Cloudburst and Hailstorm Detection in upper Shimla region of Himachal Pradesh using Wireless Sensors Network

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology.

in

Computer Science & Engineering

under the Supervision

of

Dr. Yashwant Singh

By

Akshit Nevatia(111331)

to

Jaypee University of Information Technology

Waknaghat, Solan – 173234, Himachal Pradesh

Certificate

This is to certify that project report entitled "**Cloudburst and Hailstorm Detection in upper Shimla region of Himachal Pradesh using Wireless Sensors Network**", submitted by **Mr. Akshit Nevatia** in partial fulfillment for the award of degree of Bachelor of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Supervisor's Name : Dr. Yashwant Singh

Designation : Assistant Proffessor (Senior Grade)

Signature :

Date :

Acknowledgement

This project report gives a detailed description of the study work done on the project topic **"Cloudburst and Hailstorm Detection in upper Shimla region of Himachal Pradesh using Wireless Sensors Network"** in the VIII semester for the partial fulfillment of the requirements for the degree of Bachelor of Technology (B.Tech) in Computer Science Engineering (CSE), under the supervision of **Dr. Yashwant Singh.**

I am very grateful to my Faculty Coordinator **Dr. Yashwant Singh** for his help and able guidance regarding the project. I am also very thankful to the Department of Computer Science Engineering, Jaypee University of Information Technology (JUIT) for providing me resources and facilities which helped a lot in the completion of the project.

Date: May 17, 2015 **Akshit Nevatia**

Table of Content

List of Figures

Abstract

A Wireless Sensor Network is a type of wireless network which basically consists of tiny devices called sensor nodes. A source node is resource constrained for battery power, storage and communication capability. These sensor nodes are set with radio interface with which they communicate with one another to form a communication network. Wireless sensor network has a very necessary application like remote environment monitoring and target tracking.

Over the last decade, the advent of WSN has enabled variety of applications for monitoring and tracking the objects. It has emerged as an important area of research because of rapid changes and advances in hardware, sensors and wireless networking technologies. These advances will enable WSN to become a key to true ubiquitous computing systems in the near future.

This project uses the previous environmental data to find out an efficient approach to analyze this data. Then an effective algorithm will be developed to be deployed in the sensors for detection of hailstorm and cloudburst in upper areas of Himachal Pradesh.

Intended functions of this project include collection of past data from Environment department office of the local Government and from weather forecasting databases maintained in environmental research organizations in Shimla (H.P.), regarding hailstorm and cloudburst on the basis of availability.

Collected data will be represented in proper format to extract the useful features to find relations among them to create a knowledge base, which will then be pre-processed to make the data useful for the learning algorithms.

The information system will be implemented which acquires environmental data such as humidity, temperature, pressure, rainfall and wind-speed from a low-power wireless sensor network consisting of "sensor unit". This data is transmitted to a base station where it is then stored and compared with the existing database to provide timely warning.

CHAPTER- 1

INTRODUCTION

1.1 Overview

In this project an information system will be implemented which acquires environmental data such as light, temperature and humidity from a low-power wireless sensor network consisting of "sensor unit". This data is transmitted to a base station where it is then stored in a database.

Intended functions of this project include collection of past data from Environment department office of the local Government and of various other countries and from weather forecasting databases maintained in environmental research organizations all over the world regarding hailstorm and cloudburst, on the basis of availability.

Collected data will be represented in proper format to extract the useful features to find relations among them to create a knowledge base, which will then be pre-processed to make the data useful for the learning algorithms.

Detecting the hailstorm and cloudburst prone zone through analysis of the past data. Deployment of Sensors and creation of Wireless Sensor Network to monitor environmental parameters. Sensors are placed on telephone towers located nearest to those zones which will send information to the base station. Each sensor unit consists of low-power components including a microcontroller with integrated RF transceiver, environment sensors, and power circuitry.

Development of useful algorithms to learn from the available information. Sampling and data mining techniques of streams will be adopted to process information and generate prediction regarding the hailstorm and cloudburst. Some fusion techniques to ensemble the streams of several sensors will be used.

1.2 Motivation:

Cloudburst and hailstorm are global environmental problems that threatens millions of peoples specially lives in hilly and desert regions. Some notable disasters are:

1. On August 15, 1997, 115 people were killed when a cloud burst came bustling and trail of death are all that is left behind in Chirgaon in Shimla district, Himachal Pradesh.
2. On August 17, 1998 — A massive landslide following heavy rain and a cloudburst at Malpa village killed 250 people including 60 Kailash Mansarovar pilgrims in Kali valley of the Kumaon division, Uttarakhand. Among the dead was Orissi dancer Protima Bed.
3. On July 16, 2003, About 40 persons were killed in flash floods caused by a cloudburst at Shilagarh in Gursa area of Kullu, Himachal Pradesh.
4. On August 16, 2007, 52 people were confirmed dead when a severe cloud burst occurred in Bhavi village in Ghanvi, Himachal Pradesh.
5. On August 6, 2010, in Leh, a series of cloudbursts left over 1000 persons died (updated number) and over 400 injured in the frontier Leh town of Ladakh region in Jammu and Kashmir.
6. On September 15, [year missing] cloud burst in Almora in Uttrakhand has drowned away two villages one of them being Balta, leaving a few people alive and rest entire village dead and drowned. Almora has been declared as a town suffering from the brunt of cloudburst by authorities of Uttrakhand. Had there been a bit more swaying of clouds, town of Ranikhet must have drowned also.
7. On 20 July 2011, a cloudburst in upper Manali, 18 km away from Manali town in Himachal Pradesh state left 2 dead and 22 missing.
8. On June 9, 2011, near Jammu, a cloudbursts left 4 persons dead and over several injured in Doda-Batote highway, 135 km from Jammu.Two restaurants and many shops were washed away.

1.3 Deliverables from the project:

An active alert mechanism for the detection of hailstorm and cloudburst in the region.

1.4 Objectives

1. Detection of cloudburst and hailstorm prone zone in upper Shimla region of Himachal Pradesh.
2. Collection of past years data from the environment departments on the basis of availability.
3. Analyzing past data to determine a threshold range of environmental parameters.
4. Learning algorithms to detect parameters values of cloudbursts and Hailstorms.
5. Development of efficient methodologies to predict cloudburst and hailstorm from streams.
6. Implementing the algorithm on the software.
7. Development of a broadcasting message android application to deliver the messages in case of warning alerts.

1.5 Importance of the project

The weather predictions for specified region in term of time duration are issued by various forecasts. The severe hailstorm and cloudburst often occurred for small duration and which falls in short range forecast. Mostly the extreme events developed on small scale over few hundred sq. km. and only for short duration of 3 to 48 hours, which is localized in nature. It is very difficult to pinpoint the occurrence of severe hailstorm and cloudburst within unstable area. These are most destructive phenomenon in nature.

It is very difficult for Meteorologist to calculate and determine potential of theses severe events, prepare timely forecast and issue warning to disaster management agencies which can minimize losses pertaining to life of people, financial losses due to damage of fruits and crops, protection of environment properties like forests, wild life sanctuaries and orchards, and protection of major projects and plants. Moreover some modern and advanced techniques and methodologies are required for the improvement in prediction of severe events like hailstorm and cloudburst.

1.6 Existing approaches:

The approach for the detection of cloudburst and hailstorm is completely based on analysis of the metrological data and satellite images available.

1.7 What is new in the project?

This project makes use of WSN using humidity, light, temperature and wind-speed as parameters for detection of hailstorm and cloudburst which is completely new.

CHAPTER- 2

WIRELESS SENSOR NETWORKS AND NETWORK TOPOLOGIES

2.1 Introduction

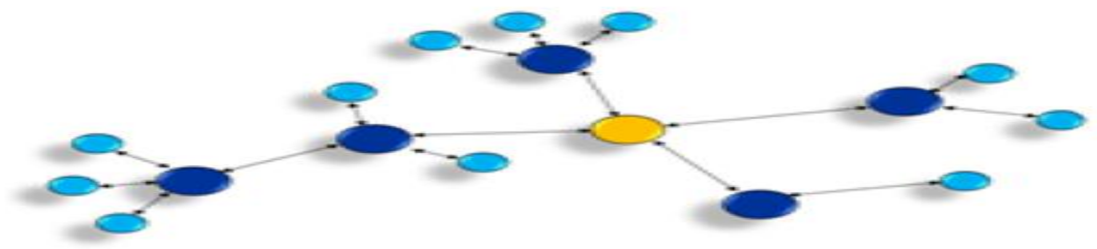

Fig 1: Wireless Network

Wireless sensor networks (WSN) of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure etc. and to cooperatively pass their data through network to a main location. The more modern networks are bi-directional which also enable control of sensor activity. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance. Today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring etc. The WSNs are built of "nodes" from a few to several hundreds or even thousands, where each node is connected to one sensor.

There are various topologies in which wireless sensor networks can be divided which are discussed in the next section.

2.2 Network Topologies in WSN

The basic issue in communication networks is the transmission of the messages to achieve a prescribed message throughput and Quality of Service (QoS). QoS can be specified in terms of message delay, message due dates, bit error rates, packet loss, economic cost of transmission, transmission power, etc. Depending on QoS, the installation environment, economic considerations and the application, one of several basic topologies can be used.

A communication network is composed of nodes, each of which has computing power and can transmit and receive messages over communication links, wireless or cabled. The basic network topologies include fully connected, mesh, star, ring, bus and tree topologies. A single network may consist of several interconnected subnets of different topologies. Networks are further classified as Local Area Network (LAN), e.g. inside one building, or Wide Area Network (WAN), e.g. between buildings.

2.2.1 Star Network Topology

Star networks are one of the most common computer network topologies. In its simplest form, a star network consists of one central switch, hub, computer or cluster head, which acts as a conduit to transmit messages. This consists of a central node, to which all other nodes are connected; this central node provides a common connection point for all nodes through a hub. In star topology, every node is connected to a central node. The switch is the server and the peripherals are the clients. Thus, the hub and leaf nodes, and the transmission lines between them, form a graph with the topology of a star. If the central node is passive, the originating node must be able to tolerate the reception of an echo of its own transmission, delayed by the two-way transmission time plus any delay generated in the central node. An active star network has an active central node that usually has the means to prevent echo-related problems.

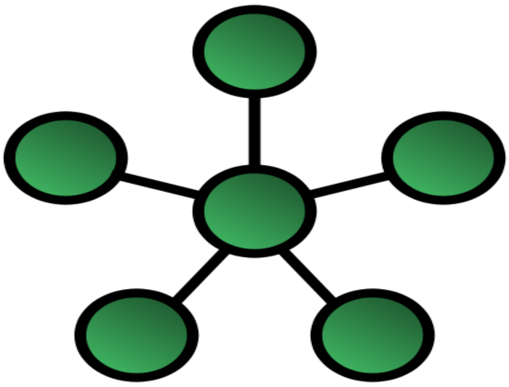

Fig 2: Star Network Topology

The star topology reduces the damage caused by line failure by connecting all of the systems to a central node. When applied to a bus-based network, this central hub rebroadcasts all transmissions received from any peripheral node to all peripheral nodes on the network, sometimes including the originating node. All peripheral nodes may thus communicate with all others by transmitting to, and receiving from, the central node only. The failure of a transmission line linking any peripheral node to the central node will result in the isolation of that peripheral node from all others, but the rest of the systems will be unaffected.

2.2.2 Ring Network Topology

A ring network is a network topology in which each node connects to exactly two other nodes, forming a single continuous pathway for signals through each node - a ring. Data travel from node to node, with each node along the way handling every packet.

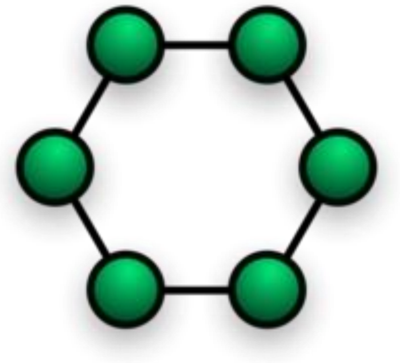

Fig. 3: Ring Network Topology

Because a ring topology provides only one pathway between any two nodes, ring networks may be disrupted by the failure of a single link. A node failure or cable break might isolate every node attached to the ring. In response, some ring networks add a "counter-rotating ring" (C-Ring) to form a redundant topology: in the event of a break, data are wrapped back onto the complementary ring before reaching the end of the cable, maintaining a path to every node along the resulting C-Ring. Such "dual ring" networks include Spatial Reuse Protocol, Fiber Distributed Data Interface (FDDI), and Resilient Packet Ring. 802.5 networks - also known as IBM token ring networks - avoid the weakness of a ring topology altogether: they actually use a star topology at the physical layer and a media access unit (MAU) to imitate a ring at the data-link layer.

2.2.3 Bus Network Topology

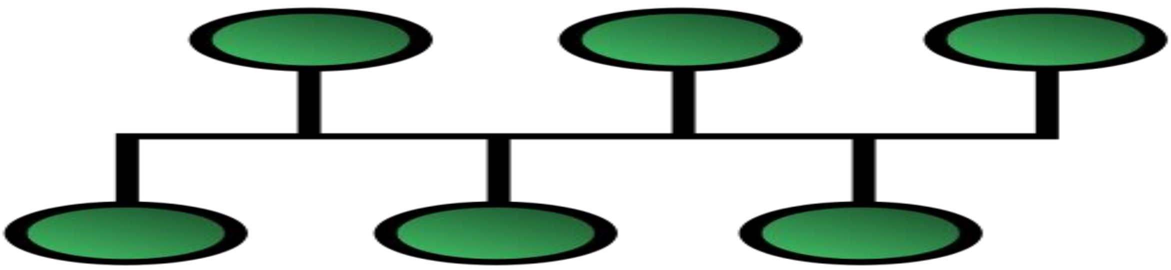

Fig. 4: Bus Network Topology

A bus network is a network topology in which nodes are connected in a daisy chain by a linear sequence of buses. The bus is the data link in a bus network. The bus can only transmit data in one direction, and if any network segment is severed, all network transmission ceases. A host on a bus network is called a station or workstation. In a bus network, every station receives all network traffic, and the traffic generated by each station has equal transmission priority. Each network segment is, therefore, a collision domain.

2.2.4 Tree Network Topology

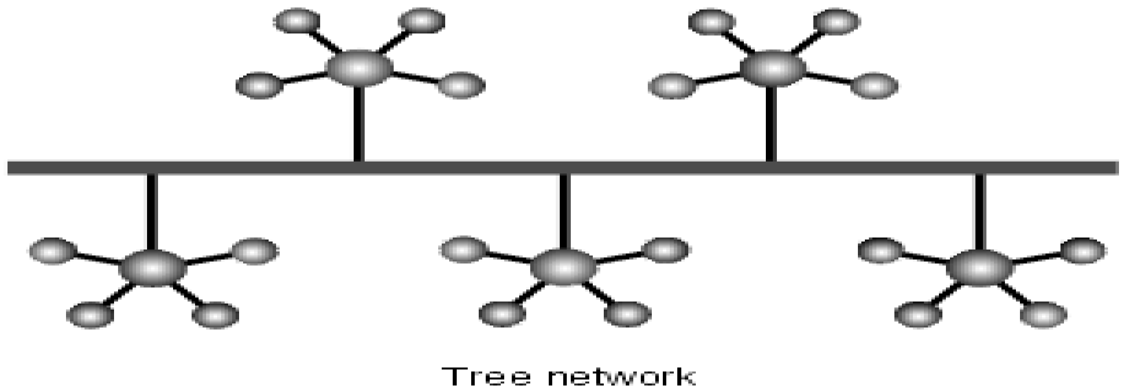

Fig. 5 Tree Network Topology

Tree Topology integrates the characteristics of Star and Bus Topology. Earlier we saw how in Physical Star network Topology, computers (nodes) are connected by each other through central hub. And we also saw in Bus Topology, work station devices are connected by the common cable called Bus. After understanding these two network configurations, we can understand tree topology better. In Tree Topology, the number of Star networks are connected using Bus. This main cable seems like a main stem of a tree, and other star networks as the branches. It is also called Expanded Star Topology. Ethernet protocol is commonly used in this type of topology.

2.2.5 Fully Connected Network Topology

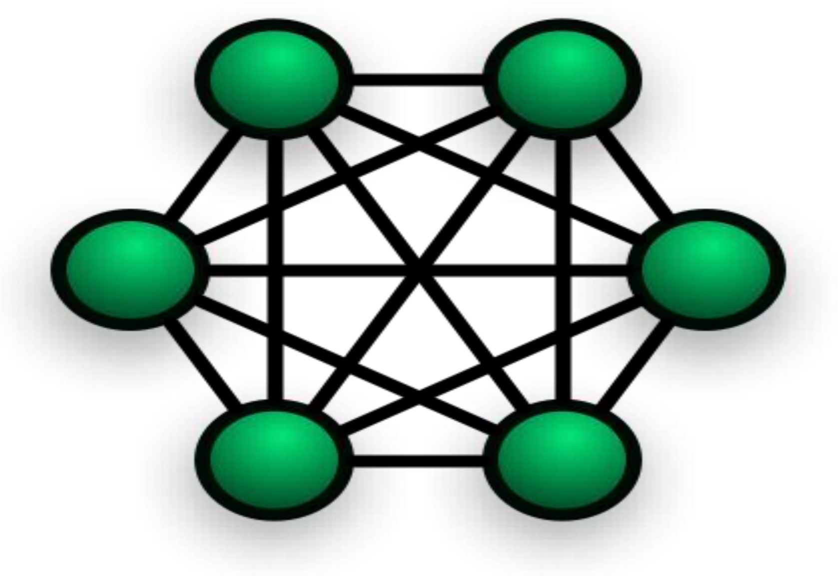

Fig. 6: Fully Connected Network Topology

A fully connected network, complete topology, or full mesh topology is a network topology in which there is a direct link between all pairs of nodes. In a fully connected network with n nodes, there are n(n-1)/2 direct links. Networks designed with this topology are usually very expensive to set up, but provide a high degree of reliability due to the multiple paths for data that are provided by the large number of redundant links between nodes. This topology is mostly seen in military applications."

2.2.6 Mesh Network Topology

A mesh network uses a network topology in which each node (called a mesh node) relays data for the network. All nodes cooperate in the distribution of data in the network.

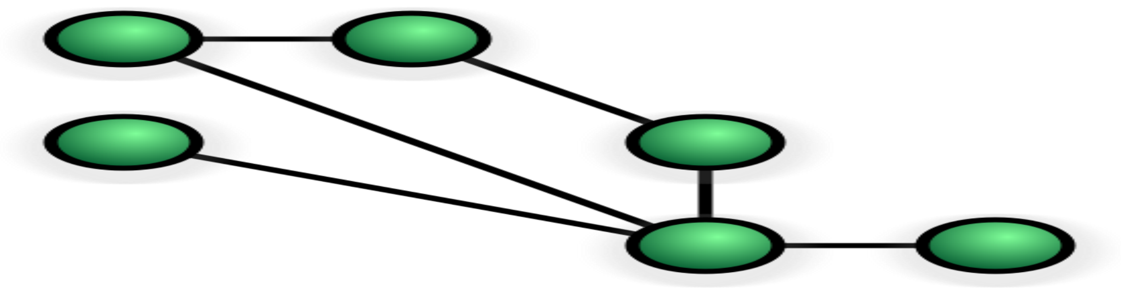

Fig. 7: Mesh Network Topology

Designers of mesh networks can employ a flooding technique or a routing technique. When using a routing technique, the message is propagated along a path, by hopping from node to node until the destination is reached. To ensure all its paths' availability, a routing network must allow for continuous connections and reconfiguration around broken or blocked paths, using self-healing algorithms. Self-healing capability enables a routing-based network to operate when one node breaks down or when a connection becomes unreliable. As a result, the network is typically quite reliable, as there is often more than one path between a source and a destination in the network. Although mostly used in wireless situations, this concept can also apply to wired networks and to software interaction.

CHAPTER- 3

CLOUDBURST AND HAILSTORM

3.1 Cloudburst

Cloudbursts are primarily seen in the dry places like deserts and also in the hilly surroundings and mountainous regions. Besides these, cloudbursts can also be perceived in the areas like inner parts of various continents. The cloudbursts are mostly seen during the monsoon months. Himachal Pradesh in India is one of the places that suffer from cloudbursts. Human life is disturbed when the cloudbursts occur. Especially the villages suffer the most due to the dangers of cloudbursts.

Cloudburst is kind of rainfall that is sudden and violent in nature and is also very short. Cloudbursts continue for only a short period of time. Cloudburst can also be observed as rainstorm. It is a kind of weather phenomenon. It is perceived in a particular region within a specific geographical climate. Cloudbursts often come with thunderstorm, sleet, hail and all that violent rainstorms. Though it lasts for only a short time, the results of cloudbursts are often floods.

During the thunderstorms, the air mass that goes up from the lower level carries a certain amount of water in it. Sometimes that air current abruptly stops moving and the water mass falls down forcefully on the surface of earth. This natural phenomenon is known as cloudburst. In fact, cloudbursts fall from a very high altitude. That is why the force of cloudbursts is so strong. The primary reason behind cloudburst is the rapid concentration of the pieces of clouds in the sky. The result of cloudbursts sometimes causes great harm to the certain place it appears.

3.2 Hailstorm

Hail is a form of precipitation that falls from the sky as pellets of ice. The pellets can range in size from small pea-sized pellets, to hailstones as large as grapefruits. Hail is especially damaging to crops. In the central US, where many hail storms are reported each year, delicate wheat and other crops are often ruined. Annually, hail can cause over 1 billion dollars in damages.

Hail forms as a result of the strong updrafts common in severe weather systems. When a strong convective cell forms, warm air rises and cool air sinks. If there is a sufficient amount of super cooled water, accumulation of ice can begin in the clouds.

Rising air will often reach a point in the atmosphere that is below freezing (hence, ice will form). The ice is suspended in the air by the strong updrafts and will later fall back down. This process will occur over and over adding layer upon layer to the hailstone. If you cut a hailstone in half, you would see alternating concentric layers inside it. As the hail falls, it may melt to varying degrees only to be picked up again and carried high into the atmosphere to re-freeze. Therefore, very large hailstones form with many repeated cycles.

Hail can cause serious damage, notably to automobiles, aircraft, skylights, glass-roofed structures, livestock, and most commonly, farmers' crops. Hail damage to roofs often goes unnoticed until further structural damage is seen, such as leaks or cracks. It is hardest to recognize hail damage on shingled roofs and flat roofs, but all roofs have their own hail damage detection problems. Metal roofs are fairly resistant to hail damage, but may accumulate cosmetic damage in the form of dents and damaged coatings.

Hail is one of the most significant thunderstorm hazards to aircraft. When hail stones exceed 0.5 inches (13 mm) in diameter, planes can be seriously damaged within seconds. The hailstones accumulating on the ground can also be hazardous to landing aircraft. Hail is also a common nuisance to drivers of automobiles, severely denting the vehicle and cracking or even shattering windshields and windows. Wheat, corn, soybeans, and tobacco are the most sensitive crops to hail damage. Hail is one of Canada's most expensive hazards. Rarely, massive hailstones have been known to cause concussions or fatal head trauma. Hailstorms have been the cause of costly and deadly events throughout history. One of the earliest recorded incidents occurred around the 9th century in Roopkund, Uttarakhand, India. The largest hailstone in terms of diameter and weight ever recorded in the United States fell on July 23, 2010 in Vivian, South Dakota; it measured 8 inches (20 cm) in diameter and 18.62 inches

(47.3 cm) in circumference, weighing in at 1.93 pounds (0.88 kg). This broke the previous record for diameter set by a hailstone 7 inches diameter and 18.75 inches circumference which fell in Aurora, Nebraska in the United States on June 22, 2003, as well as the record for weight, set by a hailstone of 1.67 pounds (0.76 kg) that fell in Coffeyville, Kansas in 1970.

The present project proposal is thus originated keeping in view the importance of automated detection of cloudburst and hail to save the people (villagers) from the attack of those disastrous environmental hazards. Objective of this project is implementation of Wireless Sensor Network to monitor the environmental parameters and forecasting the cloudburst and hail by using streaming and fusion analysis techniques.

3.3 Review of R & D in the proposed area, national and International Status

Applications using wireless sensor networks have become increasingly common. Over the past few years, the capabilities of these deployments has evolved tremendously from the initial single-hop 'sense and send' deployments to scalable multi-hop deployments.

From the onset of sensor network technology, monitoring environmental conditions or habitat monitoring has been at the forefront of the application space. In one of the first successful demonstrations of a sensor network deployment, researchers at the University of California, Berkeley deployed a sensor network at Great Duck Island off the coast of Maine. They placed their sensors in burrows and used heat to detect the presence of nesting birds, providing invaluable data to biological researchers. Additionally, their work provided helpful observations about many deployment aspects such as performance, routing, and topology construction.

Similarly, researchers for the Center for Embedded Networked Sensing deployed a sensor network into the James Reserve Forest in California with purposes from soil temperature monitoring to tracking wildlife. Their work extended sensor network research by using multi-hop routing and multiple, heterogeneous nodes.

Other habitat monitoring deployments that have been used for monitoring specific species include a system to monitor Cane Toad populations, and a system for tracking the movements of Zebras.

On a smaller scale, a sensor network was recently deployed on a single redwood tree using 33 nodes to cover roughly 50m. With this unique deployment researchers were able to map the differences in the microclimate over a single tree. Deployments in rugged terrain and under extremely harsh conditions have just begun to be developed. A group of researchers from Harvard recently deployed a sensor network on an active volcano in South America to monitor seismic activity and similar conditions related to volcanic eruptions.

Most relevant to our project, researchers from the University of California, Berkeley demonstrated the feasibility of sensor networking technology in a fire environment with their FireBug application. They deployed a 10 node network in a field and successfully measured important environmental conditions such as relative humidity and temperature as a flame front passed during a prescribed burn. Though our nodes are designed for the encounter, we stayed away from attempting to track or measure flame fronts. The fire community currently utilizes high-tech airborne infrared sensors to track flame fronts and intensities over very large scale areas. Research in security issues in sensor networks has received much attention recently, e.g. secure data communication, secure routing, secure data aggregation, and pairwise key setup etc. has been studied for the application in proposed project.

A software engineering project on Wireless sensor networks (WSN). WSN technology can be potentially applied in many area including manufacturing, agriculture, construction and transportation which is explained in webpage. Habitat and environmental monitoring promises enormous scientific benefits hence is a significant driver for WSN research. Measuring the soil moisture in Pinjar in order learn more about the soil conditions. Monitoring the microclimate of the Redwood trees in order to learn more about the tree from the amount of the moisture absorbed. Monitoring the microclimate in vineyards to aid in the production of better quality wine.

In contrast to many of these application deployments, our sensor network is distinguished by its especially rugged and unique deployment environment, its emphasis on robust design, and its relatively sparse deployment. The task of deploying our network was particularly severe, given the rugged mountainous and forested terrain over which our WSN was spread. Our network covers a unique topology which has not been studied before, ranging from sub-stantial and sharp elevational differences to a fairly wide coverage area spanning about 200 square Kilometers. The network had to be capable of providing both wide area communication coverage and fine-grained local weather sensing coverage. The large elevational differences between our nodes resulted in very different radio propagation models from typical flat-ground, short distance, or in-building deployments and telephone tower deployment. An important design point was therefore robustness in all aspects of the system, ranging from robust physical equipment to robust network routing protocols. Weather sensor networks sought to maximize the information return for each placed sensor node, i.e. our networks were strategically and sparsely deployed to cover as much meaningful terrain with as few nodes as possible. We could not fall back upon dense sensor deployments to provide fault tolerance and redundancy both for the sensed data and for relaying of that sensed data.

CHAPTER- 4

DIFFERENCE FROM EXISTING APPROACHES

4.1 Originally:

The detection system is completely based on metrological data due to which the information is delayed. In the upper Shimla region, there is high yield of apples crop, tons of which gets destroyed due to heavy hailstorm because of lack of efficient alert system every year. Since this system is based on environment parameters and in local to the region, it would help to provide an effective and non-delaying alert system.

4.2 Performance:

Implementing of detection system based on WSN would provide a timely warning based on humidity, light, temperature and wind-speed. Since these sensors are local to the area, hence would provide an efficient and non-delaying alert system which would help to protect the crops from getting destroyed.

4.3 Costs/benefits:

The cost of setting up of sensors is very less compared to the benefits it would provide by saving the huge amount of crops.

CHAPTER- 5

APPROACH FOR SOLVING PROBLEM

5.1 Model

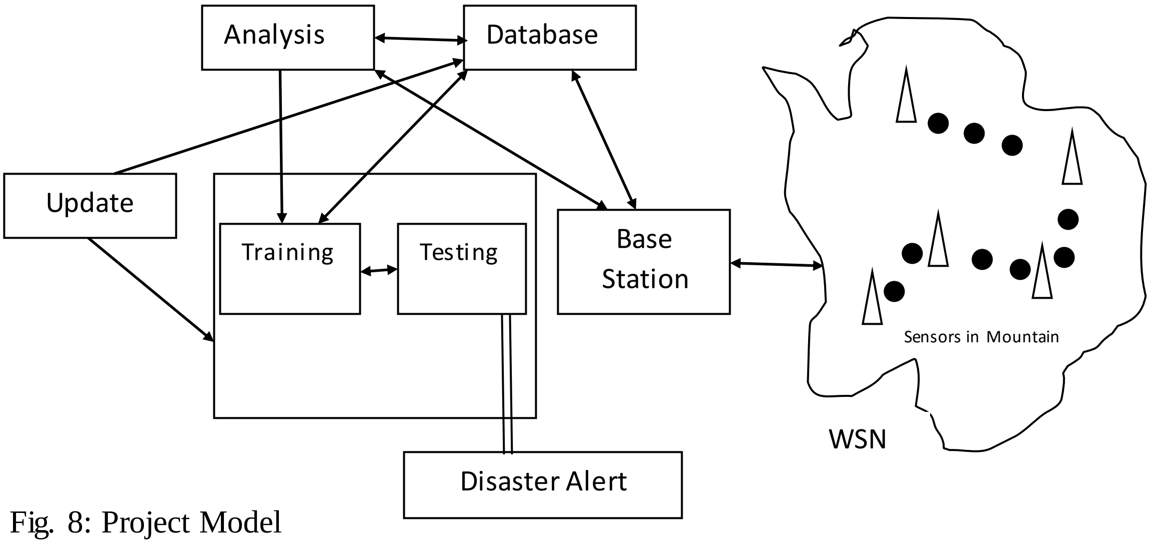

Fig. 8: Project Model

5.2 Data comparison for last 5 years in month of Jan

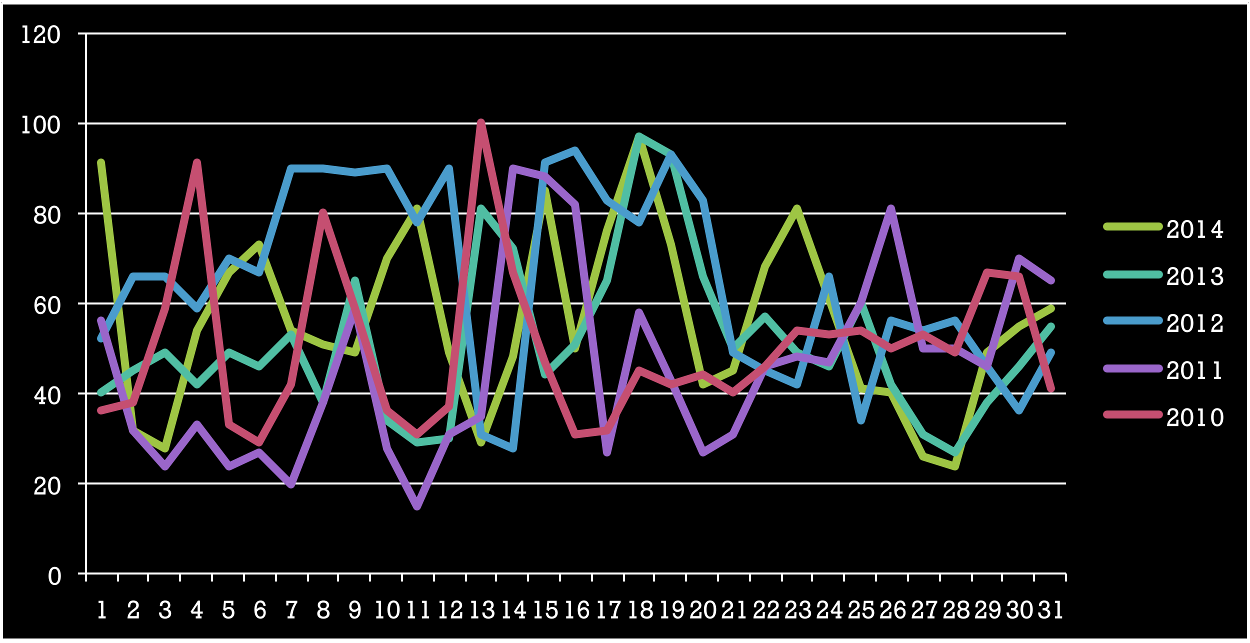

Fig. 9: Comparison on the basis of Humidity

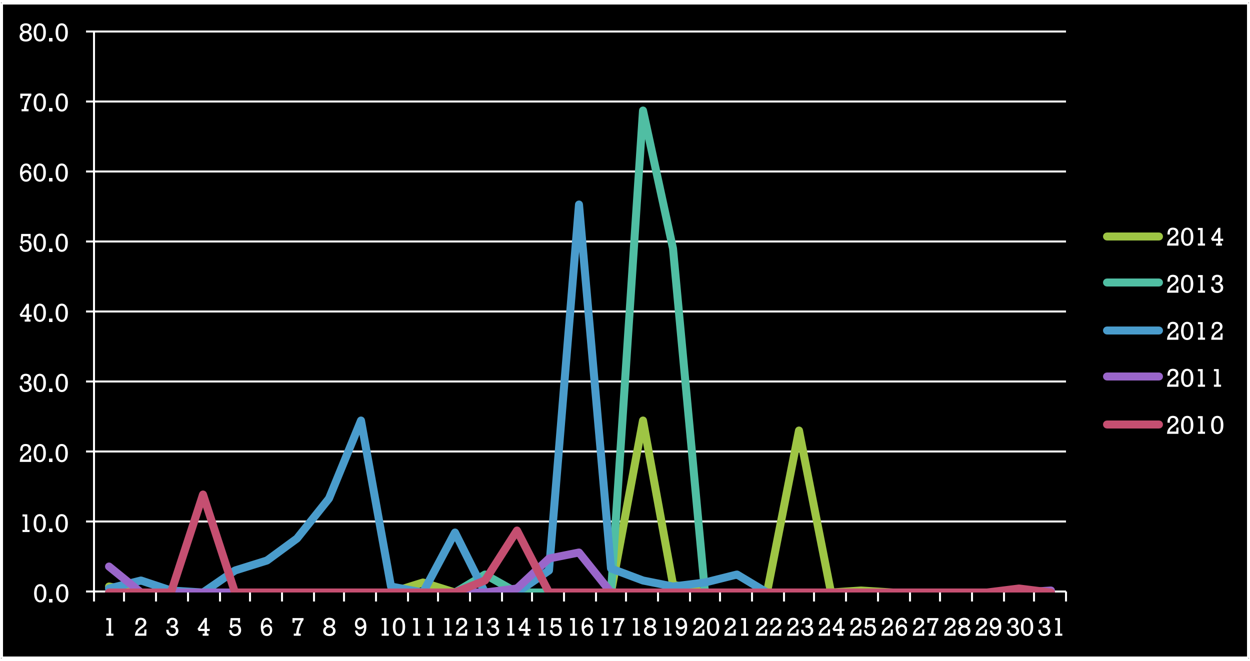

Fig. 10: Comparison on the basis of Rainfall

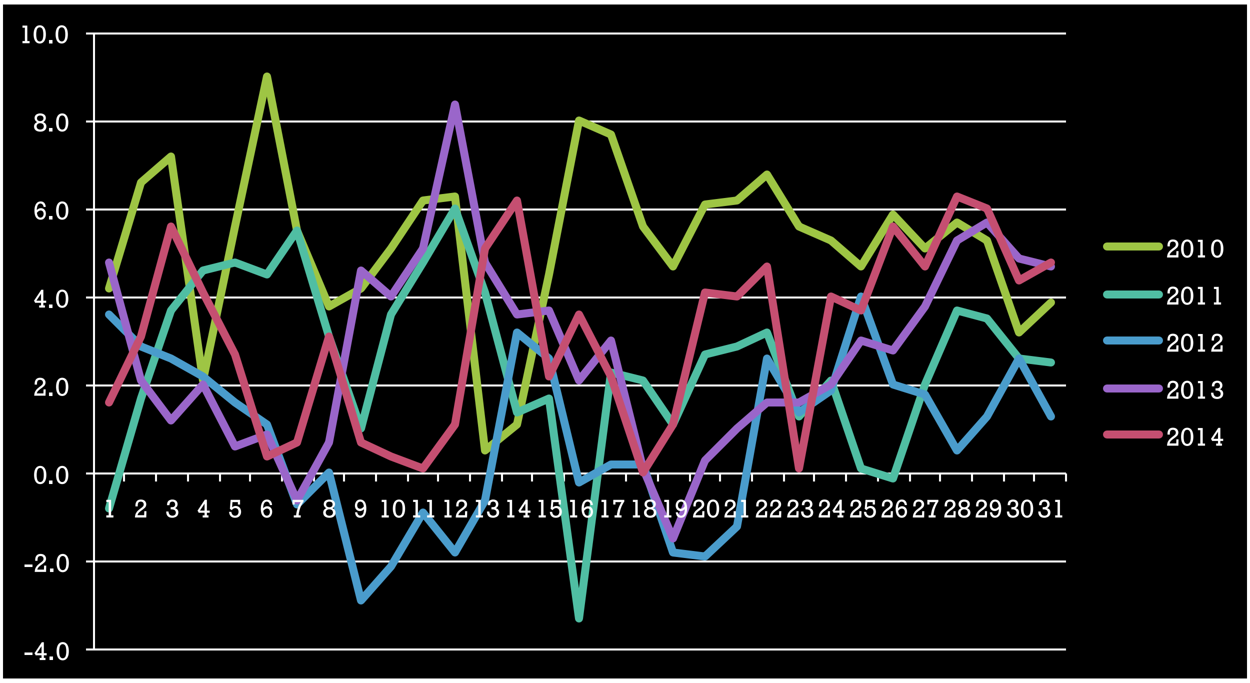

Fig. 11: Comparison on the basis of Min. Temperature

5.3 Data Analysis using Weka

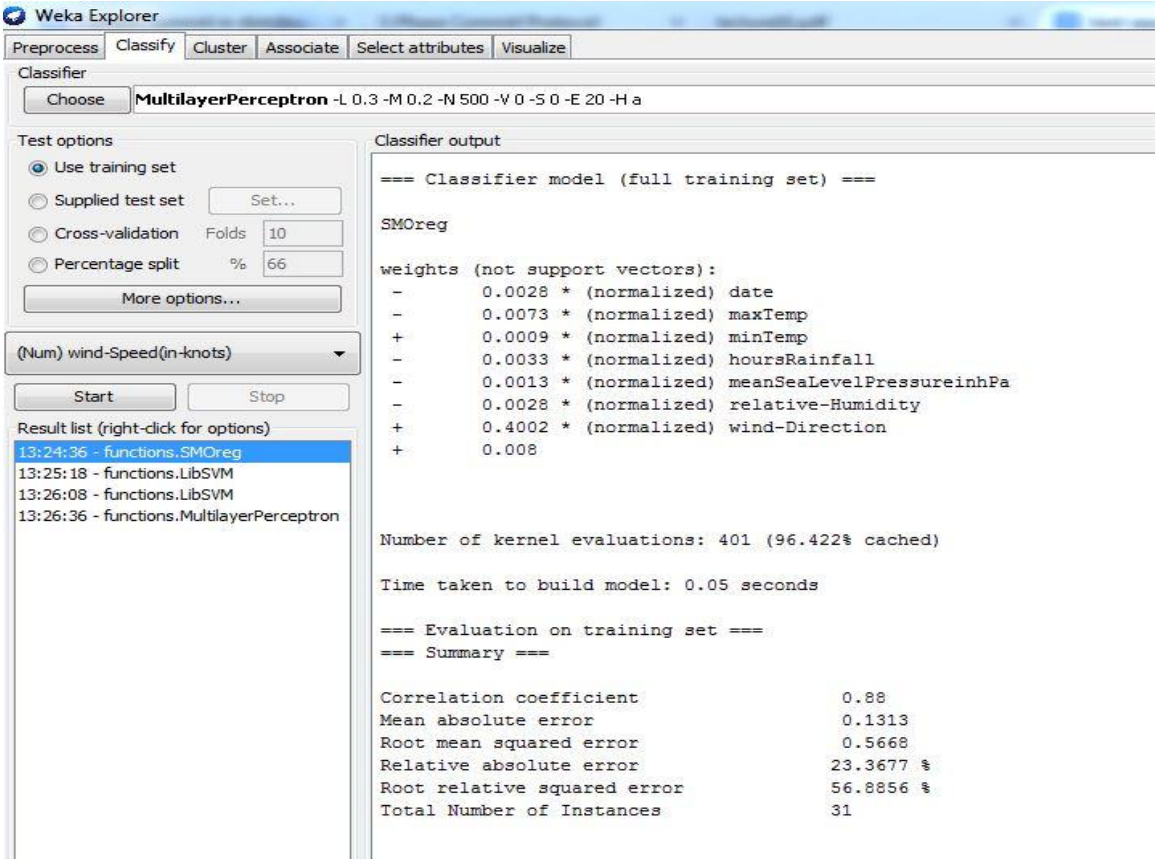

Fig. 12: SMOreg

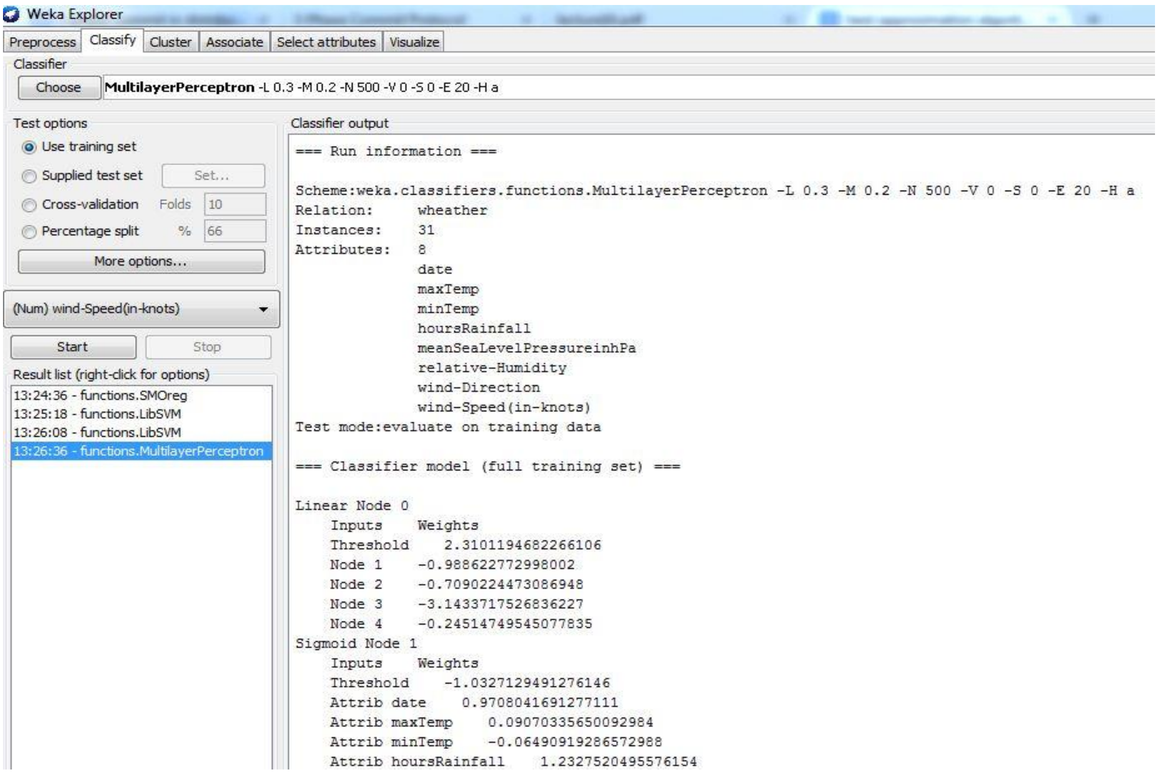

Fig. 13: MultilayerPerceptron

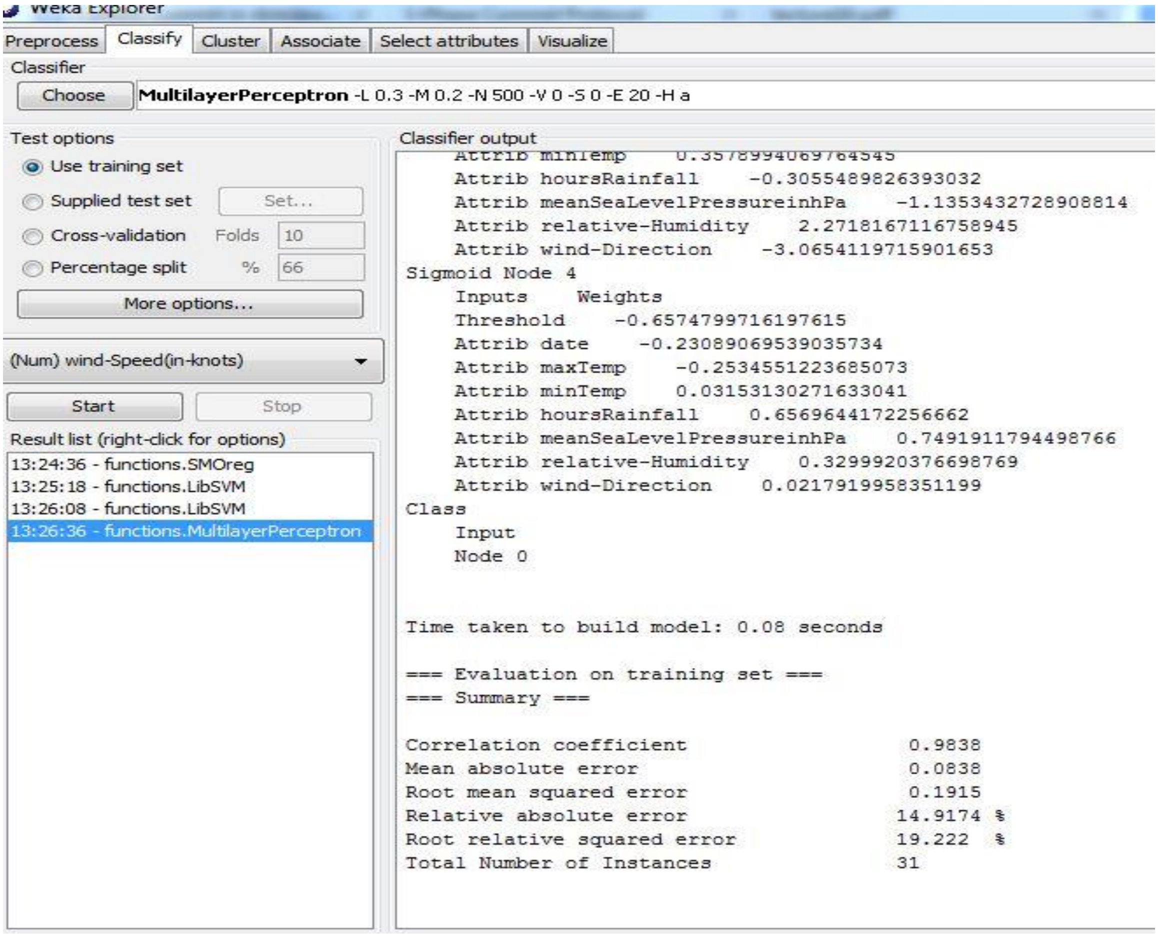

Fig. 14: MultilayerPerceptron (Contt.)

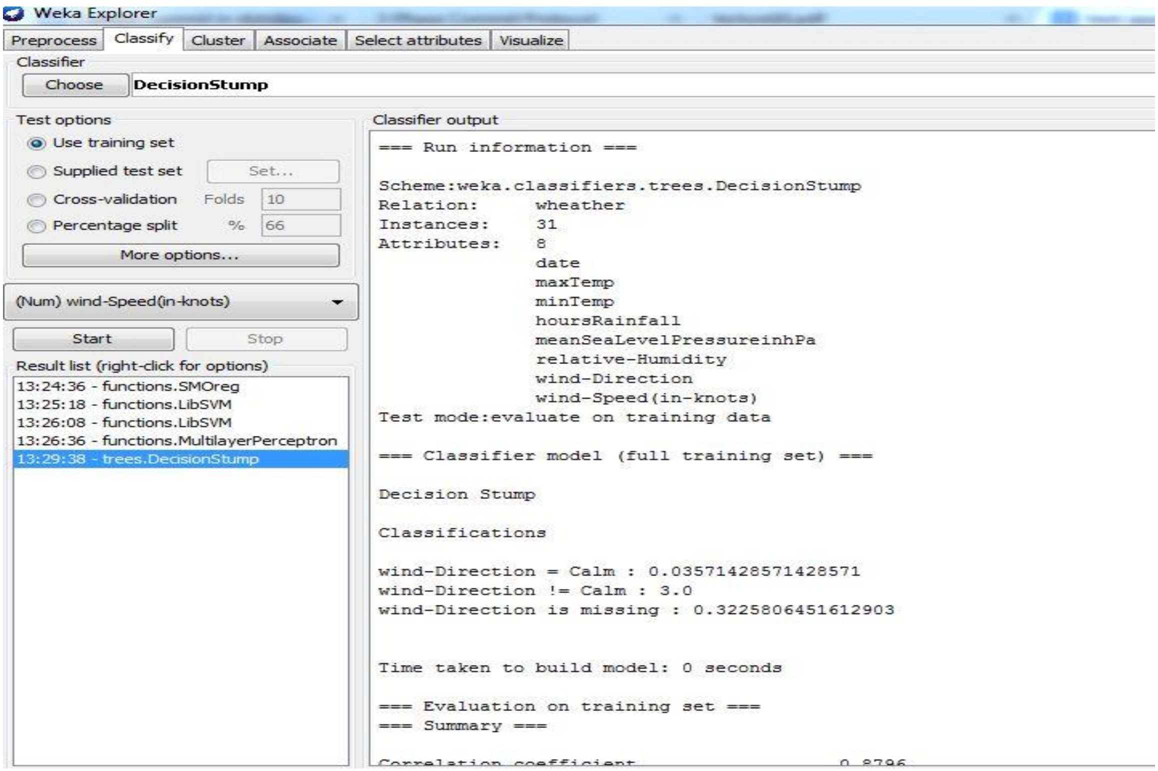

Fig. 15: Decision Stump

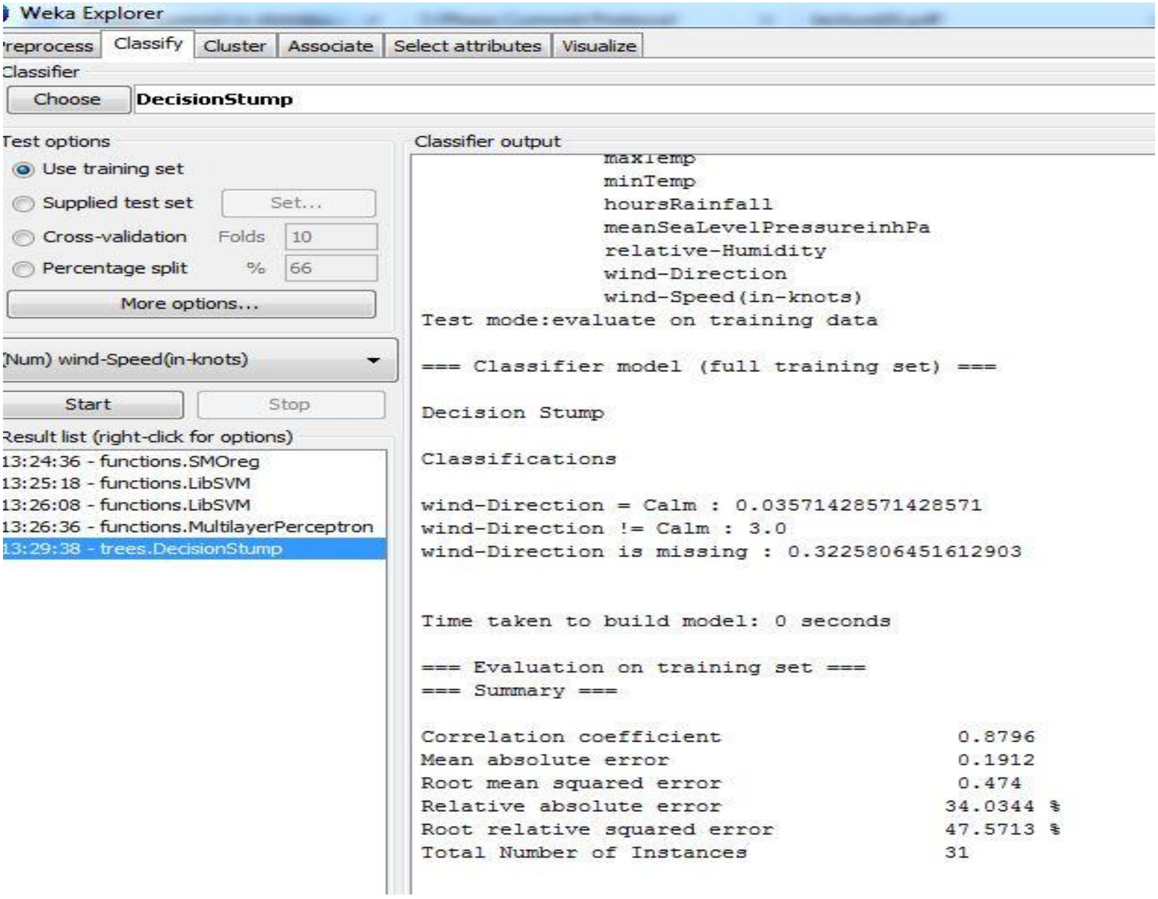

Fig. 16: Decision Stump (Contt.)

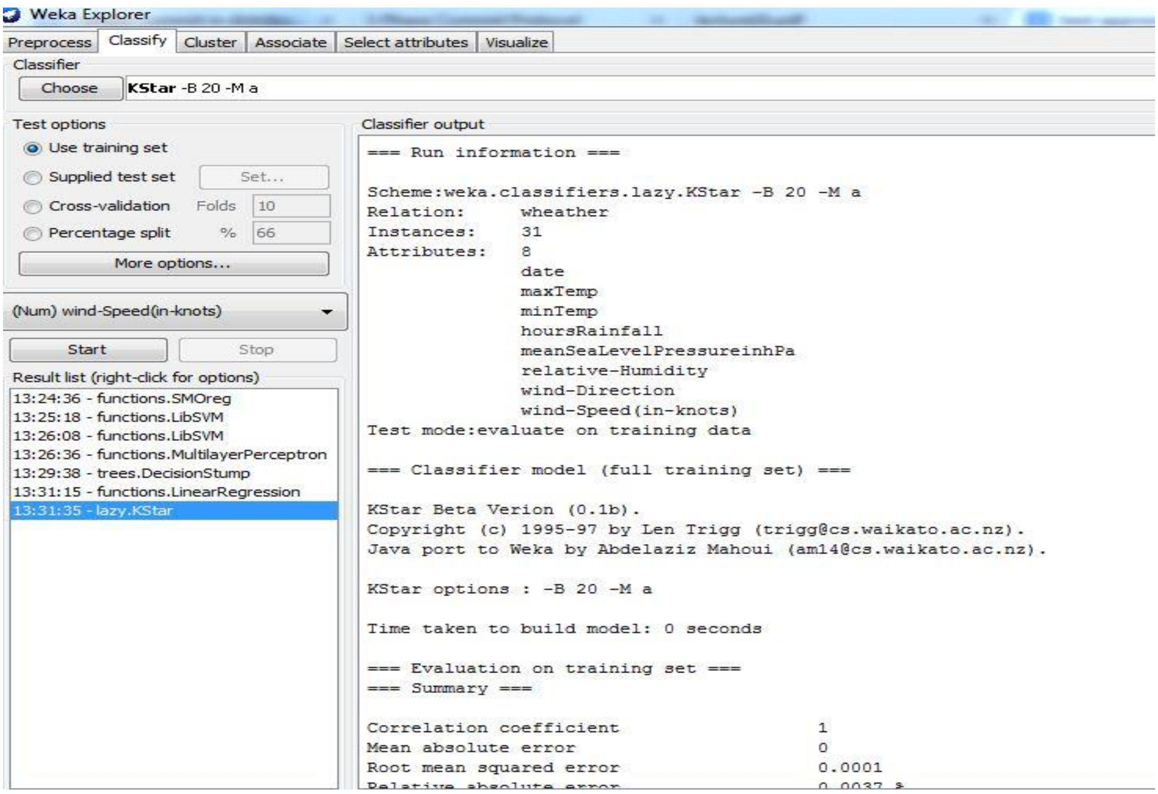

Fig. 17: KStar

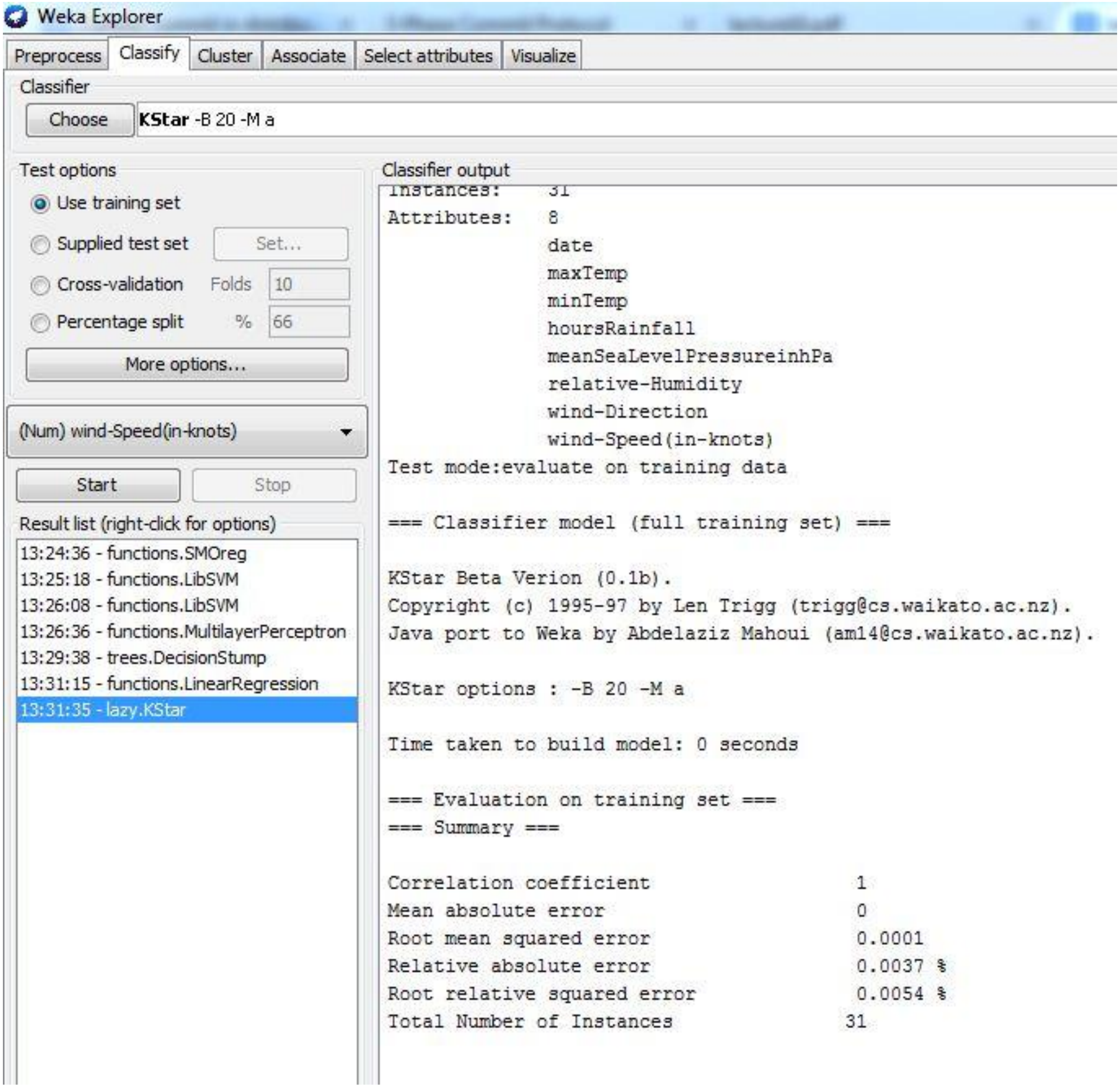

Fig. 18: KStar (Contt.)

5.4 Rainfall Terminology by Meteorological Department

S.No.	Rainfall amount in mms	Descriptive Term Used
1	0.0	No Rain
2	0.1-2.4	Very Light Rain
3	2.5-7.5	Light Rain
4	7.6-35.5	Moderate Rain
5	35.6-64.4	Rather Heavy Rain
6	64.5-124.4	Heavy Rain
7	124.5-244.4	Very Heavy Rain
8	244.5 and above	Extremely Heavy Rain

5.5 Approach for developing algorithm

- Analyzing, comparing the data values to research out threshold values of environmental parameters of last years data collected.
- Taking the input of environmental data and updating the database.
- Comparing the current input to the threshold values.
- Deriving out different combinations of environmental parameters.
- Updating the alert system in case of warning.
- Broadcasting the warning message to the localites.

5.6 Threshold values on the basis of analysis

Rainfall : 110 mms

Humidity : 95 %

Temperature : 4 Celsius

Wind speed : 9 Kmph (4.5 Knots)

5.7 Results

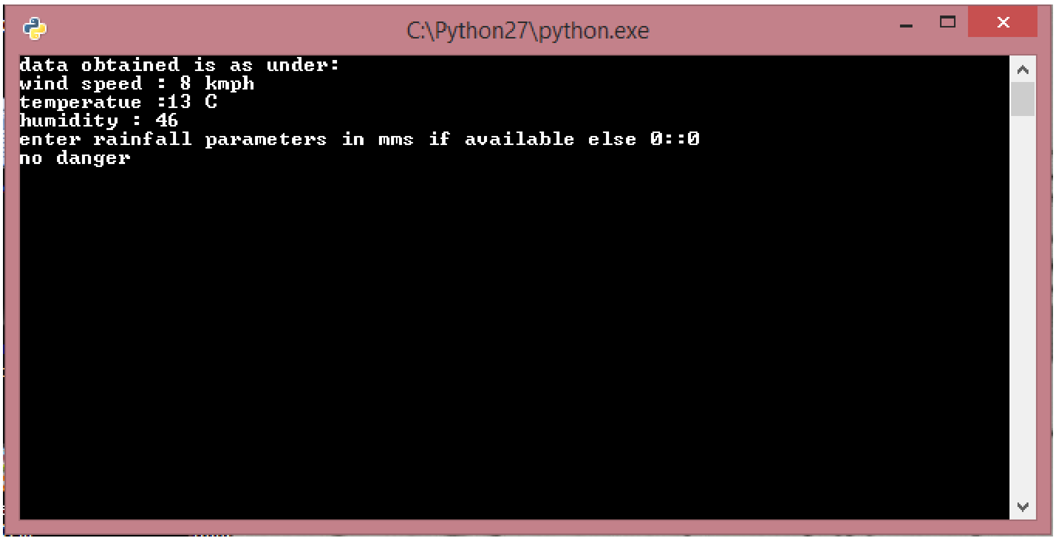

Fig 19: Screenshot of python application

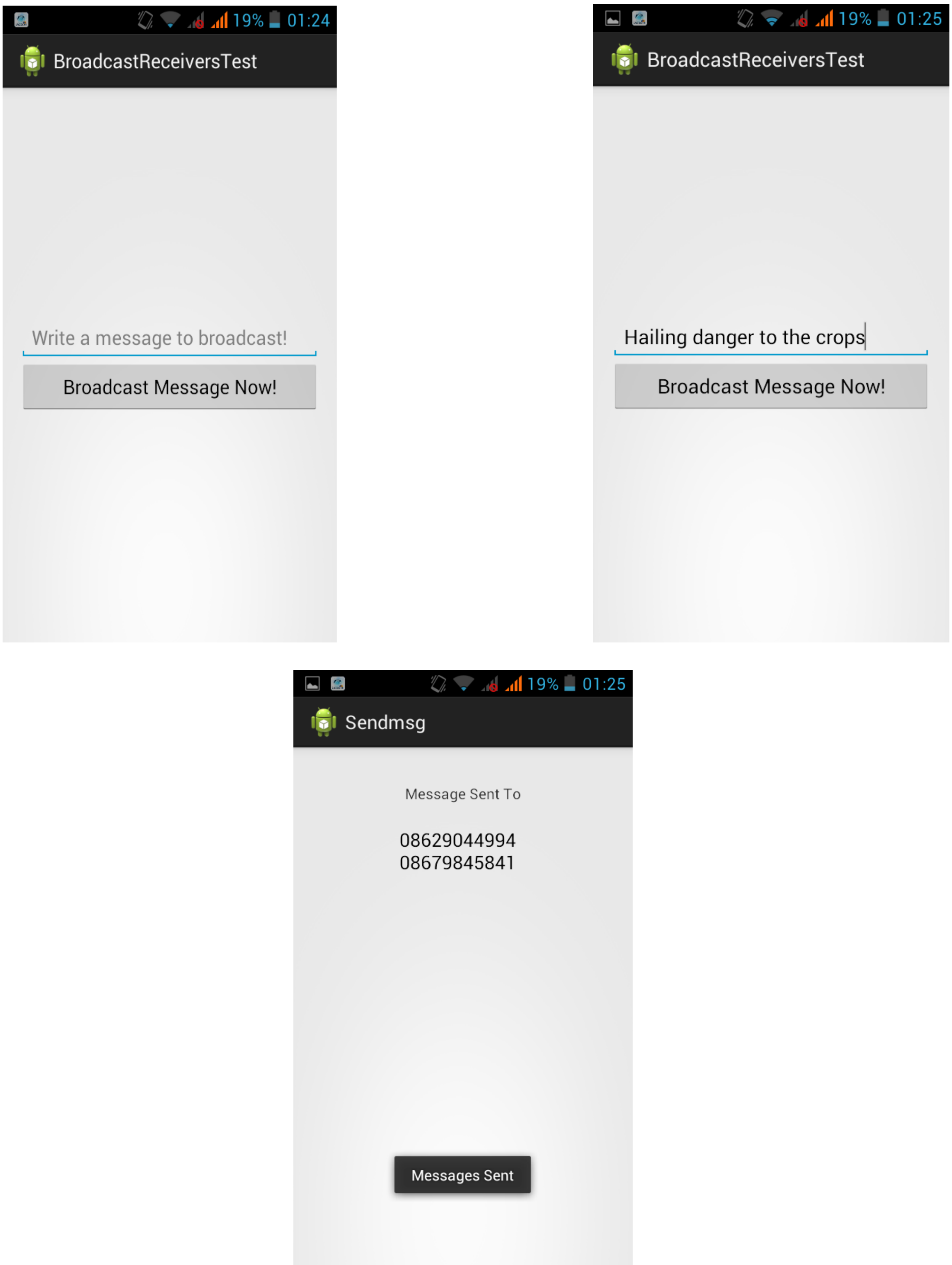

Fig 20: Broadcasting of warning alert messages using android app.

5.8 Problems faced in implementation

- Analysis of data to find out appropriate combinations of environmental parameters which varies for different locations.
- Gathering of past years environmental information.
- Developing mechanism for the alert based system.

CHAPTER- 6

PROS AND CONS

6.1 Critiques – Applications, Advantages

- This application helps to provide timely warning for cloudburst and hailing.
- The Broadcast application available helps to reach the locals to the region instantaneously.
- This would help to save major apple crops in Upper Shimla region which gets destroyed due to inefficient warning mechanism.

6.2 Disadvantages

- The environmental combinations of parameters may vary as per different regions due to height and location.
- The environmental parameters' threshold values cannot be generalized

Conclusion

An efficient detection system for hailstorm and cloudburst detection would help to provide timely warning and broadcasting through message alert would help in providing warning to the members of locality with an ease depending on previously stored database of contact numbers. The mechanism for detection would serve as an efficient mechanism for cloudburst and hailing detection as the system depends on local environment parameters such as rainfall, humidity, temperature and wind-speed for its prediction.

PROPOSED FUTURE WORK

- Connecting of the software to the Wireless Sensors that update the software for the input environment parameters.
- Developing a generalized approach for threshold parameters of all locations based on altitude and location coordinates.

BIBLIOGRAPHY

[1] Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson. Wireless sensor networks for habitat monitoring. In 1st ACM International Workshop on Wireless Sensor Networks and Applicatoins (WSNA 2002), Atlanta, GA, September 2002.

[2] Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao. Habitat monitoring: Application driver for wireless communications technology. In ACM Sigcomm Workshop on Data Communications, San Jose, Costa Rica, April 2001.

[3] W. Hu, V. Tran, N. Bulusu, C. tung Chou, S. Jha, and A. Taylor. The design and evaluation of a hybrid sensor network for cane-toad monitoring. In Information Processing in Sensor Networks (IPSN 2005), Los Angeles, CA, April 2005.

[4] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-S. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: Design tradoffs and early experiences with zebranet. In Arthicetrual Support for Programming Languages and Operating Systems (ASPLOS 2002), October 2002.

[5] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong. A macroscope in the redwoods. In Third International Conference on Embedded Networked Sensor Systems (Sensys 2005), San Diego, CA, November 2005.

[6] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh. Monitoring volcanic eruptions with a wireless sensor network. In Second European Workshop on Wireless Sensor Networks (EWSN ’05), January 2005.

[7] D. Doolin and N. Sitar. Wireless sensors for wildfire monitoring. In SPIE Symposium on Smart Structures & Materials, San Diego, CA, March 2005

[8] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. Tygar. Spins: Security protocols for sensor networks. *Wireless Networks Journal(WINET)*, 8(5):521–534, September 2002.

[9] C. Karlof and D.Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. *AdHoc Networks*, 1(2-3), September 2003.

[10] L. Hu and D. Evans. Using directional antennas to prevent wormhole attacks. In *11th Annual Network and Distributed System Security Symposium*, San Diego, CA, USA, February 2004.

[11] J. Deng, R. Han, and S. Mishra. Inrusion tolerance and anti-traffic analysis strategies in wireless sensor networks. In *IEEE 2004 International Conference on Dependable Systems and Networks (DSN'04)*, Florence, Italy, June 2004.

[12] B. Przydatek, D. Song, and A. Perrig. Sia: Secure information aggregation in sensor networks. In *ACM SenSys'03,* Los Angeles, CA, USA, November 2003.

[13] L. Eschenauer and V. Gligor. A key-management scheme for distributed sensor networks. In *Conference on Computer and Communications Security, (CCS'02),* Washington DC, USA, November 2002.

[14] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *IEEE Symposium on Security and Privacy*, May 2003.

[15] W. Du, J. Deng, Y. Han, and P. Varshney. A pairwise key pre-distribution scheme for wireless sensor networks. In *10th ACM Conference on Computer and Communications Security (CCS'03)*, Washington D.C, USA, October 2003.

[16] D. Liu and P. Ning. Establishing pairwise keys in distributed sensor networks. In *CCS'03*, Washingon D.C, USA, October 2003.

[17] S. Zhu, S. Setia, and S. Jajodia. Leap: Efficient security mechanisms for large-scale distributed sensor networks. In *10th ACMConference on Computer and communications Security*, Washington D.C, USA, October 2003.

[18] A. Parsons. Software Engineering a Wireless Sensor Network for Environmental Monitoring. Honours Project, December 2003.

[19] D. Culler, D. Estrin and M. Srivastava. Overview of Sensor Networks. webpage, http://www.motelab.org/papers/sn-culler.pdf

[20] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler and J. Anderson. ”Wireless Sensor Network for Habitat Monitoring. Proceedings of ACM International Workshop on Wireless Sensor and Application, 2002.

[21] R. Cardell-Oliver, K. Smettem, M. Kranz and K. Mayer. Field Testing a Wireless Sensor Network for Reactive Monitoring. Technical Report UWA-CSSE-04-003, August 2004.

[22] Redwoods go high tech: Researchers use wireless sensors to study California's state tree. webpage, http://www.motelab.org/papers/sn-culler.pdf

[23] J. Burrell, T. Brooke and R. Beckwith. Vineyard Computing: Sensor Networks in Agricultural Production. Pervasive Computing, 2004.

APPENDIX

Detection Software code in Python

```
import urllib2
proxy_support = urllib2.ProxyHandler({"http":"http://172.16.73.12:8090"})
opener = urllib2.build_opener(proxy_support)
urllib2.install_opener(opener)
html =
urllib2.urlopen("http://in.weather.com/weather/10day/Shimla+HP+India+INXX0195:
1:IN").read()

humid_thresh=95
rain_thresh=110
temp_thresh=4
wind_thresh=9

temp_index=html.index('p class="wx-temp"')
temp_index=temp_index+16+3

if html[temp_index+1] != ' ':
    temp=html[temp_index]+html[temp_index+1]
else:
    temp=html[temp_index]

temp=int(temp)



wind_index=html.index('ESE')
wind_index=wind_index+7
if html[wind_index+1] != ' ':
    temp=html[wind_index]+html[wind_index+1]
```

```
else:
    wind=html[wind_index]

wind=int(wind)

humid_index=html.index('Humidity')
humid_index=humid_index+18
if html[humid_index+1] != ' ':
    humid=html[humid_index]+html[humid_index+1]
else:
    humid=html[humid_index]

humid=int(humid)

print 'data obtained is as under:'
print 'wind speed : %d kmph' %wind
print 'temperatue :%d C' %temp
print 'humidity : %d ' %humid
rain=input("enter rainfall parameters in mms if available else 0::")

if rain>rain_thresh:
    print("warning alert")

elif wind > wind_thresh and temp<temp_thresh and humidity > humid_thresh :
    print("warning alert")
else:
    print("no danger")


raw_input()
```

Main activity of Broadcast Receiver android application

```
package com.javacodegeeks.android.broadcastreceiverstest;

import com.javacodegeeks.android.broadcastreceiverstest.R;

import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;
import android.view.View;
import android.widget.EditText;

public class MainActivity extends Activity {

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
   }



   @Override
   public boolean onCreateOptionsMenu(Menu menu) {
      // Inflate the menu; this adds items to the action bar if it is present.
      getMenuInflater().inflate(R.menu.main, menu);
      return true;
   }

   public void broadcastCustomIntent(View view)
   {
     Intent intent = new Intent(MainActivity.this,Sendmsg.class);
```

```
        EditText  et = (EditText)findViewById(R.id.extraIntent);
        // add data to the Intent
        intent.putExtra("message",et.getText().toString());
        startActivity(intent);
    }

}
```

Receivers List of Broadcast Receiver android application

```
package com.javacodegeeks.android.broadcastreceiverstest;

import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.telephony.SmsManager;
import android.view.Menu;
import android.widget.TextView;
import android.widget.Toast;

public class Sendmsg extends Activity {
	TextView tv1,tv2;
	@Override
	protected void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.activity_sendmsg);
		tv1=(TextView)findViewById(R.id.tv1);
		tv2=(TextView)findViewById(R.id.tv2);
		Intent intent = getIntent();
CharSequence intentData = intent.getStringExtra("message");
//tv1.setText(intentData);
		String phoneNumber[] = {"08629044994","08679845841"};
		String nmbrs="";
		String smsBody = intentData+"";
		SmsManager smsManager = SmsManager.getDefault();
		for(int i=0;i<phoneNumber.length;i++)
		{
			nmbrs+=phoneNumber[i]+"\n";
		}
		tv2.setText(nmbrs);
		for(int i=0;i<phoneNumber.length;i++)
```

```
            {
                String pn=phoneNumber[i];
            smsManager.sendTextMessage(pn, null, smsBody, null, null);
            }
            Toast.makeText(getApplicationContext(), "Messages Sent",
                          Toast.LENGTH_LONG).show();

        }

        @Override
        public boolean onCreateOptionsMenu(Menu menu) {
            // Inflate the menu; this adds items to the action bar if it is present.
            getMenuInflater().inflate(R.menu.sendmsg, menu);
            return true;
        }
}
```

Weka Code File

```
@relation 2010wheather
@atrribute Date                                   numeric
@atrribute MaxTemp.                               numeric
@atrribute MinTemp.                               numeric
@atrribute 24_hours_Rainfall_(in mm)                    numeric
@atrribute Mean_Sea_Level_Pressure_(in hPa),            numeric
@atrribute Relative_Humidity_at_0830_IST_(in %)numeric
@atrribute Wind_Direction                         {calm,NE}
@atrribute Wind_Speed(in knots)                         numeric

@data
1,17.5,8.2,0.0,1642.0,28,Calm,0
2,14.0,3.1,0.0,1670.5,48,Calm,0
3,16.4,5.1,0.0,1685.3,48,Calm,0
4,15.2,4.8,0.0,1666.4,65,Calm,0
5,17.1,5.6,0.0,1655.0,54,Calm,0
6,17.9,6.3,0.0,1674.0,27,Calm,0
7,18.0,6.2,0.0,1674.0,29,Calm,0
8,18.0,6.1,0.0,1651.8,38,Calm,0
9,15.9,3.9,0.0,1644.0,62,Calm,0
10,12.0,3.8,0.0,1617.0,84,Calm,0
11,13.2,4.3,0.0,1608.5,72,Calm,0
12,15.7,4.0,0.0,1569.4,21,NE,2
13,13.7,3.9,0.0,1612.8,55,Calm,0
14,15.3,3.2,0.0,1650.0,45,Calm,0
15,17.7,5.3,0.0,1635.6,20,Calm,0
16,17.5,6.2,0.0,1644.0,16,Calm,0
17,15.1,4.4,0.0,1650.0,37,Calm,1
18,13.1,2.7,0.0,1650.0,86,Calm,0
19,14.6,4.2,0.0,1668.5,27,Calm,0
20,15.1,5.8,0.0,1672.0,11,NE,5
```

21,17.6,7.6,0.0,1649.0,12,NE,2

22,15.1,6.4,0.0,1612.0,17,Calm,0

23,13.8,3.2,0.0,1619.0,48,Calm,0

24,14.0,2.7,0.0,1644.0,43,Calm,0

25,13.4,3.1,0.0,1635.8,41,Calm,0

26,12.7,3.0,0.0,1643.0,55,Calm,0

27,15.9,3.4,0.0,1652.0,21,Calm,0

28,14.2,4.5,0.0,1664.4,19,Calm,0

29,10.5,6.2,0.0,166305.0,44,Calm,0

30,5.1,4.1,18.6,1650.0,97,Calm,0

31,4.4,0.1,63.2,1620.8,94,Calm,0