A Novel Framework for Road Transportation Systems using Cloud Computing

Md. Rafeeq[1] , C. Sunil Kumar[2] , N. Subhash Chandra[3]

1. CMR Technical Campus, Kandlakoya,Medchal Road, Hyd, TS. India
2. SNIST, Yamnampet, Ghatkesar,Hyderabad, TS. India
3. CVR CE ,Manglapalli, Ibrahimpatnam R.R(D) TS. India

***Abstract* - Highly populated cities around the world are facing problems of better traffic management. Technologically advanced cities deploy agent based multiple multimedia sensor based networks to collect and analyse traffic data to provide better solutions for management and prediction of road conditions. However, the collected data reaches a high volume and became very difficult to manage. Moreover, the predictive analysis also demands a high computational power to run the predictive analysis algorithms. The data collections methods include automatic and manual collections of large amount of data and then the analysis is often done in legacy system or manually. Cloud providers are deploying large-scale data centres across the globe to meet Cloud customers' compute, storage and network resource demands, and better efficiency, scalability and performance of the hosted applications. The outcomes of this research also considers the most effective cloud based storage for traffic data with the knowledge of most popular cloud based storage service providers. The accumulation of the data is also followed by a predictive system for road traffic data analysis. Hence, in this work we also explore the use of standard machine learning techniques to identify the most suitable technique with performance consideration.**

Keywords - road traffic management; performance evaluation; cloud storage; erasure; framework components; framework functions.

I. INTRODUCTION

The recent research outcomes demonstrate the use of agent based sensor networks to accumulate the road traffic data. However a multipurpose framework for accumulating and managing the traffic data is still a demand. Hence this work demonstrates a mobility agent based network for road traffic data management. The outcomes of this research is also to consider the most effective cloud based storage for the traffic data with the knowledge of most popular cloud based storage service providers. The accumulation of the data is also followed by the predictive system for road traffic data analysis[1].

With the tremendous growth of population and the increasing road traffic, the demand for optimized traffic data collection and management framework is also increasing. The collection of traffic data using multiple sensors and other capture devices are been addressed in multiple researches deploying the mechanism using geodetically static sensor agents. However the sheath factors for the parallel research outcomes have significantly ignored the fact of data replication control during processing. This work proposes a novel framework for capturing and storage of traffic data. During the multi node traffic data analysis, controlling the replication in order to reduce the cost is also been a challenge. This work also addresses this problem using Erasure encoding for low cost replication [2].

The extensive practical demonstrations of virtualization and migration benefits are also carried out in this work. With the extensive experimental setup the work furnishes the comparative analysis of simulations for popular existing techniques and the proposed framework.

Data collected from more than 5000 production servers over a six-month period have shown that although servers usually are not idle, the utilization rarely approaches 100% [3]. Most of the time the server is only 10-50% utilized leading to wastage of energy. Even a completely idle server consumes 70% of its peak power [4]. So keeping the server underutilized is inefficient from energy saving point of view. Moreover for each watt of power consumed by computing resources, additional 0.5-1 W is required for the cooling system [5]. The high energy consumption also leads to release of green house gas CO_2 [6]. So, this wastage of energy is not only cost ineffective but also harmful to the environment.

II. PROBLEM STATEMENT

Across the world ranging from city to urban the tremendous growth of road traffic is leading towards a major problem. The highly populated cities around the world are facing the problem of better traffic management. The technologically advanced cities deploy agent based multiple multimedia censor based networks to collect and analyze the traffic data to provide better solutions for management and prediction of the road conditions.

Advanced sensing and surveillance technologies often collect traffic information with high temporal and spatial resolutions. The volume of the collected data severely limits the scalability of online traffic operations. To overcome this issue, we propose a low-dimensional network representation where only a subset of road segments is explicitly monitored. Traffic information for the subset of roads is then used to estimate and predict conditions of the entire network. Numerical results show that such approach provides 10 times faster prediction at a loss of performance of 3% and 1% for 5- and 30-min prediction horizons, respectively.

Hence considering the situations and identified the following problems to be addressed in the current traffic management situations:

- Management of Agent based Censor network to implement a low cost infrastructure and normalization of the data under pre-processing.
- Identification of most suitable algorithm and propose an optimal VM Migration Technique for Load Balancing.
- Comparison and identification of most suitable cloud storage architecture for replication of data considering the low cost Erasure models.

III. PROPOSED ROAD TRAFFIC DATA MANAGEMENT FRAMEWORK

Hence forth with the detailed understanding of the generic traffic data monitoring and management framework, we propose the novel framework for road traffic data management control with replication control on cloud storage. In the proposed framework we have considered the layer based approach for better controlling and management of the agent based components.

The agents in the wireless network are single function oriented but the collective network is multi-purpose. In this study we propose the framework consisting of deployed network layer, monitoring layer, application management layer, storage layer and finally the server based server layer [Figure-2].

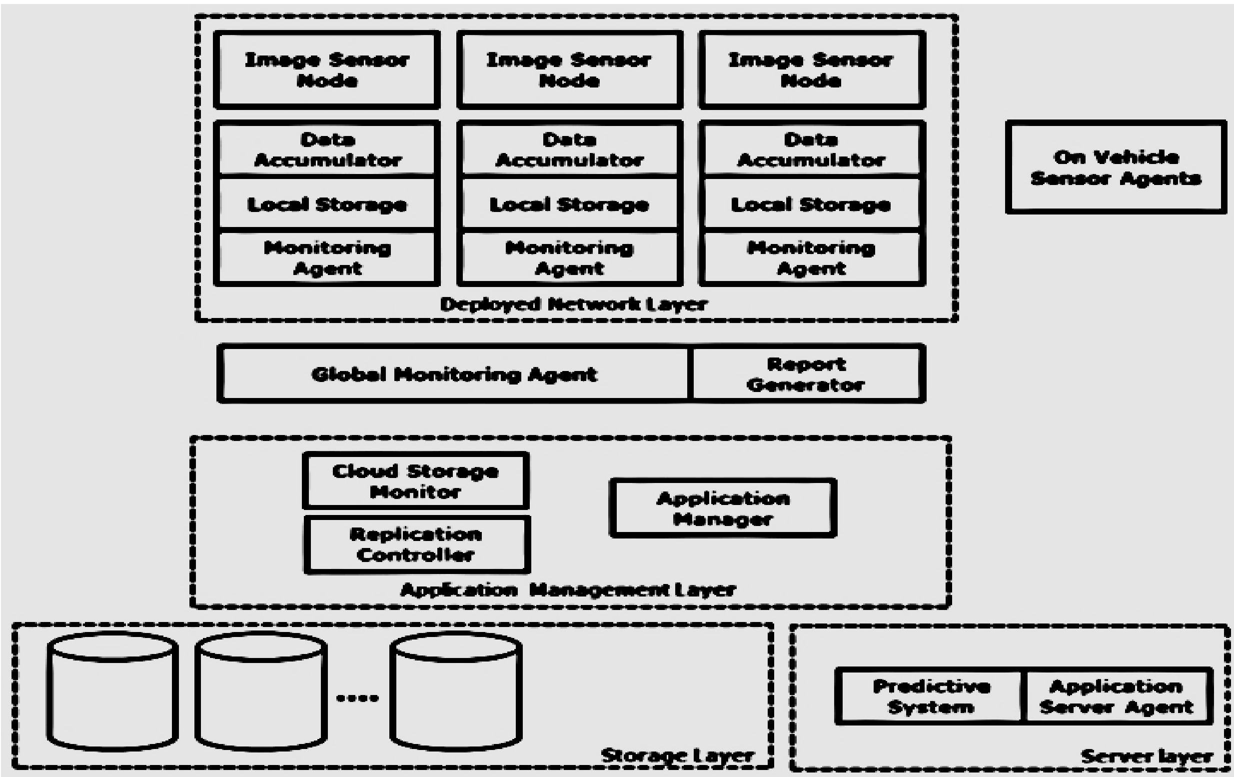

Figure – 1: Proposed Framework for Road Traffic Data Management with replication control

The components of all layers in the network are:

A. Network Layer: The First and onsite layer of this proposed framework is the network layer. The layer includes the agents required to collect the traffic data, monitor the data collection with connectivity and the significantly small amount of local storage. The deployed layer also includes the on vehicle sensor agents. The components of the network layer are:

A1. Image Sensor Node: The imaging agents installed on the site are connected to the network with high performance capturing capabilities. The imaging agents are also capable of functioning in various weather conditions. Here we propose the optimal configuration considering the power and performance balance.

The capture devices must satisfy the same configuration scheme to reproduce the same performance.

A2. Data Accumulator Agent: The programmable data accumulator agents are configured to capture the image and video data. The optimizer components into these agents are configured to capture the video or image data while the motion in the object is detected. This makes the captured data optimized in size.

The algorithm used for motion detection for multiple objects consider each object as collective vector object as O_t at any given time instance t. The collection of the detectable features based on the time instance are noted as $F_1, F_2....F_t$. The algorithm is deploying a probabilistic function to estimate the next state of the specified object denoted as $P(O_t \mid F_1, F_2, ...F_t)$. The outcome of this probabilistic function is the next possible coordinates for the object under motion tacking. Further the probabilistic function at any given time is considered to function on a weighted and time depended sample set of video data. The sample set is defined as $\{S_t^{(v)}, v = 1, ...V\}$ with the considered weight for corrected predictive location is $\pi_t^{(v)}$. The V parameter denotes the number of video data in the selected data set.

Henceforth we understand the relation between the learning features and the probabilistic prediction as:

o Firstly we consider the sample data set with V number of video data as

$$\{S_0^{(v)}, v = 1, ...V\} \quad \text{...Eq. 1}$$

o The weight component for enhanced learning is considered as

$$\{\pi_0^{(v)}, v = 1, ...V\} \quad \text{...Eq. 2}$$

o Hence, the final probabilistic function for next location determination is

$$P(O_t \mid F_1, F_2, ...F_t) \quad \text{...Eq. 3}$$

o Henceforth we consider the change in enhanced learning factor as

$$P(O_t \mid O_{t-1}) \quad \text{...Eq. 4}$$

o The next step is to apply the Eq. 4 on Eq. 1and obtain a new dataset for prediction and obtain $\{S_t^{(v)}\}$.

Hence the final step is to consider the updated feature tracking for iterative calculation of the motion as

$$\pi_t^{(v)} = \frac{P(F_t \mid S^{(v)})}{\sum_{i=1}^{V} P(F_t \mid S^i)} \quad \text{...Eq. 5}$$

TABLE I: DATA COLLECTION ALGORITHM

Proposed Algorithm 1: Data Collection through Capture Agents
Step-1. Accumulation of the Video Data
Step-2. Classification of Video Data into Recognizable frames
Step-3. Extraction of the objects from each frame
Step-4. For all objects in each frame a. Analyse the next possible location b. Predict the new location of the object c. Update the learning weight d. Update the feature tracking matrix
Step-5. Update the local storage
Step-6. Determine the change in the local data
Step-7. Update the incremental change in the cloud based data centre storage

A3. Local Storage: The local storage container is designed to store data locally from the capture agents in case of loss of connectivity between the network layer and other layers. The local storage containers are desired to perform the on board compression and store small amount of data. However the storage containers are expected to deliver the low latency. Here we propose the factors to be considered for a local storage to balance the performance and cost implications [Table II].

TABLE II: IMAGE SENSOR NODE CONFIGURATION

Parameter Type	Proposed Optimal Value
Technology	ZIP or JAZ or USB
Preferred Mode of Operation	No or Less Mechanical Parts
Read Speed	20 to 200 MBPS
Write Speed	30 to 280 MBPS
Form Factor	2.5 to 3.5
Storage Capacity	5 to 12 GB
Supported Storage Format	JPEG

A4. Monitoring and Connectivity Agent: The monitoring and connectivity agents are deployed to monitor inter and intra connectivity of the network layer.

On Vehicle Sensor Agents (Optional): The optional components of the network layer are the on-vehicle sensor components for better performance of this framework. However these components are not compulsory in this proposed framework. Here we list the sensors:

a) Positional Sensor for Latitude and Longitude information

b) Speed Sensor for acceleration and deceleration information

c) Temperature Sensor

d) Obstacle Detection Sensor

B. Monitoring Layer: The monitoring layer consists of two major components as Global monitoring agent and report generator. The global monitoring agent works same as the local monitoring agent with the enhanced performance to manage and detect the connectivity and

DOI 10.5013/IJSSST.a.19.04.19 ISSN: 1473-804x online, 1473-8031 print

data transmission rates. The report generator is a programmable software packet on the network to filter the data and generate reports based data for the queries generated from the server components.

C. Application Management Layer: The application management layer is equipped with three components as cloud storage monitor, replication controller and application protocol manager for higher management and performance enhancement of the complete framework. The cloud storage monitor is the software agent to analyse and keep track of the data storage meter for cost and scalability control for the framework data. However the framework data is also minimized in the previous layers by data accumulator and global monitoring agents.

TABLE III. SYSTEM FUNCTIONALITIES

Feature ID	Feature Description
UI – 1	The system providers the feature to graphically provide information from the user interface
UI – 2	The system provides the user interface to manage the system view for complete observation
UI – 3	The system provides the visual interpretation of the system on click for every node in the traffic model
STAT–1	The system provides the statistics related to the traffic monitoring
STAT–2	The system also provides the statistics related to each vehicle registered into the framework
ST–1	The system provides the in detail view of the storage system overall estimation
PRED–1	The system connects with the predictive analysis component of the framework
PRED–2	The system manages all predictive query and reports
FUNC–1	The performance of the system is at least equal or greater to the real time data capture speed
FUNC–2	The system is in sync always with the traffic lights deployed on the road
FUNC–3	The system provides the on road vehicle detail statistics like travel time and all
FUNC–4	The system is also capable of providing multi-lane road traffic prediction

The replication controller provides the assurance of low cost replication of data for fault tolerance and timely recovery. The mechanism employs an effective erasure mechanism called Reed Solomon framework. The Erasure framework is discussed in the later part of this work.

The Application Manager agent is deployed to maintain the software component protocols and ensure the security aspects of the data during application execution.

D. Storage Layer: The storage layer is the third part stack of the storage solutions based on cloud service providers. This work also considers the selection parameters of a cloud storage solution for cost effective storage. The detail of the storage solutions comparative study is discussed in the later part of this work.

E. Server Layer: The server layer is comprised of two major components as application server agent and the predictive system for traffic prediction. Here we discuss the components:

- Application Server Agent: The application server agent is an application for analysing the traffic data for making the data ready for predictive system module. The features of the system are also described in this work [Table III].
- Predictive System: Another component of the framework is the predictive analytic system. This component provides the system prediction for the queries generated from the application server agent. In the later part of the work we analyse the functionalities of this module with the detail understanding of road traffic data machine learning techniques. The collection of road traffic data is completely depended on the deployed sensor nodes. However the data collection is also depends on the parameters based on which the traffic analysis is done. Here in this work we analyse the most effective parameters for traffic data and apply the processing techniques based on the proposed framework [Table IV].

TABLE IV. TRAFFIC DATA COLLECTION PARAMETERS

Parameter ID	Parameter Description
Link ID	The ID of the connector node between two or more roads
Link Name	The Name of the connector node between two or more roads
Road ID	The ID of the road. Every road in the network will be assigned an unique id
Start Link	The first link of the map from which the proposed framework is starting
End Link	The last link of the map from which the proposed framework is starting
Link Description	The number of roads connecting to the same link
Road Name	The geographic name of the road. This name is the same name assigned to the road connected with locality and PIN Code
Road Length	The end to end length of the road as per the geographic measurement
Total Traffic Flow	The number of vehicles detected at any specified link in a given time slot n
Traffic Flow type – 1	The flow of the traffic for small or type – 1 vehicles
Traffic Flow type – 2	The flow of the traffic for medium or type – 2 vehicles
Traffic Flow type – 3	The flow of the traffic for big or type – 3 vehicles

Apart from the generic parameters, the system is also equipped with some optional parameters collections based on the on board sensors with the vehicles.

Once the vehicles are registered with the framework, the system will start collecting the optional parameters [Table V].

Here we present the data collected from a sample run of the system over duration of 30 mins for the sample configured map with the simulation support of NetSIM [Table VI].

TABLE V. OPTIONAL DATA COLLECTION PARAMETERS

Parameter ID	Parameter Description
Travel Time	The total duration of the trip in the configured map
Average Travel Time	The average travel time for the vehicle for a time slot of t
Average Speed	The average speed of the vehicle over a time slot of t
Quality Index	The number of valid data connections from the vehicle to the framework

TABLE VI. SAMPLE DATA COLLECTED FROM THE PROPOSED FRAMEWORK

Link ID	Link Name	Road ID	Start Link	End Link	Link Description	Road Name	Road Length	Traffic Flow type – 1	Traffic Flow type – 2	Traffic Flow type – 3	Total Traffic Flow
28	Node-28	37	Node-1	Node-99	3	Road-37	244	5	3	5	9
35	Node-35	45	Node-1	Node-99	5	Road-45	204	5	5	3	9
12	Node-12	20	Node-1	Node-99	2	Road-20	190	5	4	4	11
11	Node-11	18	Node-1	Node-99	2	Road-18	104	4	5	4	10
31	Node-31	35	Node-1	Node-99	4	Road-35	165	4	5	5	9
34	Node-34	36	Node-1	Node-99	4	Road-36	265	3	4	3	10
15	Node-15	21	Node-1	Node-99	2	Road-21	235	3	5	4	11
24	Node-24	27	Node-1	Node-99	3	Road-27	113	3	5	3	11
23	Node-23	25	Node-1	Node-99	4	Road-25	282	5	4	3	11
24	Node-24	32	Node-1	Node-99	2	Road-32	106	3	4	3	11
10	Node-10	17	Node-1	Node-99	2	Road-17	182	4	5	5	10
27	Node-27	37	Node-1	Node-99	2	Road-37	115	4	3	3	9
19	Node-19	25	Node-1	Node-99	3	Road-25	240	3	5	4	9
37	Node-37	46	Node-1	Node-99	5	Road-46	116	3	5	4	10
27	Node-27	28	Node-1	Node-99	3	Road-28	288	3	4	3	11
16	Node-16	25	Node-1	Node-99	2	Road-25	128	3	5	3	9

F. Performance Evaluation Matrix: The evaluation of the road traffic monitoring systems is continuous. The outcomes from the parallel researches have demonstrated availability of the number of application and frameworks for the same purpose. The performance of the road traffic monitoring and management systems need to be measured on a common platform as performance evaluation matrix, however the availability of the performance evaluation matrix is still thrive for the present research trends.

Hence in this work we also propose the performance evaluation matrix with various properties ranging from dash board or monitoring to I/O device performances [Table VII on next page].

Henceforth we apply the same performance evaluation matrix for evaluating the performance of the proposed framework.

IV. DATA REPLICATION CONTROL USING ERASURE

The proposed framework also proposes the failure recovery by employing the erasure replication control over the cloud storage services. Hence in this part of the work, we understand the Erasure code for cost effective replication control of the road traffic data. The most effective and popular framework under Erasure Coding is Reed-Solomon framework.

The most important factor that makes Reed-Solomon framework to implement is the simplicity. Here in this work we consider the scenario to compare the performance of Reed – Solomon and Proposed Encoding technique [8-12].

We consider there will be K storage devices each hold n bytes of data such that,

$$D = \sum D_1, D_2.D_3.....D_k \quad \text{…Eq 6}$$

Where D is the collection of storage devices

Also there will be L storage devices each hold n bytes of check sum data such that,

$$C = \sum C_1, C_2, C_3....C_L \quad \text{…Eq 7}$$

Where C is the collection of Checksum devices. The checksum devices will hold the calculated values from each respective data storage devices.

The goal is to restore the values if any device from the C collection fails using the non – failed devices.

The Reed – Solomon deploys a function G in order to calculate the checksum content for every device in C. Here for this study we understand the example of the calculation with the values as K = 8 and L = 2 for the devices C1and C2 with G1 and G2 respectively [9].

TABLE VII: PERFORMANCE EVALUATION MATRIX

Type	Name	Description
Dash Board Parameters (Over All Monitoring)	Name of Node	Unique Name of the Node
	Type	Architecture Type
	Overall Health Indicator	Running, Stopped, Critical
	Last Accessed	Last Accessed Date and Time
	Total Availability	Time of Total Availability
	Memory Utilization	Total Local Sensor Memory Utilization
	Disk Utilization	Total Local Disk Utilization in MB
	Network Utilization	Total Network Utilization in time
Memory Parameters	Active Memory	Amount of Active Memory in KB
	Over heading Memory	Amount of Over heading Memory in KB
	Swappable Memory	Amount of Swappable Memory in KB
	Total Shared Memory	Amount of Total Shared Memory
	Memory Temperature	Temperature of the Memory Units
Storage Parameters	Container Name	Unique name of the Storage Container
	Container Size	Container Size in GB
	Container Utilization	Container Utilization in GB
Type	**Name**	**Description**
Networking Parameters	Network Device ID	Unique id for the Network Devices
	Up Time	Total Up Time
	Down Time	Total Down Time
	IP Address	Unique assigned IP Address
	MAC Address	Unique assigned MAC Address
	Data Transfer Rate	Data Transfer Rate in Megabytes per second
Peripheral Parameters (I/O Device)	Device ID	Unique Device ID
	Type	Read or Write
	Read Count	Number of Read Operations
	Write Count	Number of Write Operations

The core functionalities of Reed – Solomon is to break the collection of storage devices in number of words [10] [11]. Here in this example we understand the each number of words is of u bits randomly. Hence the words in each device can be assumed as v, where v is defined as:

$$v = (nbytes).\left(\frac{8bits}{byte}\right).\left(\frac{1word}{u\,Bits}\right) \quad \text{… Eq 8}$$

Furthermore, v is defined as

$$V = \frac{8n}{u} \quad \text{…Eq 9}$$

Henceforth, we understand the formulation for checksum for each storage device as

$$C_i = W_i.(D_1, D_2, D_3 ... D_k) \quad \text{…Eq 10}$$

Where the coding function W is defined to operate on each word

After the detail understanding of the Erasure fault tolerance scheme, we have identified the limitations of the applicability to the cloud storage services and propose the novel scheme for fault tolerance in this work in the next section.

The framework can be applied in case of $n \leq 2^z$, where n denotes number of disks and z denotes number of customer data. To understand the framework for 256 storage containers or disks are considered. For a 256 disks, a Reed – Solomon code can be defined and implemented using Galois Field Arithmetic or GF (2^8). The coefficient "a" can be defined in various ways. The basic implementation of Reed – Solomon is Couchy construction. To understand Couchy construction, we select any n unique numbers in the space of GF (2^Z) [10].

Hence the selected n number are distributed in two sets called X and Y, where X contains m elements and Y contains k elements. Hence:

$$a(_{i,j}) = \frac{1}{x_i \oplus y_j} \text{ with the help of GF } (2^z) \quad \text{….Eq 11}$$

The most important factor that makes Reed-Solomon framework to implement is the simplicity. In this framework selecting k and m is random and does not depend on any factors and can be selected independently.

The performance can be questioned as the time complexity for performing an XOR operating is less compared to GF. However the modern processors rely on vector instruction sets for performing array based multiplication operation. Hence the reduction in time for computation can be achieved. Moreover with the improvement of latency time for the I/O devices and cache memory is also been improving to match with the highly complex Erasure Codes. The implementation of Reed – Solomon is simple as many open source solutions are readily available for storage solutions [9-12].

A. Machine Learning Techniques for Road Traffic Data: The most effective functionalities of a road traffic management system also may include the predictive analysis system as an alternative to the guided navigation system. The prediction system must be enabled to predict the road traffic conditions specially the congestion control [8]. Hence in this work we understand the most popular machine learning predictive approaches for road traffic data analysis (continued after table VIII below):

TABLE VIII. PROPERTY AVAILABILITY MATRIX

Parameter Observation	Availability of the Properties		
	Test Process – 1 (Duration – 60 Mins)	Test Process – 2 (Duration – 90 Mins)	Test Process – 3 (Duration – 200 Mins)
Name of Node	Available	Available	Available
Type	Available	Available	Available
Overall Health Indicator	Available	Available	Available
Last Accessed	Available	No Continuous Availability	Available
Total Availability	Available	Available	No Continuous Availability
Memory Utilization	Available	Available	Available
Disk Utilization	Available	Available	Available
Network Utilization	Available	No Continuous Availability	No Continuous Availability
Active Memory	Available	No Continuous Availability	No Continuous Availability
Over heading Memory	Available	Available	No Continuous Availability
Swappable Memory	Available	Available	Available
Total Shared Memory	Available	No Continuous Availability	No Continuous Availability
Memory Temperature	Available	Available	Available
Container Name	Available	Available	Available
Container Size	Available	Available	Available
Container Utilization	Available	Available	Available
Network Device ID	Available	Available	Available
Up Time	Available	Available	No Continuous Availability
Down Time	Available	Available	No Continuous Availability
IP Address	Available	Available	Available
MAC Address	Available	Available	Available
Data Transfer Rate	Available	Available	Available
Device ID	Available	Available	Available
Type	Available	Available	Available
Read Count	Available	Available	No Continuous Availability
Write Count	Available	Available	No Continuous Availability

B. Gaussian Naïve Bayes: In case of continuous parameter evaluation of number of parameters, the Gaussian Naïve Bayes probabilistic solution plays the major role. The Gaussian Naïve Bayes is formulated and understood here:

$$P(x=v\,|\,c)=\frac{1}{\sqrt{2\Pi\sigma^2{}_c}}e^{-\frac{(v-\mu_c)^2}{2\sigma^2{}_c}} \quad \text{…. Eq 7}$$

Where,

$P(x=v\,|\,c)$ is defined as any class for distribution

x denotes the continuous attributes used for the classification

μ_c is defined as mean value of the attributes

$\sigma^2{}_c$ is defined as variance of the attributes

-*C. Multinomial Naïve Bayes:* In case of the event generation probabilistic models, the Multinomial Naïve Bayes probabilistic model is widely used. Here the Multinomial Naïve Bayes is formulated and understood:

$$P(x\,|\,C_K)=\frac{(\sum_i x_i)!}{\prod_i x_i!}\coprod_i P_{k_i}{}^{x^i} \quad \text{…. Eq 8}$$

Where,

P denotes the probability of any event to occur or K number of multinomial cases in multiple event classes. X denotes the number of times any event I took place in the space.

Bernoulli Naïve Bayes: In case of the document or generic text classification the use of Bernoulli Naïve Bayes is highly adopted. Here we formulate and understand the Bernoulli Naïve Bayes probabilistic model:

$$P(x\,|\,C_K)=\prod_{i=1}^{n}P_{k_i}{}^{x_i}(1-P_{k_i})^{(1-x_i)} \quad \text{… Eq 9}$$

Where,

P_{k_i} denotes the probability for any class C_K producing any term.

x_i denotes the probability of producing any term from the possible set.

The above mentioned mining techniques are widely employed by multiple researchers in the past and recent researches. Henceforth this understanding will help us to extend the work of proposed framework by proposing the traffic prediction system [2-4].

V. RESULTS AND DISCUSSION

The results of simulation testing for the proposed framework under the NetSIM tool are satisfactory. The framework is been tested on the parameters proposed in the performance evaluation matrix [Table VIII on last page].

The results demonstrates the most effective and sustainable nature of the framework. However the due to the network congestion during the data transmission, it is been observed that the availability of the parameter values are seems not to be available for longer runs. Hence this issue will be addressed in the further research for this work.

VI. CONCLUSION

This works considers the parallel research outcomes of recent time and find multiple approaches to build a scalable and sustainable framework for traffic monitoring system. Based on the parallel research outcomes we understand the focus area of this research. We also understand the generic components of any road traffic management and monitoring system consisting of Sensor Nodes, Data Collector, and Monitoring agents, Storage Controller, Information Server and Application Server. Based on this understanding of the application framework, we propose the novel proposed framework consisting of Image Sensor Nodes, Data Accumulator, Local Storage, Local Monitoring Agent, Global Monitoring agent, Report generator, cloud storage monitor, replication controller, application manager, cloud storage solutions and Application server program. For the proposed framework we also build the application program for monitoring. The functionalities are listed in this work. During the framework building process, we have also demonstrated the optimal data collection process for the proposed framework. This work also proposes the performance evaluation matrix for evaluating the performance of any generic road traffic management and monitoring system. For the storage of the traffic data, we propose the cloud storage solutions with Erasure replication control and the comparative study of major cloud storage specialist service providers are also been carried out in this work. With the consideration of the results, we understand the proposed framework is highly sustainable and reliable. However the network data congestion demands further research. This work does not consider the predictive analysis of the traffic data and will be considering it for the future research on the same framework.

ACKNOWLEDGEMENT

The authors wish to express their sincere thanks to Dr. C. Sunil Kumar Supervisor and Dr. N. Subhash Chandra Co-Supervisor for constant help and guidance in the PhD work and also to the Organization CMR Technical Campus for encouragement in the research.

REFERENCES

[1] J. Lee and F. Mannering, “Impact of roadside features on the frequency and severity of run-off-roadway accidents: An empirical analysis,” Accid. Anal. Prev., vol. 34, no. 2, pp. 149–161, Mar. 2012.

[2] Anton Beloglazov∗ and Rajkumar Buyya, “Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers concurrency and computation: practice and experience, concurrency computat.: Pract. Exper. 2011; 00:1–24

[3] Guangjie Han 1,*, Wenhui Que 1 , Gangyong Jia 2 and Lei Shu 3, " An Efficient Virtual Machine Consolidation Scheme for Multimedia Cloud Computing " , Sensors 2016, 16, 246; doi:10.3390/s16020246

[4] Y. Su, “Behavior analysis of urban traffic participants and intervention measures,” Ph.D. dissertation, Dept. Autom., Tsinghua Univ., Beijing, China, 2009.

[5] J.R. Douceur, J. Howell, J.R. Lorch, M. Theimer, and R. Wattenhofer, “Farsite: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment,” Proc. Fifth Symp. Operating System Design and Implementation, (OSDI), pp. 1-14, 2002.

[6] A. Haeberlen, A. Mislove, and P. Druschel, “Glacier: Highly Durable, Decentralized Storage Despite Massive Correlated Failures,” Proc. Second Symp. Networked Systems Design and Implementation (NSDI), pp. 143-158, 2005.

[7] Q. Tang, “Type-Based Proxy Re-Encryption and Its Construction,” Proc. Ninth Int’l Conf. Cryptology in India: Progress in Cryptology (INDOCRYPT), pp. 130-144, 2008.

[8] G. Ateniese, K. Benson, and S. Hohenberger, “Key-Private Proxy Re-Encryption,” Proc. Topics in Cryptology (CT-RSA), pp. 279-294, 2009.

[9] J. Shao and Z. Cao, “CCA-Secure Proxy Re-Encryption without Pairings,” Proc. 12th Int’l Conf. Practice and Theory in Public Key Cryptography (PKC), pp. 357-376, 2009.

[10] J. Bellorado and A. Kavcic , "Low-complexity soft-decoding algorithms for Reed-Solomon codes: An algebraic soft-in hardout chase decoder" , IEEE Trans. Inf. Theory. , vol. 56 , no. 3 , pp.945 -959 , 2010

[11] F. Garc-Herrero, J. Valls and P. K. Meher , "High-speed RS (255, 239) decoder based on LCC decoding" , Circuits Syst. Signal Process. , vol. 30 , no. 6 , pp.1643 -1669, 2011.

[12] W. Zhang, H. Wang and B. Pan , "Reduced-complexity LCC Reed-Solomon decoder based on unified syndrome computation" , IEEE Trans. Very Large Scale Integr. (VLSI) Syst. , vol. 21 , no. 5 , pp.974 -978 , 2014.