Research proposal under PhD program of Mumbai University
Academic year 2019-20

1	**Faculty**	Information Technology
2	**Constituent College**	Finolex Academy of Management and Technology, Ratnagiri
3	**Department**	Information Technology
4	**Name of Research Guide**	Dr. Vinayak Ashok Bharadi
5	**Research topic**	Distributed Decomposed Analytics of IoT, SAR and Social Network data
6	**Name of research student**	Mr. Shashank Shashikant Tolye
7	**Signature of Guide with Date**	
8	**Signature of Research Student with date**	
9	**Name & Signature of Research Centre Head with Date**	
10	**Table of content**	1. Abstract 2. Introduction 3. Motivation 4. Literature review 5. Research Problem 6. Research Objectives 7. Research Design 8. System Evaluation 9. Expected outcomes 10. Conclusion References

Name of the student: **Mr. Shashank Shashikant Tolye**

Course/Branch: **Ph. D. Information Technology**

Research Title: **Distributed Decomposed Analytics of IoT, SAR and Social Network data**

Name of the Research Guide: **Dr. Vinayak Ashok Bharadi**

Date of Submission:

Mr. S. S. Tolye	**Dr. V.A.Bharadi**
Research Scholar	**Guide**
Head of Research Center	**Principal**

Table of Contents

Sr. No.	**Topic Name**	**Page No.**
01	Abstract	07
02	Introduction	08
03	Motivation	17
04	Literature Survey	18
05	Research problem	20
06	Research Objectives	20
07	Research Design	21
08	System Evaluation	27
09	Expected Outcomes	28
10	Conclusion	28
	References	29

List of Figures

Sr.No.	**Name of figure**	**Page No.**
Figure 1	Relationship between big data analytics and IoT	10
Figure 2	The Internet of Things and Fog Computing	11
Figure 3	SAR basic block diagram	12
Figure 4	The proposed architecture of Distributed Decomposed Analytics of IoT, SAR and Social Network data	21
Figure 5	Decomposition of data analytics computing program into decomposed computing units and placing those computing units onto different fog nodes in the infrastructure	22
Figure 6	Big Data analysis framework for remote sensing data using parallel platforms.	23
Figure 7	Sentiment Analysis: Training & Evaluation pipeline	24
Figure 8	Proposed Conceptual Model of Distributed Decomposed Analytics of IoT, SAR and Social Network data	25

List of Tables

Sr.No.	**Title of Table**	**Page No.**
Table 1	Literature review	18
Table 2	System evaluation	27

Abbreviations and Symbols

IoT	Internet of Things
SAR	Synthetic Aperture Radar
RDF	Resource Description Format
GPU	Graphics Processing Unit
HPC	High Performance Computing
MPI	Message Passing Interface
RCMC	Range Cell Migration Correction
SRC	Secondary Range Compression
FNN	Feed Forward Neural Network
RNN	Recurrent Neural Network
RBFNN	Radial Basis Function Neural Network
KSONN	Kohonen Self Organizing Neural Network
MNN	Modular Neural Network
CNN	Convolutional Neural Network
DNN	Deep Neural Network
OWL	Web Ontology Language

Abstract:

Deep learning techniques and advanced computing infrastructures have enabled us to process data more efficiently and make meaningful predictions on weather data. There are a variety of sources through which we can obtain the information about the calamities; however, three major sources of providing very crucial weather data are IoT sensors, SAR data and social media posts from a particular location. In this research, a system is proposed that takes IoT sensor data, SAR images and twitter feeds from a geographical location and creates a learning model that will provide a decision-making system for anomaly detection in order to minimize or nullify any casualties.

Keywords: IoT Analytics, SAR data, Twitter feeds, RDF, GPU, MPI

1. Introduction

1.1. Topic area

IoT is a network of interconnected physical or virtual 'things' which are having defined interfaces to be integrated as an information network in order to communicate with one another, with other devices and with services over the Internet to accomplish stated objective.

In recent years, big data and Internet of Things (IoT) implementations started getting more attention. Researchers focused on developing big data analytics solutions using machine learning models. Machine learning is a rising trend in this field due to its ability to extract hidden features and patterns even in highly complex datasets.

IoT sensors are deployed everywhere in variety of applications. Weather monitoring and forecasting is a widely used IoT application area that can be further utilized in application areas such as Traffic monitoring and management, Agriculture, Social Engineering. In contrast to previous frameworks that used statistical methods for analysis the new research techniques use more big data solutions and machine learning methods for prediction using weather and other IoT sensor data[3].

Another prominent data source for real life related events is SAR data. Voluminous data (Multispectral, Hyperspectral) from Variety of sensors (Airborne sensors, space borne sensors) with Velocity (high temporal resolution) is available for analysis.

People are also sharing the critical information related to natural disasters, extreme conditions or such related events are social media, though the authenticity is questionable semantic data analytics tools can be used on the data collected from various users for understanding the nature of the situation.

The proposed system aims to gather the large amount of the data generated from the IoT Sensors, SAR Images and Social Network such as Twitter feeds, process the data and use it for detection of anomalies, extreme whether conditions in real time. When used for decision making to support natural disasters such as earthquakes, floods, oil-spills etc., for near real time accurate responses, is a problem that needs Big Data Analytics. To extract a real time information from this big data, high performance computing (HPC) with a kind of scalable solution that reduces the execution time are in extreme demand. To serve this real time need, scalable hybrid parallelism approach based on state of art multi-core GPUs and Message Passing Interface (MPI) is possible to be designed for analyzing remote sensing disaster data. Further the accuracy of the prediction can be improved if we combine or corroborate the findings with IoT sensors data.

1.2. Distributed Decomposed IoT Data Analytics

1.2.1. IoT

Internet of Things (IoT) is a network of Internet connected devices, sensors, and computers. In the literature, IoT is defined as follows:" The Internet of Things allows people and things to be connected Anytime, Anyplace, with Anything and Anyone, ideally using Any path/network and Any service" [1][2].

IoT devices are increasing in numbers day by day. U.S. National Intelligence Council states that" by 2025 Internet nodes may reside in things that we use every day, food packages, furniture, paper documents, and more" [3][2]. These developments lead researchers to develop IoT platforms, frameworks to process and analyze huge IoT sensor data.

The basic idea of this concept is the pervasive presence around us of a variety of things or objects – such as Radio-Frequency Identification (RFID) tags, sensors, actuators, mobile phones, etc. – which, through unique addressing schemes, are able to interact with each other and cooperate with their neighbors to reach common goals [4].

1.2.2. IoT Analytics

The development of big data and the Internet of things (IoT) is rapidly accelerating and affecting all areas of technologies and businesses by increasing the benefits for organizations and individuals. The growth of data produced via IoT has played a major role on the big data landscape.

Big data can be categorized according to three aspects: (a) volume, (b) variety, and (c) velocity [5]. The widespread popularity of IoT has made big data analytics challenging because of the processing and collection of data through different sensors in the IoT environment.

IoT big data analytics can be defined as the steps in which a variety of IoT data are examined [6] to reveal trends, unseen patterns, hidden correlations, and new information [7]. IoT big data analytics aims to assist business associations and other organizations to achieve improved understanding of data, and thus, make efficient and well-informed decisions. Big data analytics enables data miners and scientists to analyze huge amounts of unstructured data that can be harnessed using traditional tools. Moreover, big data analytics aims to immediately extract knowledgeable information using data mining techniques that help in making predictions, identifying recent trends, finding hidden information, and making decisions.

Big data analytics is rapidly emerging as a key IoT initiative to improve decision making. One of the most prominent features of IoT is its analysis of information about "connected things." Big data analytics in IoT requires processing a large amount of data on the fly and storing the data in various storage technologies. Given that much of the unstructured data are gathered directly from web-enabled "things", big data implementations will necessitate performing lightning-fast analytics with large queries to allow organizations to gain rapid insights, make quick decisions, and interact with people and other devices. The interconnection of sensing and actuating devices provide the capability to share information across platforms through a unified architecture and develop a common operating picture for enabling innovative applications. The relationship between Big Data analytics and IoT is shown in figure 1.

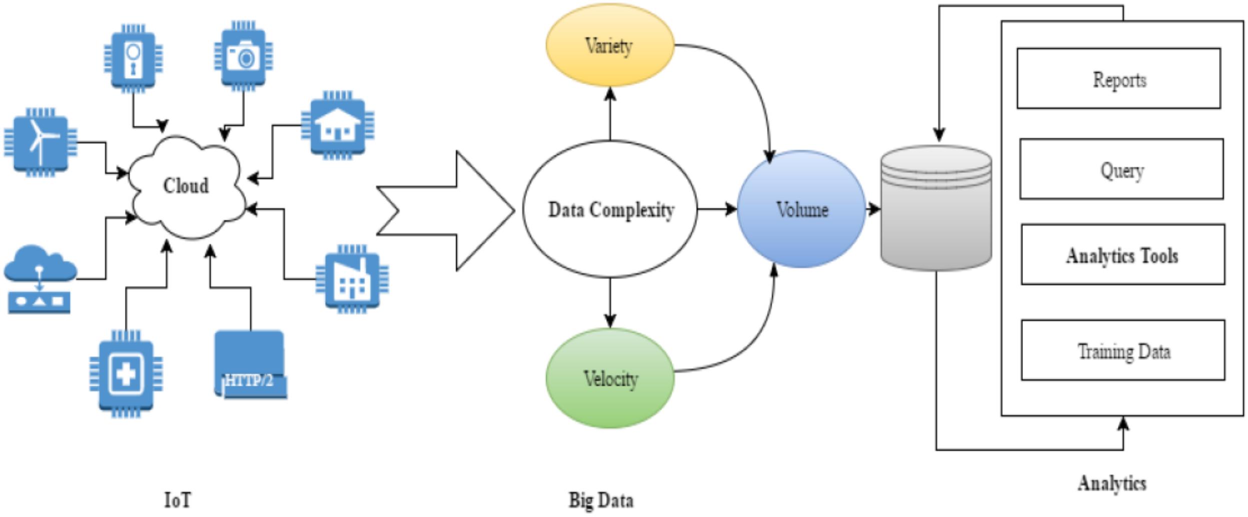


Figure 1: Relationship between big data analytics and IoT [12]

1.2.3. Fog Computing

Fog computing extends the Cloud Computing paradigm to the edge of the network, thus enabling a new breed of applications and services.

Characteristics of Fog Computing [9]:

- Low Latency and Location Awareness
- Wide-spread geographical distribution
- Mobility
- Very large number of nodes
- Predominant role of wireless access
- Strong presence of streaming and real time applications
- Heterogeneity

Fog Computing is a highly virtualized platform that provides compute, storage, and networking services between end devices and traditional Cloud Computing Data Centers, typically, but not exclusively located at the edge of network. Figure 2 presents the idealized information and computing architecture supporting the future IoT applications, and illustrates the role of Fog Computing.

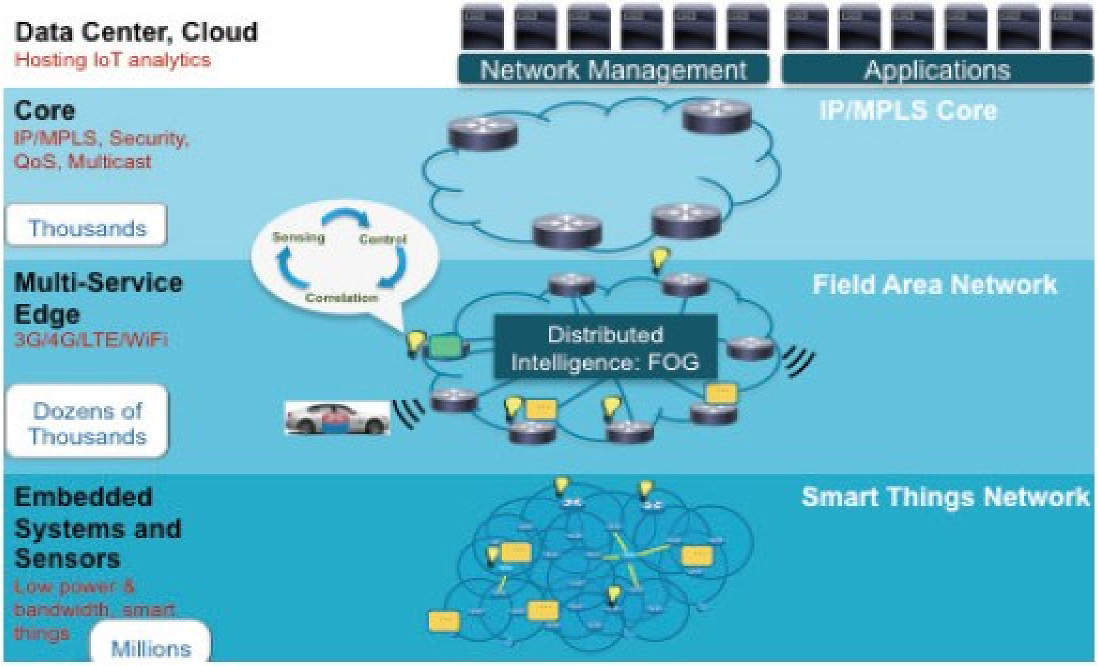


Figure 2: The Internet of Things and Fog Computing [14]

If the requirements and behavior of the IoT applications are analyzed, it can be seen that a two-layer architecture (cloud-IoT devices) can hardly support all the communication and data processing required by all these billions of connected devices. If we get it to support them, the scalability, latency, and response time would be very limited. Usually, IoT applications have stringent requirements. Most of them require almost real-time responsiveness while, at the same time, the Quality of Service (QoS), the security and privacy and the location-awareness of the response have to be achieved.

1.2.4. Distributed Decomposed Data Analytics in Fog environment

The increasing range of real-world IoT deployments essentially increases the sources of data generation, thereby globally strengthening the challenges already being faced in the Big Data space, particularly regarding moving data from one end (i.e. from data sources such as sensor/IoT devices at the edge level of infrastructure) to the other extreme end (i.e. centralized data centers at the cloud) in the network infrastructure.

Fog computing and edge computing, which distribute traditionally centralized datacenter operations closer to the end users, have been receiving considerable attention as enablers of the next level of interactivity and cognition in the Internet of Things (IoT)[11]. Executing parts of computing operations on local gateways can reduce bandwidth consumption and cloud computing costs, and hence architectures that allow executing services in multiple points have been recently made available [12] [13]. However, decomposing computing programs between the cloud and the computing gateways is not straightforward, particularly for complex algorithms in data analytics applications. In particular, traditional parallelization methods, developed for distributing operations to homogeneous nodes, are insufficient for fog and edge computing settings due to the heterogeneity of capabilities of gateways and cloud nodes.

Data in IoT deployments moves from things to cloud, and along this continuum passes through a number of network devices such as routers, gateways, etc. Each of these devices can be a potential candidate to host partial computing analytics capability to analyze the data, and further sending the calculated partial results instead of sending the raw data to cloud [14].

1.3. SAR Data Analytics

1.3.1. SAR Data Processing

Synthetic Aperture Radar (SAR) is a remote sensing system used to obtain high-resolution images of target. The Synthetic Aperture Radar (SAR) is an ingenuous radar system which can acquire data with very high resolution. In a standard monostatic architecture, the system is composed of a platform (i.e. airborne or satellite) with the same antenna for transmitter and receiver, mounted on a single beam-forming antenna on a moving platform such as an aircraft or spacecraft, from which a target scene is repeatedly illuminated with pulses of radio waves. The many echo waveforms received successively at the different antenna positions are detected and stored and then post-processed together [15].

Due to physical limitations, it is not possible to manufacture an antenna of long length and mount it on an airborne platform. Now when the RADAR irradiates the target, the beam increases as it makes way to the target. So wider the beam, lesser the details or it is easy to discern the intended target. The main concept of SAR is that there are two directions involved, range direction in which transmitted pulses travel and Azimuth direction, its the direction of sensor movement. The basic block diagram of SAR is as shown in Figure 3.

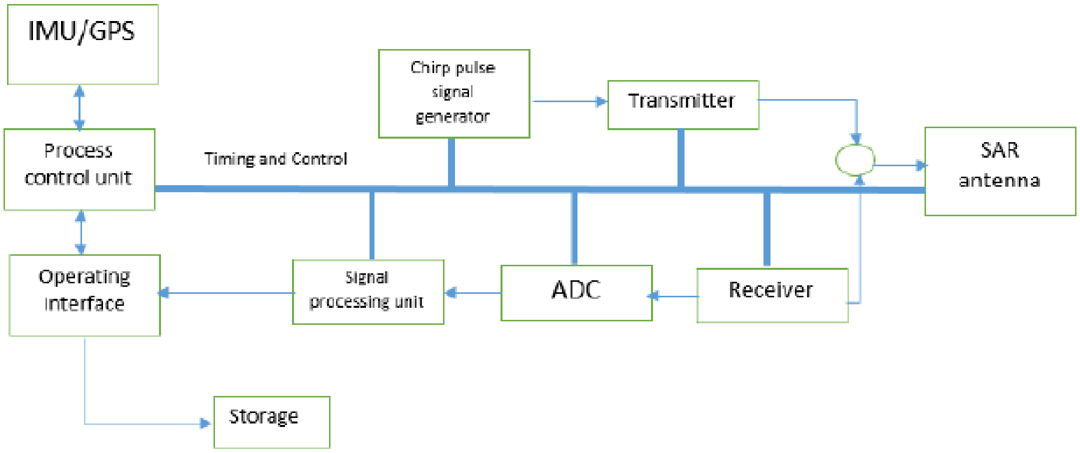


Figure 3: SAR basic block diagram [15]

There are mainly three raw data processing algorithms [15]:

a) Range Doppler Algorithm

The range Doppler algorithm is one of the techniques used in transforming raw SAR data into a SAR image. The received signal is stored in a form of two dimensional array, both in range and azimuth directions. So these algorithms are used in two dimensional scenarios.

b) Chirp Scaling Algorithm

The Chirp Scaling Algorithm [16] eliminates the interpolation and processes the data in the range Doppler domain and also in the 2-D frequency domain. In the 2-D frequency domain, the processing is based on a scaling principle, whereby a frequency modulation is applied to a chirp encoded signal to achieve a shift in the signal.

c) Omega-K Algorithm

The Omega-K algorithm has a special type of interpolation induced known as Stolt interpolation or Stolt mapping. In this one operation, the residual Range Cell Migration Correction (RCMC), secondary range compression (SRC) and azimuth compression are done simultaneously.

1.4. Social Media Data (Twitter platform)

Nowadays, people from all around the world use social media sites to share information. Twitter for example is a platform in which users send, read posts known as ‘tweets’ and interact with different communities. Users share their daily lives, post their opinions on everything such as brands and places. Companies can benefit from this massive platform by collecting data related to opinions on them.

As soon as a natural disaster occurs, people experiencing the hazard turn to social media to share it with the world. Immediately, posts reporting the disaster spread around the globe, sometimes even before the hazard is recorded by the governments or reported by the media, as it usually takes time for traditional media and official organizations to locate and validate the accuracy of news [17].

1.4.1. Twitter and its importance:

Twitter is an online social network (OSN) used by millions of people all over the world. It enables people to stay connected with their friends, family and colleagues. With advancement in technology, it has become easier to access Twitter using mobile devices like iPhones and iPads. It enables its users to post messages which are 140 characters or less which are called tweets. Users can also retweet messages, which is posting the message posted by other users. This can be thought of as email forwarding. These tweets can be displayed to all users or only to the people following the user. A user can follow other users but it is not necessary for the user who is being followed to follow back. This makes the links in Twitter directed. Currently, Twitter has 288 million monthly active users with an average of 500 million tweets being sent per day [18].

Twitter is categorized as a micro-blogging service. Microblogging is a form of blogging that allows users to send brief text updates or micromedia such as photographs or audio clips. Microblogging services other than Twitter include Tumblr, Plurk, Emote.in, Squeelr, Jaiku, identi.ca, and so on [19].

1.4.2. Real Time event detection from twitter:

An event is an arbitrary classification of a space/time region. An event might have actively participating agents, passive factors, products, and a location in space/time [20]. We target events such as earthquakes, typhoons, and traffic jams, which are visible through tweets. These events have several properties:

a) They are of large scale (many users experience the event)

b) They particularly influence people's daily life (for that reason, they are induced to tweet about it)

c) They have both spatial and temporal regions (so that real-time location estimation would be possible).

Twitter users write tweets several times in a single day. Users can know how other users are doing and often what they are thinking about now, users repeatedly return to the site and check to see what other people are doing. The large number of updates results in numerous reports related to events. They include social events such as parties, baseball games, and presidential campaigns. They also include disastrous events such as storm, fire, traffic jam, riots, heavy rainfall, and earthquakes. Actually, Twitter is used for various realtime notification such as that necessary for help during a large-scale fire emergency and live traffic updates [19, 20].

1.5. Deep Learning:

Deep learning is about automatically learning multiple levels of representations of the underlying distribution of the data to be modeled [22]. In other words, a deep learning algorithm automatically extracts the low- and high-level features necessary for classification. By high level features, one means feature that hierarchically depends on other features. For instance, in the context of computer vision, this implies that a deep learning algorithm will learn its own low level representations from a raw image (such as edge detector, gabor filters, etc...), then build representations that depend on those low level representations (such as a linear or non-linear combinations of those low-level representations), and successively repeat the same process for higher levels.

1.5.1. Neural Networks:

Neural Network is a machine learning (ML) technique that is inspired by and resembles the human nervous system and the structure of the brain. It consists of processing units organized in input, hidden and output layers. The nodes or units in each layer are connected to nodes in adjacent layers. Each connection has a weight value. The inputs are multiplied by the respective weights and summed at each unit. The sum then undergoes a transformation based on the activation function, which is in most cases is a sigmoid function, tan hyperbolic or rectified linear unit (ReLU).

The implementation of neural networks consists of the following steps:

a) Acquire training and testing data set

b) Train the network

c) Make prediction with test data

1.5.2. Classification of the Neural Networks:

Neural Networks can be classified into the following different types: [22]

a) Feedforward Neural Network

In feedforward neural network, information flows in just one direction from input to output layer (via hidden nodes if any). They do not form any circles or loopbacks.

b) Recurrent Neural Network (RNN)

The processing units in RNN form a cycle. The output of a layer becomes the input to the next layer, which is typically the only layer in the network, thus the output of the layer becomes an input to itself forming a feedback loop. This allows the network to have memory about the previous states and use that to influence the current output.

c) Radial Basis Function Neural Network

Radial basis function neural network is used in classification, function approximation, time series prediction problems, etc. It consists of input, hidden and output layers. The hidden layer includes a radial basis function (implemented as gaussian function) and each node represents a cluster center. The network learns to designate the input to a center and the output layer combines the outputs of the radial basis function and weight parameters to perform classification or inference [24].

d) Kohonen Self Organizing Neural Network

Kohonen self-organizing neural network self organizes the network model into the input data using unsupervised learning. It consists of two fully connected layers, i.e., input layer and output layer. The output layer is organized as a two dimensional grid. There is no activation function and the weights represent the attributes (position) of the output layer node. The Euclidian distance between the input data and each output layer node with respect to the weights are calculated.

e) Modular Neural Network

Modular neural network breaks down large network into smaller independent neural network modules. The smaller networks perform specific task which are later combined as part of a single output of the entire network [25].

1.5.3. Deep Neural Networks' implementation:

DNNs are implemented in the following popular ways [25]:

a) Sparse Autoencoders

Autoencoders are neural networks that learn features or encoding from a given dataset in order to perform dimensionality reduction. Sparse Autoencoder is a variation of Autoencoders, where some of the units output a value close to zero or are inactive and do not fire.

b) Convolution Neural Networks (CNNs or ConvNets)

One of the most popular deep neural networks is the Convolutional Neural Network (CNN) [26]. CNN have multiple layers; including convolutional layer, non-linearity layer, pooling layer and fully connected layer. The convolutional and fully- connected layers have parameters but pooling and non-linearity layers don't have parameters. The CNN has an excellent performance in machine learning problems. Especially the applications that deal with image data, such as largest image classification data set (Image Net), computer vision, and in natural language processing (NLP) and the results achieved were very amazing.

c) Restricted Boltzmann Machines (RBMs)

Restricted Boltzmann Machine is an artificial neural network where we can apply unsupervised learning algorithm to build non-linear generative models from unlabeled data. The goal is to train the network to increase a function (e.g., product or log) of the probability of vector in the visible units so it can probabilistically reconstruct the input. It learns the probability distribution over its inputs. RBM is made of two-layer network called the visible layer and the hidden layer. Each unit in the visible layer is connected to all units in the hidden layer and there are no connections between the units in the same layer.

d) Kohonen Self Organizing Neural Network (KSONN):

It self organizes the network model into the input data using unsupervised learning. It consists of two fully connected layers, i.e., input layer and output layer. The output layer is organized as a two-dimensional grid. There is no activation function and the weights represent the attributes (position) of the output layer node.

e) Modular Neural Network (MNN):

Modular neural network breaks down large network into smaller independent neural network modules. The smaller networks perform specific task which are later combined as part of a single output of the entire network.

2. Motivation

In recent years, big data and Internet of Things (IoT) implementations started getting more attention. Researchers focused on developing big data analytics solutions using machine learning models. Machine learning is a rising trend in this field due to its ability to extract hidden features and patterns even in highly complex datasets. IoT sensors are deployed everywhere in variety of applications. Weather monitoring and forecasting is a widely used IoT application area that can be further utilized in application areas such as Traffic monitoring and management, Agriculture, Social Engineering. In contrast to previous frameworks that used statistical methods for analysis the new research techniques use more big data solutions and machine learning methods for prediction using weather and other IoT sensor data.

Another prominent data source for real life related events is SAR data. Voluminous data (Multispectral, Hyperspectral) from Variety of sensors (Airborne sensors, space borne sensors) with Velocity (high temporal resolution) is available for analysis.

People are also sharing the critical information related to natural disasters, extreme conditions or such related events are social media, though the authenticity is questionable semantic data analytics tools can be used on the data collected from various users for understanding the nature of the situation.

3. Literature Review

An extensive literature review was carried out by referring to the reputed journals. The following table summarizes the literature referred and the key findings.

Sr. No	Research Paper Name with Author	Publication	Description	Key Findings
01	**“Real Time Analysis of Sensor Data for the Internet of Things by means of Clustering and Event Processing”** Hugo Hromic, et al (IEEE)	2015 IEEE	1. It introduces an approach to sensor data analytics using OpenIoT middleware 2. The air quality conditions have been analyzed using mobile crowd data from wearable sensors.	1. Sensor data analytics using real time event processing and clustering algorithms. 2. Use of intelligent servers running in cloud environments and edge servers for real time data acquisition, annotation and processing of sensor data.
02	**“Integration of IoT, Transport SDN, and Edge/Cloud Computing for Dynamic Distribution of IoT Analytics and Efficient Use of Network Resources”** Raul Munoz, et al (IEEE)	2018 IEEE	1. It presents the first IoT-aware multilayer transport SDN. 2. Deployment of IoT traffic control and congestion avoidance mechanism for dynamic distribution of IoT processing to the edge of the network.	It enables to offload the transports networks by dynamically and efficiently distributing the processing of IoT analytics from core datacenters to the network edge.
03	**“Weather Data Analysis and Sensor Fault Detection Using An Extended IoT Framework with Semantics, Big Data, and Machine Learning”** Aras Can Onal, et al (IEEE)	2017 (IEEE)	1. It presents a weather clustering model implemented using the Big Data IoT framework. 2. IoT-related ontologies are proposed and developed to provide a common language to express things, relationships towards higher levels of interoperability.	1. It presents an extended IoT framework that integrates data retrieval, processing and learning layers. 2. Learning layer utilizes clustering unsupervised learning method to best utilize the associated big data.

04	**“Distributed Decomposed Data Analytics in Fog Enabled IoT Deployments”** Mohit Taneja et al (IEEE)	2019 (IEEE)	Presented a fog specific decomposition method and its application to analytics model to run in a distributed manner for IoT deployments.	A generic methodology that distributes data analytics to fog-based layers to reduce the resource utilization.
05	**“Accelerating Big Data Processing Chain In Image Information Mining Using A Hybrid HPC Approach”** K. R. Kurte, et al (IEEE)	2016 (IEEE)	Presented a hybrid MPI+GPU approach to overcome big data processing limitation in Remote-sensed data.	1. Applicability of hybrid MPI+GPU approach towards rapid processing RS archives in disaster situation. 2. Offline modules in Spatial Image Information Mining (SIIM) can be pipelined to achieve task level parallelism.
06	**“Big Data Processing Using HPC For Remote Sensing Disaster Data”** U.M. Bhangale, et al (IEEE)	2016 (IEEE)	Proposed a big data analytics framework for remote sensing data.	A disaster data analytics framework to suit Big data’s high computational resources requirement.
07	**“Accelerating Time-Domain SAR Raw Data Simulation for Large Areas Using Multi-GPUs”** F. Zhang, et al (IEEE)	2014 (IEEE)	Proposed an improvement on the traditional GPU-based algorithm by access conflict optimization for SAR raw data simulation for large locations.	Multi-GPUs-based time-domain SAR raw data scalable parallel simulation method.
08	**“Sentiment Analysis of Twitter Data ”** S. El_Rahman, et al,(IEEE)	2019 (IEEE)	Proposed a model for performing sentiment analysis of real data of twitter feeds.	This approach shows a strong performance on mining texts directly from twitter.

09	**“Towards Natural Disasters Detection from Twitter using Topic Modelling”** M Hagras, et al (IEEE)	2017 (IEEE)	Discusses results of using the Latent Dircherilet Allocation LDA topic modelling technique to detect tweets related to the 2011 Japan Tsunami.	An evaluation algorithm was designed to test the effectiveness of the LDA algorithm with 76% accuracy of successful detection.

Table 1: Literature review

4. Research Problem

The purpose of research is to enhance the prediction of weather forecasting systems (which rely on data from the satellites on their own networks) as well as detection of such anomalies by analyzing the data from IoT, Remote Sensing Satellites (SAR), and Twitter feeds. For the analysis the deep learning and semantics-based technique will be used.

5. Research Objectives

The research aims to create an integrated approach that will take the input from the IoT sensor data, remote sensed SAR data and live twitter feeds and process the analytics to produce meaningful prediction about extreme weather conditions or anomalies in real time.

a) Gather data from various sources; like IoT sensors, SAR and live social network feeds in real time.
b) Perform the analytics of IoT using Distributed Decomposition
c) Perform high performance analytics of Remote Sensed (RS) SAR data.
d) Use semantic algorithms on twitter data to extract specific phrases describing extreme events.
e) Create a learning model that learns from the gathered and processed data from above mentioned three sources to predict the extreme events or anomalies in real time.

6. Research Design:

The proposed architecture design is as shown in Figure-4:

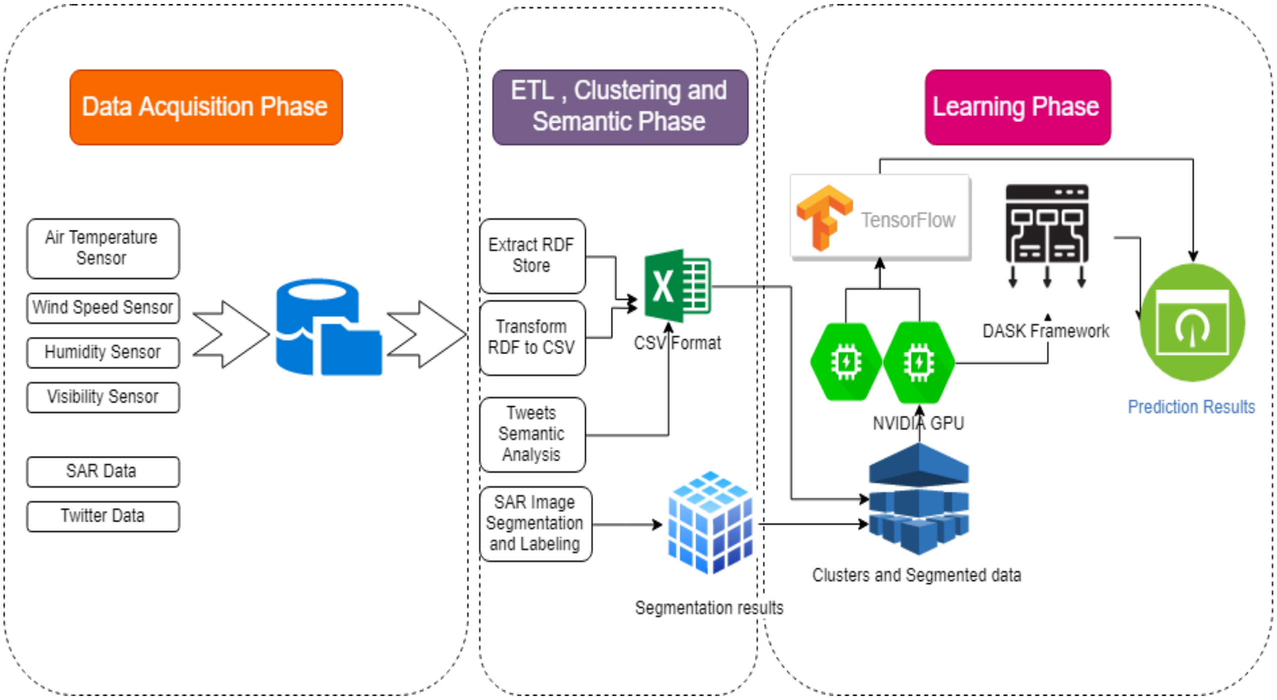


Figure 4: The proposed architecture of Distributed Decomposed Analytics of IoT, SAR and Social Network data

The proposed research design consists of three phases:

6.1. System Architecture Design:

6.1.1. Data Acquisition Phase:

The data is gathered from the various sources as discussed, this will be parsed and sent for stored locally. Sensors data is gathered over IoT network interface. SAR data from the Satellite network and the Social network data from Twitter API.

6.1.2. ETL, Clustering and Semantics Phase:

To cope with the interoperability problem, semantic Web solutions are used, such as modeling data using Resource Description Framework (RDF), Web Ontology Language (OWL) protocols, using standard data formats such as Turtle, N3, and JSON-LD. Location Specific Twitter feeds will be processed with semantic algorithms along for finding out specific phrases describing extreme events.

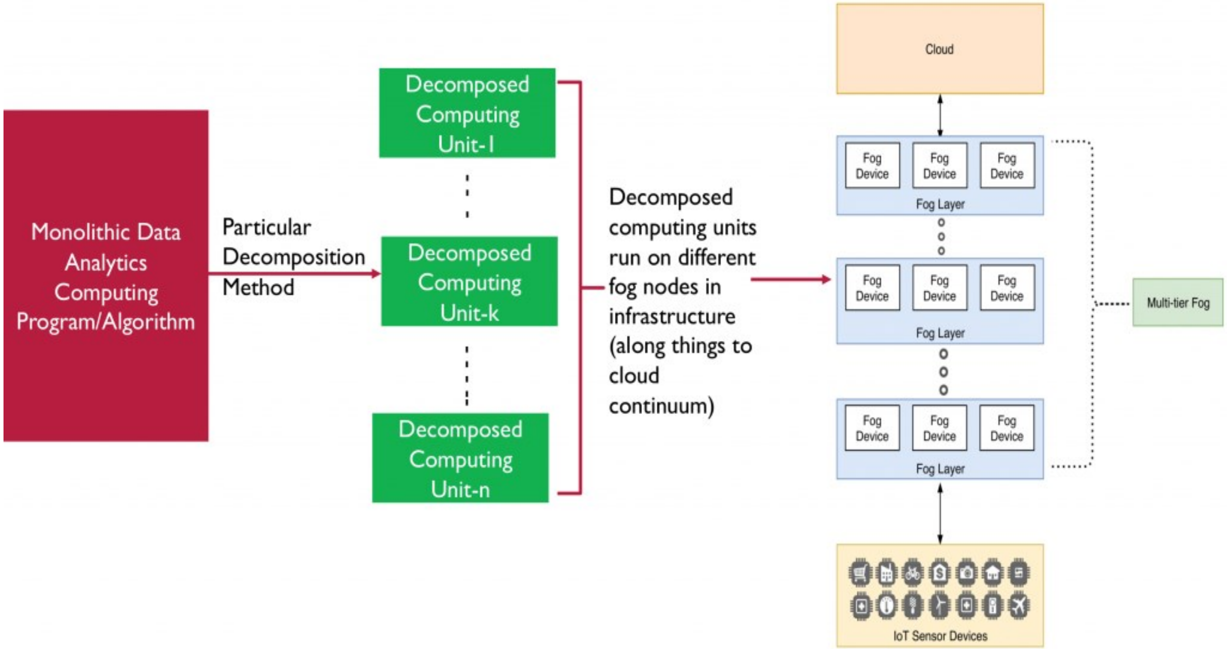
Further SAR images are processed with K-means clustering for and the clusters will be labeled. Our chosen learning model within the library is ScikitLearn based k-means clustering. We aim to cluster data to detect hidden information from a relatively complex dataset using our framework.

6.1.3. Learning Phase:

Learning algorithm will be implemented in Python on Scikit-learn, Pandas, Numpy and Dask [27] libraries and frameworks will be used. The Dask library supports the scaling on GPUs hence the performance improvement in multi GPU environment can be tested on Dask framework.

6.1.4. Distributed Decomposition of IoT Data:

In the existing approaches for data analytics in IoT, all data from an IoT deployment is collected at a centralized location such as server(s) in datacenter (i.e. cloud) and is then subjected to the desired data analytics model. Data in these IoT deployments moves from 'things' to cloud, and along this continuum passes through a number of network devices such as routers, gateways, etc. Each of these devices can be a potential candidate to host partial computing analytics capability to analyses the data, and further sending the calculated partial results instead of sending the raw data to cloud. The edge of the network in such deployments can act as a potential site to host what we call 'decomposed analytic computing units' (Figure 5) to reduce the amount of data being transferred to cloud, and provide the same quality of analytics results.



Monolithic Data Analytics Computing Program/Algorithm
Particular Decomposition Method
Decomposed Computing Unit-I
Decomposed Computing Unit-k
Decomposed Computing Unit-n
Decomposed computing units run on different fog nodes in infrastructure (along things to cloud continuum)
Cloud
Fog Device
Fog Device
Fog Device
Fog Layer
Fog Device
Fog Device
Fog Device
Fog Layer
Fog Device
Fog Device
Fog Device
Fog Layer
Multi-tier Fog
IoT Sensor Devices

Figure 5: Decomposition of data analytics computing program into decomposed computing units and placing those computing units onto different fog nodes in the infrastructure [31]

The justification for the above involves resource constraints. Contrary to the cloud which can be thought of as ‘resource rich’, the fog devices are resource constrained in nature whereby resource scaling (up/down and horizontal/vertical) cannot be done dynamically. The fog devices are already performing their fundamental computing/network operation (for e.g. in case of router as a fog device, it is already forwarding the packets to the set destination), so these operations are already utilizing the available resources (CPU, RAM and bandwidth) on it. An additional deployment of a complete data analytics computing program/algorithm on the said resource might lead to full utilization of resources on device as the workload or data input increases and also affect its fundamental network operation. Hence, the approach of decomposed computing units seems ideal in an IoT environment with fog assistance.

The predictive analytic model to be used is multivariate linear regression using statistical query model and summation form. Real world dataset from UCI repository will be used to test the performance of the proposed technique of decomposition.

6.1.5. SAR Data Analytics

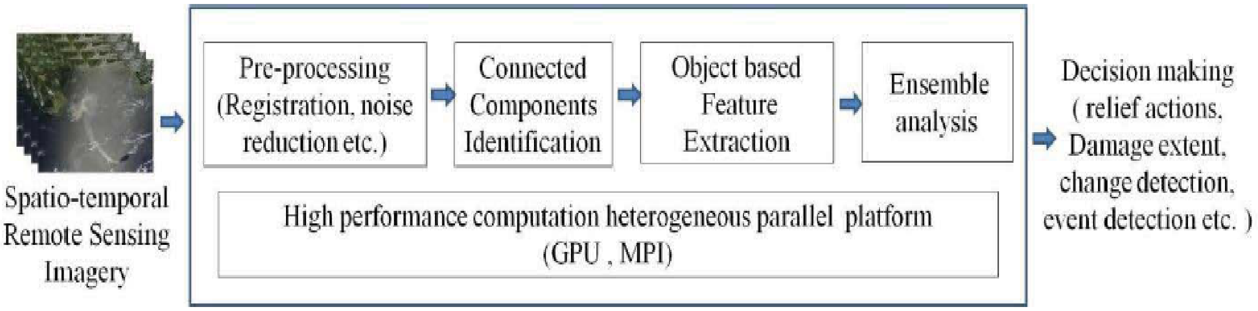

Spatio-temporal Remote Sensing Imagery
Pre-processing (Registration, noise reduction etc.)
Connected Components Identification
Object based Feature Extraction
Ensemble analysis
High performance computation heterogeneous parallel platform (GPU , MPI)
Decision making (relief actions, Damage extent, change detection, event detection etc.)

Figure 6: Big Data analysis framework for remote sensing data using parallel platforms.

It is experienced that, the image segmentation stage plays a central role in big data processing of RS data, which forms the basis for further analysis such as, feature extraction, classification and spatial relations extraction. However, for big RS archives (e.g. high-resolution RS archive) segmentation becomes a time-consuming process. Hence, a high-performance analytics approach (hybrid MPI+GPU) for image segmentation module is preferred.

Preprocessing such as noise reduction, gap filling etc. can be performed. For each image, Segmentation/connected components are identified. Further, from each component, relevant and robust features can be extracted, suitable to the type of disasters. Finally, ensemble analysis is required, which will include ensemble classification approaches to produce most accurate results to support decision making.

Features from each connected component are extracted in parallel using MPI technology as shown in figure 6. Root process is responsible for distributing the input data and uniform number of components at each core from each node. All cores work in parallel, extracts the features from the connected components assigned to it; after finishing with all components, each core sends back the extracted features to root process. Root process collects the features after finishing with all components, and store it at appropriate location of feature matrix, which holds all features of all connected components together.

6.1.6. Twitter sentiment Analysis

Now the methodology for the Sentiment Analysis of the twitter data will be discussed. The process has following stages.

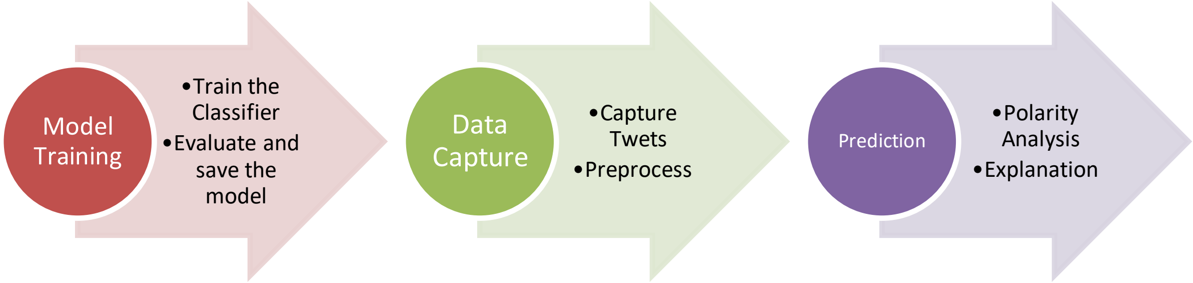


Figure 7: Sentiment Analysis: Training & Evaluation pipeline

Flair framework [32] is used here for sentiment analysis purpose. Flair allows combination of various kinds of word embeddings together; this results in greater contextual awareness of the classifier model.

Flair framework consist of contextualized representation called string embeddings at its core. Flair generates the character sequences by breaking the bigger sentences into smaller character sequences. This data is then fed to a pre-trained bidirectional language model for learning of the character level embeddings. Using this process, the model learns to identify the case-sensitive characters (for example, proper nouns from similar sounding common nouns) and other natural language patterns like syntactic patterns accurately. This process makes Flair framework capable for efficient named entity recognition (NER).

6.2. Conceptual Design

The conceptual model is as shown in Figure-8.

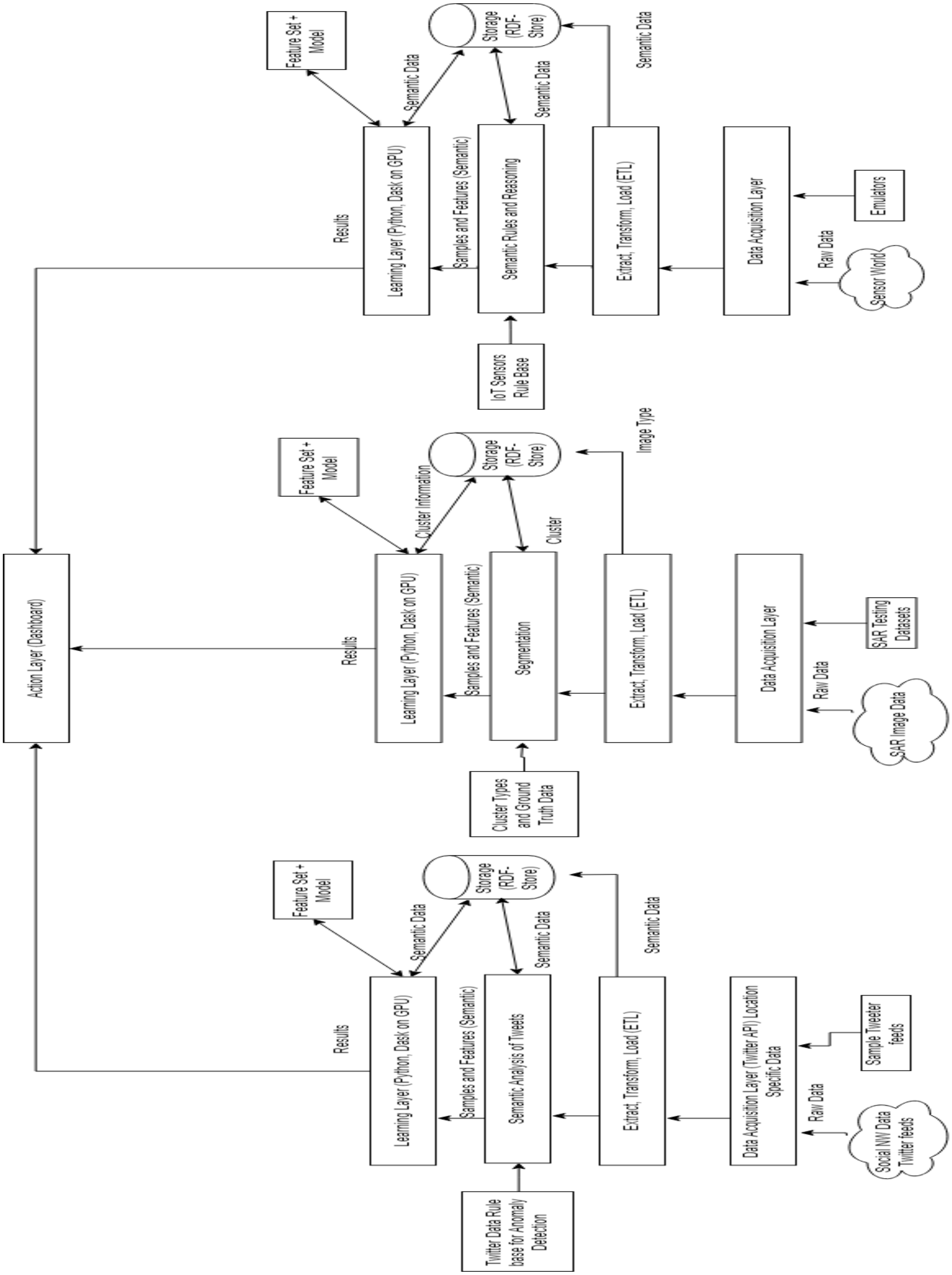

Feature Set + Model
Storage (RDF-Store)
Semantic Data
Semantic Data
Semantic Data
Results
Learning Layer (Python, Dask on GPU)
Samples and Features (Semantic)
Semantic Rules and Reasoning
Extract, Transform, Load (ETL)
Data Acquisition Layer
Emulators
Raw Data
Sensor World
IoT Sensors Rule Base
Feature Set + Model
Storage (RDF-Store)
Cluster Information
Cluster
Image Type
Results
Learning Layer (Python, Dask on GPU)
Samples and Features (Semantic)
Segmentation
Extract, Transform, Load (ETL)
Data Acquisition Layer
SAR Testing Datasets
Raw Data
SAR Image Data
Cluster Types and Ground Truth Data
Action Layer (Dashboard)
Feature Set + Model
Storage (RDF-Store)
Semantic Data
Semantic Data
Semantic Data
Results
Learning Layer (Python, Dask on GPU)
Samples and Features (Semantic)
Semantic Analysis of Tweets
Extract, Transform, Load (ETL)
Data Acquisition Layer (Twitter API) Location Specific Data
Sample Tweeter feeds
Raw Data
Social NW Data Twitter feeds
Twitter Data Rule base for Anomaly Detection

Figure 8: Proposed Conceptual Model of Distributed Decomposed Analytics of IoT, SAR and Social Network data

6.2.1. Data Acquisition Layer:

The first layer in the framework is data acquisition layer, which is responsible for collecting data from resources, more specifically sensors, twitter data and Remote Sensed Satellite Data from the outside world. It can be seen as input layer since the framework uses this layer to interact with sensors. The incoming data is raw data and the only task this layer accomplishes is acquiring and conveying raw data to ETL layer for processing. This layer does not touch its contents or parse it in any way. But, it makes sure that the data sent from sensors are not lost. To accomplish this robust data acquisition mechanisms, such as multi-threaded queues, should be employed.

6.2.2. ETL (Extract, Transform, Load) Layer:

The second layer in the framework is ETL (Extract, Transform, Load) layer. The incoming data from data acquisition layer is received by the ETL layer for parsing purposes. Since different kinds of input sources send different types and formats of data, ETL layer consists of compatible drivers for each input type to receive and parse data appropriately. For instance, a humidity sensor and temperature sensor may send data in different formats. Furthermore, each sensor driver is responsible for producing data in the right type, right unit, and format depending on the vendor, type, and version. For example, a temperature sensor from vendor A may produce data in Celsius unit while another sensor from vendor B may produce data in Fahrenheit unit. This should be differentiated in the platform in all layers. For this purpose, ETL layer is responsible for keeping data in the right type and format regardless of the sensor type via semantic technologies. Data is converted to a semantic format in the form of RDF (Resource Description Framework) protocol, the very basic semantic protocol to describe statements. At this point, artifacts from SSN ontology will be used along with our ontology constructs. The next step is to store this data in RDF format.

6.2.3. Semantic Analysis Layer:

The third layer is semantic-rule reasoning layer. This layer makes use of the data coming from ETL layer that is already in semantic form and properly parsed along with the domain specific parsing rules defined in drivers. The main purpose of reasoning layer is to designate the limits of the domain and make basic inferences from the RDF data using a reasoning engine embedded.

Two types of rules are executed on the semantic sensor data. The first is the semantic reasoning rules inherent to the semantic web protocols (RDF, RDFS, OWL). These are protocol or language specific rules, inferred automatically using the rule engine. The second is the domain specific or user specific rules.

Semantic analysis layer processes RDF data according the rules and produces inferred data. A CSV (Comma Separated Values) file containing resulting data is transferred to the learning layer.

6.2.4. Learning Layer:

The fourth layer is learning layer. This layer basically extracts features from the data and builds learning models by applying machine learning methods. This layer consists of two sub steps as preprocessing and learning. The features coming from semantic-rule reasoning layer can be excessive and using all of them without any preprocessing or filtering method can end up with low success rates in learning algorithms. Therefore, feature selection methods should be used to weed out irrelevant features. Principal Component Analysis or subset selection can be used for this task. After selecting the most relevant features, the next step is designating a learning algorithm. Various deep learning algorithms can be used and the most successful one can be selected or the result can be determined by combining various algorithms in a voting classifier style. Also, the learning layer does not need to know about domain specific rules.

6.2.5. Action Layer:

The last layer is action layer. The results that the learning layer produces will be evaluated and necessary actions will be taken in this layer. There will be predetermined actions for learning algorithm's output values that are defined by user.

7. System Evaluation

1.	**Distributed decomposed IoT analytics**		
	i	**Classification Accuracy**	It is the ratio of correct predictions of weather parameters (temperature, air pressure, humidity) to the total number of input samples.
	ii	**Confusion Matrix**	A matrix showing the performance of model used for classification.
2.	**Remote Sensing (RS) Data analytics**		
	i	**Segmentation Accuracy**	It is the percentage of pixels in the image which are correctly classified.
	ii	**Anomaly Detection Accuracy**	It is the percentage of correct identification of unexpected patterns at a location.
3.	**Twitter Polarity Accuracy**		It is a measure of how correctly the tweet has been identified and classified to its expected polarity.
4.	**Overall System evaluation metrics**		
	i	**Availability**	The time for which system is functioning properly without any failure.
	ii	**Reliability**	The system's capability to function under given environmental conditions, for a particular amount of time.
	iii	**Response Time**	The time system takes to classify the input.
	iv	**Latency**	The time system takes from submission of input to produce the classified output.
	v	**Throughput**	Number of requests handled per second.

Table 2 : System evaluation

8. Expected Outcomes

The research aims to create an integrated approach that will take the input from the IoT sensor data, remote sensed SAR data and live twitter feeds and process the analytics to produce meaningful prediction about anomalies in real time.

1. Data from various sources like IoT sensors, remote sensing SAR images and live twitter feeds will be fetched in real time.
2. Distributed decomposition of IoT analytics will be done with the help of fog nodes to offload big data analytics at cloud infrastructure.
3. Remote Sensed (RS) SAR data analytics will be done using MPI and emerging GPU technology to identify anomaly situation.
4. Twitter data will be segmented to extract specific phrases describing extreme events.
5. A learning model that learns from the gathered and processed data from above mentioned three sources to predict the extreme events or anomalies in real time.

9. Conclusion

In this proposal, a framework for distributed decomposed analytics of IoT, SAR and Social Network data is proposed. IoT data processing will be done in a distributed decomposed manner with the assistance of fog-based devices for basic analytics. This will ensure offloading of cloud based infrastructure. Parallel processing of huge amount of remote sensing data using multi-core GPUs will be managed using MPI platform for efficient feature extractions. Further, to improve accuracy, the sentiment analysis will be performed on the Twitter data using the Named Entity recognition. The location, hashtag and organization-based analysis can be further used for planning about the counter actions for specific events and further actions. This will result in fast and accurate system for meaningful information extraction for taking strategic decisions.

Paper publication on the research topic:

Dr. Vinayak Ashok Bharadi, Shashank Tolye presented a paper on **"Distributed Decomposed Data Analytics of IoT, SAR and Social Network data" at IEEE conference** of the **2020 3rd International Conference on Communication System, Computing and IT Applications (CSCITA) from 3rd to 4th April, 2020.**

References

[1] C. Perera, A. Zaslavsky, M. Compton, P. Christen, and D. Georgakopoulos, “Semantic-Driven Configuration of Internet of Things Middleware,” in 9th International Conference on Semantics, Knowledge and Grids, IEEE, 2013, pp. 66–73.

[2] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu, “Contextaware computing, learning, and big data in internet of things: A survey,” IEEE Internet of Things Journal, vol. 5, no. 1, pp. 1–27, 2018.

[3] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” Computer networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[4] D. Giusto, A. Iera, G. Morabito, L. Atzori (Eds.), “The Internet of Things, Springer, 2010”. ISBN: 978-1-4419-1673-0.

[5] Tiainen, P., New opportunities in electrical engineering as a result of the emergence of the Internet of Things. 2016.

[6] Mital, R., J. Coughlin, and M. Canaday. “Using Big Data Technologies and Analytics to Predict Sensor Anomalies”. in Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference, held in Wailea, Maui, Hawaii, September 15-18, 2014, Ed.: S. Ryan, The Maui Economic Development Board, id. 84. 2015.

[7] Golchha, N., Big Data–The information revolution. IJAR, 2015. 1(12): p. 791-794.

[8] Mohsen Marjani, Fariza Nasaruddin, Abdullah Gani, Ahmad Karim, Ibrahim Abaker Targio Hashem, Aisha Siddiqa, Ibrar Yaqoob, “Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges ”, 2169-3536 (c) 2016 IEEE

[9] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog Computing and Its Role in the Internet of Things, in: Proc. First Ed. MCC Workshop Mob. Cloud Comput., ACM, New York, NY, USA, 2012: pp. 13–16.doi:10.1145/2342509.2342513.

[10] M. Taneja, N. Jalodia, and A. Davy, “Distributed decomposed data analytics in fog enabled IoT deployments,” IEEE Access, vol. 7, pp. 40969–40981, 2019.

[11] Ta-Cheng Chang, Liang Zheng, Maria Gorlatova, Chege Gitau, Ching-Yao Huang, Mung Chiang, “Demo Abstract: Decomposing Data Analytics in Fog Networks”, SenSys’17, November 6–8, 2017, Delft, The Netherlands

[12] Amazon AWS. 2017. AWS Greengrass. (2017). http://aws.amazon.com/greengrass

[13] Microsoft Inc. 2017. Azure IoT Edge. (2017). http://github.com/Azure/iot-edge.

[14] M. Taneja and A. Davy, ``Poster abstract: Resource aware placement of data stream analytics operators on fog infrastructure for Internet of Things applications," in Proc. IEEE/ACM Symp. Edge Comput. (SEC), Oct. 2016, pp. 113114.

[15] Vasanthkumar Joshi, Dr. S. Manikandan, H. Srinivasaiah, “Overview of airborne SAR data processing alogorithms”, International Conference on Innovative Mechanisms for Industry Applications (ICIMIA 2017), 978-1-5090-5960-7, 2017 IEEE

[16] Digital processing of Synthetic aperture RADAR data Algorithm and Implementation By by Ian Cumming and Frank Wong, Artech House, Norwood, MA, January 2005

[17] Mohammed Hagras, Ghada Hassan, Nadine Farag, “Towards Natural Disasters Detection from Twitter using Topic Modelling”, 2017 European Conference on Electrical Engineering and Computer Science, DOI 10.1109/EECS.2017.57, 2017 IEEE

[18] Jain, Saloni, ”Real-time social network data mining for predicting the path for a disaster.” Thesis, Georgia State University, 2015.

[19] T. Sakaki, M. Okazaki and Y. Matsuo, "Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors," in WWW2010, Raleigh, 2010.

[20] Y. Raimond and S. Abdallah. The event ontology, 2007. http://motools.sf.net/event/event.html

[21] Bhuvaneswari Anbalagan, Dr. Valliyammai. C,"#ChennaiFloods: Leveraging Human and Machine Learning for Crisis Mapping during Disasters using Social Media", 2016 IEEE 23rd International Conference on High Performance Computing Workshops.

[22] Y. Bengio, "Deep Learning of Representations for Unsupervised and Transfer Learning," in Proceedings of the Unsupervised and Transfer Learning challenge and workshop, 2011

[23] Ajay Shrestha, Ausif Mahmood, "Review of Deep Learning Algorithms and Architectures", IEEE April 2019

[24] M. D. Buhmann, Radial Basis Functions. Cambridge, U.K.: Cambridge Univ. Press, 2003, p. 270.

[25] K. Chen, ``Deep and modular neural networks," in Springer Handbook of Computational Intelligence, J. Kacprzyk and W. Pedrycz, Eds. Berlin, Germany: Springer, 2015, pp. 473494

[26] Saad ALBAWI , Tareq Abed MOHAMMED, Saad AL-ZAWI, "Understanding of a Convolutional Neural Network", International Conference on Engineering and Technology (ICET), 2017, 978-1-5386-1949-0/17/©2017 IEEE

[27] Dask- a flexible library for parallel computing - https://docs.dask.org/en/latest/

[28] https://tssg.org/2019/06/publication-distributed-decomposed-data-analytics-in-fog-enabled-iot-deployments/

[29] UCI Machine Learning Repository: Air Quality Data Set. Accessed: Dec. 24, 2018. [Online]. Available: http://archive.ics.uci.edu/ ml/datasets/air+quality

[30] Ganea , O.; Hofmann, T.; Deep joint entity disambiguation with local neural attention, *In Proc. EMNLP, 2017*, pp. 2619-2629, 2017

[31] https://tssg.org/2019/06/13/publication-distributed-decomposed-data-analytics-in-fog-enabled-iot-deployments/

[32] Akbik, A; Bergmann, T; Blythe, D;Rasul, K.; Schweter,S.; Vollgraf, R; FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations) 2019, doi: 0.18653/v1/N19-4010, 54–59