Asian Journal of Information Technology 15 (24): 5106-5119, 2016
ISSN: 1682-3915

Metropolitan Traffic Research and Analysis

Kumar Abhishek, Rohit Kashyap, Divya Upadhyay and M.P. Singh
Department of Computer Science and Engineering, National Institute of Technology,
Patna, India

Abstract: Metropolitan traffic research and analysis is a heuristic based system that analyses traffic congestion in metropolitan cities of India like Delhi using input from twitter feeds. It provides information regarding the level of congestion between source and destination in a route and correlations among events that cause traffic. The inputs from cyber social system are analyzed according to different locations of the city listed in a separate record. From the data collected, we extract crucial information pertaining to analysis of congestion patterns. Using the obtained results, we analyze the traffic and generate a heuristic based opinion. Bayesian networks and relevant histograms are used for finding out the relationships and patterns among variables involved and to answer probabilistic queries about them and further for graphical representation of evaluated patterns.

Key words: Cyber-social system, event extraction, event correlation, twitter mining, geo-fencing

INTRODUCTION

Motorized transportation system is undoubtedly one of the biggest revolutions of the twentieth century. It made commuting from one place to another an easy, convenient yet a controlled mechanism with its ever so complex evolution all over the world. However, as the number of vehicles increased and traffic systems became more and more complex, the necessity of monitoring it clearly became important and crucial. To address this need, numerous systems have been proposed by researchers in the past few decades. Many countries have started implementation of such systems from quite early on and have made them an integral part of their overall urban planning. However, for developing countries, who are beginning to adopt such systems, challenges are galore for wide-scale implementation of such systems.

Probabilistic Graphical Model like Bayesian network allow us to handle dynamism, uncertainty and incompleteness in multiple domains. These data built models ignore declarative knowledge which is already existing about the domain such as Linked Open Data (LOD) (Yu, 2011) and ontologies which exist online. In this paper, an approach is presented to leverage such "top -down " domain knowledge to enhance "bottom-up "construction of graphical models.Specifically,the operations on the graphical model s tructure is enriched with nodes, edges and edge directions. Enrichment process is done using traffic data obtained from the social stream of official twitter accounts such as @dtptraffic (official twitter handle of Delhi Traffic Police). Resulting model obtained can lead to better predictions of traffic delays. India being a developing nation faces an acute problem of traffic jams and clogging on its existing crowded road infrastructure. Lack of a proper tracking and monitoring system aggravates this issue and necessitates the need of a framework that can track real time traffic updates to inform commuters in making their travel decisions based on latest movement updates sourced from physical cyber social systems.

New Delhi, the capital city of India has come up with a unique solution of posting updates on social channels regarding traffic movement and providing a notification service to commuters. This system takes into consideration the official source of traffic information, Delhi Traffic Police in this case, to alert commuters regarding traffic.

Through this study, a system is developed that crawls real time traffic information to generate a knowledge base which can be used for probabilistic modelling and with the help of appropriate front end modules, will be able to cater navigation related queries of users.

Literature review: This study builds upon previous research work done in the field of traffic Analytics in "Traffic analytics using probabilistic graphical models enhanced with knowledge bases "(Anantharam *et al*., 2013 a,b).The researcher of reference (Anantharam and Srivastava, 2013) have dealt with the lack of certainty, incompleteness and dynamism in the domain knowledge of traffic data with probabilistic graphical Models. In order to build knowledge base supported from the ground up, a "top-down" appro ach to leverage available knowledge has been proposed which

Corresponding Author: Kumar Abhishek, Department of Computer Science and Engineering, National Institute of Technology, Patna, India

Table 1: World statistics for cities with worst traffic [Source:http://auto.ndtv.com/news/three-indian-cities-in-the-top-10-list-of-worst-traffic-conditions-in-the-world-735439

City	Traffic index	Time index (min)	Time exp index	Inefficiency index	Co emission index
Mumbai, India	320.24	66.18	24560.83	263.030	6581.18
Nairobi, Kenya	317.24	65.2	23012.87	253.960	7123.60
Puna, India	316.92	60.86	16878.88	244.010	12215.71
Cairo, Egypt	309.28	58.61	14164.83	309.210	13010.67
Kolkata, India	299.77	58.00	13477.89	397.769	11179.60
Miami FL, United state	299.04	59.20	14847.40	340.410	9908.80
Tahran, Iran	292.98	59.84	15615.00	237.800	8604.11
Recite, Brazil	285.17	57.85	13308.35	474.750	8130.77
Pretoria, South africa	268.00	52.25	8027.250	355.080	11515.00
manila, philippines	266.39	54.33	9797.540	248.100	9471.67
Denver Co, United states	262.99	52.50	8227.950	171.830	11380.00
Jakarta, Indonesia	262.18	50.70	6852.060	277.110	12556.00
Istanbul, Turkey	250.54	55.39	10786.98	219.440	5847.63
Rio De janeiro, Brazil	248.28	56.05	11430.86	281.820	4695.26
Saint petersburg, Russia	245.11	53.64	9180.330	257.560	6337.27
Banbkok, Thailand	244.18	48.90	5631.990	301.220	10584.00
Maxico city, Maxico	233.62	48.00	5077.910	266.580	9610.00

results in better traffic delay predictions.an extension of this strategy can be found in "dynamic update of public transport schedules in cities lacking traffic instrumentation" (Anantharam *et al*., 2014) where a dyn amic system has been proposed using public transportation information as source (which is largely static in nature) that can reason about traffic delays through quantitative estimation.

In "City notifications as a data source for traffic management" (Anantharam and Srivastava, 2013 a, b), the authors talk about utilizing data fro authorized, city-initiated data sources to implement an alert system that will be helpful in informing citizens to make informed travel in places lacking instrumentation for traffic monitoring. The push notification system is built using SMS updates being sent by Delhi Traffic Department using mass SMS alert service SMS Gupshup. A major challenge here is of information extraction.

A major breakthrough in the extraction of information from Physical Cyber Social systems existing in a citywide infrastructure has been proposed in "Extracting city traffic events from social streams "(Anantharam *et al*., 2015 a,b).It talks about accessing instance level domain knowledge by annotating it through a trained sequence labelling program that can extract event from text. Since the updates can vary in terms of structure and content to a large extent preparing a sequence labelling tool that can take into consideration such large and varied dataset is a challenging task.

This study aims at utilizing physical cyber social Systems for obtaining information about metropolitan traffic. The obtained information will be analyzed, after rigorous classification and probabilistic reasoning, to develop a model that can compute delay probability and give valuable results in helping travellers to make an informed travel.

Analyzing the problem statement: A person spends a very considerable amount of time in the traffic. This precious time could be utilized elsewhere. The data in Table 1 shows estimated time spent by people stuck in traffic residing in major cities of the world. Three Indian cities, Mumbai, Pune and Kolkata take up respectively 1st, 3rd and 5th clearly showing a need for such system in Indian metropolitan cities.

Problems faced by commuters: There is very limited information available on the real time traffic as well as historical patterns for the same. This prevents commuters from taking right routing decisions on a daily basis. Another problem is that very few sources are actually available that give information about the traffic updates. These sources are not timely and not even available for most areas.

There is also the problem of authenticity of the data received on traffic updates. Reliable and official sources are very few in number. Further, there is no record of traffic updates that can give us a predictable time or routes.

Review of current measures: There are some solutions available for this problem. Some of them are as follows.

Twitter and facebook posts by traffic authorities: In case of any specific events, there are some official accounts that provide information regarding any specific events that is causing discontinuity in the traffic. Most of the traffic managements hold a portal where they timely update the traffic related work. In Delhi, Delhi Traffic Police (DTP) maintains Twitter and Facebook accounts where they post daily traffic activities. Those users who follow this account get regular information. Major problem with this approach is that the user gets a lot of

information which might not be of his/her use. It is also possible that the necessary information is not available to users.

SMS gupshup service: Bulk messages have been used in the past to provide information regarding traffic updates. Due to TRAI (Telecom Regulatory Authority of India) regulation on the bulk messages its use by @dtptraffic has been discontinued.

Google maps: Google maps also provide information about the traffic events and updates. In a country like India where routes are not well defined and distributed, it is difficult to get the real time information behind the cause of traffic events, this information, if available can greatly impact the way we commute. As on Google Maps, only traffic is visible in color codes representing the density and if users get to know about the cause behind a particular traffic jam which is either short-lived like a red signal or adverse like breakdown or vehicle, roads or bridges that requires considerable amount of time, they can make better decision in choosing their path based on their past experiences with such events in the region. Like, if in a particular scenario on shortest path between source and destination, the events behind jam is likely to resolve soon, then it can be better to continue on the shortest path rather than changing the course and vice versa.

Paid traffic monitoring services: There are some paid services such as Ridlr (http://ridlr.in/) that provides information regarding traffic updates. Ridlr is currently providing real time traffic information in various cities in India such as Mumbai, Delhi NCR, Hyderabad, Bangalore, Kolkata, Chennai and Pune. A major limitation is that it is a paid service and people are not very interested to pay money for such services.

Radio updates: Radio channels provide updates about traffic in different parts of the city. These updates are periodic so it is difficult to get information in times of need.

User updated forums: There are many user updated forums that post traffic events and updates for a particular city. Here major problem is that the information authenticity can 't be verified.

Limitations of current measures: There are many shortcomings associated with existing solutions. Some of these limitations are as follows: With official social media account posts user can only get information when there is any specific event that is triggering traffic disturbance.The user can't get information about the daily traffic updates. Bulk Message services are now discontinued because of the TRAI (Telecom Regulatory Authority of India) regulations on bulk messages. Hence, the cost of bulk messages has increased, so the option of sending updates through Bulk Messaging services has now been discontinued.

Google maps provide traffic updates but they do not give information about the cause of the events which can be used for decision making by the commuter for travelling based on event it's longevity and effect on traffic.

There are some paid services which allow bulk messaging like mVaayoo but people are reluctant to pay for such services. Hence, these kinds of services do not attract user's attention in any significant way. Radio channels provide updates about traffic but they are not very timely and limited to local regions. A user can't switch to a radio channel whole day to get information about a particular area.So this is also not very feasible for users. There are some user updated forums but authenticity of data is a big problem. They are not very reliable.

MATERIALS AND METHODS

Here, we propose prediction of the traffic condition based on historical data. Users will be getting a suggestion for alternate routes allowing lesser congested commute. Figure 1 shows the workflow of the proposed solution. Data source under consideration is tweets from @dtptraffic-Police. Elements of the workflow are explained in Fig. 1

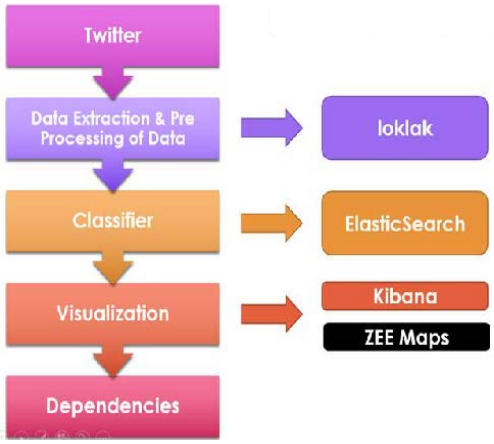

Fig. 1: Workflow of traffic analysis (proposed model)

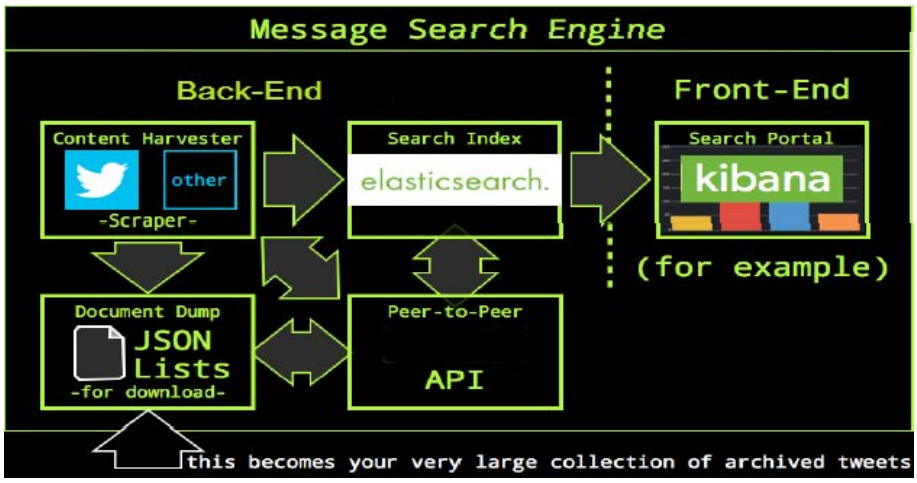

Fig. 2: Linking of all components

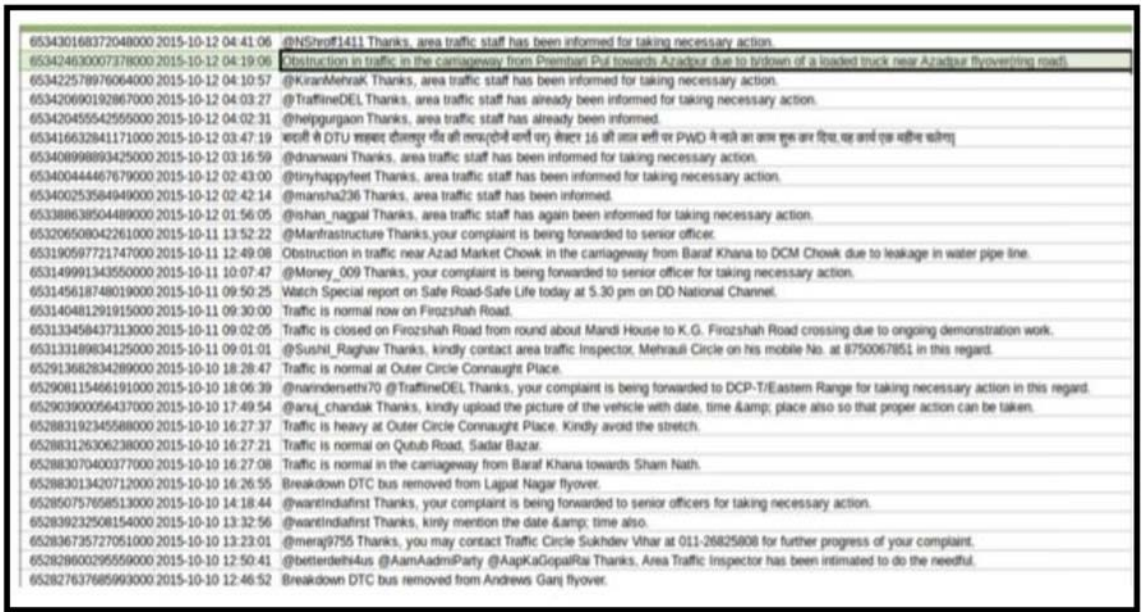

Fig. 3: Snapshot of tweets extracted

Data extraction and pre-processing of data: First step is to extract tweets from the DTP twitter account. There is a limitation on the number of calls on the twitter API. Twitter API will only entertain 15 requests per window. To overcome this limitation, a servlet that scrapes twitter based on url params passed and builds JSON response that can be stored in a database has been implemented, this way collection and simultaneously storage of huge amounts of Tweets is possible. It is used to create a search portal for evaluating tweets statistically. Then, extracted tweets are converted to JSON format in order to extract sections of the tweets such as Tweet ID, Timestamp and Content (Fig. 2).

Classification: After tweet extraction, classification is done on the collected data to mine and store the required content. Classification is done for the analysis of data. It is a long process. Classification will analyze the tweets and will store the traffic event associated with the location found in tweet. The tweets scraped from twitter are tagged in a separate JSON key with location provided in tweet response payload. This helps in classification and analysis of tweets in based on location. The whole classification process has been divided into a number of smaller algorithms for easier understanding.

Devised algorithm: A snapshot of tweets collected from @dtptraffic shown below (Fig. 3): Some

observed patterns in structures of tweets from @dtptraffic are as follows:

- Obstruction to traffic in …….from <source> to/towards <destination> due to <event>
- Traffic is normal at <location>
- Traffic is heavy at <location>
- Breakdown of vehicle at <location>
- Traffic is closed on <location> to <location> for <event>

Strings such as from, to, towards, at, due to, because of and several others are markers or stopwords. These help to identify sources, destinations and events in the tweets. After observing tweets generated by Delhi Traffic Police through their official twitter account we were able to identify possible templates used for traffic event announcement. Let all such tweet structures be stored in array TS. Let s, d, tm, e be source, destination, timestamp and event respectively. The overall process of classification can be divided into three parts. The output of one part is fed as input to the next. The algorithms corresponding to each part are discussed next

Algorithm 1: Algorithm to collect relevant tweets from DTP
Crawler based on Loklak has been used to collect tweets from the Twitter account of @dtptraffic. The algorithm for this process can be described as:

Input: twitter account of Delhi Traffic Police (DTP)
Process:1. Initiate connection to requested URL through ClientConnection() through a Servlet program
2. Parsing the search query in a form that can be understood by Twitter API through
prepareSearchURL() in Servlet
3. Making the search call to Twitter API through Timelines[]Search
4. Parsing the response obtained through BufferedReader supported by Java to generate output in the
form of JSON that can be processed and stored in database

Output: Database ds containing all the tweets of DTP account.
For each tweet tw do
For all the tweet templates in TS do
Match tw with TS
If there is a match
Extract s, d, e, tm and store it in the database ds
Break;
end if
end for
end for

This process will collect all the relevant tweets from DTP account. It will store all those tweets in JSON format.For example,consider the following tweet is encountered with timestamp "2015 -10-12 4:19:06 ": ts ="Obstruction in traffic in carriageway from Pitambari Pul to Azadpur due to the breakdown of a truck near Azadpur flyover(ring road)." This tweet matches the following structure which is one of the several identified structures in TS array. Tweet Structure: Obstruction to traffic in ... from <source> to/towards <destination> due to <event>Hence as match occurs, this tweet is stored in a database and the following results are extracted:
s :"Pitambari Pul "
d :"Azadpur "
e :"breakdown of a truck near Azadpur flyover(ring road)"
tm : 2015-10-12 4:19:06

Algorithm 2: Algorithm to find frequency of the different areas:
Let lc denote the variable used to store location currently being queried for frequency computation from loc database:
Input: Database loc stores a list of all locations of Delhi
Process:
All scraped tweets are indexed
Using the location attribute associated with tweets are traversed
To find frequency the searching is done on database based on location index stored in database
Output: A database containing locations with their respective frequency.
frequencyOf(lc|loc)
{
For all location lc in loc do
Find the frequency of lc from database ds
end
}

Delhi region is divided into several areas which is stored in database loc. The frequencies of these locations are determined from the tweets stored in a database. These frequencies help to predict traffic congestion of different areas.

Algorithm 3: Algorithm to measure the traffic density

Input: lc denotes the location whose density needs to be measured

Output: Traffic density for location lc
Density(lc)
{
f=frequencyof(lc)
If f<10 AND f>0
return(low)
Else if f>10 AND f<20
return(medium)
Else if f>30
return(high)
}

The frequency of locations prone to traffic jam is calculated as they appeared in our data set. This will help us to decide the density of traffic (Fig. 4). The locations with their traffic densities are computed. The output is is a graph of the frequency of the different locations of the city; this helps to analyze the traffic density of different areas. The output of this computation is visualized using Kibana which is discussed in the next section.

Visualization: After classification, traffic densities of the locations are obtained. The results obtained are shown in the form of charts and maps. These are achieved by a tool,

ID	NAME	FREQUENCY	STATUS
1	Tilak Marg	25	heavy
2	India gate	16	Medium
3	Ram Manohar Lohia Hospital	23	heavy

ID	NAME	FREQUENCY	STATUS
4	Rajghat	17	Medium
5	Nehru Place	25	heavy
6	Chirag Delh	24	heavy

ID	NAME	FREQUENCY	STATUS
7	Rajiv Chow	21	Heavy
8	India Gate	9	Low
9	Red Fort	5	Low

ID	NAME	FREQUENCY	STATUS
10	Akshar Dham	23	Heavy

Fig. 4: Snapshot of the dataset consisting of a location with their frequency and traffic density

Kibana (Sheth *et al*., 2014ab, 2013). Kibana uses the results obtained from the Elasticsearch and visualize them in the form of charts easily. Kib ana is used as the front-end module for the proposed system; it will generate several useful graphical patterns for study and analysis.

RESULTS AND DISCUSSION

Frequency of locations prone to traffic jam is calculated as they appeared in our data set. This will help to decide the density of traffic. The obtained frequency distribution enumerating occurrence of location in the data source can be as shown in Fig. 5. This proves to be a very informative metric to understand the distribution of traffic based on locations.

Map generation: According to the coordinates of locations, a database for Geo-fencing was generated (Fig. 6). Employing Geo-fencing and using KML (Keyhole Markup Language) (Yu, 2011), a map is generated for showing locations with a degree of frequency of traffic (Fig. 7). The different icons denote different traffic density. This is the graphical representation of the nodes generated after frequency collection, where nodes have been highlighted to represent a particular range of traffic density with a marker. Details about a range of traffic represented by marker can be found in Table 2.

Table 2: Marker representation

Marker	Represents
A	Low traffic
Blue	Medium traffic
B	Heavy traffic

Traffic density at a particular location is calculated on the basis of the range of traffic handled by that particular location. In order to categorize locations based on frequency of Traffic density, following categories have been taken into consideration as given in Table 3.

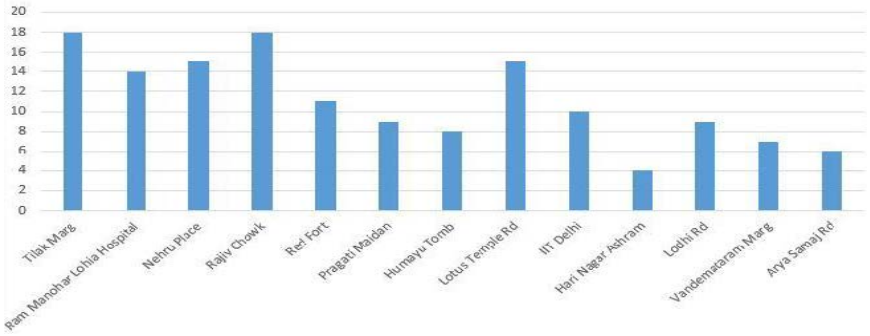

Fig. 5 : Frequency distribution of 'lc ' in the 'loc

ID	LATITUDE	LONGITUDE	NAME	DESCRIPTION	ICON
1	28.631813	77.219091	Rajiv Chow	Heavy	12
2	28.612268	77.225094	India Gate	Low	11
3	28.65483	77.241568	Red Fort	Low	11
ID	LATITUDE	LONGITUDE	NAME	DESCRIPTION	ICON
4	28.614198	77.27519	Akshar Dham	Heavy	12
5	28.618759	77.23977	Pragati Maidan	Medium	192
6	28.649103	77.23281	Jama Masjid	Medium	192
ID	LATITUDE	LONGITUDE	NAME	DESCRIPTION	ICON
7	28.590238	77.247389	Humayu Tomb	Heavy	12
8	28.598401	77.256404	Gurjar Samrat Mihir Bhoj Marg	Low	11
9	28.552921	77.257499	Lotus Temple Rd	Medium	192
ID	LATITUDE	LONGITUDE	NAME	DESCRIPTION	ICON
10	28.554775	77.21257	Balbir Saxena Marg	Heavy	12

Fig. 6: Database for Geo-fencing

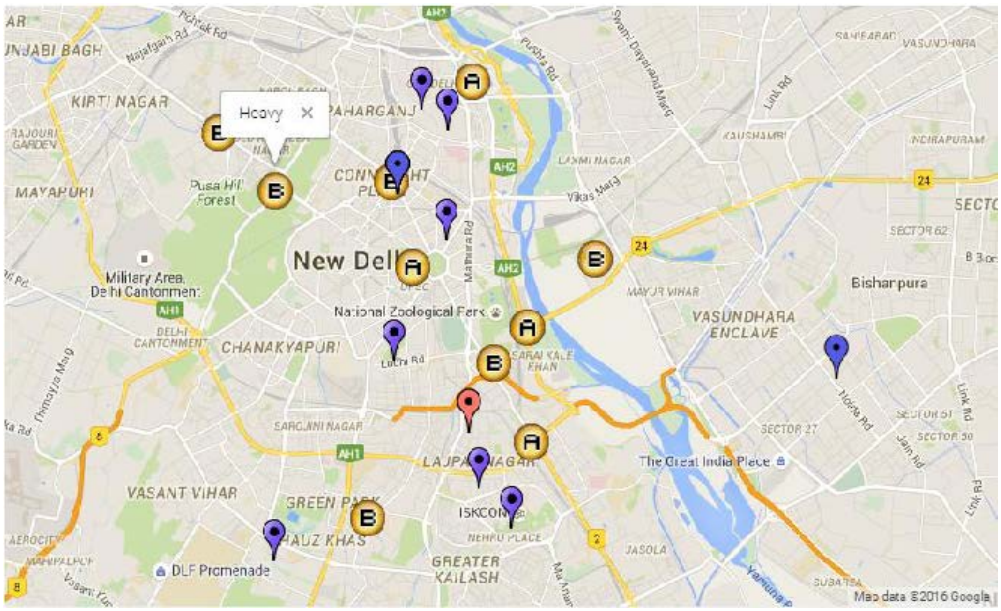

Fig. 7: Map Generated showing traffic density

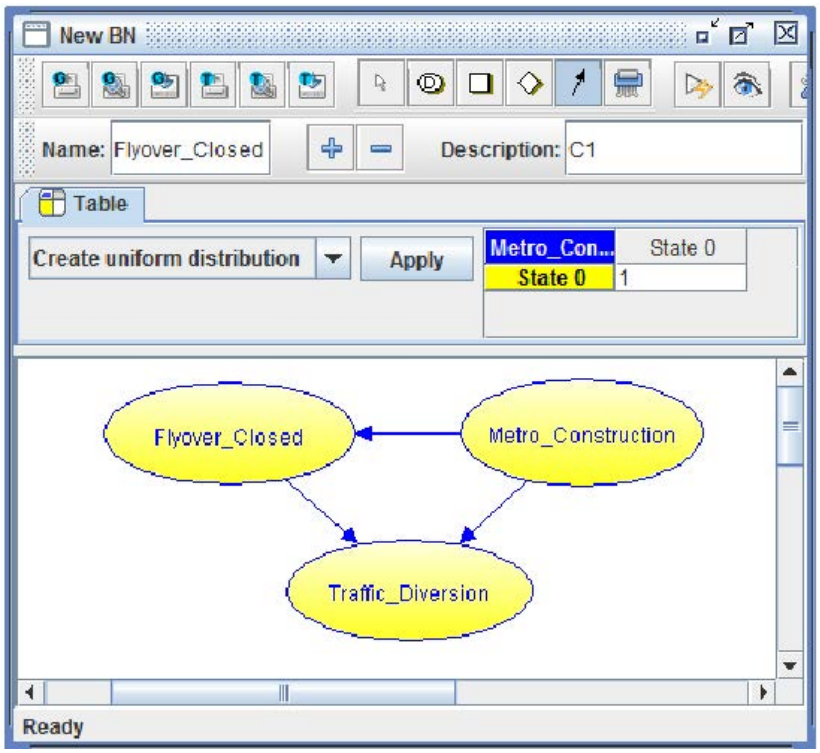

Fig. 8: A sample event correlation using UnBBayes

Table 3: Traffic density

Range	Degree
20<x≤10	Low
10<x≤10	Medium
x>20	Heavy

Event correlation: Bayesian Network refers to a probabilistic graphical model that represents a set of random variables along with their conditional dependencies through Directed Acyclic Graphs (DAG). Conditional dependencies are represented by edges, whereas conditionally independent variables are represented by nodes. Every node is associated with its probability function. Since our data set is sparse, hence generating such a graph is helpful to understand event correlation. UnBBayes (Anantharam, 2016) has been used for this purpose which is an open source software for modelling, learning and reasoning upon probabilistic networks and has support for Bayesian networks.

The conditional probability function is calculated by using baye's theorem. Baye's theorem describes the event probability, taking into consideration the events which might trigger them. Mathematically, Baye' s theorem can be represented as:

$$P(A/B) = \frac{P(B/A)P(A)}{P(B)} \quad (1)$$

Where:

A, B = Refer to events

P(A) as well as P(B) = Refers to probabilities of observing A and B regardless of each other. P(B).0 must be true

P(A | B) = Refers to conditional probability which means the probability of observing an event A given event B has occurred

P(B | A) = Shall refer to the probability of observing event B given that event A has occurred

Here, we are taking into consideration a scenario where we will calculate the relative effect of two traffic events on Traffic Diversion. We have selected events "Metro_Construction "and "Flyover_Closed " where "Flyover_Closed "based on the tweet received for this particular event from our collected database which is quoted as below (Fig. 8). "Metro Pillar construction work is being carried out by DMRC near Mayapuri flyover. Due to this".

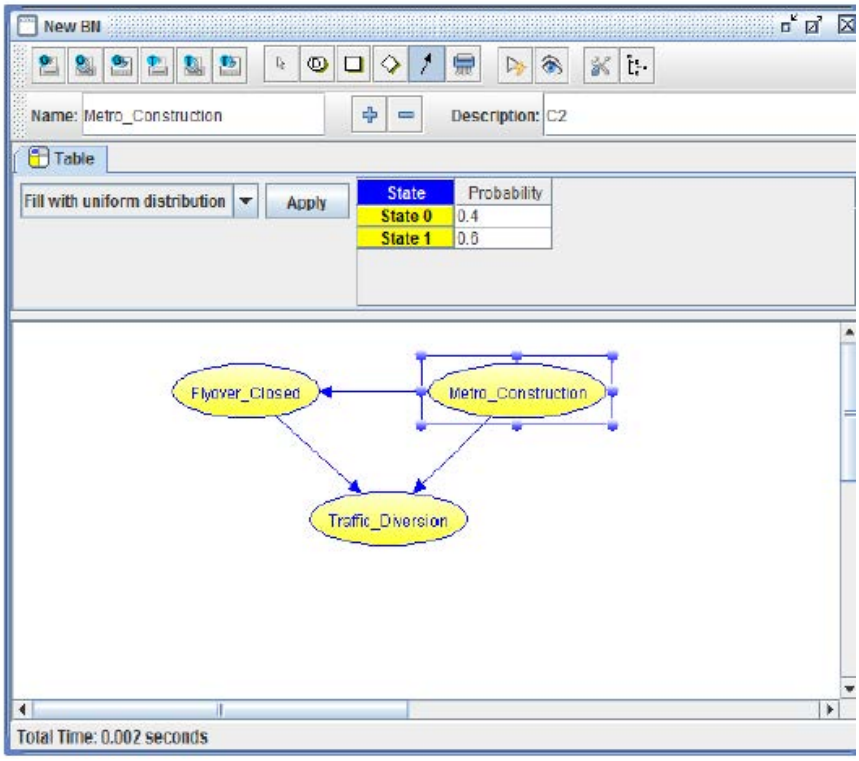

Fig. 9: Cpt (conditional probability table) for metro_construction

We were able to find out that "Flyover_Closed "is dependent on "Metro_Construction "and both cause an effect on "Traffic_Diversion "as well.Here edges represent conditional dependencies between all three event variables. In this manner we have compiled an exhaustive list of events and lined out their possible dependencies based on the reasons mentioned in extracted tweets containing information about dependencies of a particular event and generated a dependency graph manually. The probability value for Independent variables are calculated using Equation:

$$\text{State } 0 : P(\text{Event}) = \frac{\text{Event Instances causing Traffic Jam}}{\text{Total number of Event Instances}}$$

For example, to calculate probability of Traffic jam for variable Metro-construction we would apply the following approach:

$$\text{State } 0 : P(\text{Metro} - \text{Construction}) = \frac{\text{Metro} - \text{Construction causing Traffic Jam}}{\text{Total number of Metro} - \text{Construction Instances}}$$

$$\text{State } 1 : 1 - P(\text{Metro} - \text{Construction})$$

Conditional probability tables were obtained for Metro_Construction as above (Fig. 9) by considering True (State 1) and False (State 0) values for Metro_Construction. Succeeding values are computed using conditional dependency based on Bayes Formula.

Similarly, conditional Probability Tables were obtained for Flyover_Closed as above (Fig. 10) by considering True (State 1) and False (State 0) values for Flyover_Closed. Succeeding values are computed using conditional dependency based on Bayes Formula. From the CPT obtained for Metro_Construction (Fig. 9) and Flyover_Construction, (Fig. 10) Action Table for Traffic_Diversion was obtained (Fig. 11) whose probability is dependent on events Metro_Construction and Flyover_Closed. Finally, the obtained CPTs (Conditional Probability Tables) for Metro_Construction, Flyover_Closed and Traffic_Diversion can be combined for visualization and modelling as shown in Fig. 12.

A Bayesian network has been formulated after doing multiple iterative scans of obtained data from @dtptraffic to enlist frequent events existing in the data set which helps in finding out their dependency related to other events as it is mentioned in tweets and measuring the extent to which they affect traffic movement in a particular region. To quantify these events in order to calculate their impact on Traffic movement, a decision variable

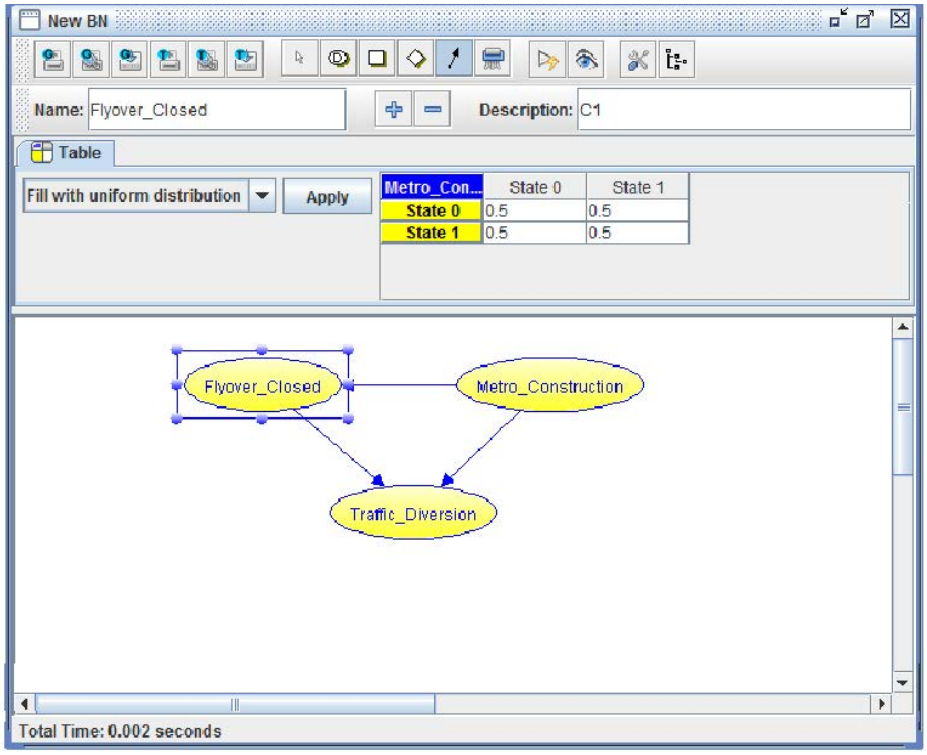

Fig. 10: CPT (Conditional probability table) for flyover_closed

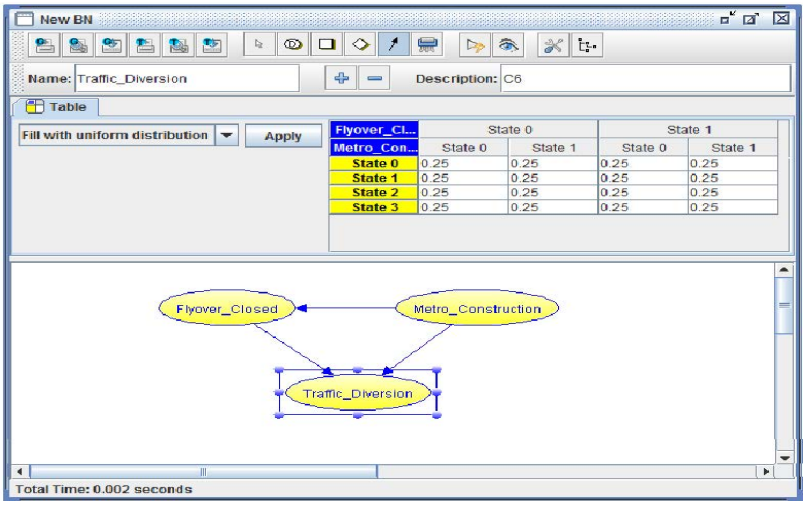

Fig. 11: CPT (Conditional probability table) for traffic_diversion

delay_probability is defined. Delay probability takes into consideration Avg_Volume_of_Vehicles (variable to understand the volume of traffic moving in and out from a region) and Avg_Speed_of_Vehicles (variable to understand the average speed to traffic moving in a region that helps in determining if the area is experiencing a traffic jam if speed is slow) which are in turn computed using Event random variables viz. Public_Event, Time_of_Day, Sport_Event etc. using Eq. 1. Above visualization (Fig. 13) of Bayesian Network for Event Correlation of Traffic has been obtained by compilation of Bayesian Network constructed earlier which shows the

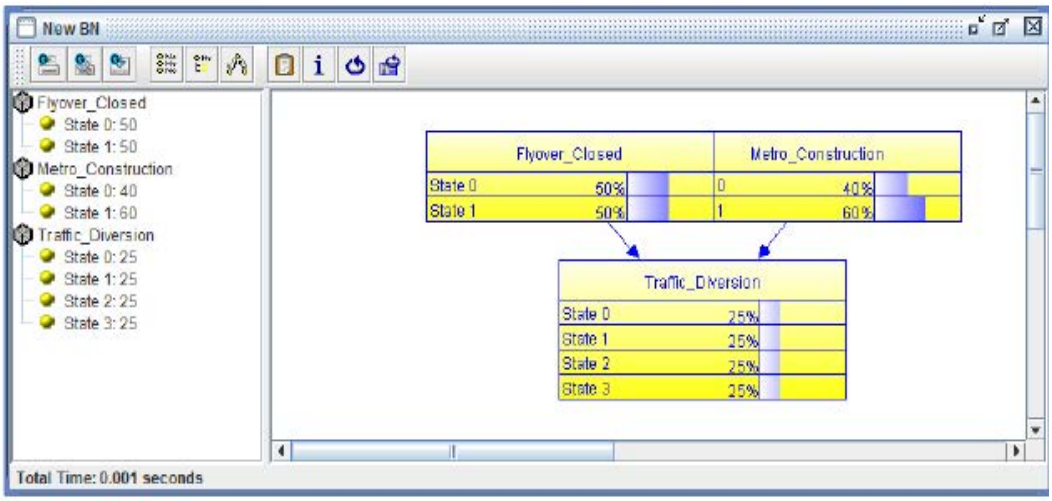

Fig. 12: Visualization of CPT (Conditional Probability Table)

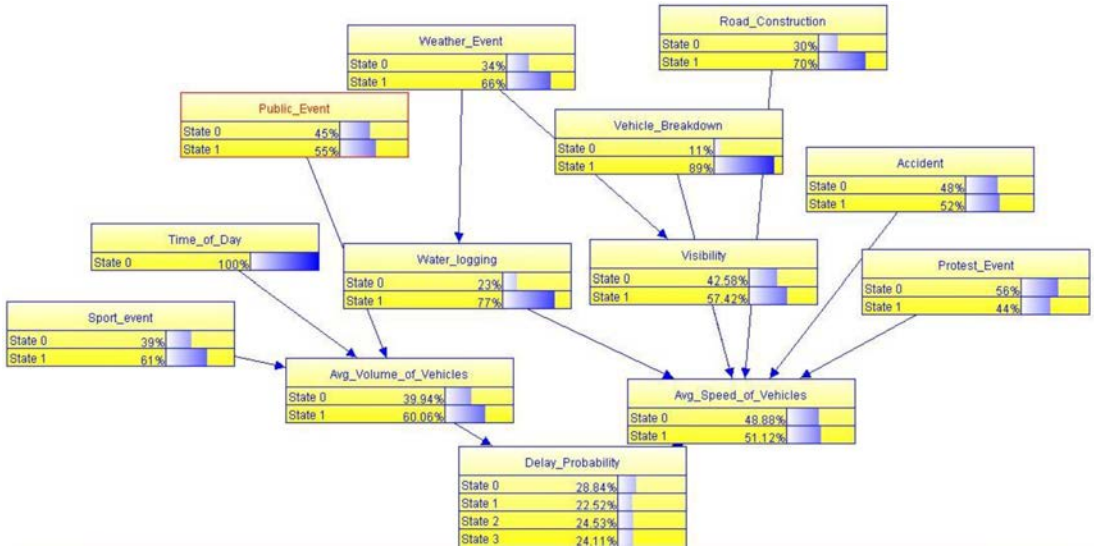

Fig. 13: Visualization of CPT (conditional probability tables) for delay probability model

effect of Probability of Events depicted in the form of Random variables on delay probability (variable to store the computed outcome of probability of an event of traffic jam in a particular route being considered for calculating delay probability). Extending on previously obtained Bayesian Networks, event correlation can be achieved to compute and predict the propagation of traffic delay, its effect on surrounding regions based on their location and calculated as a function of two utility variables (Avg_Volume_of_Vehicles, Avg_Speed_of_Vehicles) that carry cumulative effect of each random event variable.

How have Bayesian networks helped our purpose?: Bayesian networks have helped us to obtain meaningful probabilistic inferences from relatively unstructured data. This helps in finding out the propagation and triggering of events in a probabilistic perspective. Like, by varying the parameters of variables and events on which a particular event depends we can find out probable effect on that particular event. This can be utilized in several ways by users as well as traffic authorities to take better decisions for traffic control. One of the many possible ways can be to decide for how long an event will affect traffic movement based on it 's past records and current expected time to resolve or complete, it can be probabilistically inferred.

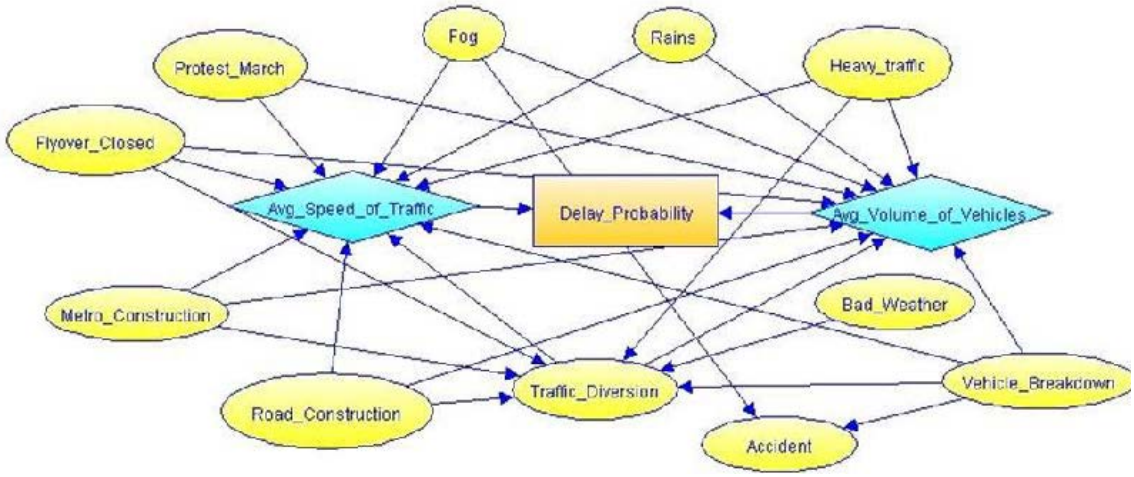

Fig. 14: Bayesian network for event correlation

ID	LOCATION	PUBLIC_EVENT	SPORT_EVENT	WEATHER_EVENT	VEHICLE_BREAKDOWN_EVENT	ACCIDENT_EVENT	ROAD_REPAIR
1	Hauz Khas	3	0	6	5	4	5
2	Vikaspuri	3	2	4	4	5	4
3	Chirag Delhi	4	2	5	3	4	6
ID	LOCATION	PUBLIC_EVENT	SPORT_EVENT	WEATHER_EVENT	VEHICLE_BREAKDOWN_EVENT	ACCIDENT_EVENT	ROAD_REPAIR
4	Delhi Cantt	6	2	3	2	6	5
5	Nehru Place	7	4	4	3	3	6
6	Ram Manohar Lohia Hospital	3	0	5	4	4	6
ID	LOCATION	PUBLIC_EVENT	SPORT_EVENT	WEATHER_EVENT	VEHICLE_BREAKDOWN_EVENT	ACCIDENT_EVENT	ROAD_REPAIR
7	Rajghat	2	0	6			

Fig. 15: Frequency according to event type

Example: It has been recorded that MG Road connecting Delhi to Gurgaon stays clogged for 1900-2100 hrs due to people commuting back to Delhi after office, user can be informed to avoid this route on weekdays.

Classification according to event types: Events are classified according to types mentioned in Fig. 14 for event correlation and analysis is done on events that are highly responsible for causing traffic at a particular location (Fig. 15 and 16). Various event types that we found during our study are:

- Rains
- Fog
- Protest march
- Flyover closed
- Metro construction
- Road construction
- Traffic diversion

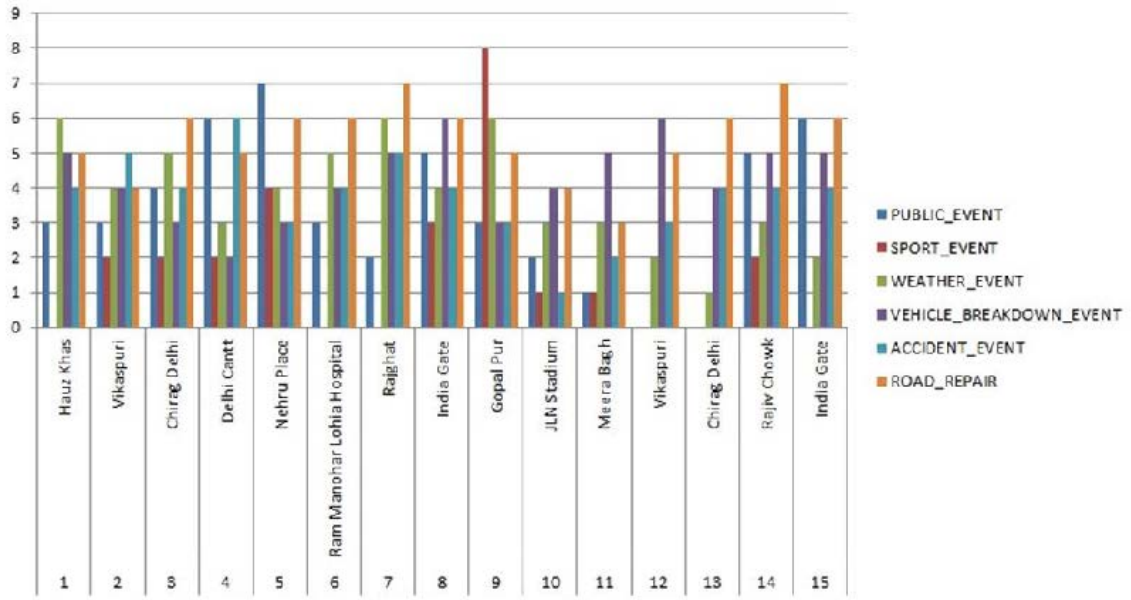

Fig. 16: Graphical visualisation for event type based frequency

- Accident
- Vehicle breakdown
- Bad weather
- Heavy traffic

Hence, using the classifier, we obtain categorization of events and the frequency with which they are matched with locations in the data set (Fig. 15 and 16). This will be extremely useful for traffic authorities to perform traffic control by heuristically determining which kind of events shall affect particular locations more. Like for example: Khel Gaon will be more prone to congestion and traffic jams on occasion of sports events and a region like Nehru Place is likely to witness traffic jam over weekends as it is a market where movement of people is increased during weekends.

Dependencies and limitations: Data sources used for the proposed system are very few and sparse information can be extracted from them. Unstructured nature of the already limited amount of data poses a huge challenge. This problem has been overcome to some extent by the implementation of Bayesian networks. It shall help greatly in heuristic predictions if several such authentic and reliable sources are present for data collection.

CONCLUSION

This study outlined an efficient way of implementing an effective heuristic based traffic congestion measuring system, address all the aspects of such a system and develops a proof of concept application. Based on all prior discussions and related work in this study, implementing such a system is not very difficult. While the system can work even if there is only one source of data available, any given time, more the number of available data-sources, better the system will reearch.

Hence, for that there is a need of more reliable and authenticate sources that will give us data so that we could apply proposed methods to produce superior results.

There are some shortcomings with this methodology as discussed in dependencies but we plan to overcome these issues in the future and follow up on the work we have done so far. Future scope semantic analysis will be done to further analyse hidden patterns in the traffic events. Event correlation can be studied in a broad spectrum this will help us to predict future possibilities of event occurrence. Other social media sources can also be included for data collection. We can provide live traffic and the density of traffic on the route. Density of traffic can also be provided in different parts of the route. User can also suggest path and such crowd sourced information could be integrated in our system. We can track user activities to provide more useful and reliable results.

RECOMMENDATIONS

Semantic analysis will be done to further analyse hidden patterns in the traffic events. Event correlation can be

studied in a broad spectrum; this will help us to predict future possibilities of event occurrence. Other Social Media sources can also be included for data collection. We can provide live traffic and the density of traffic on the route. Density of traffic can also be provided in different parts of the route. User can also suggest path and such crowd sourced information could be integrated in our system. We can track user activities to provide more useful and reliable results.

REFERENCES

Anantharam, P. and B. Srivastava, 2013. City notifications as a data source for traffic management. Proceedings of the 20th Conference on ITS World Congress, October 14-18, 2013, Wright State University, Tokyo, Japan, pp: 500-608.

Anantharam, P., 2016. Knowledge-empowered probabilistic graphical models for physical-cyber-social systems. Master Thesis, Wright State University, Dayton, Ohio.

Anantharam, P., B. Srivastava and R. Gupta, 2014. Dynamic update of public transport schedules in cities lacking traffic instrumentation. Master Thesis, Wright State University, Dayton, Ohio, USA.

Anantharam, P., P. Barnaghi and A. Sheth, 2013a. Data processing and semantics for advanced Internet of Things (IoT) applications: Modeling, annotation, integration and perception. Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics, June 12-14, 2013, ACM, New York, USA., ISBN:978-1-4503-1850-1, pp: 1-5.

Anantharam, P., K. Thirunarayan and A.P. Sheth, 2013b. Traffic analytics using probabilistic graphical models enhanced with knowledge bases. Proceedings of the 2nd International Workshop on Analytics for Cyber-Physical Systems (ACS-2013), May 2-4, 2013, Wright State University, Dayton, Ohio, pp: 13-20.

Anantharam, P., P. Barnaghi, K. Thirunarayan and A. Sheth, 2015a. Extracting city traffic events from social streams. ACM. Transac. Intell. Syst. Technol., 6: 43-43.

Anantharam, P., T. Banerjee, A. Sheth, K. Thirunarayan and S. Marupudi et al., 2015 b. Knowledge-driven personalized contextual mhealth service for asthma management in children. Proceedings of the IEEE International Conference on Mobile Services (MS), June 27-July 2, 2015, IEEE, New York, USA., ISBN:978-1-4673-7284-8, pp: 284-291.

Sheth, A., P. Anantharam and C. Henson, 2013. Physical-cyber-social computing: An early 21st century approach. IEEE. Intell. Syst., 28: 78-82.

Sheth, A., P. Anantharam and K. Thirunarayan, 2014a. kHealth: Proactive personalized actionable information for better healthcare. Proceedings of the Workshop on Personal Data Analytics in the Internet of Things (PDA@ IOT 2014), Collocated at VLDB 2014, September 5th, 2014, Wright State University, Hangzhou, China, pp: 1-8.

Sheth, A., P. Anantharam and K. Thirunarayan, 2014b. Applications of Multimodal Physical (IoT), Cyber and Social Data for Reliable and Actionable Insights. Proceedings of the International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), October 22-25, 2014, IEEE, New York, USA., ISBN:978-1-63190-043-3, pp: 489-494.

Yu, L., 2011. Linked Open Data. In: A Developer's Guide to the Semantic Web, Y. Liyang (Ed.). Springer, Berlin, Germany, ISBN:978-3-642-15970-1, pp: 409-466.