Industrial Internet of Things

Chapter 1: Introduction to IIoT

Dr Abhishek Hazra

abhishek.h@iiits.in

Definition of IoT

"It's a global infrastructure for the information society, enabling advance services by interconnecting physical and virtual things based on existing and evolving interoperable information and communication technologies"

Characteristics of IoT

- Dynamic & Self-Adapting
- Self-Configuring
- Interoperable Communication Protocols
- Unique Identity
- Integrated into Information Network

Physical Design of IoT

- The "Things" in IoT usually refers to IoT devices which have unique identities and can perform remote sensing, actuating and monitoring capabilities.
- IoT devices can:
  - Exchange data with other connected devices and applications (directly or indirectly), or
  - Collect data from other devices and process the data locally or
  - Send the data to centralized servers or cloud-based application back-ends for processing the data, or
  - Perform some tasks locally and other tasks within the IoT infrastructure, based on temporal and space constraints

Case Study 1: Smart Home

Major Components:

- Connected radiator/heater
- Connected air conditioner
- Connected camera
- Connected door
- Connected window
- Connected microwave
- Connected refrigerator
- Connected water boiler

Illustration of smart home system.

Case Study 2: Smart Healthcare

Major Components:

- Connected medicine
- Body sensor network
- Healthcare robot
- Healthcare-specific smart home
- Home-hospital system
- User API
- Storage and processing
- Data Analytics

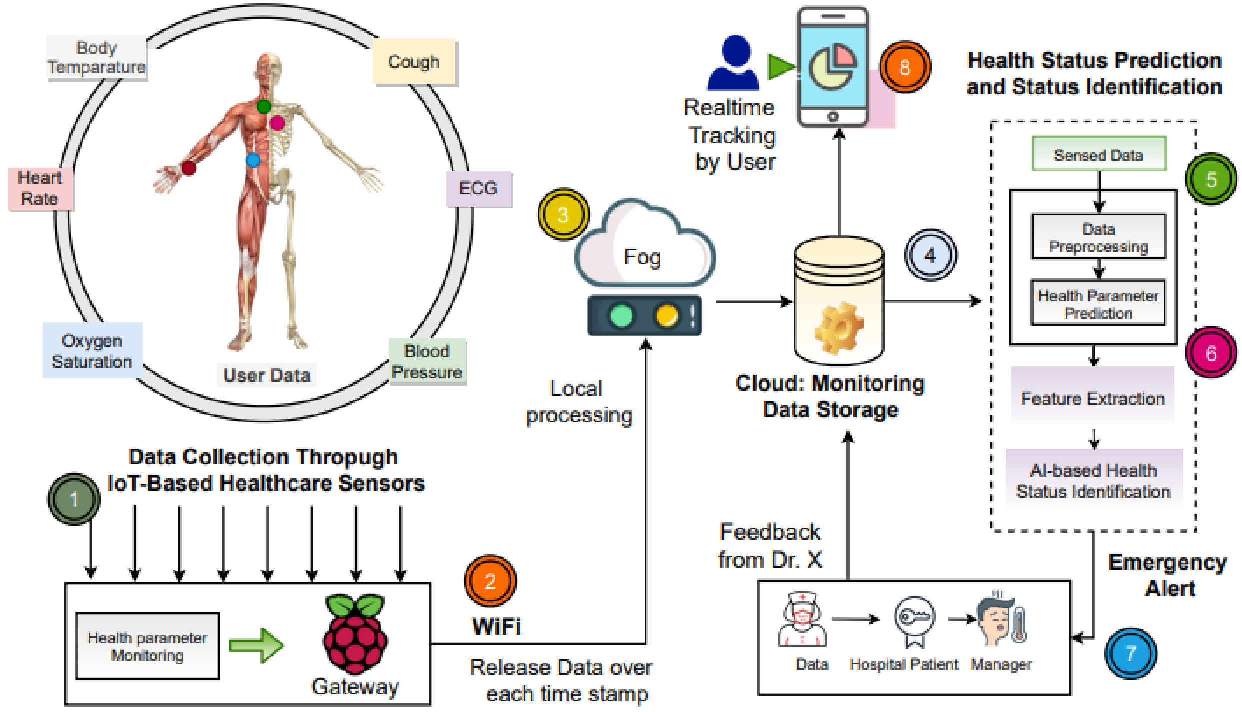


Illustration of smart healthcare system.

Case Study 3: Smart Agriculture

Major Components:

- Connected plants
- Connected animals
- Connected barn
- Connected agricultural equipment
- Connected packing and shipping
- Weather forecast
- Prediction (Artificial Intelligence)
- Cloud data analytics

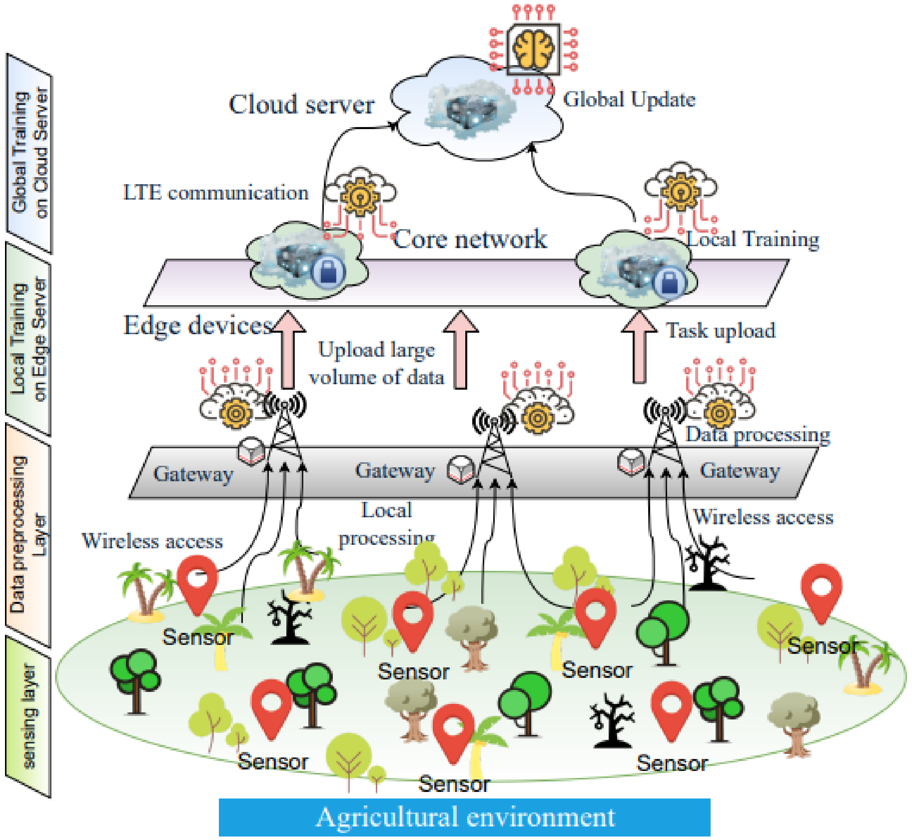


Illustration of smart agriculture system.

Case Study 4: Smart Transportation

Major Components:

- Smart connected vehicle
- Connected base station
- Connected roadside Unit
- Connected fog-cloud computation
- Connected traffic information
- Safety, security, and environment

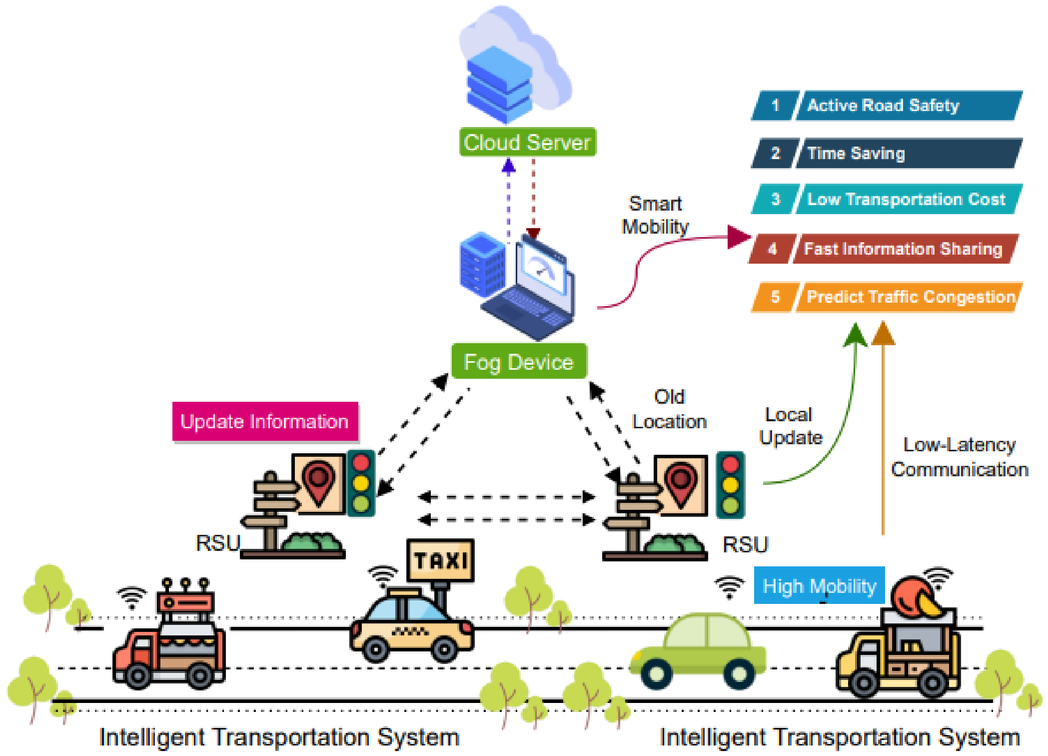
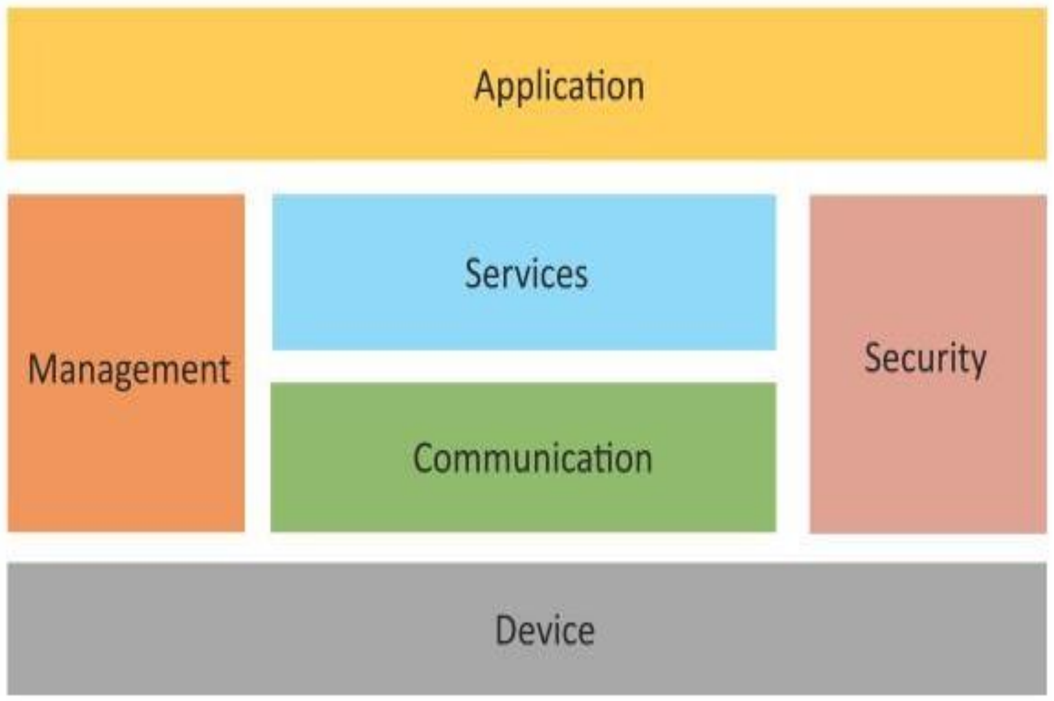


Illustration of smart transportation system.

Logical Design of IoT

- Logical design of an IoT system refers to an abstract representation of the entities and processes without going into the low-level specifics of the implementation.

- An IoT system comprises of a number of functional blocks that provide the system the capabilities for identification, sensing, actuation, communication, and management.



IoT enabling technologies

- Wireless connectivity
- Sensors and Actuators
- Cloud Computing
- Edge Computing
- Data Analytics and Machine Learning
- Blockchain

Wireless connectivity

- **Wi-Fi:** Commonly used for high-speed connections, suitable for applications in homes, offices, and urban areas.
- **Bluetooth:** Ideal for short-range connections between devices, often used for wearable devices and smart home applications.
- **Zigbee:** Designed for low-power, low-data-rate communication, suitable for applications requiring energy efficiency, like home automation.
- **LoRaWAN:** Provides long-range, low-power connectivity, making it suitable for applications in agriculture, smart cities, and Industrial IoT.
- **Cellular Networks (5G, LTE-M, NB-IoT):** Cellular technologies offer wide coverage and support for various IoT use cases, from connected vehicles to remote monitoring.

Sensors and Actuators

- **Temperature Sensors:** Measure ambient temperature and help control heating, cooling, and climate systems.
- **Motion Sensors:** Detect movement, enabling applications like security systems, smart lighting, and activity tracking.
- **Humidity Sensors:** Measure moisture levels in the air, important for HVAC systems, agriculture, and storage environments.
- **Proximity Sensors:** Detect the presence of objects or people, used in touchless interfaces, parking assistance, and industrial automation.
- **Actuators (e.g., Servos, Motors):** Convert electrical signals into mechanical motion, enabling devices to perform physical actions like opening doors, moving robotic arms, etc.

Cloud Computing

- **Scalability:** Cloud platforms can scale resources up or down according to demand, ensuring optimal performance and cost-efficiency.
- **Storage:** Cloud storage provides a centralized and accessible repository for storing and managing IoT-generated data.
- **Data Processing:** Cloud platforms offer powerful processing capabilities to analyze and extract insights from massive datasets.
- **Remote Access:** IoT devices can be monitored and managed remotely through cloud-based dashboards and applications.
- **Integration:** Cloud services enable seamless integration of data and functionality across diverse IoT devices and applications.

Edge Computing

- **Low Latency:** Edge computing reduces data processing time by processing data closer to where it's generated, improving real-time responsiveness.
- **Bandwidth Optimization:** Only relevant data is sent to the cloud, reducing the amount of data transferred and conserving network bandwidth.
- **Privacy:** Processing data locally reduces the need to send sensitive information to the cloud, enhancing privacy and security.
- **Offline Operation:** Edge devices can continue functioning even when disconnected from the cloud, ensuring uninterrupted operations.
- **Real-time Analytics:** Immediate data analysis at the edge allows for quick decision-making without waiting for data to travel to the cloud.

Data Analytics and Machine Learning

- **Predictive Maintenance:** ML algorithms can predict equipment failures before they happen, minimizing downtime and maintenance costs.
- **Anomaly Detection:** ML models identify unusual patterns in data, helping to detect security breaches, fraud, and abnormal behaviors.
- **Pattern Recognition:** ML can recognize complex patterns in data, assisting in image and speech recognition, as well as predictive analysis.
- **Personalization:** ML algorithms can analyze user behavior to provide personalized recommendations and experiences.
- **Optimization:** ML can optimize processes by analyzing data to identify inefficiencies and suggest improvements, such as supply chain optimization.

Blockchain

- **Decentralization:** Blockchain operates on a decentralized network, enhancing security and reducing the risk of single points of failure.
- **Immutability:** Once data is added to a blockchain, it cannot be altered or deleted, ensuring data integrity and auditability.
- **Security:** Transactions on a blockchain are encrypted and validated by a consensus mechanism, making it highly secure against unauthorized access and tampering.
- **Transparency:** All participants in a blockchain network have access to the same data, fostering transparency and trust.
- **Smart Contracts:** Blockchain supports self-executing smart contracts, which automatically execute predefined actions when specific conditions are met.

Predecessors of IIoT

The Industrial Internet of Things (IIoT) has evolved from earlier technologies and concepts in the industrial and manufacturing sectors. Here are some key predecessors that laid the groundwork for the development of IIoT:

- SCADA (Supervisory Control and Data Acquisition)
- PLC (Programmable Logic Controller)
- M2M (Machine-to-Machine) Communication
- Embedded Systems
- Enterprise Resource Planning (ERP) Systems
- Wireless Sensor Networks (WSN)

Thank You!

Industrial Internet of Things

Chapter 1: Introduction to IIoT

Dr Abhishek Hazra

abhishek.h@iiits.in

Industrial internet of things

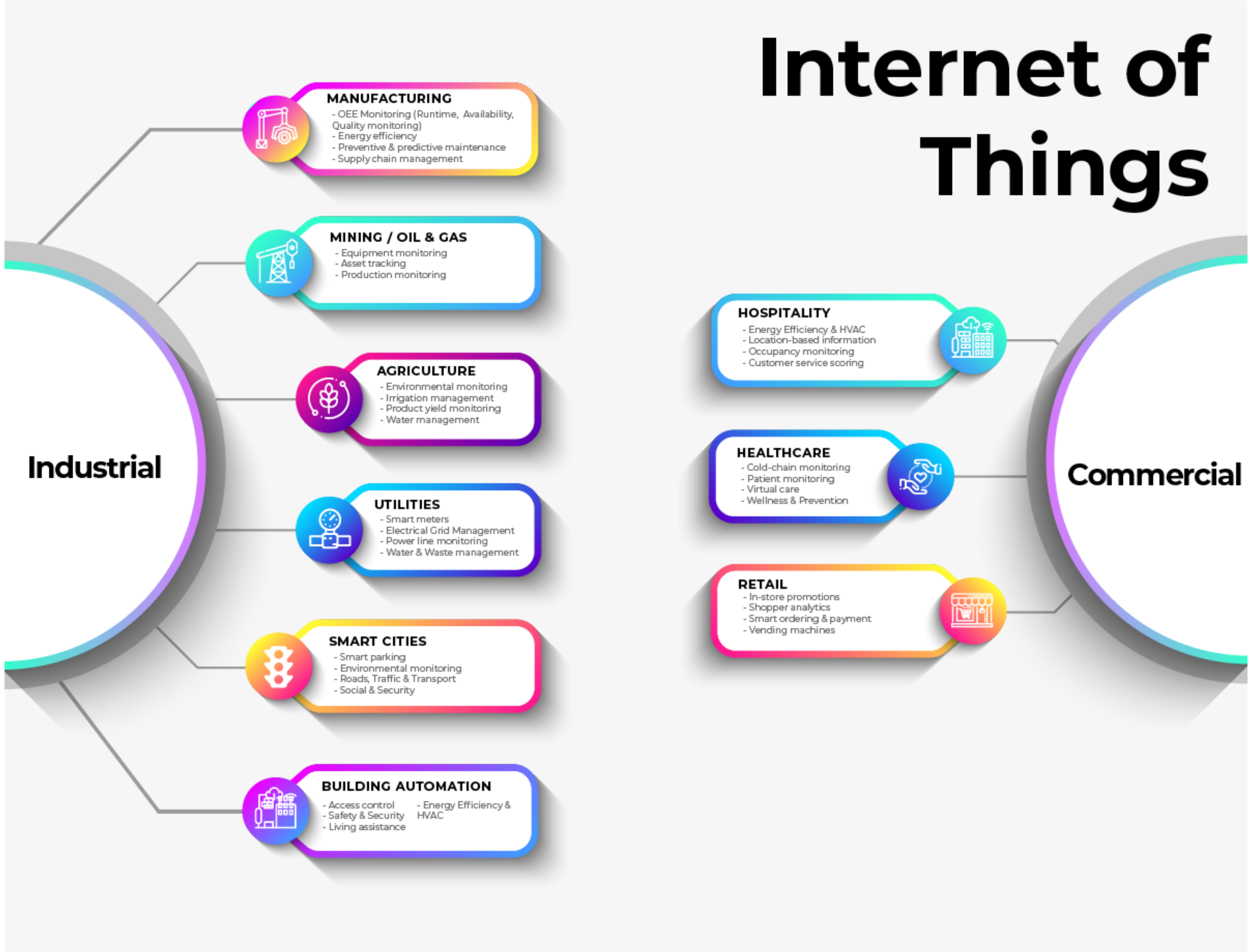
The **Industrial Internet of Things** (IIoT) refers to interconnected sensors, instruments, and other devices networked together with computers' industrial applications, including manufacturing and energy management.

- This connectivity allows for data collection, exchange, and analysis, potentially facilitating improvements in productivity and efficiency as well as other economic benefits

- The IIoT is enabled by technologies such as cybersecurity, cloud computing, edge computing, mobile technologies, machine-to-machine, 3D printing, advanced robotics, big data, Internet of Things, RFID technology, and cognitive computing

Industrial internet of things

Industrial internet of things



Predecessors of IIoT

The Industrial Internet of Things (IIoT) has evolved from earlier technologies and concepts in the industrial and manufacturing sectors. Here are some key predecessors that laid the groundwork for the development of IIoT:

- SCADA (Supervisory Control and Data Acquisition)
- PLC (Programmable Logic Controller)
- M2M (Machine-to-Machine) Communication
- Embedded Systems
- Enterprise Resource Planning (ERP) Systems
- Wireless Sensor Networks (WSN)

Predecessors of IIoT

SCADA (Supervisory Control and Data Acquisition):

- Real-time Monitoring and Control
- Data Acquisition
- Critical Infrastructure Management

PLC (Programmable Logic Controller):

- Industrial Automation
- Reliability and Robustness
- Programmability

M2M (Machine-to-Machine) Communication:

- Efficiency Improvement
- Data Exchange
- Remote Monitoring and Control

Predecessors of IIoT

Embedded Systems:

- Dedicated Functionality
- Power Efficiency
- Integration with Sensors and Actuators

Enterprise Resource Planning (ERP) Systems:

- Holistic Business Integration
- Data Centralization
- Streamlined Workflows

Wireless Sensor Networks (WSN):

- Remote Sensing
- Cost-Effective Monitoring
- Scalability

Emergence of IIoT

The emergence of the Industrial Internet of Things (IIoT) represents a significant evolution in the industrial and manufacturing sectors. Several factors have contributed to the rise of IIoT, transforming traditional industries and shaping the way businesses operate. Here are key factors and drivers behind the emergence of IIoT:

1. **Advancements in Connectivity:** The availability and affordability of advanced connectivity technologies, such as high-speed internet, wireless networks, and communication protocols, have enabled seamless data exchange between devices and systems in industrial environments.

Emergence of IIoT

2. Sensor Technology: The development of sensors with improved accuracy, reduced costs, and enhanced capabilities has been crucial in collecting real-time data from various industrial assets. These sensors enable the monitoring of physical parameters, equipment conditions, and environmental factors.

3. Data Analytics and Cloud Computing: The growth of data analytics and cloud computing technologies has provided the infrastructure needed to process and analyze vast amounts of data generated by industrial devices. Cloud platforms offer scalable and cost-effective solutions for storing and processing data from distributed industrial systems.

Emergence of IIoT

4. Interoperability Standards: The establishment of interoperability standards, such as MQTT (Message Queuing Telemetry Transport) and OPC UA (Open Platform Communications Unified Architecture), has facilitated the integration of diverse industrial devices and systems. Standardization promotes compatibility and interoperability across different vendors and platforms.

5. Cost Reduction in Technology: The decreasing costs of hardware components, including sensors, actuators, and embedded systems, have made it more feasible for industries to implement IIoT solutions. This cost reduction has lowered barriers to entry and encouraged broader adoption.

Emergence of IIoT

6. Big Data and Analytics: IIoT leverages big data analytics to derive valuable insights from the massive volumes of data generated by industrial processes. Analyzing this data allows businesses to optimize operations, predict maintenance needs, and make data-driven decisions.

7. Increased Computing Power at Edge Devices: Edge computing capabilities have improved, allowing processing and analysis of data closer to the source (at the edge of the network). This reduces latency, enhances real-time decision-making, and minimizes the need for sending all data to centralized servers.

Predecessors of IIoT

8. Demand for Operational Efficiency: Industries are constantly seeking ways to enhance operational efficiency, reduce downtime, and improve overall productivity. IIoT solutions offer real-time visibility into operations, enabling proactive decision-making and optimization of processes.

9. Security Measures: As the number of connected devices increases, so does the importance of cybersecurity in industrial settings. The development of robust security measures and protocols has been crucial in ensuring the protection of sensitive data and critical infrastructure.

10. Industry 4.0 Initiative: The concept of Industry 4.0, which emphasizes the integration of digital technologies into industrial processes, has played a pivotal role in driving the adoption of IIoT. Industry 4.0 envisions the "smart factory" where machines, and humans collaborate more intelligently.

Thank You!

Industrial Internet of Things

Chapter 1: Introduction to IIoT

Dr Abhishek Hazra

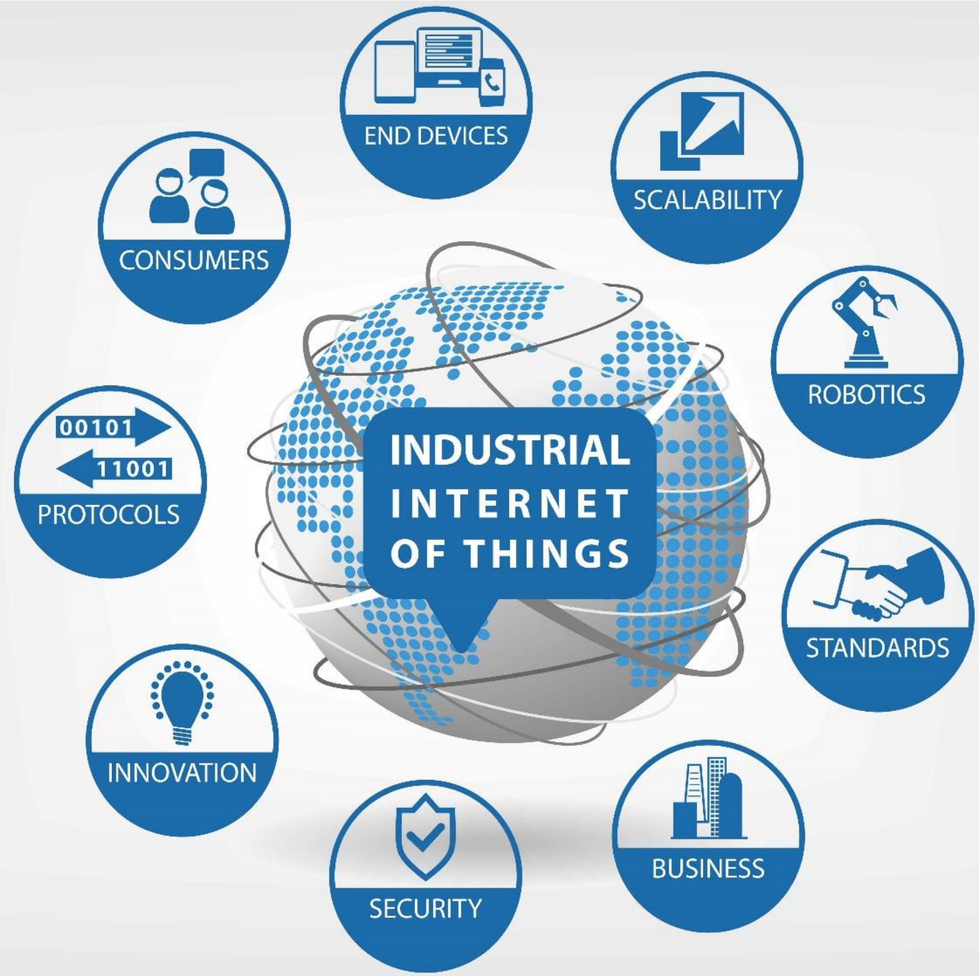
abhishek.h@iiits.in

Importance of IIoT

- Real-time monitoring and control of industrial processes, optimizing efficiency and reducing downtime.
- Predictive maintenance, minimizing equipment failures and improving overall reliability.
- Informed decision-making to enhance productivity and quality.
- Enhances supply chain visibility, optimizing logistics, reducing costs, and improving overall supply chain efficiency.
- Smart energy monitoring and management in IIoT lead to reduced energy consumption and more sustainable industrial practices.
- Continuous monitoring throughout the manufacturing process ensures high-quality production and early detection of deviations from standards.

Importance of IIoT

- IIoT enables remote monitoring of hazardous environments, improving workplace safety by minimizing human presence in high-risk areas.
- Precise tracking of assets and inventory minimizes losses, optimizes asset utilization, and streamlines inventory control.
- IIoT facilitates human-machine collaboration, integrating collaborative robots to enhance overall operational efficiency.

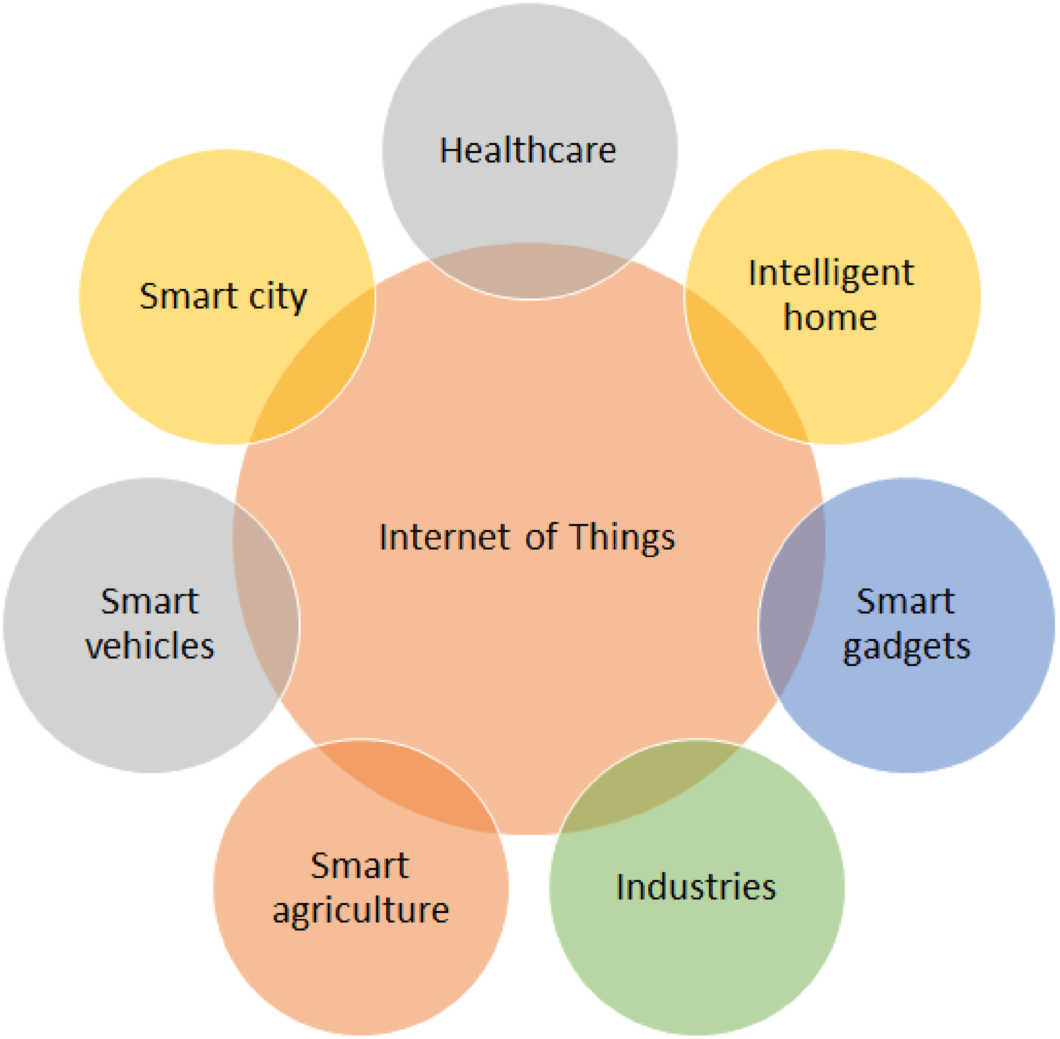


IoT vs IIoT

Perspective	Internet of Things	Industrial Internet of Things
Connected things	consumer-level devices, usually less expensive	critical machines, sensors, systems, usually with a high degree of complexity
Service model	human-centric	machine-centric
Applications	consumer-oriented applications	industry-oriented applications
Communication infrastructure	essentially wireless	wireless and wired
Communication capabilities	a small number of communication standards	a high number of connectivity standards and technologies
Amount of data	medium to high	high to very high
Criticality	not stringent	mission critical (timing, reliability, security, privacy)
Real-time requirement	usually no, dealing with less time-sensitive systems	most often has a key role

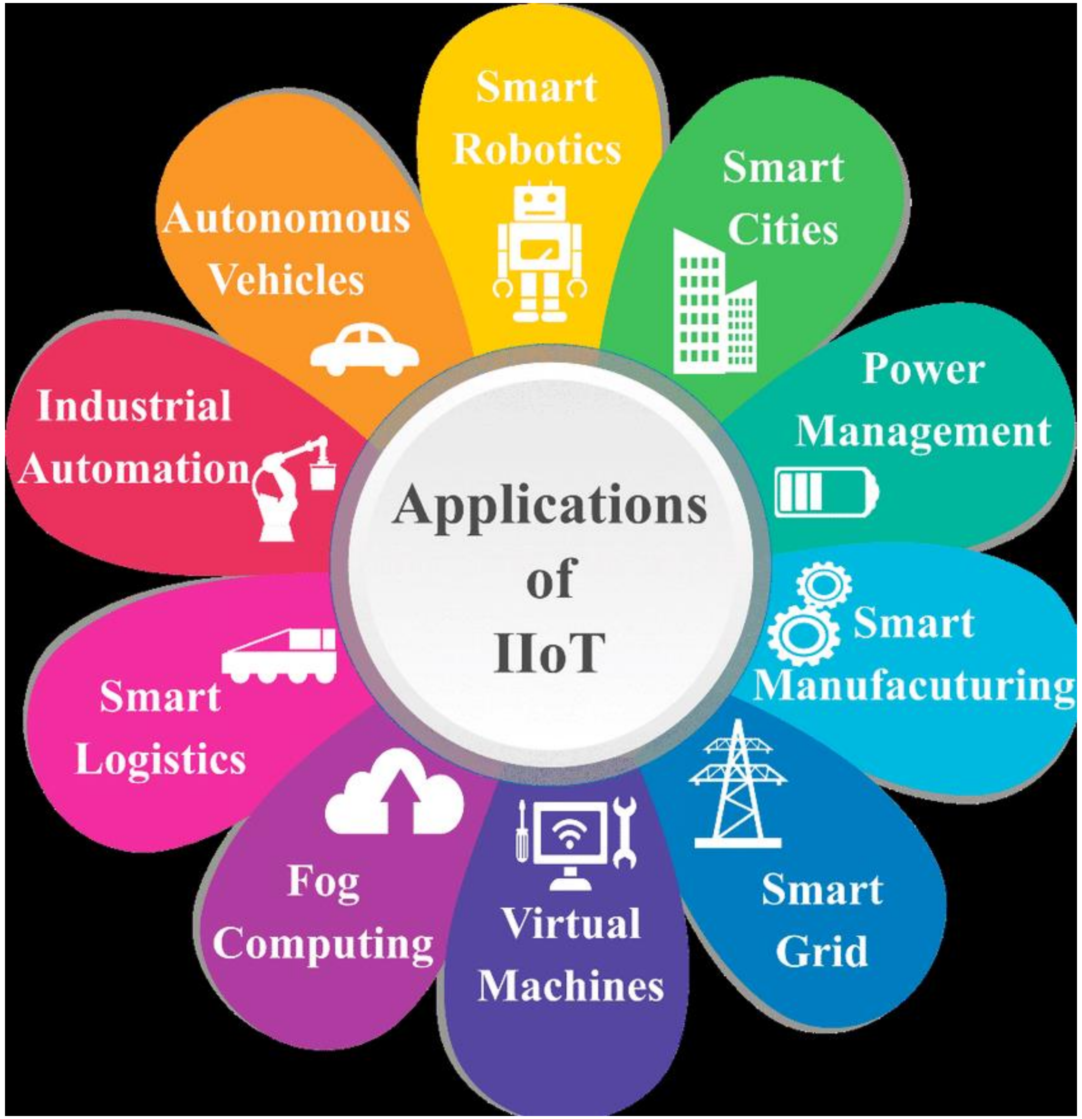
Industrial safety signs

IoT Applications



IIoT Applications

- Remote Diagnostics in Healthcare
- Smart Grids and Energy Management
- Condition-Based Monitoring
- Quality Control and Management
- Supply Chain Visibility
- Industrial Transportation Systems

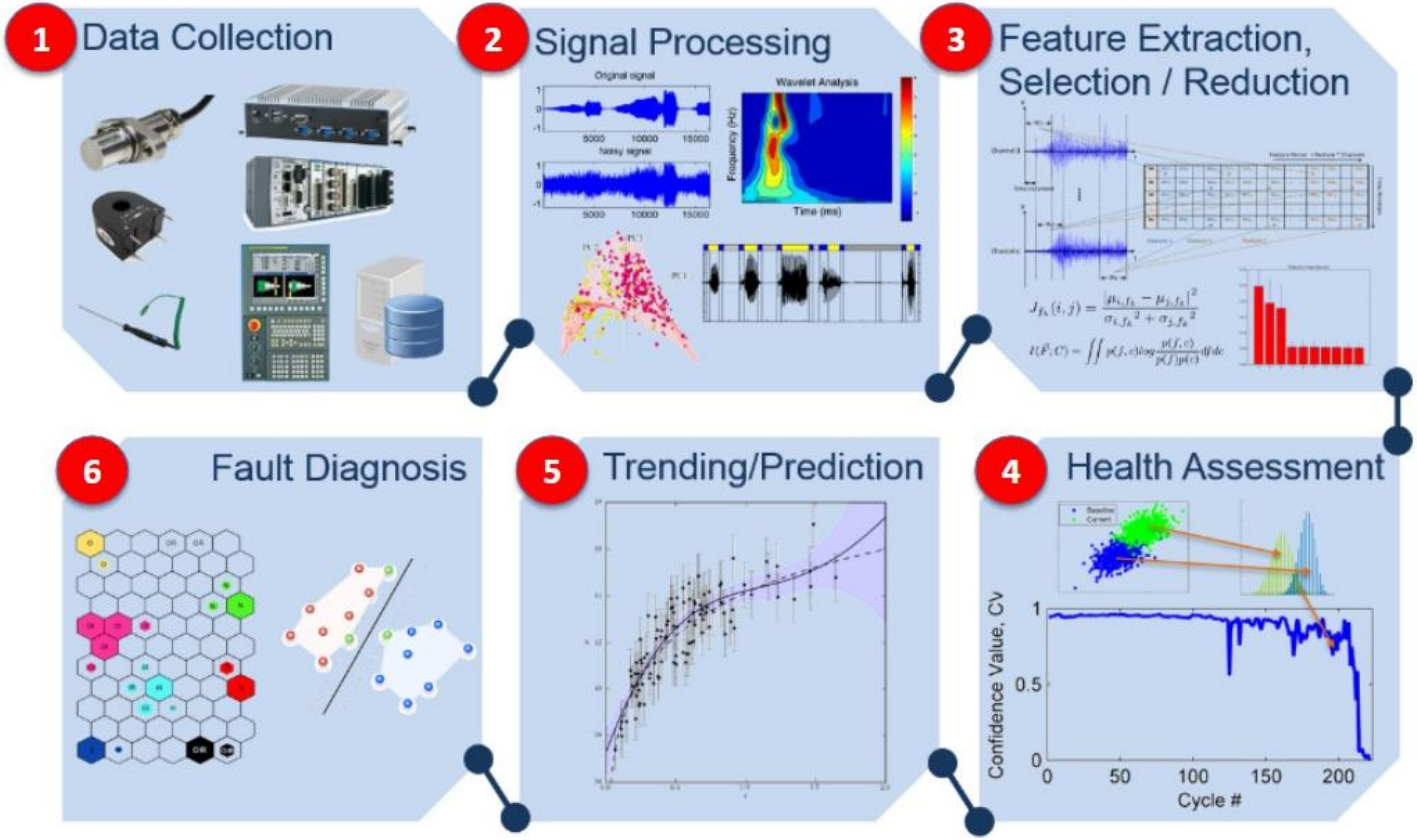


IIoT Applications

Asset Tracking and Management

IIoT Applications

Predictive Maintenance:



Predictive Analysis and Downtime Estimation

- Increasing applicability of IIoT applications.
- Challenges include product lifetime, predictive maintenance, limited resources and heavy data traffic in industrial devices.
- Adoption of Industrial Edge Computing (IEC) for local processing of latency-critical tasks at the edge.
- Design of an offloading and ML-based predictive maintenance framework.

Scope and Motivation

- Importance of predictive maintenance for strategic planning and cost optimization.
- **Challenges in predictive maintenance**: data quality, integration, handling substantial data, and initial financial implication.
- **Motivation**: Challenges in heavy machinery usage, need for IEC framework, and the role of ML-based data analytics.

Predictive Maintenance System

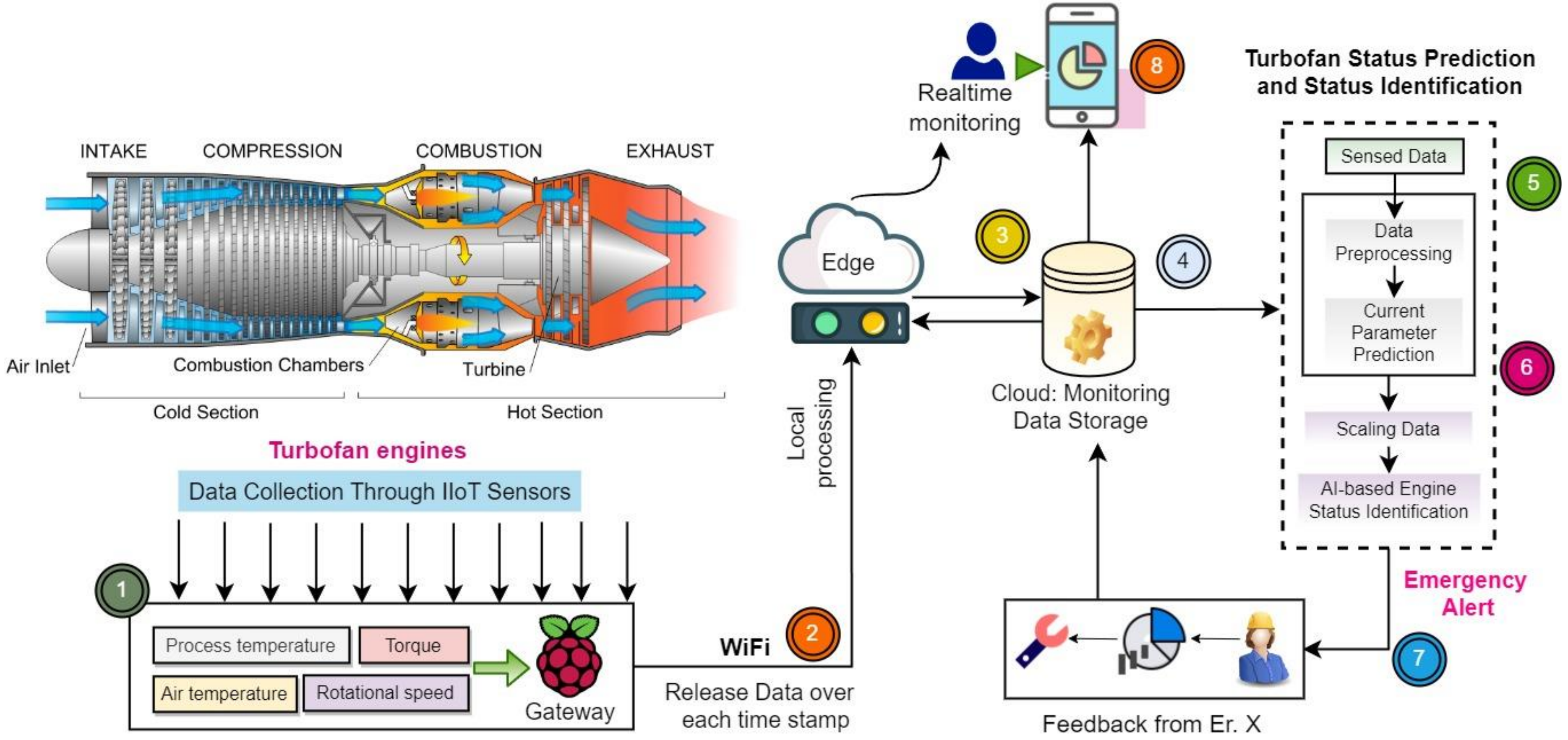


Fig. : Illustration of industrial edge networks for predictive maintenance.

Contribution

Design of a predictive maintenance framework using ML-enabled data analytics with IEC resources.

Objective: Estimate subsequent downtime and schedule maintenance for complex industrial machines.

Key contributions: Optimization function, task offloading decision-making, and comprehensive performance study.

Practical dataset used: "*NASA Turbofan Jet Engine Dataset*" from 3 different types of industrial machines.

Proposed Methodology

Our proposed predictive maintenance framework involves several phases: *task classification*, *queue allocation*, and *ML-based downtime estimation*.

Step 1: Task Classification

We use a heuristic strategy to categorize tasks into *delay-sensitive* and *computation-intensive* based on factors like input size and execution deadline.

Step 2: Task Offloading and Queue Allocation

IEC servers manage two queues for different tasks.

Tasks are either *locally executed* or *offloaded* to the cloud based on computational capacity.

Task Offloading and Queue Allocation

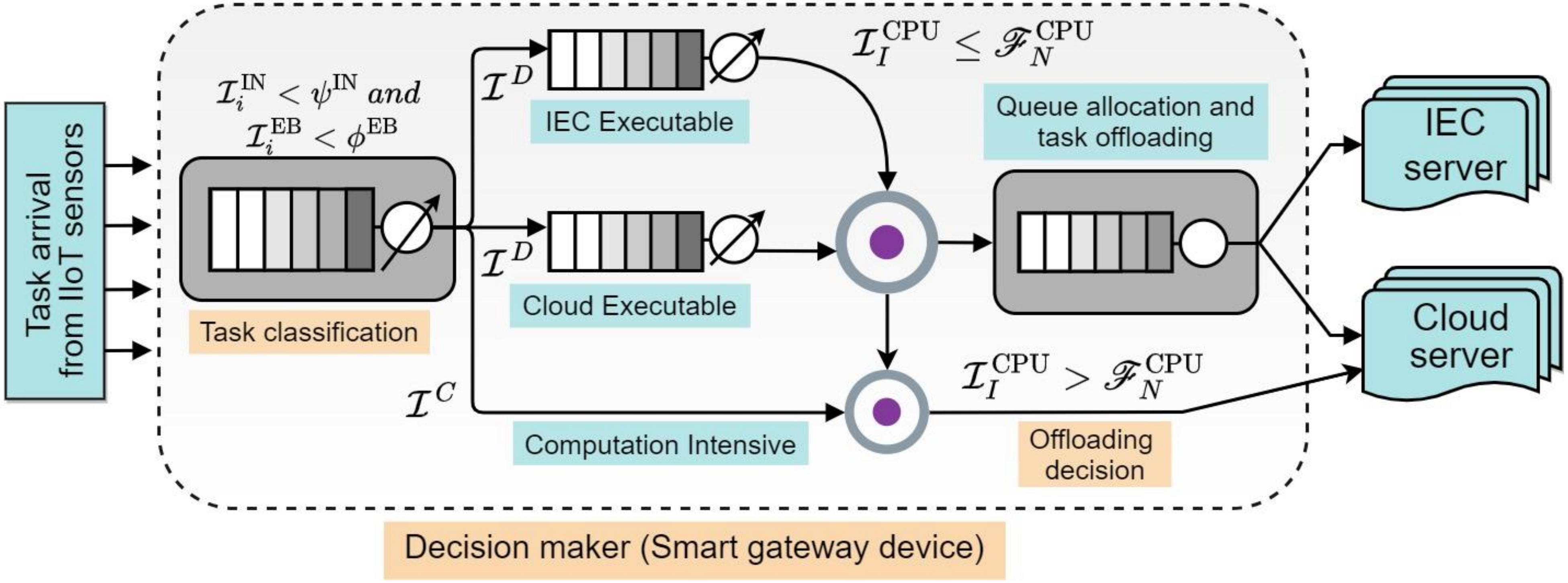


Fig: Illustration of task offloading and queue allocation strategy.

Proposed Methodology

Step 3: ML-Based Downtime Estimation

ML techniques (*Random Forest, Bagging, Boosting, Ensemble Learning*) accurately estimate downtime.

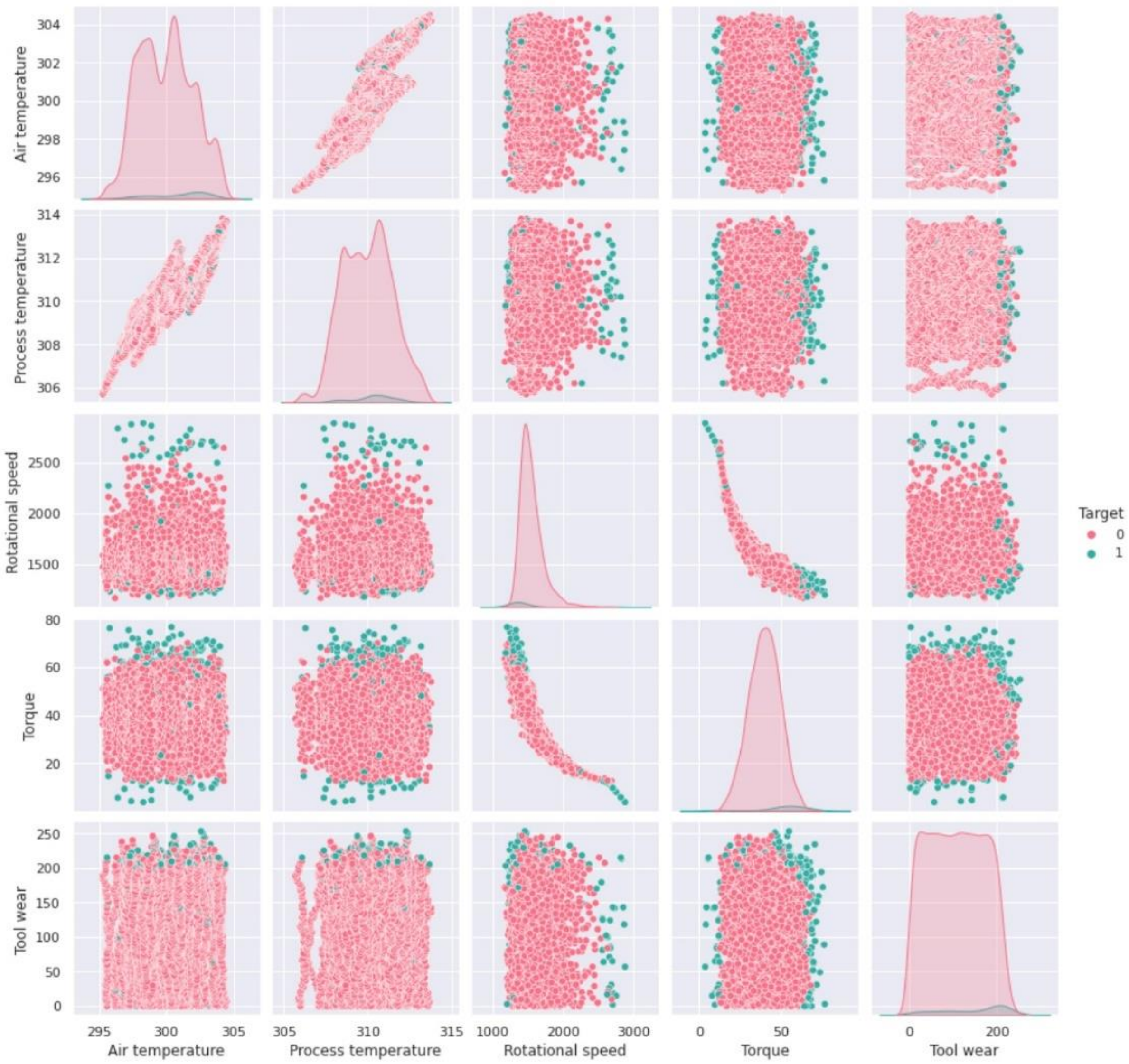
Supervised ML involves training models on labeled datasets for better decision-making.

Simulation Setup

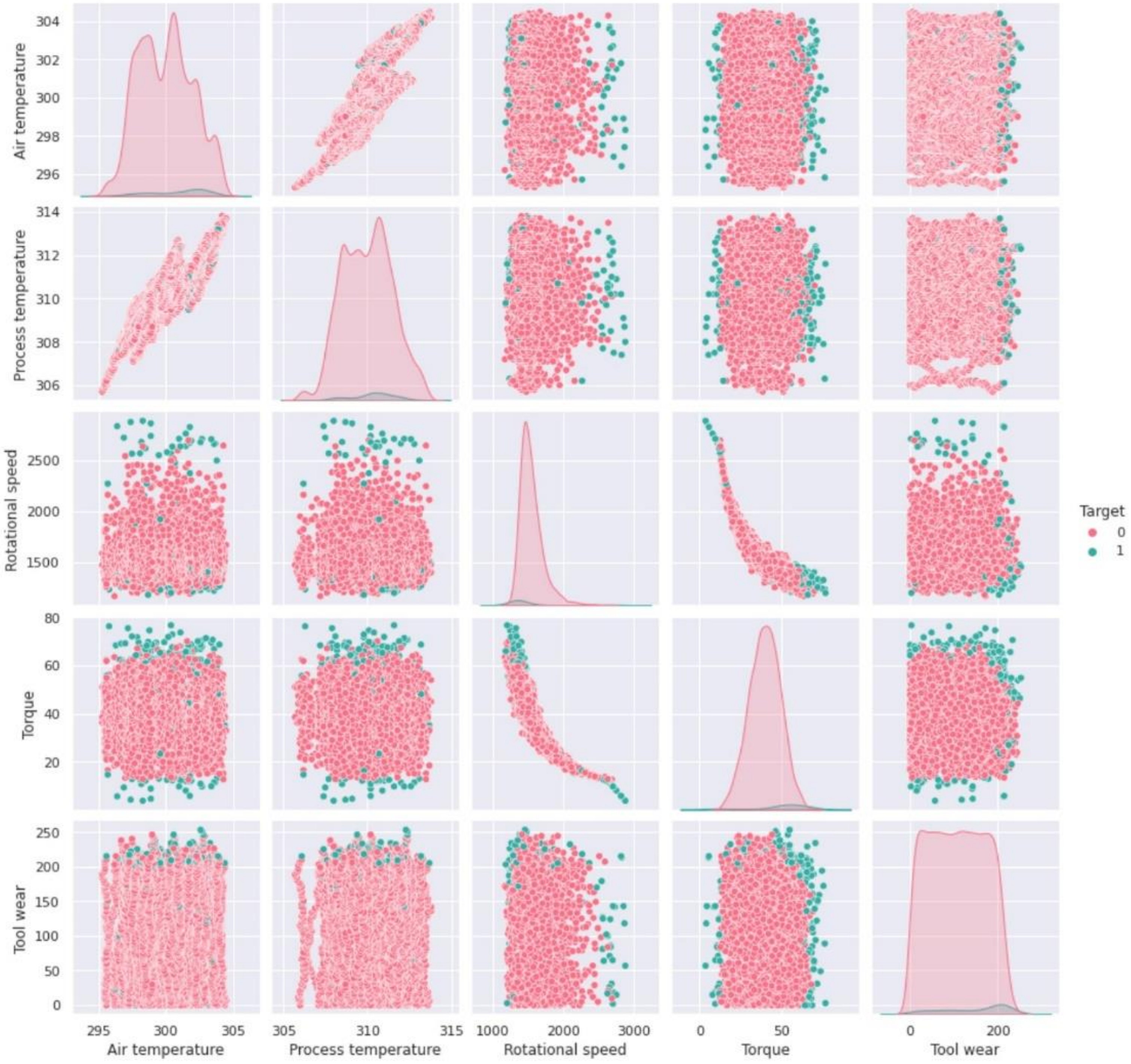
Number of tasks: 50, 100, 150, 200, 250, 300.
Number of Industrial Machines: 20.
Number of IEC Servers: 10.
Number of Cloud Servers: 2.
Task Offloading Decision Threshold: 0.5.
Available bandwidth between devices: 20 MBps.
Power consumption for computation at IEC server: 0.5 mW.
Power consumption for computation at Cloud server: 10 mW.
“NASA Turbofan Jet Engine Dataset” is used for estimating the lifetime and next maintenance of industrial machines.
Google Colab environment utilized for the entire experiment.

Dataset

- **NASA Turbofan Jet Engine Dataset**: 10,000 entries from 3 industrial machine types.
- **Engine Size Distribution**: 60% large, 30% medium, 10% small engines.
- **Class Imbalance**: 96.7% no failure, 3.3% failure labels.
- **Failure Reasons**: 1.1% power, 1.0% toolware, 0.8% overstrain, 0.5% high heat dissipation.
- **Dataset Features**: 6 key parameters (Air temperature, Process temperature, Rotational speed, Torque, Tool wear, Target), 2 output levels (failure, no failure), facilitating comprehensive machine behavior evaluation



Dataset


Air temperature
Process temperature
Rotational speed
Torque
Tool wear
Target
0
1
304
302
300
298
296
314
312
310
308
306
2500
2000
1500
80
60
40
20
250
200
150
100
50
0
295
300
305
305
310
315
1000
2000
3000
0
50
0
200

Performance Analysis

Execution Delay Analysis:
Task completion rate impacts execution delay on different devices.
IEC servers regulate end-to-end execution delay within a limit.
Proposed offloading strategy reduces overall execution delay with task volume.

Energy Consumption Trends:
IIoT sensors attempt execution on devices but offload due to limitations.
Delay-sensitive tasks directed to IEC servers, reducing computation overhead.
Computation-intensive tasks assigned to cloud servers, increasing energy consumption.

Computation Cost Evaluation:
Total computation cost comparison with existing strategies (Wang's, Mao's, Din's).
Significant improvement in performance with lower overall computation cost.
Consideration of end-to-end costs and execution costs on various computing devices

Predictive Analysis and Downtime Estimation

- Aim to enhance machine longevity and reduce downtime in industrial operations.
- Usage of balanced ML models: andom forests, bagging, boosting, ensemble learning.
- Training the model involved 1000 iterations for balanced prediction.
- Dataset divided into 80% training and 20% test samples for model evaluation.
- Accuracy Results: Random Forests (97.8%), Bagging (97.33%), Boosting (86.7%), Ensemble Learning (96.15%).
- Standard scaling and sampling techniques employed to enhance estimation accuracy.
- Bagging technique shows higher performance improvement in cross-validation.
- Despite overall accuracy enhancement, average prediction accuracy increased from 83% to 87%.

Thank You!

Industrial Internet of Things

Sensor and Actuator

Dr Abhishek Hazra

abhishek.h@iiits.in

Industrial Networking and Network Security

Importance:

- Enhances connectivity, efficiency, and automation in industrial processes.
- Facilitates real-time monitoring and control of industrial systems.

Challenges:

- Security vulnerabilities due to increased connectivity.
- Compatibility issues among diverse industrial devices.
- Ensuring network reliability in harsh industrial environments.

Types of Industrial Networking

Industrial Ethernet:

- Utilizes Ethernet protocols for communication in industrial environments.
- Provides higher data transfer rates and reliability compared to traditional fieldbuses.

Fieldbuses:

- Common in industrial automation, connecting sensors and actuators.
- Examples include Profibus, Modbus, and CANopen.

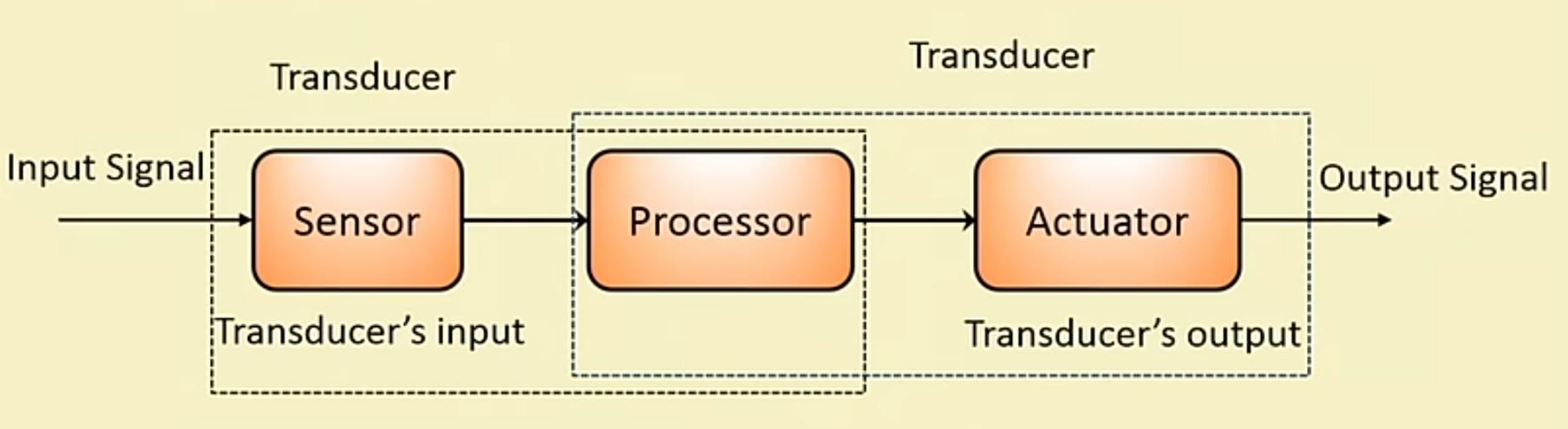
Wireless Industrial Networks:

- Utilizes wireless communication technologies like Wi-Fi or Bluetooth.
- Enables flexible and mobile industrial setups.

Industrial IoT (IIoT):

- Integrates IoT devices for enhanced data collection and analysis.
- Facilitates real-time monitoring and control.

Transducer


Transducer
Transducer
Input Signal
Sensor
Processor
Actuator
Output Signal
Transducer's input
Transducer's output

Transducer

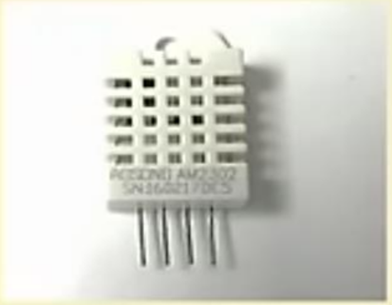
- Transducer:
  - Converts a signal from one physical form to another physical form
  - Physical form: thermal, electric, mechanical, magnetic, chemical, and optical
  - Energy converter
  - Example:
    - Microphone : Converts sound to electrical signal
    - Speaker : Converts electrical signal to sound
    - Antenna : Converts electromagnetic energy into electricity and vice versa

Sensor

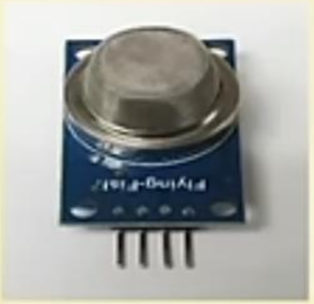
- The characteristic of any device or material to detect the presence of a particular <u>physical quantity</u>
- The output of sensor is signal, which is converted to human readable form

- Performs some function of input by sensing or feeling the physical changes in the characteristic of a system in response to <u>stimuli</u>
- Input: Physical parameter or stimuli
  - Example: Temperature, light, gas, pressure, and sound
- Output: Response to stimuli

Sensor



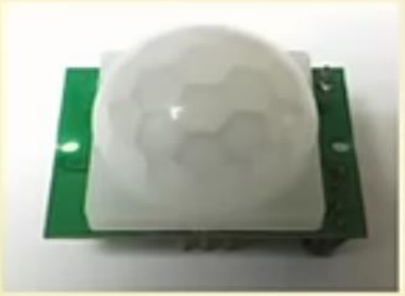
Temperature and Humidity sensor – DH22



Gas (LPG, CH4, and CO) detector sensor - MQ-5

Ultrasonic sensor - HC-SR04

CMOS Camera



PIR sensor

Rain detector sensor

Fire detector sensor

Characteristics of sensors

- Static characteristics
  - After steady state condition, how the output of a sensor change in response to an input change
- Dynamic characteristics
  - The properties of the system's transient response to an input

Static characteristics

- Accuracy
  - Represents the correctness of the output compared to a superior system
  - The different between the standard and the measured value
- Range
  - Gives the highest and the lowest value of the physical quantity within which the sensor can actually sense
  - Beyond this value there is no sensing or no kind of response

Static characteristics

- Resolution
  - Provides the smallest change in the input that a sensor is capable of sensing
  - Resolution is an important specification towards selection of sensors.
  - Higher the resolution better the precision
- Errors
  - The difference between the standard value and the value produced by sensor

Static characteristics

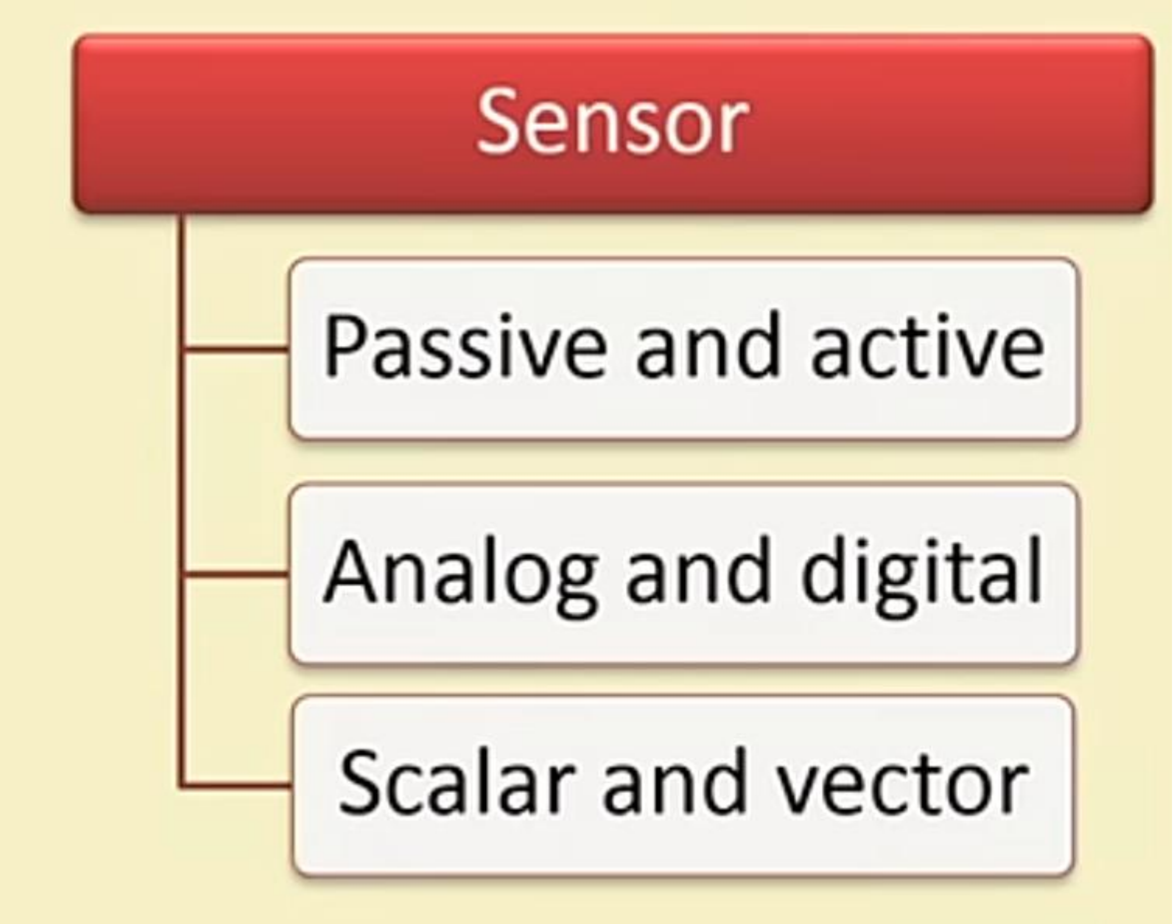
- Sensitivity
  - Sensitivity indicates ratio of incremental change in the response of the system with respect to incremental change in input parameter.
  - It can be found from slope of output characteristic curve of a sensor
- Linearity
  - The deviation of sensor value curve from a particular straight line
- Drift
  - The difference in the measurements of sensor from a specific reading when kept at that value for a long period of time
- Repeatability
  - The deviation between measurements in a sequence under same conditions

Dynamic Characteristics

How well a sensor responds to changes in its input

- Zero order system
  - Output shows a response to the input signal with no delay
  - Does not include energy-storing elements
  - Example: Potentiometer measures linear and rotary displacements
- First order system
  - When the output approaches its final value gradually
  - Consists of an energy storage and dissipation element
- Second order system
  - Complex output response
  - The output response of sensor oscillates before steady state

Sensor Classification



Passive Sensor

- Cannot independently sense the input
- Example: Accelerometer, soil moisture, water-level, and temperature sensors

Active Sensor

- Independently sense the input
- Example: Radar, sounder, and laser altimeter sensors

Analog Sensor

- The response or output of the sensor is some continuous function of its input parameter
  - Example: Temperature sensor, LDR, analog pressure sensor, and Analog Hall effect/Magnetic Sensor
    - A LDR shows continuous variation in its resistance as a function of intensity of light falling on it

Digital Sensor

- Responses in <u>binary</u> nature
- Designs to overcome the disadvantages of analog sensors
- Along with the analog sensor it also comprises of extra electronics for bit conversion
- Example: Passive infrared (PIR) sensor and digital temperature sensor (DS1620)

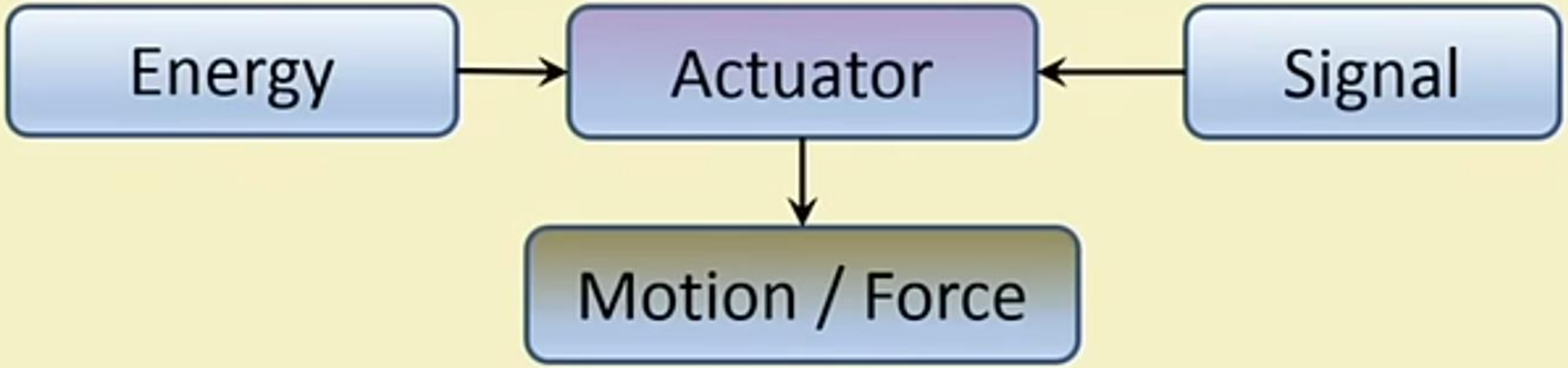
Scalar Sensor

- Detects the input parameter only based on its <u>magnitude</u>
- The response of the sensor is a function of magnitude of the input parameter
- Not affected by the direction of the input parameter
- Example: Temperature, gas, strain, color, and smoke sensors

Vector Sensor

- The response of the sensor depends on the magnitude of the direction and orientation of input parameter
- Example : Accelerometer, gyroscope, magnetic field, and motion detector sensors

Actuator



- An actuator is part of the system that deals with the control action required (mechanical action)
- Mechanical or electro-mechanical devices

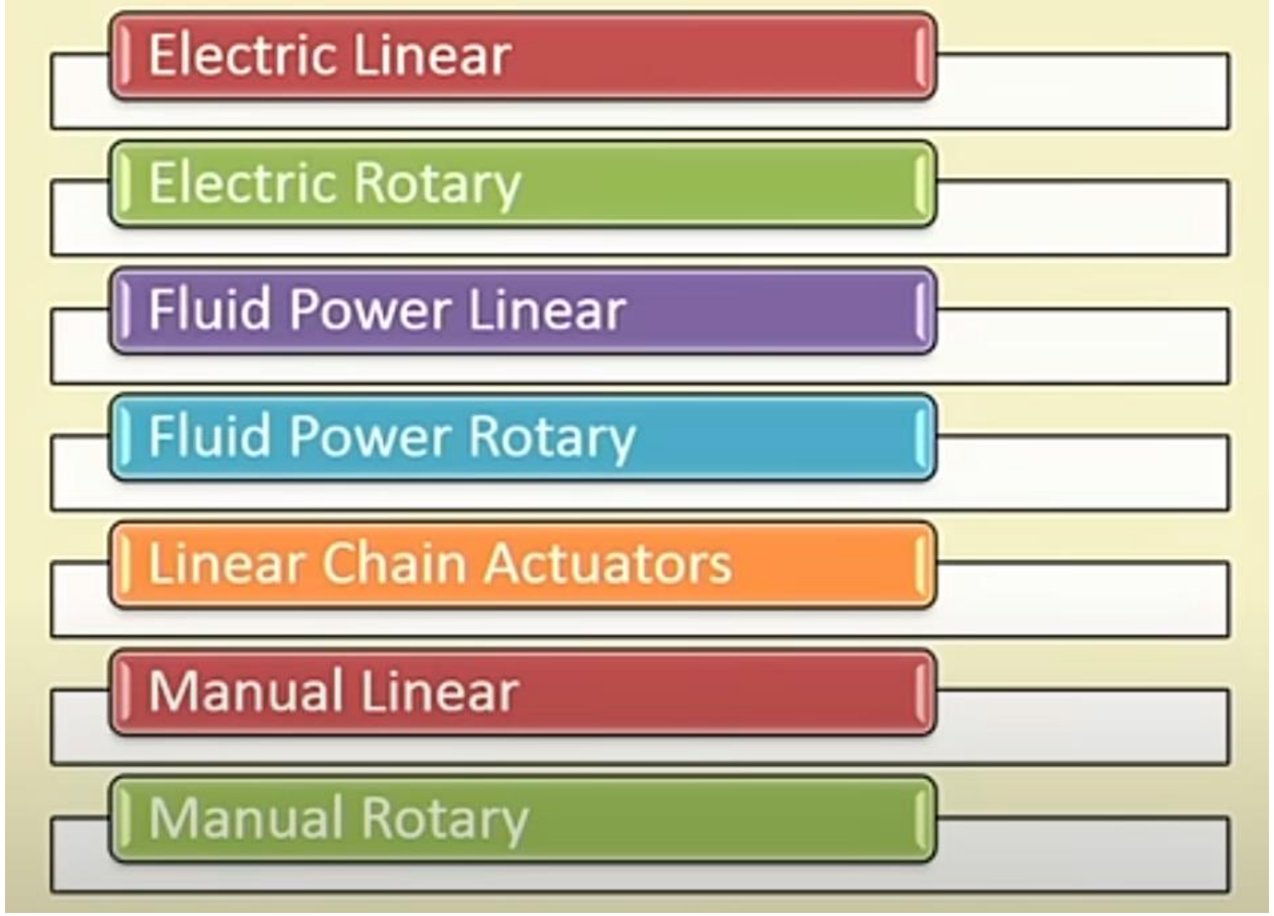
Actuator

- A control signal is input to an actuator and an energy source is necessary for its operation
- Available in both micro and macro scales
- Example: Electric motor, solenoid, hard drive stepper motor, comb drive, hydraulic cylinder, piezoelectric actuator, and pneumatic actuator

DC Motor

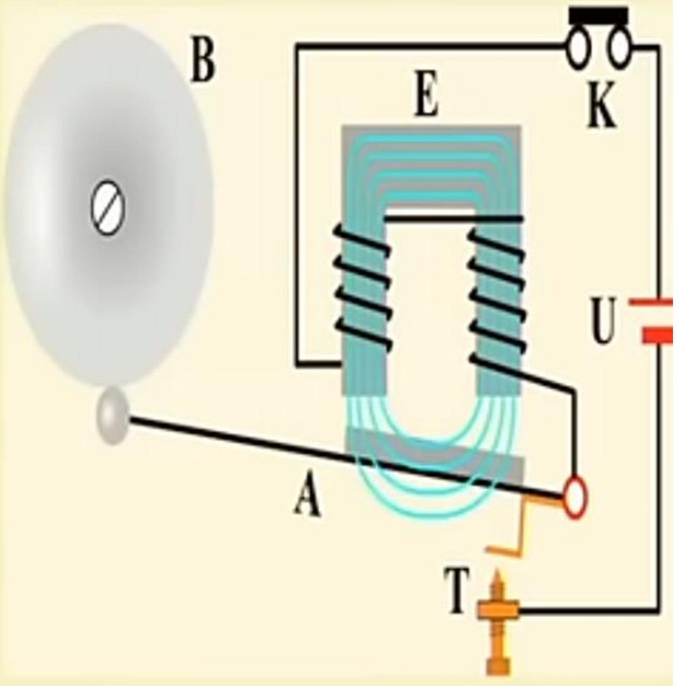
Relay

Classifications of Actuators



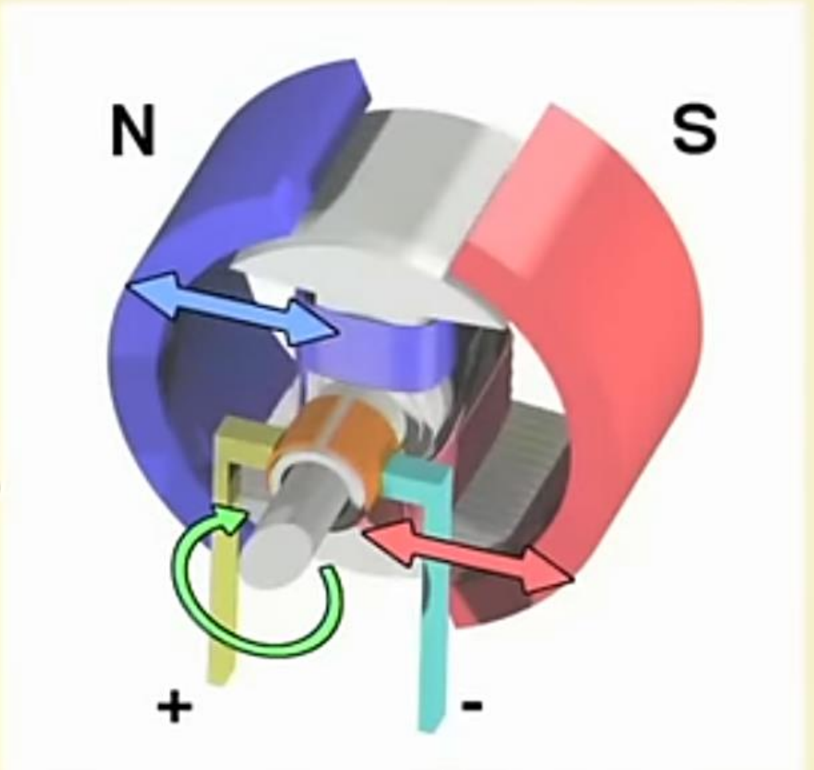
Electric Linear Actuator

- Powered by electrical signal
- Mechanical device containing linear guides, motors, and drive mechanisms
- Converts <u>electrical energy</u> into <u>linear displacement</u>
- Used in automation applications including electrical bell, opening and closing dampers, locking doors, and braking machine motions



Electric Rotary Actuator

- Powered by electrical signal
- Converts <u>electrical energy</u> into <u>rotational motion</u>
- Applications including quarter-turn valves, windows, and robotics

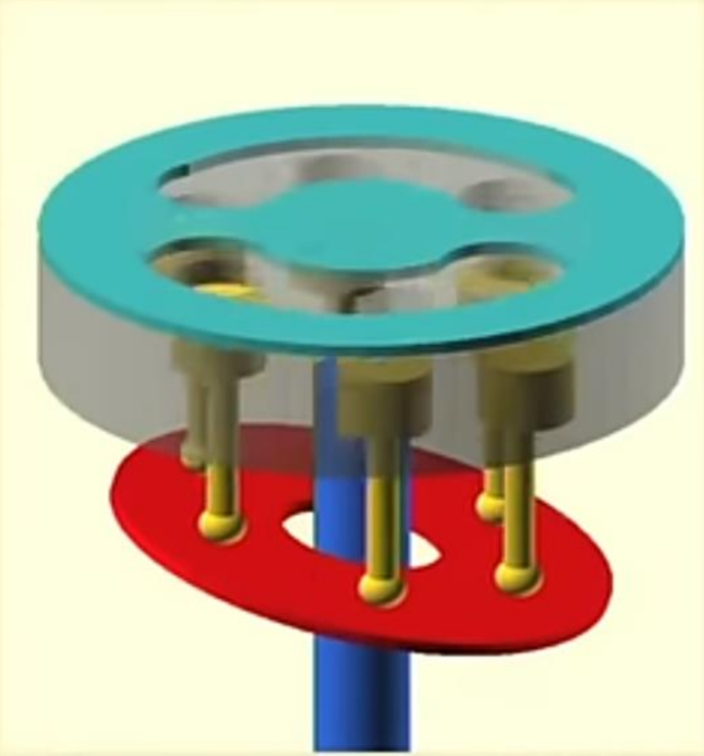


Fluid Power Linear Actuator

- Powered by hydraulic fluid, gas, or differential air pressure
- Mechanical devices have cylinder and piston mechanisms
- Produces linear displacement
- Primarily used in automation applications including clamping and welding

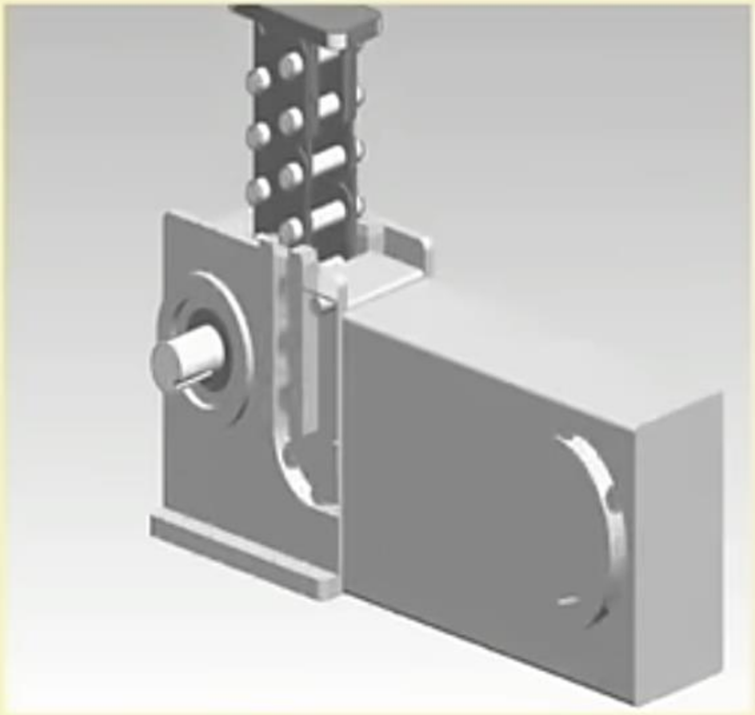
Fluid Power Rotary Actuator

- Powered by fluid, gas, or differential air pressure
- Consisting of gearing, and cylinder and piston mechanisms
- Converts hydraulic fluid, gas, or differential air pressure into rotational motion
- Primarily applications of this actuator are opening and closing dampers, doors, and



Linear Chain Actuator

- Mechanical devices containing sprockets and sections of chain
- Provides linear motion by the free ends of the specially designed chains
- Primarily used in motion control applications



Manual Linear Actuator

- Provides linear displacement through the translation of manually rotated screws or gears
- Consists of gearboxes, and hand operated knobs or wheels
- Primarily used for manipulating tools and workpieces

Manual Rotary Actuator

- Provides <u>rotary output</u> through the translation of <u>manually rotated</u> screws, levers, or gears
- Consists of hand operated knobs, levers, handwheels, and gearboxes
- Primarily used for the operation of valves

Thank You!

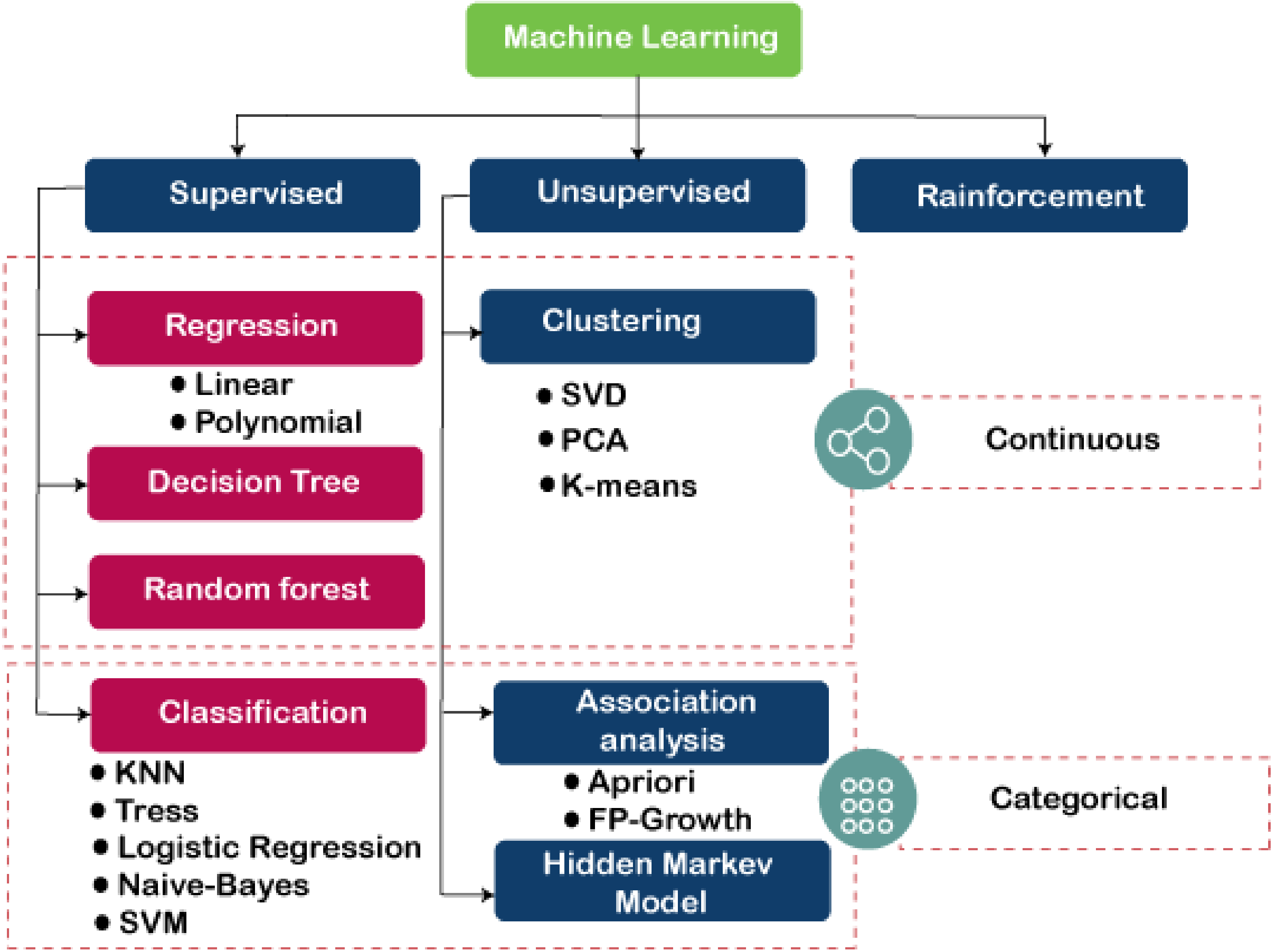
Industrial Internet of Things

Federated Learning for IIoT

Dr Abhishek Hazra

abhishek.h@iiits.in

Machine Learning for IIoT



Source: Internet

Supervised vs Unsupervised vs Reinforcement Learning

Supervised Learning :

1. Try to predict specific quantity
2. Have training example with labels

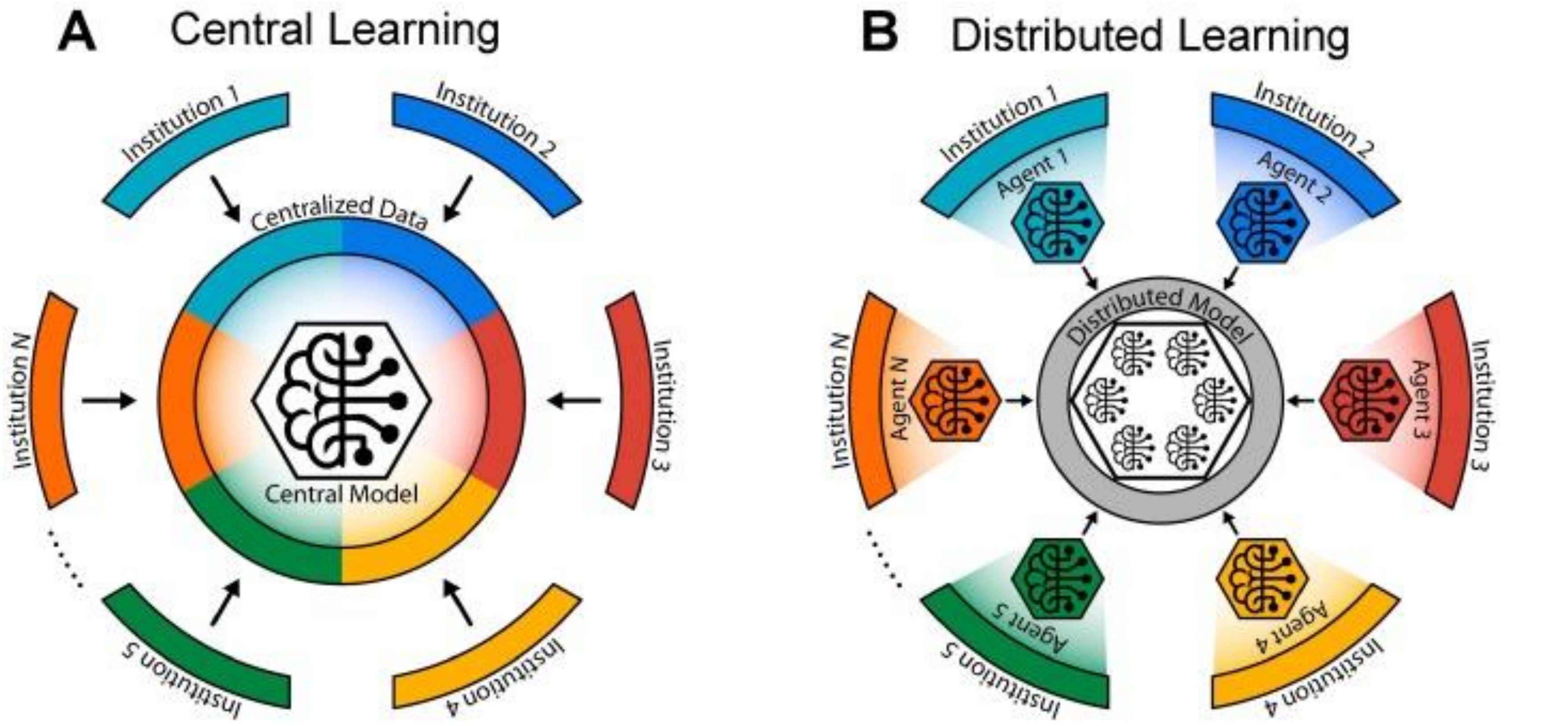
Unsupervised Learning :

1. Trying to understand the data
2. Looking for patterns in data
3. Does not required labeled data

Reinforcement Learning:

- Make a sequence of decisions
- An agent interacts with the environment and improves reward.

Centralized vs Distributed Learning



Anup et al. “Building machine learning models without sharing patient data: Asimulation-based analysis of distributed learning by ensembling”

Centralized vs Distributed Learning

Centralized Systems	
advantages	**disadvantages**
• It has a central and simple system management. • It has a relatively high degree of security, because of the existence of only one kind of access to the system.	• The system performance for each user decreases when many users try to attach simultaneously. • The system is relatively expensive (hardware and software). • The scalability of the mainframe systems is extremely low.

Shifting from centralized ML to decentralized ML

- **Enhanced Data Privacy**: Decentralized ML minimizes the need for centralizing sensitive data, addressing privacy concerns and aligning with data protection regulations.

- **Reduced Latency and Real-time Inference**: By processing data locally on edge devices, decentralized ML enables quicker decision-making and real-time inference, minimizing latency.

- **Scalability through Edge Computing**: Leveraging edge devices for computation allows for scalable and parallelized machine learning tasks, improving overall efficiency.

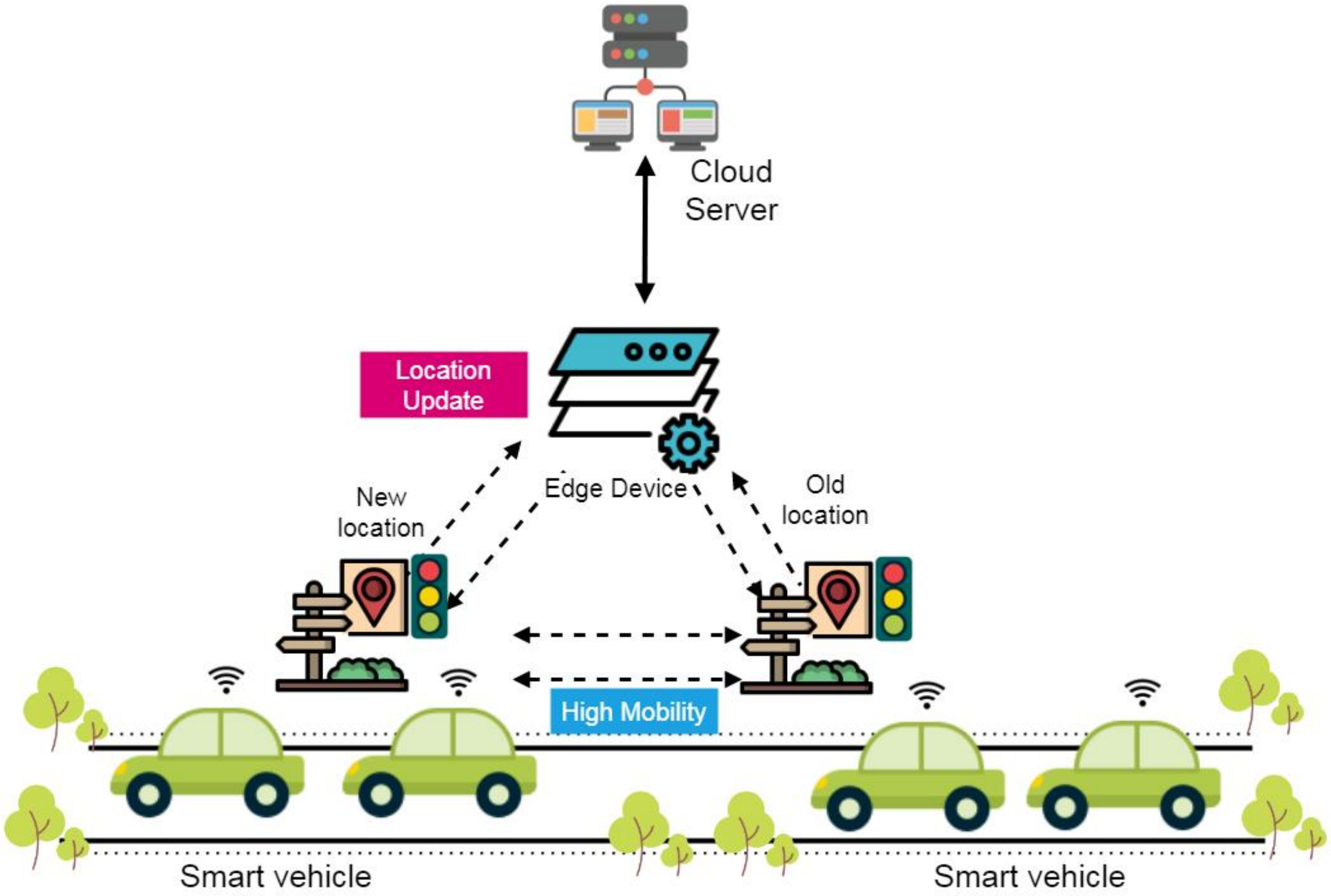
Shifting from centralized ML to decentralized ML

- **Improved Security**: Distributing the learning process across devices enhances security by reducing the risk associated with a centralized point of failure or potential security breaches.

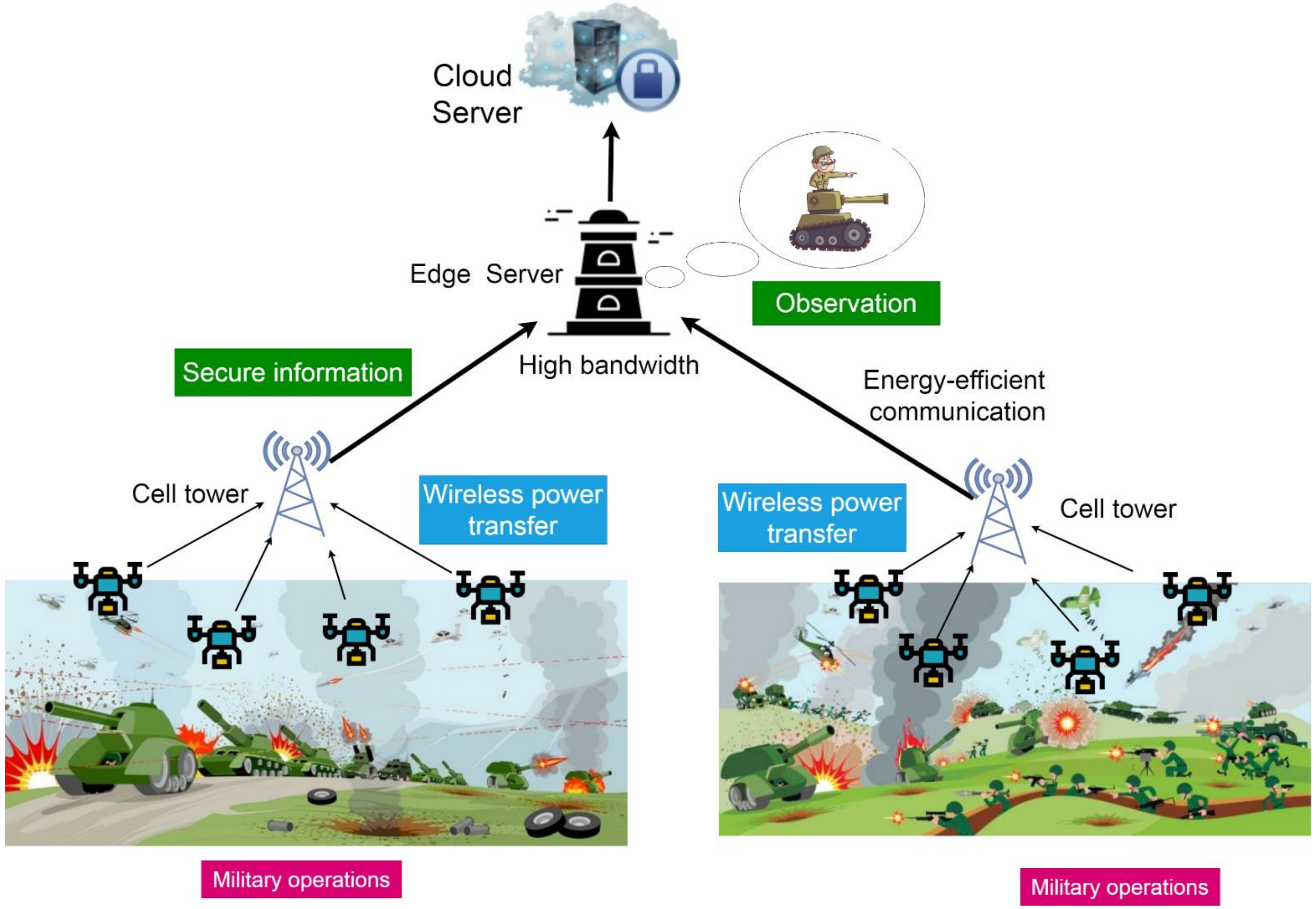
- **Energy Efficiency**: Local computation on edge devices reduces the need for continuous communication with a central server, leading to lower energy consumption.

- **Adaptability to Edge Devices**: Decentralized ML accommodates the diversity of edge devices, allowing for the deployment of machine learning models on a wide range of devices with varying capabilities.

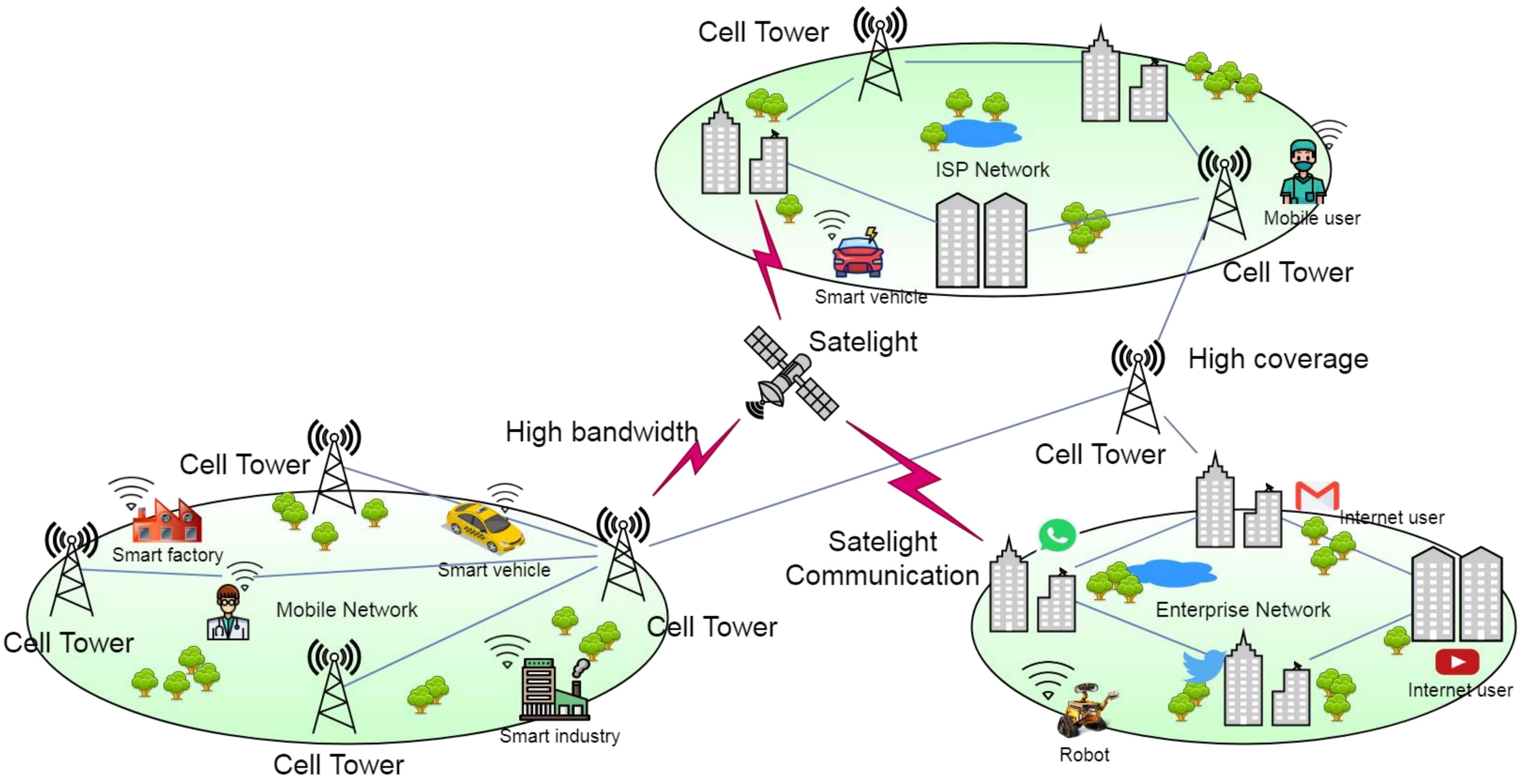
Application Area: Transportation system



Application Area: Military operation



Application Area: Industry



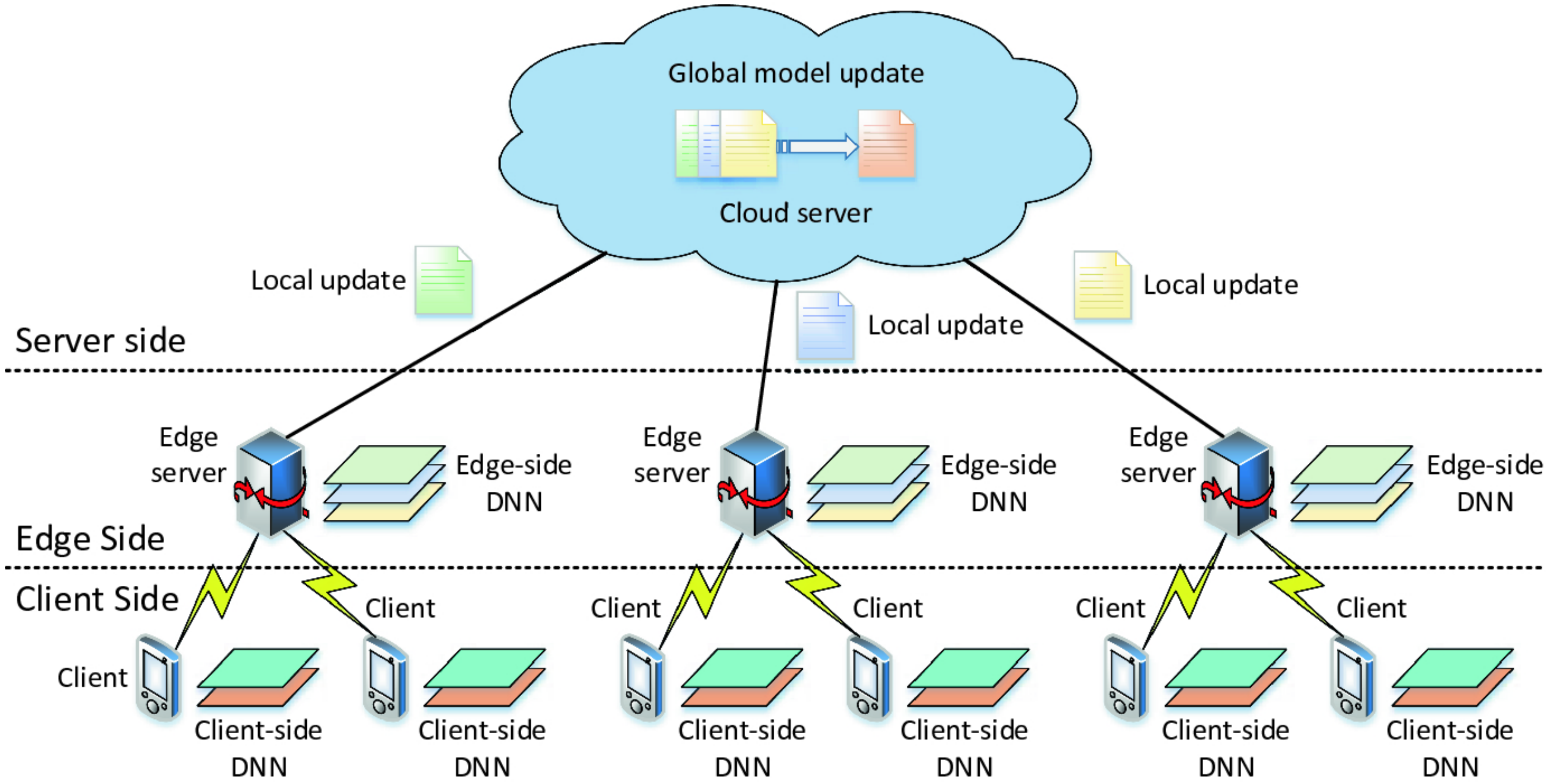
Distributed Learning

- In distributed learning, each individual agent is represented as a node in a graph.
- Edges between nodes indicate that the respective agents can exchange information.

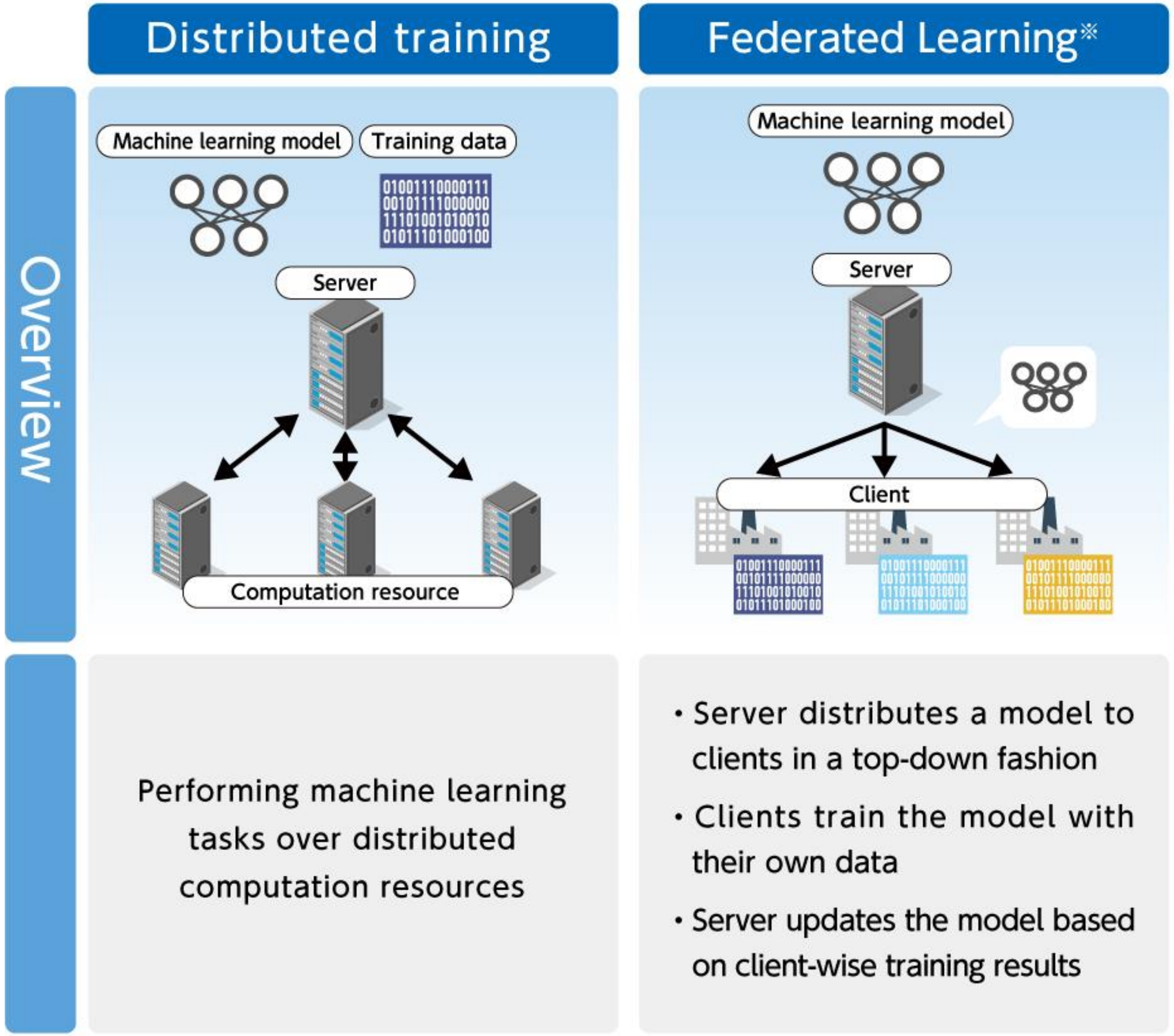
Advantage:

1. Use of multiple database
2. Performing ML tasks over distributed computing resources
3. High scalability and low maintenance.

Federated Learning



Distributed Learning vs Federated Learning



Federated Learning

Assume that each device $q \in \mathcal{Q}$ has a local dataset $\mathcal{D}_q = \{(\boldsymbol{x}_{q,1}, \boldsymbol{y}_{q,1}), (\boldsymbol{x}_{q,2}, \boldsymbol{y}_{q,2}), \ldots, (\boldsymbol{x}_{q,N_q}, \boldsymbol{y}_{q,N_q})\}$, where $\boldsymbol{x}_{q,v} \in \mathbb{R}^{N_{in}*1}$ is the v-th data sample characterized by N_{in} features collected by device q, and N_q is the total amount of data from device q, while $\boldsymbol{y}_{q,v} \in \mathbb{R}^{N_{out}*1}$ is the ground-truth label, corresponding to $\boldsymbol{x}_{q,v}$, respectively. It is noted that the data size of $\boldsymbol{x}_{q,v}$ relies on specific ML tasks.

Federated Learning

Let $\boldsymbol{w}_q$ represents the local model parameters of device q. Hence for device q, the loss function on its training dataset $\mathcal{D}_q$ can be represented as

$$F_q(\boldsymbol{w}) = \frac{1}{N_q} \sum_{v=1}^{N_q} f\left(\boldsymbol{w}; \boldsymbol{x}_{q,v}, \boldsymbol{y}_{q,v}\right), \quad \forall q \in \mathcal{Q} \tag{4}$$

where $f\left(\boldsymbol{w}; \boldsymbol{x}_{q,v}, \boldsymbol{y}_{q,v}\right)$ is the loss function of local model in the q-th device, which is defined differently relying on different ML algorithms, referring to Table II.

Federated Learning

TABLE II

LOSS FUNCTIONS OF TYPICAL ML ALGORITHMS

Model	Loss function $f(\boldsymbol{w}, \boldsymbol{x}_v, \boldsymbol{y}_v)$
Linear regression	$\frac{1}{2}\left\|\boldsymbol{y}_v - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_v\right\|^2$
Logistic regression	$-\log\left(1 + \exp\left(-\boldsymbol{y}_v \boldsymbol{x}_v^T \boldsymbol{w}\right)\right)$
K-means	$\frac{1}{2}\min_l \left\|\boldsymbol{x}_v - \boldsymbol{w}_{(l)}\right\|^2$
Squared-SVM	$\frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{2}\max\left\{0; 1 - \boldsymbol{y}_v \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_v\right\}^2$
Neural networks	Mean squared error, cross-entropy

Federated Learning

For simplicity, we assume that local datasets are statistically independent, i.e., $\mathcal{D}_q \cap \mathcal{D}_{q'} = \emptyset$ for $q \neq q'$. Let $\boldsymbol{w}$ denote the global model parameters, and hence the global loss function is denoted by

$$F(\boldsymbol{w}) \triangleq \sum_{q=1}^{Q} \frac{N_q F_q(\boldsymbol{w})}{N} = \frac{1}{N} \sum_{q=1}^{Q} \sum_{v=1}^{N_q} f\left(\boldsymbol{w}_q; \boldsymbol{x}_{q,v}, \boldsymbol{y}_{q,v}\right), \tag{5}$$

where $N = \sum_{q=1}^{Q} N_q$ is the number of all the samples from devices, while $\boldsymbol{w}_q$ is the local model parameters of device q. The objective of FL is to find the global optimal model parameters to minimize the global loss function as

$$\boldsymbol{P}1: \boldsymbol{w}^* = \arg\min F(\boldsymbol{w}), \tag{6}$$

Federated Learning

where $\boldsymbol{w}^* = \boldsymbol{w}_1 = \cdots = \boldsymbol{w}_Q$,

The procedure for one global iteration of FL is summarized as follows:

- Devices train their local models in parallel.
- Devices upload their new-updated models to the UAV.
- UAV aggregates the received local models and broadcasts the new global model to the devices.

Federated Learning

$\boldsymbol{P2}$.

$$\boldsymbol{P2}: \quad \min_{\boldsymbol{h}_q \in \mathbb{R}^{N_{in}*1}} L_q\left(\boldsymbol{w}^{(i)}, \boldsymbol{h}_q\right) \triangleq F_q\left(\boldsymbol{w}^{(i)} + \boldsymbol{h}_q\right) - \left(\nabla F_q\left(\boldsymbol{w}^{(i)}\right) - \xi \nabla F\left(\boldsymbol{w}^{(i)}\right)\right)^{\mathrm{T}} \boldsymbol{h}_q, \quad (7)$$

where $\boldsymbol{h}_q$ is the difference between the global model and the local model of device q, while ξ is a constant. To solve $\boldsymbol{P2}$, relying on the gradient method, each device q updates $\boldsymbol{h}_q$ by

$$\boldsymbol{h}_q^{(i),(j+1)} = \boldsymbol{h}_q^{(i),(j)} - \delta \nabla L_q\left(\boldsymbol{w}^{(i)}, \boldsymbol{h}_q^{(i),(j)}\right), \quad (8)$$

Federated Learning

where $\boldsymbol{h}_q^{(i),(j)}$ represents the value of $\boldsymbol{h}_q$ at the j-th local iteration, while δ is the step size. Eq. (8) will be executed repeatedly until the given local accuracy η is satisfied, i.e.,

$$\begin{aligned} L_q\left(\boldsymbol{w}^{(i)},\ \boldsymbol{h}_q^{(i),(j)}\right) - L_q\left(\boldsymbol{w}^{(i)}, \boldsymbol{h}_q^{(i),*}\right) \\ \leq \eta\left(L_q\left(\boldsymbol{w}^{(i)}, \boldsymbol{h}_q^{(i),(0)}\right) - L_q\left(\boldsymbol{w}^{(i)}, \boldsymbol{h}_q^{(i),*}\right)\right). \end{aligned} \tag{9}$$

Federated Learning

When the optimal solution $h_q^{(i),*}$ is obtained, the device will send it to the UAV. After receiving all the $h_q^{(i),*}$ from each device, the UAV updates the global model parameters by

$$\boldsymbol{w}^{(i+1)} = \boldsymbol{w}^{(i)} + \frac{1}{Q}\sum_{q=1}^{Q} \boldsymbol{h}_q^{(i),*}, \tag{10}$$

and broadcast the new value to each device. The aforementioned FL procedure will be executed repeatedly until the given global accuracy ϵ is satisfied, i.e.,

$$F\left(\boldsymbol{w}^{(i)}\right) - F\left(\boldsymbol{w}^{*}\right) \leq \epsilon\left(F\left(\boldsymbol{w}^{(0)}\right) - F\left(\boldsymbol{w}^{*}\right)\right) \tag{11}$$

Thank You!