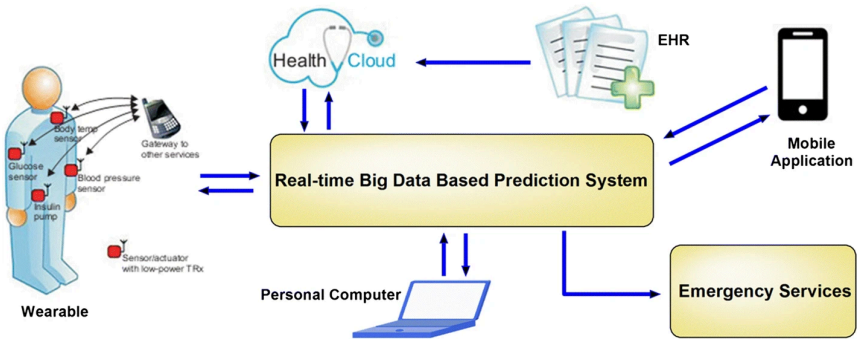

Figure 12. Workflow for the proposed system (Ed-daoudy & Maalmi, 2019)

2. Kafka - Real-Time Data Collection and Stream Management

The Apache Kafka Streaming platform acts as the data backbone, aggregating oceans of data from disparate health monitoring systems like IoT devices or Klara wearable health monitors. It can handle large amounts of data and is able to group it under certain topics, all concerning different health conditions. This seamlessly labels data streams by disease or health metric, which also allows the data to be routed to the correct processing pipelines without requiring any kind of significant integration efforts. Due to Kafka's distributed and partitioned architecture, it can be highly scalable and reliable with thousands of producers publishing real-time data in the form of topics which multiple consumers such as the Spark streaming engine subscribes to for processing upon completion. Zookeeper is a distributed system coordination and facilitation tool as shown in Figure 13 and its continuous development, turbulent introductory definition designed for maintaining large scale API across distributed systems. *(Ed-daoudy & Maalmi, 2019)*

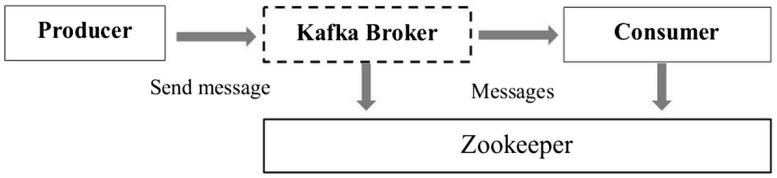

Figure 13. Kafka messaging system (Ed-daoudy & Maalmi, 2019)

3. Spark Streaming - Real-Time Data Processing

After Kafka collects the data, it forwards it to Apache Spark for processing. Apache Spark, an open-source distributed processing engine that has become famous for its speed and versatility. Due to performing on in memory and making use of resilient distributed datasets (RDD) spark is best suited for large-scale data processing and machine learning. RDD (Resilient Distributed Dataset): Immutable collections of objects spread across a cluster which makes parallel processing and faster caching of data possible. Spark has its own MLlib library for different machine learning algorithms, and it supports streaming through Spark Streaming.

The master-worker architecture of Spark is shown in figure 14, the driver process that is responsible for distributing tasks and subprocesses called executors across a cluster. This architecture, combined with its core features, positions Spark as a versatile tool for tackling complex data challenges in various domains. *(Ed-daoudy & Maalmi, 2019)*

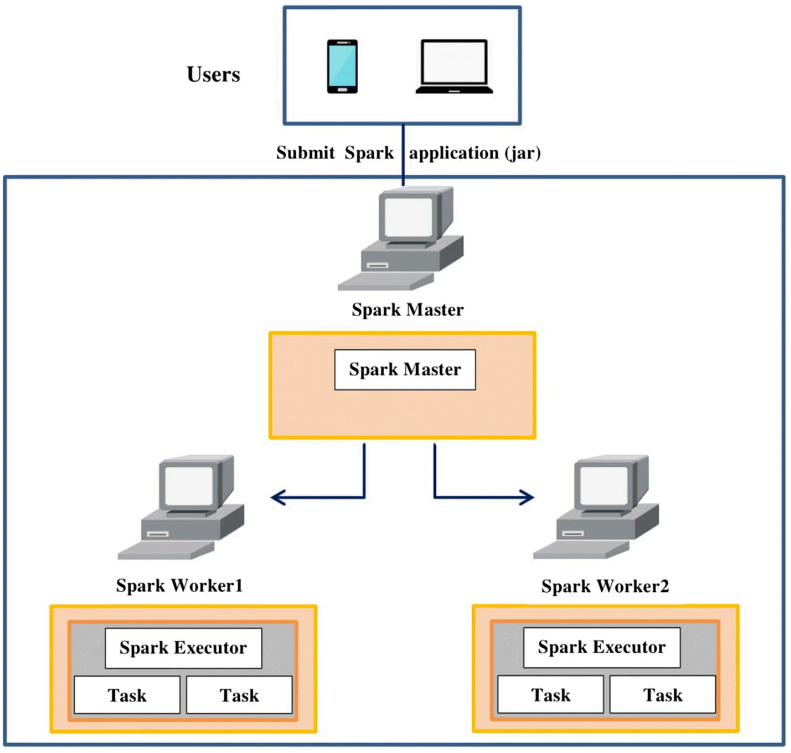

Figure 14. Spark architecture (Ed-daoudy & Maalmi, 2019)

4. Cassandra - Distributed Data Storage

Cassandra, a No SQL scalable and fault-tolerant database where processed data from Spark is saved for continual influx of healthcare data. As a distributed database, Cassandra scales to big data effectively and keeps the historical and real-time data available for additional analytics and reporting. Cassandra backing onto a structured and reliable way of storing data allows the system to monitor trends over time which gives healthcare providers an opportunity to identify patterns and make better decisions. *(Ed-daoudy & Maalmi, 2019)*

Cassandra offers several advantages over traditional database systems. They are highly scalable, providing better performance and cost-effectiveness. Features such as partitioning, replication, and rapid data transfer contribute to their efficiency. Additionally, Cassandra's seamless integration with Spark enables efficient data processing and storage, making it a valuable tool for healthcare data management. *(Ed-daoudy & Maalmi, 2019)*

Figure 15 shows us Cassandra architecture, it can run on multiple machines and these machines, or nodes, communicate via a gossip protocol. Unlike traditional databases with a master node, Cassandra is masterless, making it highly resilient. Nodes can be organized into clusters or 'rings' or even distributed across multiple datacenters. *(Cassandra Basics, n.d)*

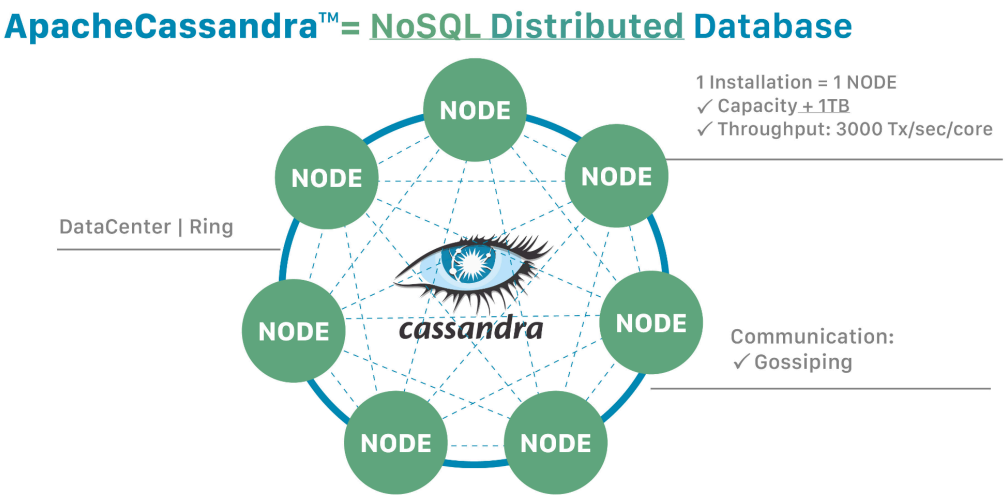

Figure 15. Cassandra architecture (Cassandra Basics, n.d)

5. Apache Zeppelin - Data Visualization and Dashboarding

To make data actionable, Apache Zeppelin is employed to visualize and analyze stored data, retrieving information from Cassandra to create real-time dashboards with charts, tables, and graphs. This allows healthcare professionals to easily track patient metrics and trends, viewing patients' health status at a glance and receiving immediate notifications for critical changes. The dashboard automatically refreshes at set intervals, ensuring up-to-date information, which is crucial for monitoring patients with chronic conditions. *(Ed-daoudy & Maalmi, 2019)*

Apache Zeppelin is a web-based, open-source notebook that enables interactive data analytics on top of Apache Spark. It provides interactive and exploratory data visualization in real-time for multiple programming languages including Scala, Python, and SparkSQL that allows you to work with your teammates. Zeppelin also allows users to develop, organize, and run analytics code within long workflows, dynamically creating input forms and sharing notebook URLs among collaborators. *(Ed-daoudy & Maalmi, 2019)*

Using Zeppelin, a real-time data dashboard has been created to retrieve data from Cassandra and display it in various visual formats, refreshing every second. This dashboard can be shared with authorized individuals such as physicians, health firms, or consultants, enabling them to access collected data and monitor patient health status remotely. *(Ed-daoudy & Maalmi, 2019)*

6. Emergency Alert System - Real-Time Response to Health Changes

Integrated within the visualization layer, the emergency alert system leverages data insights to trigger alerts when a patient's health metrics reach critical thresholds. This system ensures that care providers receive timely notifications, enabling rapid responses to medical emergencies. By combining data visualization with real-time alerts, this component directly supports the architecture's objective of saving lives by providing early intervention opportunities in cases of medical emergencies such as heart attacks or diabetic crises. *(Ed-daoudy & Maalmi, 2019)*

c) Choose and introduce a healthcare-related dataset used in the existing solution, discussing its characteristics and relevance.

The selected article uses two dataset, both extracted from different sites, the first dataset is a diabetes dataset from Kaggle, which provides access to open-source data for data science. This dataset includes 15,000 records with nine attributes: pregnancies, glucose, blood pressure, skin thickness, insulin, BMI, diabetes pedigree function, age, and a binary outcome indicating whether a patient has diabetes (1) or not (0). This dataset is particularly valuable as it provides key health metrics relevant to diabetes risk, enabling the model to analyze patterns and identify predictors of diabetes. *(Ed-daoudy & Maalmi, 2019).* The sample data of the first dataset is shown in Figure 16 below.

# Pregnancies Number of times pregnant	# Glucose Plasma glucose concentration a 2 hours in an oral glucose tolerance test	# BloodPressure Diastolic blood pressure (mm Hg)	# SkinThickness Triceps skin fold thickness (mm)	# Insulin 2-Hour serum insulin (mu U/ml)	# BMI Body mass index (weight in kg/(height in m)^2)	# DiabetesPedigre... Diabetes pedigree function	# Age Age (years)	# Outcome Class variable (0 or 1) 268 of 768 are 1, the others are 0
0 17	0 199	0 122	0 99	0 846	0 67.1	0.08 2.42	21 81	0 1
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0

Figure 16. Diabetes Dataset (Learning, 2016)

The second dataset is the Heart Disease (HD) dataset known as processed.cleveland.data. Widely used in machine learning research, it contains 303 records with 14 attributes. This labeled dataset includes a binary class label, indicating the presence (1) or absence (0) of heart disease. Originally, it used values 1-4 to denote different levels of heart disease, but for simplicity, these were consolidated into a single binary value of 1 to indicate heart disease presence. This simplification enhances the model's ability to predict heart disease in a straightforward binary format, making it suitable for real-time predictive analysis. *(Ed-daoudy & Maalmi, 2019).* Figure 17 below, is a similar dataset taken from kaggle, on heart disease patients with 14 classes/attributes

# age age in years	# sex (1 = male; 0 = female)	# cp chest pain type	# trestbps resting blood pressure (in mm Hg on admission to the hospital)	# chol serum cholestoral in mg/dl	# fbs (fasting blood sugar >120 mg/dl) (1 = true; 0 = false)	# restecg resting electrocardiographic results	# thalach maximum heart rate achieved	# exang exercise induced angina (1 = yes; 0 = no)	# oldpeak ST depression induced by exercise relative to rest
29 77	0 1	0 3	94 200	126 564	0 1	0 2	71 202	0 1	0 6.2
52	1	0	125	212	0	1	168	0	1
53	1	0	140	203	1	0	155	1	3.1
70	1	0	145	174	0	1	125	1	2.6

Figure 17. Heart Disease Dataset (Lapp, 2019)

The authors utilize both datasets in their treatment within Spark's predictive analysis by importing the data from CSV into a RDD of Strings. Each element of the RDD is mapped and transformed into a Labeled Point for the model's training or testing purposes. Currently, streaming data processing and distributed machine learning are at the center stage, the above mentioned datasets also however serve well for health status forecasting and can be substituted with other health datasets conveniently if the need arises. *(Ed-daoudy & Maalmi, 2019).*

d) Critically assess the strengths and limitations of the implementation of the existing solution using the selected dataset. Provide suggestions for potential improvements.

Strengths of the Existing Solution

The solution's scalability and real-time capabilities are key strengths, as Apache Spark efficiently handles large datasets and processes streaming data. This enables up-to-date insights critical for monitoring patients with chronic conditions. Apache Zeppelin's interactive dashboard enhances visualization by auto-refreshing, which provides a live view of patient data that is easy for providers to interpret and act upon. *(Ed-daoudy & Maalmi, 2019)*

The solution's flexibility with multiple datasets allows for adaptability across health conditions, as shown by its application to both diabetes and heart disease datasets.

Limitations of the Existing Solution

Despite these advantages, several limitations reduce the solution's effectiveness. The small dataset size may limit model accuracy and generalizability for a broader patient population. Additionally, data privacy and security measures are minimal, raising concerns given the sensitivity of healthcare data. Furthermore, binary classification for heart disease oversimplifies diagnosis by removing details on disease severity, while mini-batch streaming in Spark Streaming may introduce delays for applications requiring instantaneous responses. *(Ed-daoudy & Maalmi, 2019)*

Another drawback is the limited use of advanced machine learning. Basic models in Spark MLlib may not capture the complex patterns in health data as effectively as deep learning models, reducing predictive performance. *(Ed-daoudy & Maalmi, 2019)*

Recommendations for Improvement

To improve the solution, several enhancements are recommended. Using larger, diverse datasets or generating synthetic data would increase model accuracy and reliability. Implementing data encryption and access controls would strengthen data privacy, while multi-class classification for heart disease could provide more detailed insights into disease severity. *(Ed-daoudy & Maalmi, 2019)*

Switching from Spark's mini-batch streaming to a real-time continuous streaming framework like Apache Flink would allow for more immediate data processing, essential for time-sensitive healthcare applications. Incorporating advanced machine learning and deep learning techniques with tools like TensorFlow or PyTorch could improve predictive performance, particularly for complex health data. Finally, enhancing dashboard customization and role-based access controls would allow providers to set personalized alert thresholds and focus on data most relevant to their roles. *(Ed-daoudy & Maalmi, 2019)*