KIET JOURNAL OF COMPUTING AND INFORMATION SCIENCES

ISSN: 2616-9592

Volume : 1 | Issue : 1 | July-December | 2018

KIET JOURNAL OF COMPUTING AND INFORMATION SCIENCES

Volume 1, Issue 1, 2018
ISSN: 2616-9592
Frequency Bi-Annual

College of Computing & Information Sciences
Pakistan Air Force - Karachi Institute of Economics & Technology

College of Computing & Information Sciences

Vision

To develop technology entrepreneurs & leaders for national & international market

Mission

To produce quality professionals by using diverse learning methodologies, aspiring faculty, innovative curriculum and cutting edge research, in the field of computing & information sciences.

AIMS AND SCOPE

KIET Journal of Computing and Information Sciences (KJCIS) is the bi-annual, multi-disciplinary research journal published by **College of Computing & Information Sciences (CoCIS)** at **Pakistan Air Force - Karachi Institute of Economics and Technology (PAF-KIET)**, Karachi, Pakistan. **KJCIS** aims to provide a panoramic view of the state of the art development in the field of computing and information sciences at global level.

It provides a premier interdisciplinary platform to researchers, scientists and practitioners from the field of computing and information sciences to share their findings and contribute to the knowledge domain at global level. The journal also fills the gap between academician and industrial research community.

KJCIS focused areas for publication includes; but not limited to:

- Data mining
- Big data
- Machine learning
- Artificial intelligence
- Mobile applications
- Computer networks
- Cryptography and information security
- Mobile and wireless communication
- Adhoc and body area networks
- Software engineering
- Speech and pattern recognition
- Evolutionary computation
- Semantic web and its application
- Data base technologies and its applications
- Internet of things (IoT)
- Computer vision
- Distributed computing
- Grid and cloud computing

OPEN ACCESS POLICY

For the benefit of authors and research community, this journal adopts open access policy, which means that the authors can self-archive their published articles on their own website or their institutional repositories. The readers can download or reuse any article free of charge for research, further study or any other non profitable academic activity.

PEER REVIEW POLICY

Peer review is the process to uphold the quality and validity of the published articles. KJCIS uses double-blind peer review policy to ensure only high-quality publications are selected for the journal. Papers are referred to at least two experts as suggested by the editorial board. All publication decisions are made by the journal's Editors-in-Chief on the basis of the referees' reports. We expect our Board of Reviewing Editors and reviewers to treat manuscripts as confidential material. The identities of authors and reviewers remain confidential throughout the process.

COPYRIGHT

DISCLAIMER

The opinions expressed in **KIET Journal of Computing and Information Sciences (KJCIS)** are those of the authors and contributors, and do not necessarily reflect those of the journal management, advisory board and the editorial board. Papers published in KJCIS are processed through double blind peer-review by subject specialists and language experts. Neither the **CoCIS** nor the editors of **KJCIS** can be held responsible for errors or any consequences arising from the use of information contained in this journal, instead; errors should be reported directly to the corresponding authors of the articles.

Patron's Message

Pakistan Air Force - Karachi Institute of Economics and Technology (PAF-KIET) has been imparting education with its core values of merit, quality, and excellence since its inception in the year 1997. Higher Education Commission recognized KIET in 1998 and in the year 2000, KIET was awarded the degree-awarding status. Currently, more than 5000 students are pursuing various degree programs at PAF-KIET in the field of Computing & Information Science, Management Sciences, Engineering and Media & Arts.

College of Computing and Information Science [CoCIS] at PAF-KIET is working with a clear vision of producing high quality technology entrepreneurs & market leaders for national & international market. We are focused to improve the quality of our students by using project based learning methodology, industry aspiring faculty and innovative curriculum; consisting of theoretical understanding, hands on skills and cutting-edge research in the field of computing and information sciences.

In the realm of research and development, we believe that research is an integral part of modern development and extremely essential for improving the quality of life of the human race. In this context, the launch of KIET Journal of Computing and Information Sciences (KJCIS) is a modest effort towards contributing to and promoting a research environment within the institution and Pakistan as a whole. The Journal is a peer-reviewed and multi-disciplinary research journal. It is expected to provide a platform to publish findings and results of the latest and innovative research work in various areas of Computing and Information Sciences. We are confident that in the years to come, the Journal would achieve international recognition and make a name for itself.

I wish the Editorial team the very best in achieving their objectives.

AVM Tubrez Asif (Ret'd PAF)
President PAF-KIET
Patron KJCIS

Editor's Note

I feel immense pleasure to present before you the first issue of KIET Journal of Computing and Information Sciences (KJCIS). I would also like to point out that College of Computing and Information Sciences is dedicated on creating a research culture through various activities. These includes incorporating the most recent research areas in the curriculum, holding international conferences and inviting the industry experts to advice on the development of teaching contents and methodology. Launch of KJCIS is a serious effort in the same direction and shows our hope and commitment to create a research-based institution. The KJCIS will provide an interdisciplinary platform for researchers, scientists, practitioners and academicians for publishing their contributions in the recent technological advances and innovations specific to the area of Computing and Information Sciences and disseminate it to a large number of stakeholders, one of the key stakeholders being the students.

Original research findings in the field of Computing and Information Sciences will be ensured through double-blind peer-review process. The KJCIS's editorial board comprises of the leading academic and industrial researchers. The Journal has adopted the open access policy and will not charge any publication fees to attract wider group of interested researchers.

On behalf of the KJCIS, I welcome the submissions for upcoming issue (Volume-1, Issue-1, January - June 2018) and look forward to receive your valuable feedback.

Sincerely,

Prof. Dr. Muzaffar Mahmood
Dean PAF-KIET
Chief Editor KJCIS

Editorial Board

Language Editors

Mr. Ayub Latif, Mr. Sohail Imran, Ms. Solat Jabeen, Ms. Huma Tariq
CoCIS, PAF-KIET, Pakistan

Production Management

Mr. Furqan Abbasi
CoCIS, PAF-KIET, Pakistan

Table of Content

Optimizing Energy Consumption using Fuzzy Logic for HEMS in a Smart Grid

Qurat-ul-Ain[1] Sohail Iqbal[2] Nadeem Javaid[3]

Abstract

Energy consumption minimization and user comfort enhancement in Home Energy Management System (HEMS) are the major challenges in a smart grid. In HEMS, appliances of Heating, Ventilation, and Air Conditioning (HVAC) have a large impact on the energy consumption. For user comfort, one needs to take into account different environmental factors among which humidity plays an important role in determining the suitable temperature for optimal user comfort. In order to minimize energy consumption without compromising user comfort, fuzzy logic techniques are widely used without considering humidity. In this paper, we tune the Fuzzy Inference System (FIS) by including humidity as well as we propose a method for the automatic rule generation for FIS. Automatic rule generation is devised using combinatorics. The proposed system is evaluated by the membership functions of the input parameters and the results are compared using Mamdani FIS and Sugeno FIS. Indoor temperature, outdoor temperature, occupancy, price, initialized set points of thermostat, and humidity are the input parameters of the system. Performance metrics used for the evaluation are energy consumption, Peak-to-Average Ratio (PAR), cost, and efficiency gain. Simulation of one month energy consumption with proposed technique is performed in MATLAB®. Simulation results validate the proposed technique and show that despite all the energy savings, the proposed technique manages to be in the user comfort zone while achieving electricity cost reduction up to 24%. Moreover, optimization using FIS provides the reduced energy consumption up to 28%. The proposed technique seems to have a potential for improved demand-side energy management in a smart grid.

Keyword: Smart Grid, Fuzzy Logic, Energy Management, User Comfort

1 Introduction

Demand Response (DR) plays an important role in energy consumption minimization for the smart grid environment. HEMS is a demand response tool which is an important part of the smart grid that enables the residential users to create optimal energy consumption by considering many objectives such as energy cost, load profiles, and consumer comfort. The worldwide fossil fuel resources are declining at an escalated pace which signifies the need of energy management and minimization [1].Heating, Ventilation, and Air Conditioning (HVAC)

[1]*SEECS, National University of Sciences and Technology (NUST), Islamabad*
[2]*SEECS, National University of Sciences and Technology (NUST), Islamabad*
[3]*COMSATS Institute of Information Technology, Islamabad*
**Corresponding author: sohail.iqbal@seecs.nust.edu.pk*

appliances contribute the major part of total energy consumption worldwide in the current scenario. HVAC also shows great effects on peak load management during peak demand hours especially during summer days. Therefore, different techniques are used to schedule the HVAC to reduce peak load. Different pricing mechanisms or tariffs such as Real Time Pricing (RTP), Time of Use (TOU) [2] and critical peak pricing [3] are used to encourage the electricity user for reducing load demand during peak hours. Electricity cost is determined by the utility at different times of day based on TOU rates for the High Peak (HP), Mid Peak (MP), and Off Peak (OP). Based on these values, load control in HEMS is either shifted or curtailed especially for the HVAC.

A wind driven optimization based energy scheduling technique [4] is used to reduce the energy cost and PAR by shifting the load during peak hours to off peak hours. PAR is a useful measure which describes how peak electricity consumption affects the system. It is often seen that users find it difficult to remember that they need to update their thermostat particularly during critical situations. In [5] programmable communicating thermostat incorporated model which helps to reduce the PAR along with energy consumption.

Occupancy and user participation are the factors that directly affect the functionality of thermostats resulting in more optimal energy consumption and bill savings. Occupant's activities and presence have been observed in [6] to evaluate the energy savings. The user negligence while using thermostat effects the energy consumption in many cases where customers forget or neglect to participate in DR during peak prices [7].

Previous research shows that evaluation of energy consumption for HVAC by varying different parameters is performed. Energy evaluation is performed using different DR techniques day-ahead electricity prices, TOU rates, real time pricing is used to reduce energy consumption and electricity bills for residential consumers by shifting home appliance from high peak hours to off peak hours [8],[9]. Challenges that are often faced during the use of programmable thermostat (PT) is user's lack of communication with new technologies such as smart meters [10]. PTs are improved into Programmable Communicating Thermostat (PCT) [11] with the advancement in communication benefitting users to participate in DR program promulgated by the utility. Currently, types of thermostat that are being used to participate in DR are: 1) programmable communicating thermostat, 2) price responsive thermostats, and 3) occupancy responsive thermostat to reduce the residential HVAC energy consumption. Price responsive thermostat uses price signals from smart grid and change the thermostat set points to the values already defined by the residential users. Occupancy based thermostats sense the occupancy of the user and modifies the building or room set points. During these studies, it is observed that user comfort is heavily sacrificed while participating in the demand response programs[11].PCT is used because of its feature of communicating with the smart meter in order to read the price signals that are decided by the utility on interval basis. PCTs allows user to participate in DR programs where user can vary the thermostat set points according to the TOUs tariffs using the intervals of off peak, mid peak, and high peak. However, the constant interaction of the user with thermostat often irritates the residential users making the behavior of PCT as programmable thermostat (PT).

Keshtkar et al. [12] evaluate the load reduction in HVAC system using fuzzy logic model. Outdoor temperature, price, occupancy, and initialized set points are the input parameters of that system. However, the proposed system lacks adaptability in thermostat. The system in [13] is the extension of [12] where authors introduced the adaptive autonomous thermostat. In [13], system is made adaptive using fuzzy logic approach by training the thermostat on initialized set points. It considers the three consecutive changes of same set points for the same day of the week and then modifies the thermostat set point to the optimized set point. Although, this technique is good it is limited to cold regions of the world i.e. Canada in this case. And the results are reliable only for the country based research.

Extending the idea of [13], a model of worldwide adaptive thermostat is proposed by Javaid el al. [14]. The proposed technique uses Fuzzy Logic Controllers (FLC) to set the thermostat set points. Input parameters of this system are outdoor temperature, price, occupancy, and initialized set points for hot and cold cities. Their system is evaluated using Mamdani and Sugeno FIS. Although this technique showed good results in energy consumption minimization there are multiple parameters that can be considered for the improvement of results.

This limitation lead us to extend the existing study for further improvement in the energy consumption minimization using thermostat set points optimization.

The organization of paper is as follows: in Section II, problem considered is elaborated while the problem formulation is described in Section III. Proposed system is presented in IV and simulations along with the result analysis are discussed in Section V. The paper is concluded in Section VI.

2 Problem Statement

Residential HVAC systems contribute to a significant part of world's energy consumption. These devices are the primary electrical load during peak hours which often leads to peak load blackouts. As the energy prices increases during the peak hours such as TOU, the household electricity bill is strongly dependent on the HVAC system. Thermostat is widely used in order to save energy as well as to maintain the temperature of residential building in user desired range. PT is the kind of thermostat used in HEMS where users maintain the set point temperature on interval basis for a day which depicts their schedule and user preferences [13].

Many fuzzy logic techniques have been developed that targets to save energy without compromising the user comfort. It is observed that different factors play a significant role in determining the thermostat set points which in result affect the energy consumption. Some of them have been considered in previous research work but many of them are still missing. Humidity is an important factor that determines the way a user sets the set points. There is a need to design a model that incorporates the humidity to show the effect of humidity while setting the thermostat set points in order to reduce energy consumption without much compromising on the user comfort.

It is known that generating and writing the rules for the FIS is a very tedious and time consuming task. Measuring the effect of different parameters on the energy consumption is a beneficial methodology that will help us in setting the thermostat set points in a way that not only minimizes the energy consumption but the temperature will also remain within the user comfort range. But increase in the number of input parameters increases total numbers of rules to be defined hence increasing the complexity of defining the rules for the FIS. Defining an automatic way to set the rule base is an important task.

3 Problem Formulation

This section discuss the details of the formulation of proposed scheme, energy consumption, cost, PAR for HEMS. The proposed system is developed using fuzzy logic rules and evaluated using FIS. Fuzzy logic technique has a major advantage as compared to ON-OFF control as controlled variables used in this study varies continuously during a period of time [15]. FLC responds very well to these changes. The input and output of FLC are real variables which are connected through IF-THEN rules to achieve the desired output. The major advantage of FLC as compared to other controllers is its requirement of little mathematical modeling. Another reason for using the FLC is that the rules defined are purely on human intuition which is effective and more expressive. Mamdani and Sugeno are among the types of FIS that are most commonly used for evaluation. The input parameters used in this study are directly related to energy management and user comfort in residential buildings. Energy consumption is evaluated with the help of fuzzy rules. In this paper, energy consumption is calculated by considering with humidity and without humidity.

A. Mamdani and Sugeno FIS

FIS takes the crisp inputs, fuzzifies it, applies fuzzy operators on premise (antecedent), performs implication from premise to conclusion (consequent), aggregate conclusion across fuzzy rules to generate fuzzy output and defuzifies it to get a crisp output. The model proposed is evaluated and tested using the Mamdani and Sugeno FIS.

Mamdani FIS uses linguistic variables for the rules and its premise and conclusion are both linguistic variable. Fuzzyrules are generated using the linguistic variables. e.g.

IF Outdoor-*Temp is "Normal" AND* Indoor-*Temp is "Normal" AND* Rate *is "High Peak" AND Occpancy is "Absent" AND ISP is "Low" AND Humidity is "Low", THEN energyconsumption is "Low"*

Defuzzification method used in Mamdani FIS is centroid which is calculated using the formula [16]:

$$z = \frac{\int \mu_C(z) \,.\, z dz}{\int \mu_C(z) dz} \qquad (1)$$

Sugeno FIS takes the premise part as a linguistic variables, however, its conclusion part is function which can be zero order (constant) or first order. Fuzzy rules are generated using the function which is efficient, for example:

$$IF\ \text{Outdoor-}Temp\ is\ "Normal"\ AND\ \text{Indoor-}Temp\ is\ "Normal"\ AND\ Rates\ is\ "High\ Peak"\ AND\ Occpancy\ is\ "Absent"\ AND\ ISP\ is\ "Low"\ AND\ Humidity\ is\ "Low",\ THEN\ energyconsumption = energyconsumption\ (temp_{in}, temp_{out}, price, occupancy, ISP, humidity)$$

Defuzzification method used in Sugeno FIS is weighted average which is calculated using the formula:

$$z = \frac{\sum \mu_C \bar{z} \cdot \bar{z}}{\sum \mu_C \bar{z}} \qquad (2)$$

To conclude, Mamdani is intuition based that is well suited for the human input whereas Sugeno is computationally efficient method and well suited for the mathematical analysis [17].

In order to calculate the total cost following formula is used:

$$Cost(h) = EC(h) * Rates(h) \qquad (3)$$

Here, Cost (h) is the hourly cost whereas EC (h) is the electricity consumption on hourly basis and Rates (h) are the hourly pricing tariffs based on TOU.

PAR is calculated using the formula as follows:

$$PAR(a_n) = \frac{P_{\max}}{P_{avg}} = \frac{\max\limits_{0 \le n \le N-1} (a_n)}{\frac{1}{N} \sum_{n=0}^{N-1} a_n} \qquad (4)$$

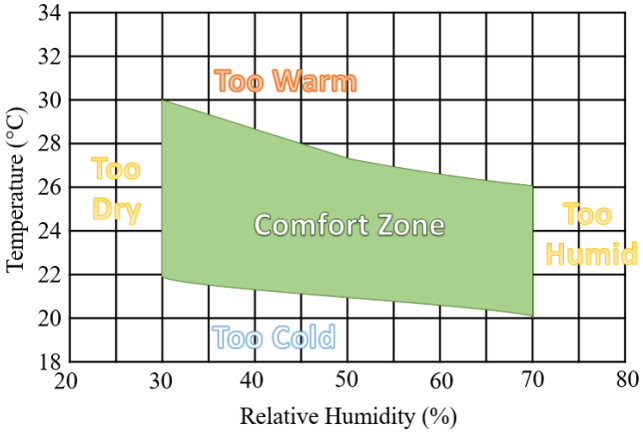


Figure 1: Comfort Zone using Psychrometric Graph [20]

4 System Model

In this manuscript, an extension to the adaptive fuzzy learning model [13] and worldwide adaptive thermostat model [14] is developed. Proposed technique introduces humidity along with the existing input parameters like outdoor temperature, indoor temperature, prices, occupancy, and set points of thermostat in hot and cold cities for energy consumption minimization without disturbing the user comfort.

The two cities considered for analyzing the cooling and heating power consumption in any residential building around the world are the WadiHalfa in Sudan and Yakutsk from Russia. WadiHalfa is one of the hottest cities in the world and Yakutsk is the coldest city in the world. Then, we have selected one of the hottest and coldest day from the respective cities. Highest temperature in WadiHalfa has been recorded during the month of June. Whereas, coldest weather in Yakutsk has been observed during January. Outdoor temperature for WadiHalfa and Yakutsk is taken form weather forecasting website [18] and [19] respectively. The initialized set points are used for controlling the indoor temperature for both cold and hot cities are defined using the psychrometric chart mentioned in [20]. Comfort zone defined in Fig. 1 shows the temperature range that can be used as thermostat set point with respect to a particular relative humidity value which results in little disturbance on user comfort. Values for occupancy and TOU pricing tariff are taken form [13].

In Fig. 2, computation model used in this system is depicted. Input parameters provided to the FIS are measured by deploying sensors. This system works for the cold and hot cities along with their thermostat heating and cooling points. The price is communicated to the user by utility using the smart meters of the residential buildings. These values are used to set thermostat set points.

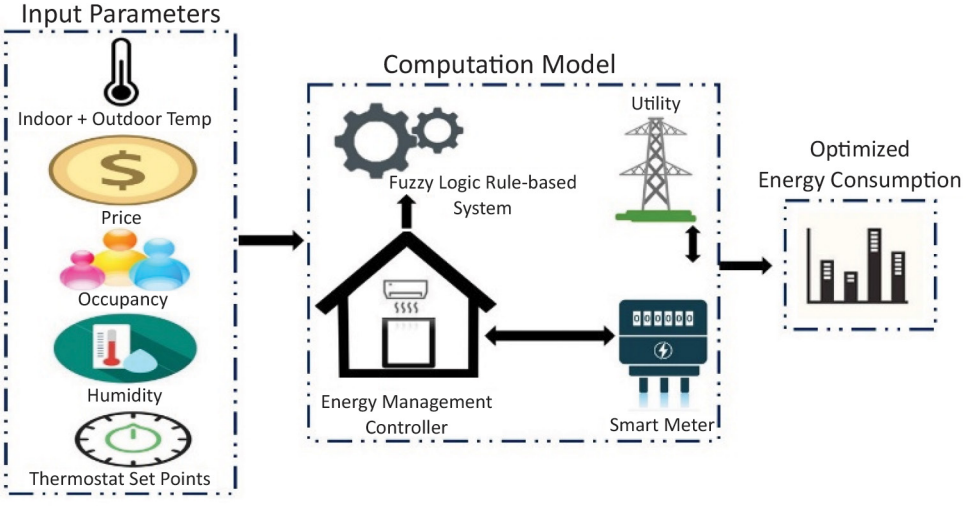


Figure 2: HVAC Control System in HEMS

Inputs of the system are indoor temperature, outdoor temperature, price, occupancy, thermostat set points, and humidity. Membership functions defined for the indoor temperature ($Temp_{indoor}$) and outdoor temperature ($Temp_{outdoor}$) are: 1) Very Cold (VC), 2) Cold (C), and 3) Normal (N) for cold cities whereas for hot cities the membership functions are 1) Normal (N), 2) Hot (H), and 3) Very Hot (VH). The user occupancy (O) has two membership functions: 1) Absent (A) and 2) Present (P). Price (P_{rates}) is defined on the basis of TOU tariff according to which membership functions are defined as: 1) Off Peak (OP), 2) Mid Peak (MP), and 3) High Peak (HP). The membership function used for thermostat set points (ISP_s) and humidity ($Humidity_{rel}$) are: 1) Low (L), 2) Medium (M), and 3) High (H). Output parameter of this system is energy consumption (EC). The output membership functions are 1) Very Low (VL), 2) Low (L), 3) Medium (M), 4) High (H), and 5) Very High (VH).Figs.3-5 shows the membership functions of some parameters used in this system as well as the defined ranges:

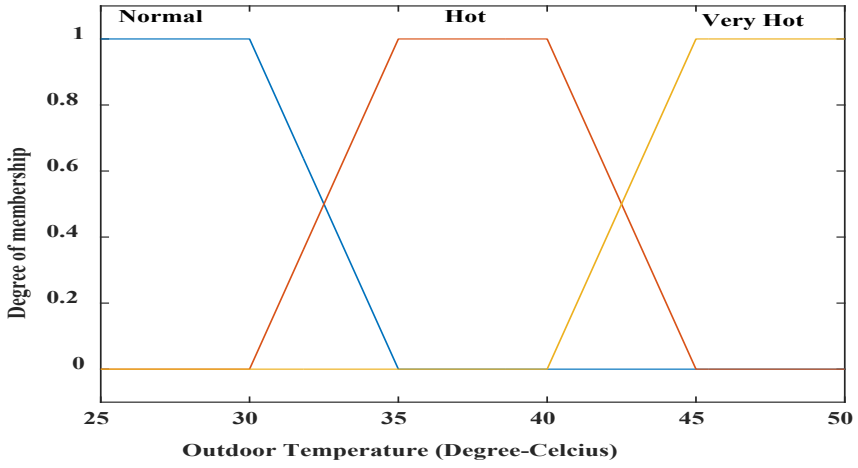


Figure 3: Outdoor Temperature for Hot Cities

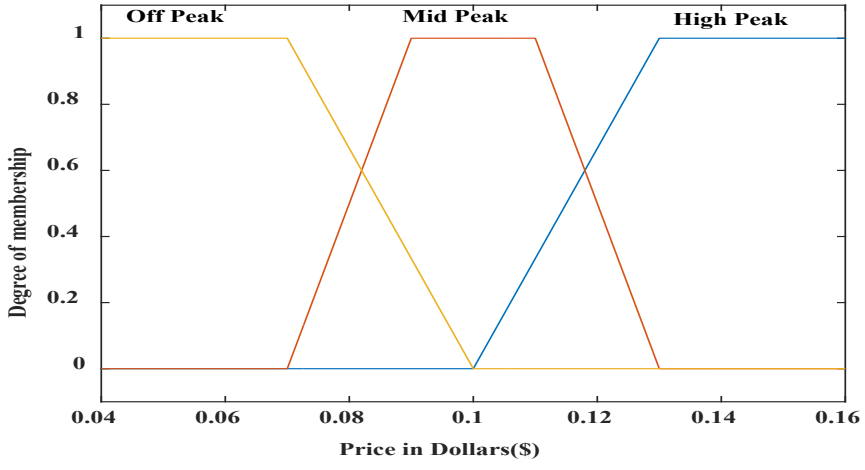


Figure 4: TOU Price Rates

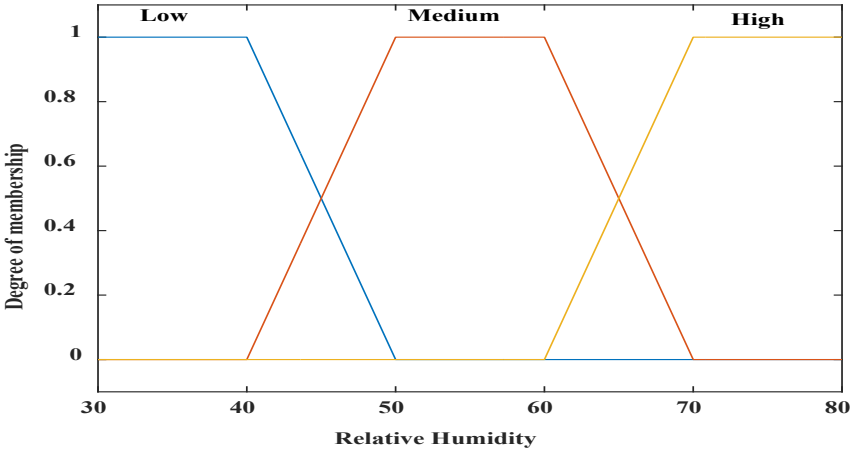


Figure 5: Relative humidity

Working of Fuzzy Logic System is heavily dependent on the selection of the membership function for the input and output parameters. When selecting a membership function for a parameter, consider the shapes that best represent the human knowledge. There has been a number of membership functions with well-known forms like triangular, left-shoulder, right-shoulder and trapezoidal.

The membership functions of input and output parameters used in this system are defined as trapezoidal which has a flat top or it can be said it is a truncated triangular membership function. Although the triangular membership function is simple to use the parameters used in this study are best defined using trapezoidal membership function since the temperature, set points, price and humidity do not suddenly drops their value and maintain the same value for a length of time. So, these flat line membership functions have the advantage of simplicity [21].

The system is evaluated with the help of fuzzy rules in order to determine the energy consumption. When FIS is defined without incorporating humidity, there are 4 variables with 3 values and fifth variable with 2 values. In this case, total number of rules defined in the rule base for both Mamdani and Sugeno FIS are 162. In the second scenario, FIS considering humidity has 5 variables with 3 values and sixth variable with 2 values resulting in total 486 rules in the rule base. Some of the fuzzy rules defined for FIS decisions making are shown in the Table 1.

Table 1: Sample Fuzzy Rules for Energy Consumption Optimization

# Rule	T_{in}	T_{out}	Rate	Occupant	ISP	Humidity	EC
1	L	L	HP	A	L	L	VL
2	L	M	OP	P	L	L	M
3	L	H	MP	P	H	H	M
4	M	H	OP	A	M	H	H
5	M	L	MP	P	H	M	M
6	H	M	OP	A	L	M	M
7	H	H	OP	P	H	H	VH

A. Automatic FIS rule base Generation

It is observed that defining rule for the rule base of FIS is a very lengthy and tedious process. Developing an automatic FIS rule generation process using combinatorics method is also proposed in this paper.

Algorithm 1 Automatic Rule Generator

```
Temp_outdoor ← {L,M,H}
Temp_indoor ← {L,M,H}
P_rates ← {OP,MP,HP}
O ← {A,P}
ISP_s ← {L,M,H}
for Temp_outdoor[1] to Temp_outdoor[n] do
    for Temp_indoor[1] to Temp_indoor[n] do
        for P_rates[1] to P_rates[n] do
            for O[1] to O[n] do
                for ISP_s[1] to ISP_s[n] do
                    Compute Score                          ▷ Defined in Eq.5
                    if Score = 0 or Score = 1 then
                        EC = VL
                    else if Score = 2 or Score = 3 then
                        EC = L
                    else if Score = 4 or Score = 5 then
                        EC = M
                    else if Score = 6 or Score = 7 then
                        EC = H
                    else
                        EC = VH
                    end if
                end for
            end for
        end for
    end for
end for
```

The major steps of FLC are as follows:

- First step is the fuzzification process in which all the membership functions of the system parameters are initialize and define.
- Second step is defining the rules in the rule base by giving weightage to membership functions of input parameters and then assigning the suitable output fuzzy value.
- Third step uses the Mamdani and Sugeno FIS to evaluate the energy consumption.
- After rule evaluation, defuzzification is performed to get the crisp value for the energy consumption. Calculation of remaining performance measures is performed.

Formula to compute the Score used in the Algorithm 1 is as follows:

$$Score = \sum_{i=0}^{n-1} Temp_{outdoor}[i] + \sum_{i=0}^{n-1} Temp_{indoor}[i] + \sum_{i=0}^{n-1} P_{rates}[i] + \sum_{i=0}^{n-1} O[i] + \sum_{i=0}^{n-1} ISP_s[i] \quad (5)$$

5 System Model

In this section, the results of the proposed FLC in HEMS are discussed. The proposed controller works for both cold and hot cities using the inputs: 1) Temp_outdoor, 2) Temp_indoor, 3) P_rates, 4) O, 5) ISP_s and 6) Humidity_rel. We have simulated the effect of these parameters for four scenarios: a) energy consumption in hot cities without humidity, b) energy consumption in hot cities considering humidity, c) energy consumption in cold cities without humidity, and d) energy consumption in cold cities using humidity. Furthermore, all these scenarios are evaluated for following performance measures: energy consumption, cost, PAR and efficiency gain.

A. Energy Consumption in Hot Cities

Energy consumption computation based on outdoor temperature and humidity variations during the 24 hours for one of the hottest city in the world and on one of its hottest day is performed and the hourly energy consumption is presented in Fig. 6. Maximum hourly energy consumption of Mamdani without humidity, Sugeno without humidity, Mamdani with humidity, and Sugeno with humidity is 5.7kWh, 5.5kWh, 4.5kWh, and 4.5kWh respectively. Our proposed FLC improves the energy consumption by effectively maintaining the user comfort up to 21% for both techniques.

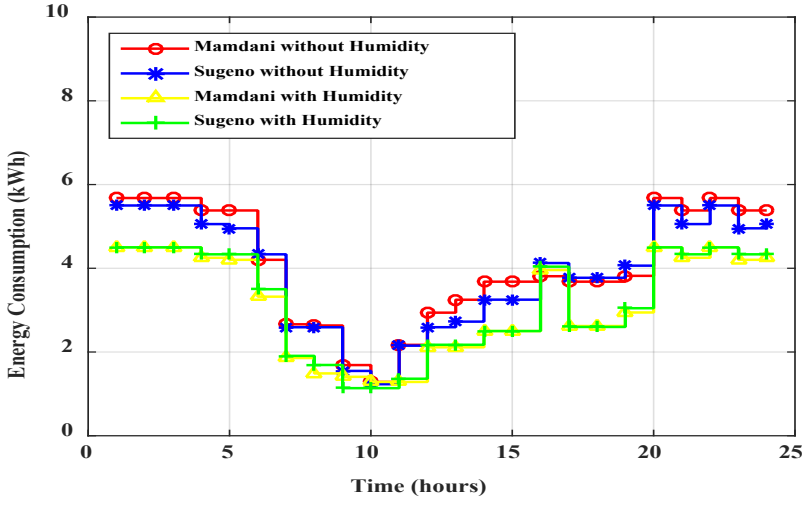


Figure 6: Energy Consumption over a day for Hot Cities

Fig. 7 shows the monthly energy consumption of our designed FLC using both FIS with and without considering humidity. Energy consumption shown is calculated by evaluating and

analyzing the fuzzy rule base. In our study, the energy consumption is low when the initialized set points are high.

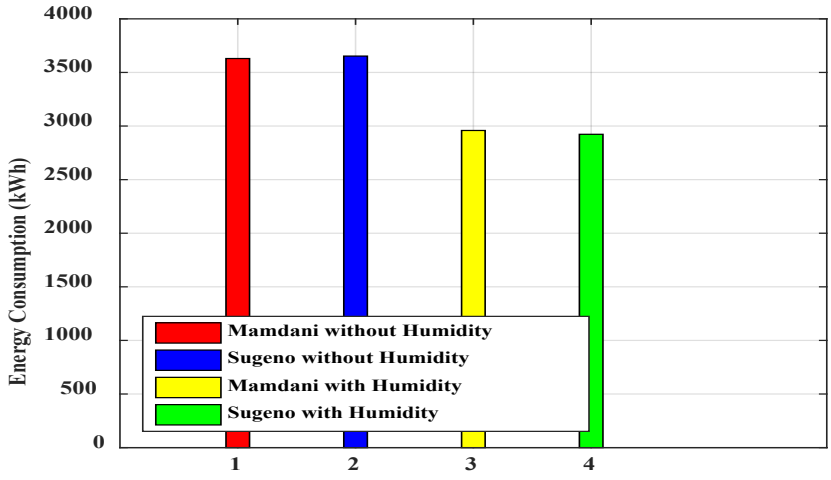


Figure 7: One month simulation of energy consumption for Hot Cities

We have run this simulation for one month. The energy consumption of FLC using Mamdani without humidity is 2954kWh, Sugeno without humidity consumes 2837kWh, Mamdani with humidity shows 2270kWh energy consumption and Sugeno with humidity consumes 2295kWh energy. Mamdani with humidity improves 23% energy consumption while Sugeno with humidity is improving 22% energy consumption as compared to the energy consumed using Mamdani without humidity.

The Mamdani FIS performs better than Sugeno FIS because it is simple in nature and have more energy efficiency. As the demand of HVAC varies on hourly basis in a residential building, the set points are modified by using temperature and humidity information. Thermostat set points are used according to Fig. 1, where it shows the range temperature that lies in the user comfort zone for a particular relative humidity value.

B. Energy Consumption in Cold Cities

Now, we are going to discuss energy management in a residential building using proposed FLC in the cold cities. Input for the occupancy, TOU prices remains same during evaluation. Outdoor temperature, relative humidity, thermostat set points, and indoor temperature are used accordingly to cold cities weather. The effect of these input parameters using cold cities is shown in the Fig. 8.

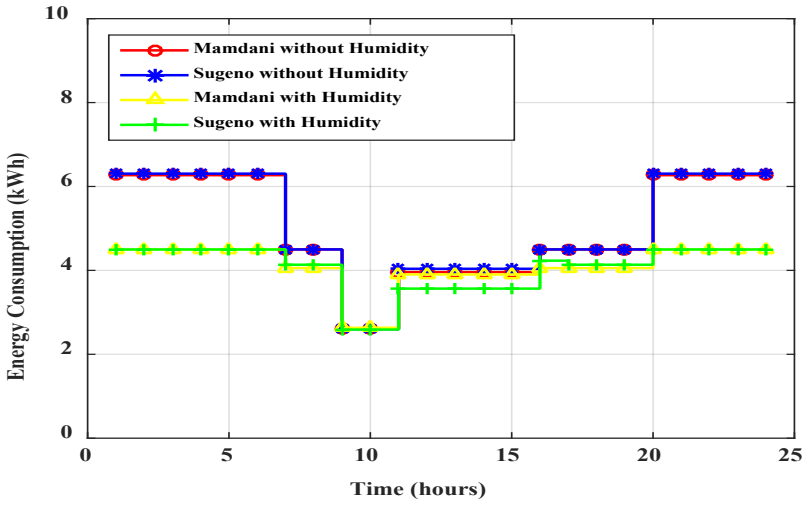


Figure 8: Energy Consumption over a day for Cold Cities

In Fig. 8, energy consumption of the techniques proposed and existing technique for comparison is presented. The maximum energy consumption in cold cities is 6.26kWh, 6.30kWh, 4.5kWh, and 4.5kWh using FIS Mamdani without humidity, Sugeno without humidity, Mamdani with humidity, and Sugeno with humidity where both proposed FIS show28% efficiency in energy consumption then existing FIS without humidity.

Fig. 9 shows the energy consumption of cold cities run for one month simulation using Mamdani and Sugeno while considering and leaving the humidity parameter. The behavior of energy consumption is maintained at a desired comfort level using ISPs. Although the energy consumption for cold cities is greater as compared to hot cities, our proposed system succeed in energy consumption minimization which shows the efficiency of proposed scheme to the earlier schemes.

The monthly energy consumption of Mamdani without humidity is 3630kWh, Sugeno without humidity is 3653kWh, Mamdani with humidity uses 2959kWh, and Sugeno with humidity consume 2922kWh of energy. Efficiency in energy consumption for Mamdani with humidity is 19% whereas in Sugeno with Humidity is 20% as compared to FIS without Humidity.

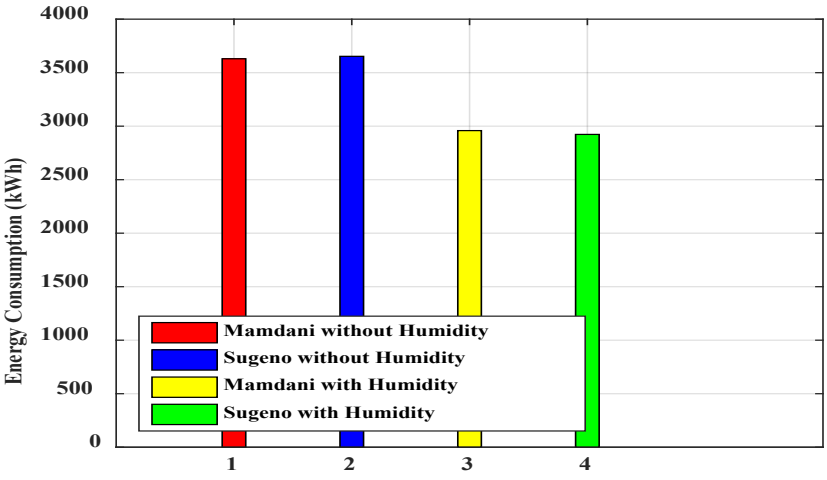


Figure 9: One month simulation of energy consumption for Cold Cities

C. PAR

PAR of the cold cities is shown in Fig. 10, which shows Mamdani with humidity achieved 12% efficiency as compared to the Mamdani without humidity whereas efficiency of Sugeno with humidity is 10%. However, if the simulations are run for the hot cities no prominent efficiency gain is observed. This is mainly because the proposed system is mainly focused on the energy consumption minimization. Improvement in the PAR efficiency for the hot cities can be regarded as the byproduct of the proposed scheme.

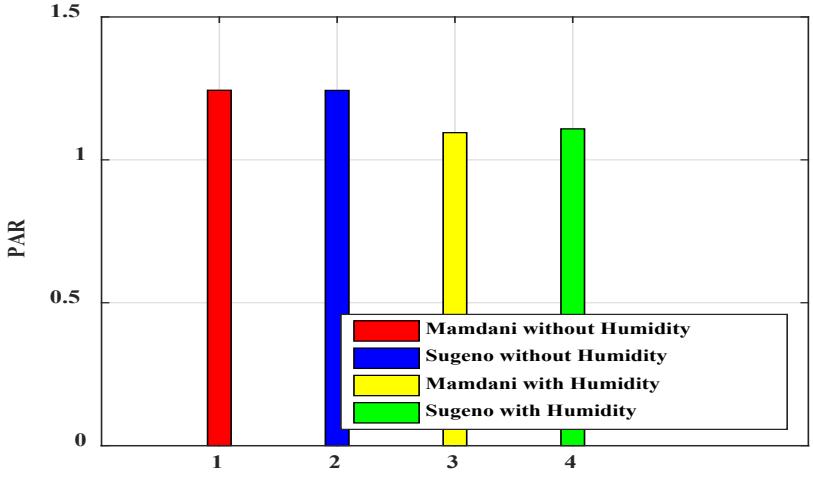


Figure 10: PAR for Cold Cities

D. User Comfort

User comfort is mostly sacrificed in previous techniques. In this scheme we initialized the thermostat according to Fig. 1, which allows setting high set points for hot cities and low set points for cold cities considering a particular relative humidity values. Therefore, values for ISPs are selected that not only reduces the energy consumption but also keeps the residential environment in user comfort zone.

E. Cost in Hot Cities

Cost reduction is an inevitable consequent of energy consumption minimization. Cost is computed using the Eq. (3) and proposed system performs best among all approaches. Using technique of Mamdani without humidity, cost is nearly8.92$, Sugeno without humidity costs 8.6 $, Mamdani with humidity approach cost 6.79 $, and Sugeno with humidity FIS costs 6.86 dollar per day as shown in the Fig. 11. Approach using Mamdani with humidity reduces the cost by 23.87% as compared to Mamdani FIS without humidity and Sugeno with humidity to 23.09% as compared to the method using Sugeno without humidity. Mamdani outperforms here because of its simple nature and so having energy efficiency.

F. Cost in Cold Cities

Cost is computed using the Eq. (3) which was also used in the cost calculation for hot cities. As shown in Fig. 12. Scheme using Mamdani FIS without humidity costs 11.19 $, Sugeno without humidity costs 11.25$, Mamdani with humidity costs 9.4 $, and Sugeno with humidity costs 9.3 $ for energy consumption in a day.

Moreover, Mamdani with humidity shows efficiency of 16.44% in cost reduction and 17.33% efficiency of Sugeno with humidity as compared to the Mamdani and Sugeno without humidity.

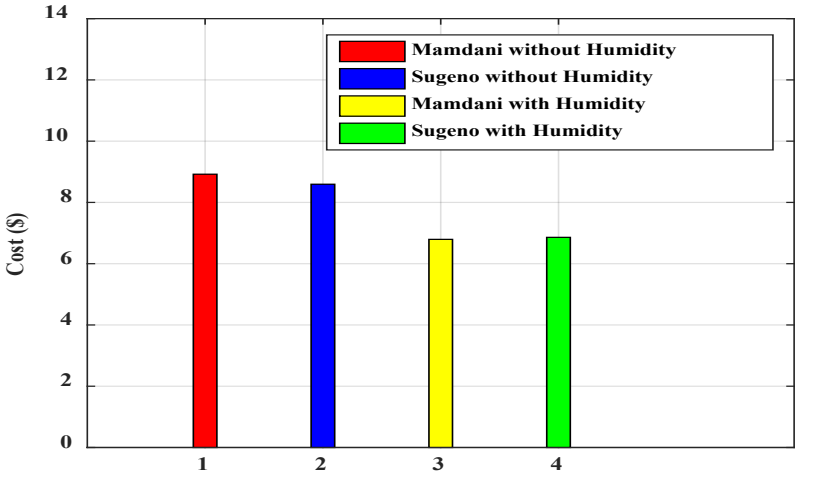


Figure 11: Cost of energy consumption in a day for Hot Cities

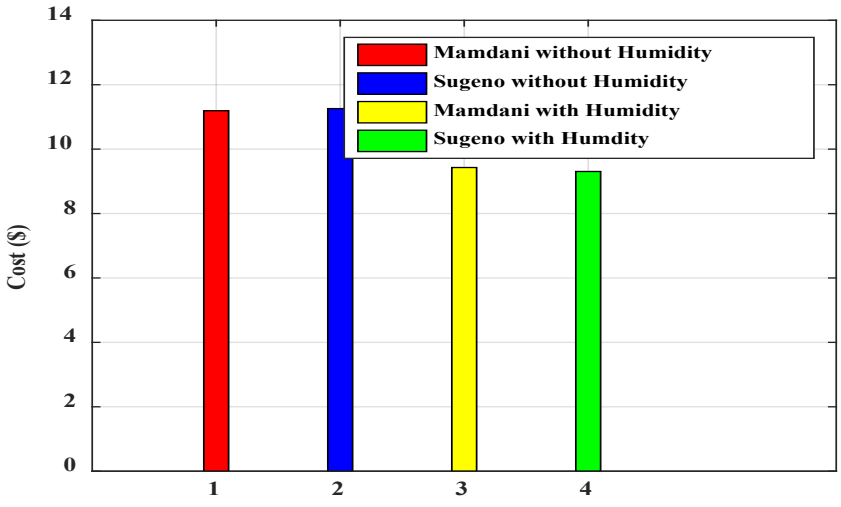


Figure 12: Cost of energy consumption in a day for Cold Cities

G. Result Analysis

Effect of humidity on energy consumption minimization and cost reduction is studied in this paper. It is observed that considering humidity while setting thermostat set points plays an important role in achieving the defined objectives. It can be concluded that information of humidity values helps user to set the ISP_s in such a way that not only reduces energy consumption but also guarantee little disturbance in user comfort. Mamdani and Sugeno, both FIS give the reasonable results and show improvement in comparison of previous techniques. However the competition between the proposed Mamdani FIS and Sugeno FIS is very close, but it can be seen that Mamdani FIS performs better in hot cities whereas Sugeno FIS outperforms in cold cities.

Following points are concluded by analyzing the performance of Mamdani and Sugeno FIS used for both hot and cold cities:

i. In order to maintain the residential building temperature as close to user comfort zone, cold regions require more energy consumption as compared to hot regions because appliances for heating requires more energy than the appliances used for cooling purpose.

ii. Reason behind the slightly better performance of Mamdani in hot cities is the simplest nature of the rules defined through Mamdani FIS.

Although the variations in outdoor temperature influence the temperature range of user comfort, proposed scheme is minimizing energy consumption without disturbing user comfort.

6 Conclusion

In this paper, we extended the worldwide adaptive thermostat model to include the parameter of humidity and observed the effect it does on the energy consumption. An algorithm for the

automatic generation of the fuzzy rules and their initialization in rule base is also proposed. Automatic generation of rules avoids the time consuming process of fuzzy rules initialization. Proposed scheme of work tracked the energy consumption of the HVAC in residential buildings throughout the world along with cost and PAR Simulation results show that proposed methodology significantly reduced the energy consumption and cost while maintaining the user comfort. Efficiency gain in energy consumption using Mamdani and Sugeno is 23% and 22% whereas cost obtained using Mamdani and Sugeno is 23.87% and 23.09% efficient when dealing with hot cities. In cold cities efficiency gain in energy consumption using Mamdani and Sugeno both is up to 28% whereas cost obtained using Mamdani and Sugeno is 16.44% and 17.33% efficient.

In future, more parameters that effect the user's decision of setting the thermostat set points should be considered along with the closed control loop. It can be extended for using more pricing schemes in order to observe the real time effect of dynamic pricing.

References

[1] Mahmood, A., Javaid, N., Zafar, A., Riaz, A., Ahmed, S., Razzaq, S., "Pakistan's overall energy potential assessment, comparison of LNG, TAPI and IPI gas projects," Renewable and Sustainable Energy Reviews, 31, 182-193, 2014.

[2] Albadi, M., El-Saadany, E.,"A summary of demand response in electricity markets," Electric Power Systems Research, 78, 1989-1996, 2008.

[3] Mahmood, A., Javaid, N., Khan, A., Razzaq, S.,"An overview of load management techniques in smart grid," International Journal of Energy Research, 39, 1437-1450, 2015.

[4] Rasheed, B., Javaid, N., Ahmed, A., Khan, Z., Qasim, U., Alrajeh, N.," An efficient Power Scheduling Scheme for Residential Load Management in Smart Homes," Applied Sciences, 5, 1134-1163, 2015.

[5] Chanana, S., Arora, M., "Demand Response from Residential Air Conditioning Load Using a Programmable Communication Thermostat," International Journal of Electrical, Computer, Energetics, Electronics and Communication Engineering, 7(12), 1670-1676, 2013.

[6] Mirakhorli, A., Dong, B., "Occupancy behavior based model predictive control for building indoor climate – A critical review," Energy and Buildings, 129, 499-513, 2016.

[7] Faruqui, A., Sergici, S., "Household response to dynamic pricing of electricity: a survey of 15 experiments," Journal of regulatory Economics, 38, 193-225, 2010.

[8] Hatziargyriou, N. D., Dimeas, A., Tsikalakis, A. G., Lopes, J. P.,Karniotakis, G., and Oyarzabal, J., "Management of microgrids in market environment," Future Power Systems,1-7, IEEE, 2005.

[9] Gangale, F., Mengolini, A. and Onyeji, I.,"Consumer engagement: An insight from smart grid projects in Europe," Energy Policy, 60, 621-628, 2013.

[10] Walker, I. S. and Meier, A. K, "Residential thermostats: comfort controls in California Homes," Lawrence Berkeley National Laboratory, California, 2008.

[11] Meier, A. Aragon, C., Peffer, T., and Marco, P., "Thermostat interface and usability: a survey," Lawrence Berkeley National Laboratory, California, 2011.

[12] Keshtkar, A., Arzanpur, S., Keshtkar, F., Anmadi, P., "Smart residential load reduction via fuzzy logic, wireless sensors, and smart grid incentives," Energy and Buildings, 104, 165-180, 2015.

[13] Keshtkar, A., Arzanpour, S., "An adaptive fuzzy logic system for residential energy management in smart grid environments," Applied Energy, 186, 68-81, 2016.

[14] Javaid, S., Javaid, N., Iqbal, S., Mughal, J., "Controlling energy consumption with the world-wide adaptive thermostat using fuzzy inference system in smart grid," International Conference on ICT convergence, 2017.

[15] Kolokotsa, D., Tsiavos, D., "Advanced Fuzzy Logic Controllers design and evaluation for buildings occupants thermal-visual comfort and indoor air quality satisfaction," Energy and Buildings, 33, 531-543, 2001.

[16] Ross, T. J. Fuzzy logic with engineering applications. John Wiley and Sons. 2009.

[17] The MathWorks, Inc. (2017). Comparison of Sugeno and Mamdani Systems[Online]. Available:https://www.mathworks.com/help/fuzzy/comparison-of-sugeno-and-mamdani-systems.html

[18] Time and Date Weather forecast website. (Accessed on 22 December 2017). Past Weather in WadiHalfa, Sudan – June 2017[Online]. Available:https://www.timeanddate.com/weather/@364132/historic?month=6&year=2017

[19] Time and Date Weather forecast website. (Accessed on 22 December 2017). Past Weather in Yakutsk, Russia – January 2017[Online]. Available:https://www.timeanddate.com/weather/russia/yakutsk/historic?month=1&year=2017

[20] Apogee Interactive. Inc. (2010). Comfort [Online]. Available: https://c03.apogee.net/contentplayer/?coursetype=res&utilityid=wppi&id=1703

[21] Gully, N., Fuzzy Logic Toolbox User's Guide. The MathWorks, Inc. 2002

Transfer Learning Techniques for Image Recognition: A Systematic Literature Review

Tariq Mahmood [1] Aleenah Khan [2] Salma Tajuddin [3]

Abstract

Deep learning is a rapidly expanding research area focusing on the use of more extended (deep) and varied neural network architectures to solve more complicated problems than traditional multi-layer perceptrons. Transfer learning is a more recent off-shoot of deep learning which focuses on using information from one machine learning task in another related task. It has primarily seen applications in image classification, for instance, when information used to recognize/classify a bicycle can be used to classify a motorcycle. In a rapidly evolving research space, it is important to summarize the research applications of different deep learning off-shoots. In this regard, this paper presents the first systematic literature review particularly targeting applications of transfer learning to image recognition. We follow the standard methodology and categorize papers on the basis of more critical KPIs. Our core finding is that this particular domain is a hot area of research these days, and most applications are related to pre-trained models learnt from convolution neural network and applied to another convolution network. Also, transfer learning has lead to significant improvements in accuracy and efficiency and facilitation, as compared to learning deep models or other machine learning approaches from scratch. From our results, we propose several future directions of research.

Keyword: Transfer Learning, Image Recognition, Image Classification, Systematic Literature Review, Convolution Neural Networks

1 Introduction

Deep learning [1], [2], [3], [4] involves the use of larger and more robust artificial neural network models, as compare to traditional single-layer or multi-layer perceptrons [5]. The word 'deep' mostly signals the use of more hidden layers containing a larger number of hidden neurons, along with novel, diverse architectures for instance, convolution neural networks, recurrent neural networks (LSTM, GRU), restricted Boltzmann machines, deep belief networks, and generative models [2]. Deep learning has successfully solved many complicated machine learning problems, for instance, image recognition, image captioning, machine translation, natural language processing, and automatic speech recognition [6], [7], [8]. Notwithstanding this, deep learning always has had a problem of efficiency with initial training times running into hours. The rapid evolution and application of GPU technology has catered for this problem to a certain extent [9], [10], efficiency still remains

[1] *Institute of Business Administration, Karachi | tmahmood@iba.edu.pk*
[2] *Institute of Business Administration, Karachi | aleenahkhan@iba.edu.pk*
[3] *Institute of Business Administration, Karachi | stajuddin@iba.edu.pk*

a major problem. Considering the complexity of deep learning problems, there is also a need to make learning easier. If we train an LSTM to recognize English speech patterns of American users, we would want to use this knowledge for recognizing speech of British users also because both of them come from the same distribution of English speaking users. Knowledge from a convolution network model that recognizes cars can be used in another convolution network that is learning to recognize trucks, because both belong to the same distribution of motor vehicles.

Transfer learning [11], [12], [13], [14], [15] is a machine learning technology which uses models (architecture with hyper parameters) from one learning problem in another related learning problem. The first research paper appeared in 1992 [15] and the field has seen significant advances since then, particularly in deep learning. In this paper, we perform a systematic literature review of the applications of transfer learning to image classification or recognition. In particular, we are motivated by the importance of image recognition to the biomedical domain. Image classification is a critical application area of deep learning and there is relatively substantial literature on its transfer learning applications which needs to be reviewed. Although there have been several reviews on transfer learning applications [16], [11], this is the first review giving a drill-down into image recognition. We adopt the standard reviewing procedure by identifying keywords, search queries, filtration of retrieved papers, and then creating appropriate labels to classify the results. Due to some limitation of time, we are able to present the most recent results till the time of conducting this survey, i.e., from January 2017 till April 2017. Even with this limited sample, we were able to extract useful results and propose concrete future directions of research.

2 Background Knowledge

A Convolution Neural Networks

Convolution Neural Networks [17], [18], [19] (CNNs) are feed-forward deep neural networks best suited to solve visual imagery learning problems, e.g., image classification and recognition. They are famous because they eliminate the need to exact image features

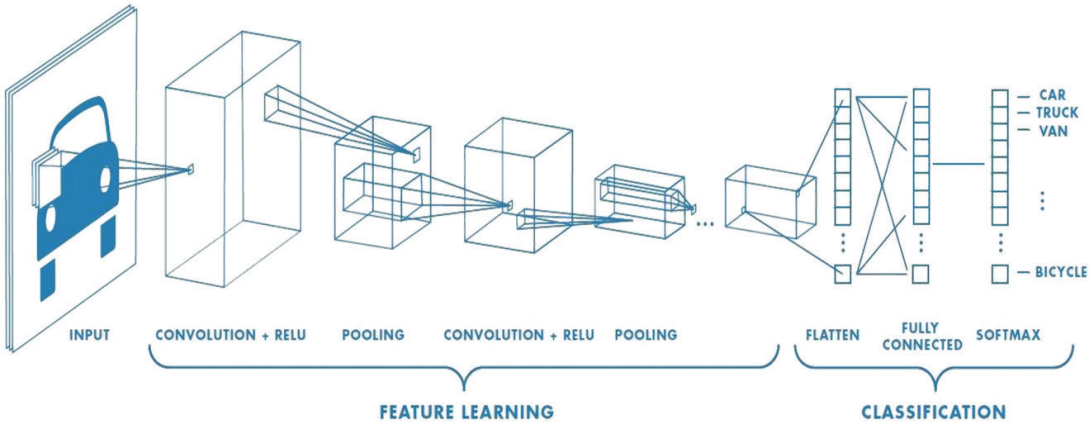


Figure 1: Generalized Architecture of Convolution Neural Network

Manually and also facilitate transfer learning. Figure 1[4] shows the basic architecture of a CNN. We can imagine all the layers in feature learning and classification sections as being stacked together from bottom (feature learning) to top (classification) in 3D, having width, height and depth. The depth is typically 3, corresponding to RGB color bands. The width and height correspond to the same dimensions of an input image in 2D. In feature learning, the convolution layer applies a particular filter on the input image to extract one or more features and the RELU activation function non-linearizes the result. Each neuron in the convolution layer might only recognize a particular feature or part of an image (e.g., an edge, circle, color shade etc.) in its receptive field, i.e., a given neuron is only connected to a small part of the image and takes no input from the remaining image parts. This latter concept is taken from research in neuroscience in how animals and humans recognize objects. The job of the pooling layer is to summarize the non-linearized convolution filters, e.g., by taking maximum or average. Typically, a large number of convolution + RELU + Pooling triples are applied, in order to automatically extract as many features as possible from the input image and summarize them. In the classification phase, the output from the final pooling layer is flattened out into a vector which is fed to a traditional fully-connected MLP. In the output, a soft max activation function is applied to assign probabilities to output, depending on the type of output being trained on.

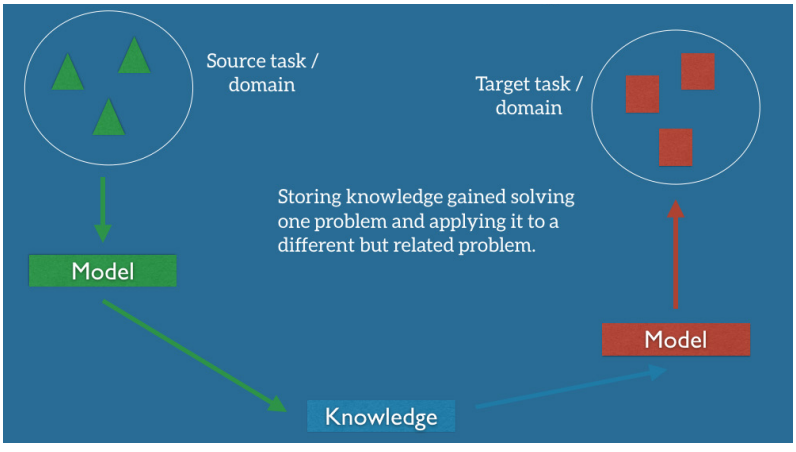


Figure 2: Concept of Transfer Learning

B Transfer Learning

Transfer learning is a technique of machine learning in which a model trained on one problem is re-purposed on another related problem. This transfer of knowledge typically improves generalization in the other setting and hence can also be considered a type of optimization of the second task. Transfer learning is generally related to multi-task learning and concept drift, and cannot be considered only related to deep learning.

However, it is prominent in deep learning because of extensive computational resources required to train and learn deep models. The requirement here is that model features learned in the initial problem should be generic in nature. Specifically, the source network is trained on a

[4] *Adapter Network HTML from mathwork.com/discovery/convolutional-neural-netwrok.html*

baseline task with a baseline dataset. Its learned features are then re-purposed, or transferred, to the target network for learning a target task with a target dataset. This situation is shown in Figure 2[5]. The transfer will work if the features used are applicable to both the source and target tasks, and not just specific to one of them. This inductive transfer considerably minimizes the hypotheses space for searching the solution for target task.

There are two approaches to deep transfer learning: 1) model development, and 2) using pre-trained models. In the first case, we first select a related source problem which could be less complicated than the target one, based on relatively similar inputs and outputs. We then learn the source task and ensure that a significant amount of feature selection has occurred, and then re-purpose this model as the starting point for the target task. Here, the model can be used partially or completely and is then finally tuned for the target. In the second case, we reuse a model pre-trained on complicated deep problems, which are provided by research organizations and academia. Again, partial or complete model can be used, which is then finally tuned to the target problem. This latter approach is more commonly used in deep learning. Nowadays, transfer learning papers are published in acclaimed conference related to data science and machine learning, for instance, ICDM, KDD, ICML, AAAI, NIPS, ECML, and NIPS.

C Formalizing Transfer Learning

Transfer learning can be formalized by answering three questions:

1) What needs to be transferred?: Knowledge can either be applicable to only a single task or it can be applicable across multiple tasks. We need to determine the partition of knowledge which will be transferred from source to target task.

2) When should the transfer take place?: We need to determine the exact situations in which transfer should happen and those in which transfer learning is not applicable. In the latter case, forcefully transferring knowledge through brute force can lead to derogatory performance, also called a "negative transfer".

3) How should the transfer be done?: Transfer learning can occur either from one type of machine learning problem into another one. We need to determine the exact categories and then the most appropriate algorithm for this type of transfer.

1) *Transfer Learning with Visual Imagery:* Visual imagery problems are most common in deep learning, particularly related to object recognition. The input is an image in colored or gray-scale mode, and the task is to recognize objects in the image, for instance, motor vehicles, animals or people. More than one object (class) needs to be detected for a given object type, for instance, cars, trucks and motorcycles in motor vehicle object type, and cats, dogs and rabbits in animal object type. For such problems, we can use models pre-trained for more challenging image recognition tasks such as the standard Image net Large Scale Visual Recognition Challenge (image-net.org/challenges/LSVRC/2016/index) challenging global researchers on a 1000-class problem. Models pre-trained on this problem are provided by several research

[5] *Adapter from ruder.io/transfer-learning/*

groups, for instance, the visual geometry group at university of Oxford[6], Microsoft's Resent CNN model[7] and Google's inception model[8]. More such models can be found at Caffe's Model Zoo[9]. These pre-trained models are more useful in the earlier layers of the target task, and more specific target dataset features are useful in later layers.

More formally, a source image recognition problem with domain D consists of feature space F and a marginal probability distribution $P(G)$ over feature space with $G = f1, f2,...,fn \epsilon F$ with G being the total set of images used in training. Given $D = F; P(G)$, problem τ has labeled space γ with the conditional probability distribution $P(Y—G)$ learned from training data consisting of pairs $fi \epsilon G$ and yi $\epsilon \gamma$. In our case, γ is the set of image object type labels, e.g., car, bus, and truck. Now suppose we have a source domain Ds, source problem τ_s, target domain D_t and target problem τ_t. The aim of transfer learning is to learn the distribution P(Yt | Xt) in Dt with knowledge acquired from Ds and τ_s where $D_s \neq D_t$ and $\tau_s \neq \tau_t$. This formulation gives rise to the following four possible scenarios:

$F_s \neq F_t$: The feature space of the source and target tasks are different from each other, e.g., the input images represent different object types (cars and truck)

$P(G_s) \neq P(G_t)$: The marginal distributions of source and target domains are different, e.g., distribution of features of cars and trucks will be different

$_s \neq _t$: The label spaces is different for each domain, e.g., images are assigned different types of labels in the target space (this case doesn't occur with a high probability)

$P(Y_s \mid G_s) \neq P(Y_t \mid G_t)$: The conditional probability distributions of the source and target tasks are different from each other, e.g., if the labeling is unbalanced in both cases with respect to the class distribution.

One method of doing transfer learning is to execute it using simulations (to avoid training of expensive hardware) and the reuse the results in the real world target task. In this case, case 2 above is applicable and not case 1 as the feature space is the same in both simulation and real-world. The simulation technique is extremely useful for complicated tasks like training self-driving car; Fig 3 shows such a simulator from udacity [20]. Another pertinent concept is that of domain adaptation, in which the source and target data could belong to different domains. This happens when we are forced to use a pre-trained model on something which is not exactly what we want. For instance, one may use a model pre-trained to recognize regular bike in shopping stores to learn to recognize mountain bikes being driven on mountain roads and tracks.

[6] *robots.ox.ac.uk/vgg/research/very_deep/*
[7] *github.com/KaimingHe/deep-residual-networks*
[8] *github.com/tensorflow/models/tree/master/research/inception*
[9] *github.com/BVLC/caffe/wiki/Model-Zoo*

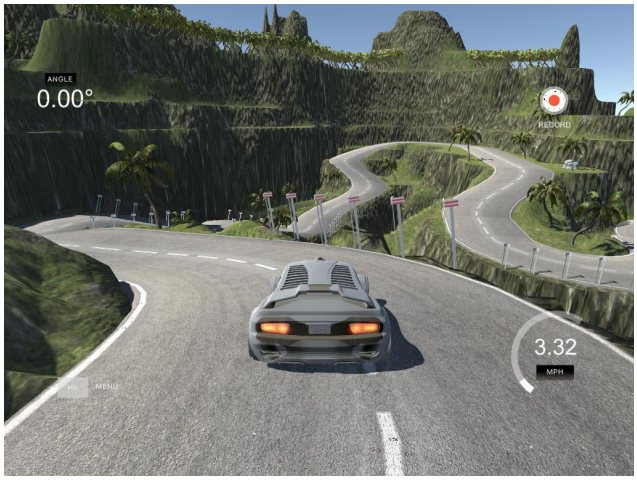


Figure 3: Udacity Self-Driving Car Simulator

3 Systematic Literature Review

A systematic literature review (SLR) is a repeatable research method which critically analyzes multiple research studies in parallel regarding a particular research domain, with the intention of answering one or more research questions according to a structured methodology [21]. In this paper, we conduct the first SLR on applications of transfer learning to image recognition which is the first of its type to the best of our knowledge. We chose an SLR because we want to investigate about this research domain rather than find some particular solution for it. Our research questions are formulated as follows:

> RQ1: How important are transfer learning applications to image recognition for the research community? We will answer this by considering the number of related publications.
> RQ2: What is the impact of transfer learning on image recognition? We will answer this by considering the results of experiments in the publications along with other related data.
> RQ3: What are the future research directions of transfer learning applications to image recognition? We will answer this by initially analyzing our review results and then deriving the potential future directions.

We limit our searched relevant papers to those provided by developed queries ()given below) from renowned computer science digital sources, i.e., IEEE, ACM, Google Scholar, Springer, Elsevier, and Wiley. Duplicates were likely due to the presence of Google Scholar in the list sources which can reiterate results from any and all other sources list. We removed these duplicates before any analysis.

A Keyword Identification and Query Formation

Based on our research domain and scope of review, we identified keywords and generated search queries based on those keywords. Although most keywords can be combined with each other, convolution neural networks is a subset of deep learning and therefore these terms may not be

used in conjunction with one another. We kept such relationships into account when forming queries. We identified ten types of logically possible conjunctions of our keywords, which are shown in Table I. It is important to note that these conjucntions represent the context on what we searched and do not represent the exact query that we used. While querying, we applied various types of rephrasing to get better results, especially in the context of inadequately sized data set. This particular concept was difficult to formulate precisely and we searched it using various alternates. The results documented were the maximum of all result counts, instead of summing up in case of repetitions. In some case when different and possibly relevant results were observed, we formed a combination of results using the Mendeley citation management tool[10].

Table I: Context of our Search Queries

No.	Search Query
1	Deep Learning Techniques and Image Classification
2	Transfer Learning Techniques and Image Classification
3	Convolution Neural Networks and Image Classification
4	Convolution Neural Networks and Transfer Learning
5	Image Classification and Transfer Learning
6	Handling Inadequate Sized Data Classification of Images using Deep learning
7	Handling Inadequate Sized Data Classification of Images using Transfer learning
8	Handling Less Data Classification of Images with Convolutional Neural networks
9	Image Classification using Convolutional Neural Networks and Transfer Learning
10	Image Classification with Transfer Learning and CNN (small data sets)

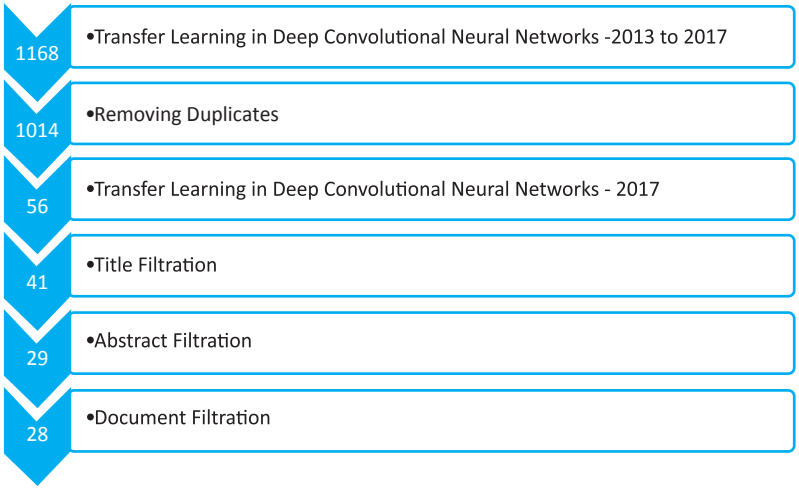


Figure 4: Article Filtration Process

[10] *www.mendeley.com*

B Article Filtration Process

On executing the search queries, we retrieved a total of 1168 relevant research publications from our digital sources. These papers were published between 2013 and 2017 (inclusive) and specifically dealt with applications of transfer learning to convolution neural networks. We then adopted an article filtration process shown in Figure 4. Removing duplicates using Mendeley gave us 1014 articles. Based on certain constraints, we considered this a large quantity of literature to review, so we decided to focus initially on the most recent publications in 2017. This left us with 56 articles up to the time of this review. We then used a three-pronged approach: we first filtered for title (remaining 41 articles), then for abstract (remaining 29 articles) and finally the body of the publication (remaining 28 articles).

C Analytics of Retrieved Articles

The number of publications per year is shown in Figure 5. As we can see, there is an almost exponential increase in the number of articles related to transfer learning from 2013-2016. It is expected that the total number of articles in 2017 will be greater than 300, but to keep our scope limited, we only consider 28 publications up to the time of writing this survey (January 2017 - April 2017).

Moreover, as can be seen in Figure 6, most of these 28 articles have been published in conference proceedings. There are no technical reports and just 1 book chapter and thesis along with 2 journals. One reason for this is the fast pace at which the transfer learning field is growing which makes it more convenient to publish in a conference with shorter review times as compared to other types.

Also, the distribution of our 28 articles with respect to digital sources (Figure 7) shows that a large number of articles were retrieved from Google Scholar, followed by IEEE, arXiv, IEEE, Elsevier and finally ACM.

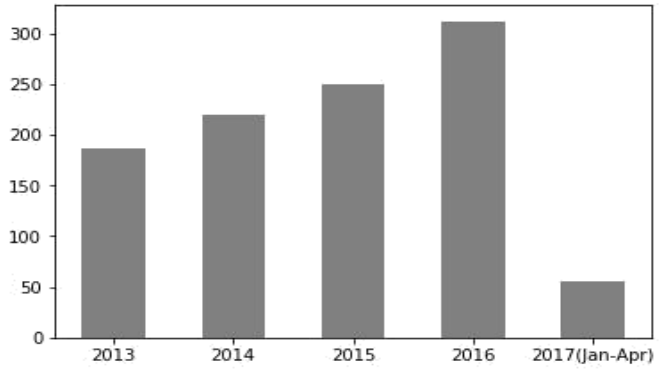


Figure 5: Annual Distribution of Articles from 2013-2017

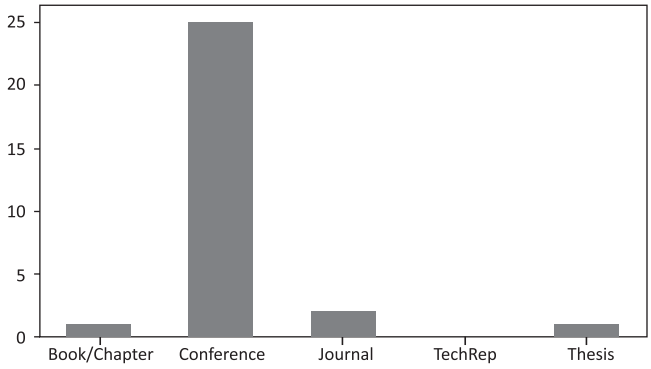


Figure 6: Distribution of Articles with respect to Type

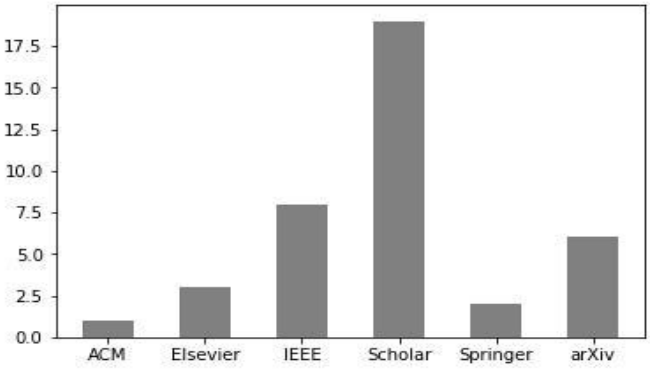


Figure 7: Distribution of Articles with respect to Digital Sources

The usage of pooling functions in our 28 articles is shown in Figure 9. Mostly, the well-known max pooling is used, followed by average pooling. However, a large majority of articles haven't mentioned their pooling function and a single paper hasn't mentioned their pooling function. The activation function distribution is shown in Figure 10. As can be expected, mostly ReLU has been used, followed by Softmax. A single paper has used Sigmoid while the remaining have not mentioned their activation function details.

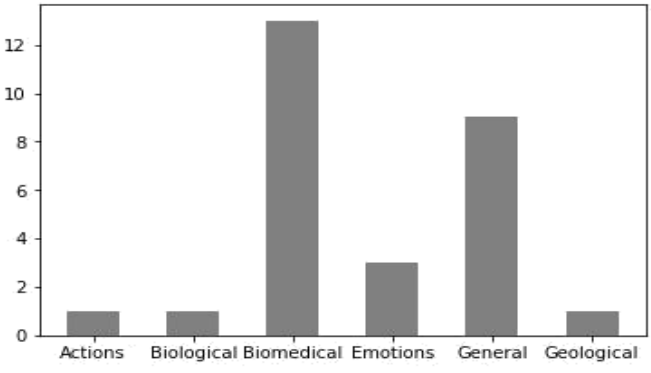


Figure 8: The Domain Type of Research in 28 Selected Articles

Table 2: Datasets Used in Selected 28 Articles

GBM	KIRC	NIH AREDS	BCDR-F03	Indian Pines
CIFAR10	MRSCVOC2007	USPS	Salinas Valley	Pavia University
MNIST	COIL20	CMU-PIE	PACS (Abdonminal Ultrasounds)	ISBI 2016 Challenge Dataset
Office	Caltech256	BUAA VIS-NIR	Chalearn-LAP First Impressions	Driver Monitoring
LifeCLEF2015	DLCST	COPDGENE1	RSD	aPY
COPDGENE2	Frederickshavn	OxfordDogs120	GENKI	AM-FED
OxfordFollowers102	MITIndoor67	Duke OCT	NJUD	STEREO
APR	EmotiW2015	EmotiW2016	AwA	UCM
CK+	MMI	RECOLA	DES	NLPR

The data sets used in our 28 papers are shown in Table II. These are classification datasets. The number of categories varies from only four entries to a set of hundreds of classes. The size of the data varies from 200 to around 50,000. This is all possible with good results due to transfer learning. Some experiments gradually increase the training set size to observe the impact of transfer learning and its ability to perform well with even a small set of labeled images. Also, the type of domain research with respect to our articles is shown in Figure 8. We see that almost 50% of articles are from Bio-Medical domain followed by general datasets. Three articles deal with human emotion recognition, while individual contributions come from human actions, land type and plant recognition domains.

In Table III, we create a classification label for each selected article and also show the overall performance results which are extracted from the paper. We found two papers dealing with unsupervised transfer learning [22], [23], and two which have presented a robust framework for learning metrics or hyper parameters through transfer learning [24], [25]. The majority of classification is understandably related to application of transfer learning in deep convolution neural networks [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42] to solve problems in diverse application domains shown in Figure 8. Two papers have addressed the use of transfer learning in non-deep convolution networks [43], [44], while the remaining have used non-CNN methods to enhance image classification performance through transfer learning [45], [46], [47], [48], [49].

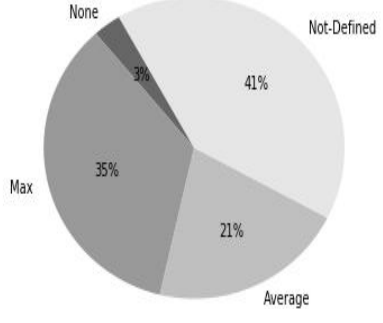


Figure 9: Distribution of Pooling Technique Usage

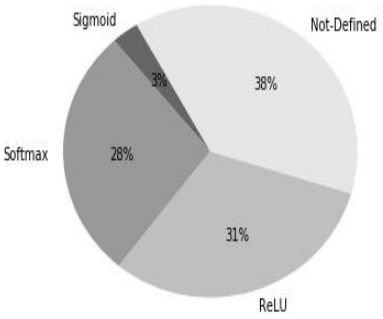


Figure 10: Distribution of Activation Function Usage

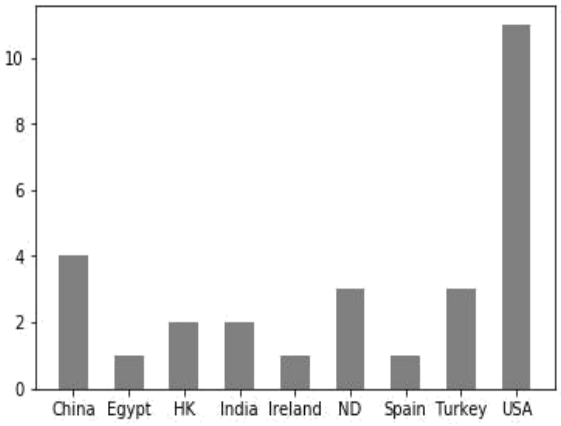


Figure 11: Distribution of Papers with respect to Country of First Author - ND=Netherlands, HK=Hong Kong

Regarding the performance results, most articles have recorded accuracy measure on their selected datasets. In Table III, we have shown the average performance across all the selected datasets for the sake of clarity and understanding. We see that most accuracies for deep CNN based transfer learning applications are satisfactory (90%). Apparently, such an architecture seems to be best validated for transfer learning support in image recognition. The other recorded measures are the Area Under the Curve (AUC) [43], Root Mean Squared Error (RMSE) [31], overall error in classification [39], and an improvement in performance [44], [47]. However, the performance in these papers is also seen to be satisfactory. In essence, it is apparent that the application of transfer learning does have a substantial effect on improving the performance, although researchers have preferred to stick more with deeper CNN based models due to their recent success, e.g., LeNet, AlexNet, GoogleNet, ResNet, VGGNet and CaffeNet.

Finally, we extracted the country of the first author for each of our selected articles to understand the global locations where more research in our domain is concentrated. These results are shown in Figure 11. Majority of the papers have been published by American authors (11), followed by China (4), Netherlands (3) and Turkey (3). Other countries with even lesser publications include Ireland, Spain, India and Egypt. We didn't have publications from Australia, South America and Africa.

4 Future Research Directions

Our research (although performed on a limited scale) has identified several crucial details. Based on this, we are able to suggest the following future research directions:

Application to Security Domain: The most frequent application of our selected papers is in the biomedical domain. However, we couldn't find any application in security or cyber security realm where face, gesture and movement recognition become extremely crucial. Applications in this domain need to increase substantially.

Increase in Journal Publications: As compared to the conference publications, journal articles are almost infrequent. This clearly shows that the transfer learning field is expanding at a rapid pace and hence is more suited to shorter review times of conference. However, in order to lay deeper scientific foundations of the field, the number of journals also needs to increase.

Performance with Different Pooling and Activation Functions: In our papers, researchers have gone with the use of standard max pooling and ReLU activation function. However, we didn't find any article offering a comprehensive comparison of different pooling techniques and activation functions. We deem it necessary considering the substantial performance improvements offered by transfer learning.

Application with Other Machine Learning Approaches: We have seen significant number of applications using deep convolution networks. However, the performance with other architectures and algorithms has been as satisfactory. There needs to be hence more research in these directions: either exploration of non-CNN architectures and algorithms or im-proving transfer learning through other machine learning approaches coupled with CNN architectures. These combinations can also be used with other deep learning approaches, for instance, recurrent neural networks for tasks such as image captioning or image description generation [2].

Experimentation with CNN hyper parameters: In [50], the authors establish a robust taxonomy of CNN hyper parameters which have generated to the huge success of CNNs in the recent past. There needs to be a comprehensive research work which experiments with all such parameters with one or more standard transfer learning approaches. Such hyper parameters include different types of convolution layers (tiled, transposed, dilated, network in network, inception module), loss functions (hinge, softmax, contrastive, triplet, KL divergence), optimization (data augmentation, weight initialization, stochastic gradient descent, batch normalization, shortcut connections), and faster processing methods (fast fourier transforms, structured transforms, low precession, weight compression and sparse convolution).

5 Answering the Research Questions

We now reiterate and answer our research questions as follows:

RQ1: How important are transfer learning applications to image recognition for the research community? Answer: Transfer learning applications are considered highly beneficial to image recognition domain. In the period 2013-2016, the numbers of related publications have increased at an exponential rate and hence, this is a hot research area these days. Mostly, the research has focused on healthcare and biomedical domains primarily to automate recognition across clinical laboratory test images, e.g., MRI, CT Scan etc.

RQ2: What is the impact of transfer learning on image recognition? Answer: The impact of transfer learning on image recognition is substantial. In our review, we considered 28 publications for only four months in 2017. We still found that satisfactory performance is achieved in all these articles, and we consider this to be better than the typical deep CNN or MLP performance based on our experience.

RQ3: What are the future research directions of transfer learning applications to image recognition? Answer: Although this is a hot area of research, still much needs to be done. We have listed our proposed directions in Section IV. Mostly transfer learning focuses on using knowledge from pre-trained CNNs in another CNN but we particularly need to investigate the effect of transferring knowledge learnt from other machine learning models and approaches (e.g. SVMs).

Table 3: Classification of Selected 28 Papers and their Performance

Paper Ref.	Performance	Classification of Paper
[22]	Accuracy: >90%	Unsupervised TL
[26]	Accuracy: >84%	Deep CNN based TL
[27]	Accuracy: 77%	Deep CNN based TL
[43]	AUC: 0.85	CNN Training + Linear Discriminate Analysis TL
[24]	Accuracy: >90%	Transfer Metric Learning Framework
[25]	Accuracy: >75%	Transfer Metric Learning Framework
[44]	65% Improvement in Performance	Improving CNN based TL for MRI
[28]	Accuracy: >80%	Deep CNN based TL
[45]	Accuracy: >90%	Logistic Classifier Training + Weighted Logistic Classifier TL
[29]	Accuracy: >90%	Deep CNN based TL
[42]	Accuracy>93%	Deep CNN based TL
[46]	Accuracy: >85%	Improving TL in Unshared Feature Space
[30]	Accuracy: >80%	Deep CNN based TL
[31]	RMSE: 5.6	Deep CNN based TL
[32]	Accuracy: >70%	Deep CNN based TL
[47]	Improved Performance	Facial Recognition Transfer Learning
[48]	Accuracy: >90%	Active Transfer Learning Approach

[49]	Accuracy: >90%	SVM Based Transfer Learning
[33]	Accuracy: >85%	Deep CNN based TL
[23]	Accuracy: >90%	Unsupervised TL
[34]	Accuracy: >72%	Deep CNN based TL
[35]	Accuracy: >90%	Deep CNN based TL
[36]	Accuracy: >90%	Deep CNN based TL
[37]	Accuracy: >90%	Deep CNN based TL
[38]	Accuracy: >84%	Deep CNN based TL
[39]	Error: 7.1%	Deep CNN based TL
[40]	Accuracy: >90%	Deep CNN based TL
[41]	Accuracy: >73%	Deep CNN based TL

6 Conclusion

This paper has presented a systematic literature review on applications of transfer learning approach to image recognition. We have followed the standard procedure for conducting such a review. Due to several constraints, we reviewed publications only in period January 2017 - April 2017. Even in this limited time frame, we retrieved 28 relevant articles and analyzed them thoroughly over the more standard metrics. Results show that transfer learning brings a significant boost in efficiency and accuracy to image recognition, particularly using convolution neural networks. Based on current domains, we motivate its application to other domains such as cyber security and also indicate the need to thoroughly present an experimentation with all transfer learning procedures for image recognition with convolution neural networks. Our future research work is to review the 1000-odd papers from 2013 to 2017 and to validate our results across this time period. We also plan to conduct an experimentation of standard transfer learning methods with changes in CNN hyper parameters as suggested in our research directions in Section IV.

Acknowledgment

The authors would like to thank Usman Ali (PhD Scholar at Institute of Business Administration) for his valuable feedback on this paper.

References

[1] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in Computer Vision – ECCV 2014, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833.

[2] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning.

[3] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in NIPS Deep Learning and Representation Learning Workshop, 2015. [Online]. Available: http://arxiv.org/abs/1503.02531

[4] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition.” CoRR, vol. abs/1310.1531, 2013.

[5] E. Alpaydin, Machine Learning: The New AI.

[6] L. Deng and D. Yu, Deep Learning: Methods and Applications. Hanover, MA, USA: Now Publishers Inc., 2014.

[7] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," Mechanical Systems and Signal Processing, vol. 107, pp. 241 – 265, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0888327017306064

[8] D. Yu and L. Deng, "Deep learning and its applications to signal and information processing [exploratory dsp]," IEEE Signal Processing Magazine, vol. 28, no. 1, pp. 145–154, Jan 2011.

[9] Y. Cai and S. See, GPU Computing and Applications, 1st ed. Springer Publishing Company, Incorporated, 2016.

[10] J. Nickolls and W. J. Dally, "The gpu computing era," IEEE Micro, vol. 30, no. 2, pp. 56–69, Mar. 2010. [Online]. Available: http://dx.doi.org/10.1109/MM.2010.41

[11] S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Trans. on Knowl. and Data Eng., vol. 22, no. 10, pp. 1345–1359, Oct. 2010. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2009.191

[12] B. Tan, Y. Song, E. Zhong, and Q. Yang, "Transitive transfer learning," in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '15. New York, NY, USA: ACM, 2015, pp. 1155–1164. [Online]. Available: http://doi.acm.org/10.1145/2783258.2783295

[13] T. Galanti, L. Wolf, and T. Hazan, "A theoretical framework for deep transfer learning," Information and Inference: A Journal of the IMA, vol. 5, no. 2, pp. 159–209, 2016. [Online]. Available: + http://dx.doi.org/10.1093/imaiai/iaw008

[14] L. Yang, S. Hanneke, and J. Carbonell, "A theory of transfer learning with applications to active learning," Mach. Learn., vol. 90, no. 2, pp. 161–189, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1007/s10994-012-5310-y

[15] L. Y. Pratt, "Discriminability-based transfer between neural networks," in Advances in Neural Information Processing Systems 5, [NIPS Conference]. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993, pp. 204–211. [Online]. Available: http://dl.acm.org/citation.cfm?id=645753.668046

[16] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," Journal of Big Data, vol. 3, no. 1, p. 9, May 2016. [Online]. Available: https://doi.org/10.1186/s40537-016-0043-6

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, ser. NIPS'12. USA: Curran Associates Inc., 2012, pp. 1097–1105. [Online]. Available: http://dl.acm.org/citation.cfm?id=2999134.2999257

[18] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in Shape, Contour and Grouping in Computer Vision. London, UK, UK: Springer-Verlag, 1999, pp. 319–. [Online]. Available: http://dl.acm.org/citation.cfm?id=646469.691875

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pp. 770–778.

[20] D. Etherington, "Udacity open sources its self-driving car simulator for anyone to use," https://beta.techcrunch.com/2017/02/08/udacity-open-sources-its-self-driving-car-simulator-for-anyone-to-use/, 2017, [Online; accessed 9-February-2017].

[21] B. Kitchenham, "Procedures for performing systematic reviews," Department of Computer Science, Keele University, UK, Keele University. Technical Report TR/SE-0401, 2004.

[22] H. Chang, J. Han, C. Zhong, A. Snijders, and J.-H. Mao, "Unsupervised transfer learning via multi-scale convolutional sparse coding for biomedical applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1, 2017. [Online]. Available:https://doi.org/10.1109/tpami.2017.2656884

[23] R. Kemker and C. Kanan, "Self-taught feature learning for hyperspectral image classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 5, pp. 2693–2705, may 2017. [Online]. Available: https://doi.org/10.1109/tgrs.2017.2651639

[24] Z. Ding and Y. Fu, "Robust transfer metric learning for image classification," IEEE Transactions on Image Processing, vol. 26, no. 2, pp. 660–670, feb 2017. [Online]. Available: https://doi.org/10.1109/tip.2016.2631887

[25] Y. Xu, S. J. Pan, H. Xiong, Q. Wu, R. Luo, H. Min, and H. Song, "A unified framework for metric transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 6, pp. 1158–1171, jun 2017. [Online]. Available: https://doi.org/10.1109/tkde.2017.2669193

[26] P. Burlina, K. D. Pacheco, N. Joshi, D. E. Freund, and N. M. Bressler, "Comparing humans and deep learning performance for grading AMD: A study in using universal deep features and transfer learning for automated AMD analysis," Computers in Biology and Medicine, vol. 82, pp. 80–86, mar 2017. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2017.01.018

[27] K. H. Cha, L. M. Hadjiiski, H.-P. Chan, R. K. Samala, R. H. Cohan, E. M. Caoili, C. Paramagul, A. Alva, and A. Z. Weizer, "Bladder cancer treatment response assessment using deep learning in CT with transfer learning," in Medical Imaging 2017: Computer-Aided Diagnosis, S. G. Armato and N. A. Petrick, Eds. SPIE, mar 2017. [Online]. Available: https://doi.org/10.1117/12.2254977

[28] M. M. Ghazi, B. Yanikoglu, and E. Aptoula, "Plant identification using deep neural networks via optimization of transfer learning parameters," Neurocomputing, vol. 235, pp. 228–235, apr 2017. [Online]. Available: https://doi.org/10.1016/j.neucom.2017.01.018

[29] W. Ge and Y. Yu, "Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning," CoRR, vol. abs/1702.08690, 2017. [Online]. Available: http://arxiv.org/abs/1702.08690

[30] H. Kaya, F. Gurpınar,¨ and A. A. Salah, "Video-based emotion recognition in the wild using deep transfer learning and score fusion," Image and Vision Computing, vol. 65, pp. 66–75, sep 2017. [Online]. Available: https://doi.org/10.1016/j.imavis.2017.01.012

[31] Y. Yuan, X. Zheng, and X. Lu, "Hyperspectral image superresolution by transfer learning," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 10, no. 5, pp. 1963–1974, may 2017. [Online]. Available: https://doi.org/10.1109/jstars.2017.2655112

[32] "RGB-D salient object detection based on discriminative cross-modal transfer learning," CoRR, vol. abs/1703.00122, 2017, withdrawn. [Online]. Available: http://arxiv.org/abs/1703.00122

[33] F. Jiang, H. Liu, S. Yu, and Y. Xie, "Breast mass lesion classification in mammograms by transfer learning," in Proceedings of the 5th International Conference on Bioinformatics and Computational Biology, ser. ICBCB '17. New York, NY, USA: ACM, 2017, pp. 59–62. [Online]. Available: http://doi.acm.org/10.1145/3035012.3035022

[34] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol. 542, no. 7639, pp. 115–118, jan 2017. [Online]. Available: https://doi.org/10.1038/nature21056

[35] P. M. Cheng and H. S. Malhi, "Transfer learning with convolutional neural networks for classification of abdominal ultrasound images," Journal of Digital Imaging, vol. 30, no. 2, pp. 234–243, nov 2016. [Online]. Available: https://doi.org/10.1007/s10278-016-9929-2

[36] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," CoRR, vol. abs/1707.07012, 2017. [Online]. Available: http://arxiv.org/abs/1707.07012

[37] M. S. Elmahdy, S. S. Abdeldayem, and I. A. Yassine, "Low quality dermal image classification using transfer learning," in 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI). IEEE, 2017. [Online]. Available: https://doi.org/10.1109/bhi.2017.7897283

[38] A. Romero-Lopez, X. G. i Nieto, J. Burdick, and O. Marques, "Skin lesion classification from dermoscopic images using deep learning techniques," in Biomedical Engineering. ACTAPRESS, 2017. [Online]. Available: https://doi.org/10.2316/p.2017.852-053

[39] M. Turan, Y. Almalioglu, E. Konukoglu, and M. Sitti, "A deep learning based 6 degree-of-freedom localization method for endoscopic capsule robots," CoRR, vol. abs/1705.05435, 2017. [Online]. Available: http://arxiv.org/abs/1705.05435

[40] G. J. Scott, M. R. England, W. A. Starms, R. A. Marcum, and C. H. Davis, "Training deep convolutional neural networks for land–cover classification of high-resolution imagery,"

IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 4, pp. 549–553, apr 2017. [Online]. Available: https://doi.org/10.1109/lgrs.2017.2657778

[41] J. Lemley, S. Bazrafkan, and P. Corcoran, "Transfer learning of temporal information for driver action classification," in 28th Modern Artificial Intelligence and Cognitive Science Conference, 04 2017.

[42] S. P. K. Karri, D. Chakraborty, and J. Chatterjee, "Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration," Biomedical Optics Express, vol. 8, no. 2, p. 579, jan 2017. [Online]. Available: https://doi.org/10.1364/boe.8.000579

[43] B. Q. Huynh, N. Antropova, and M. L. Giger, "Comparison of breast DCE-MRI contrast time points for predicting response to neoadjuvant chemotherapy using deep convolutional neural network features with transfer learning," in Medical Imaging 2017: Computer-Aided Diagnosis, S. G. Armato and N. A. Petrick, Eds. SPIE, mar 2017. [Online]. Available: https://doi.org/10.1117/12.2255316

[44] M. Ghafoorian, A. Mehrtash, T. Kapur, N. Karssemeijer, E. Marchiori, M. Pesteie, C. R. G. Guttmann, F.-E. de Leeuw, C. M. Tempany, B. van Ginneken, A. Fedorov, P. Abolmaesumi, B. Platel, and W. M. Wells, "Transfer learning for domain adaptation in MRI: Application in brain lesion segmentation," in Lecture Notes in Computer Science. Springer International Publishing, 2017, pp. 516–524. [Online]. Available: https://doi.org/10.1007/978-3-319-66179-7 59

[45] V. Cheplygina, I. P. Pena, J. H. Pedersen, D. A. Lynch, L. Sorensen, and M. de Bruijne, "Transfer learning for multi-center classification of chronic obstructive pulmonary disease," IEEE Journal of Biomedical and Health Informatics, pp. 1–1, 2017. [online]. Available: https://doi.org/10.1109/jbhi.2017.2769800

[46] W. Chang, Y. Wu, H. Liu, and Y. Yang, "Cross-domain kernel induction for transfer learning," in Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA., 2017, pp. 1763–1769.

[47] A. Anuse, N. Deshmukh, and V. Vyas, "Transfer subspace learning model for face recognition at a distance," vol. 9, pp. 27–32, 01 2017.

[48] Y. Guo, G. Ding, Y. Wang, and X. Jin, "Active learning with cross-class knowledge transfer," in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, ser. AAAI'16. AAAI Press, 2016, pp. 1624–1630. [Online]. Available: http://dl.acm.org/citation.cfm?id=3016100.3016126

[49] I. Becht, "Individualized smile detection using transfer learning," Master's thesis, Utrecht University, 2017.

[50] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," Pattern Recognition, vol. 77, pp. 354–377, may 2018. [Online]. Available: https://doi.org/10.1016/j.patcog.2017.10.013

Performance Evaluation of Unmanned Aerial Vehicles for video based Surveillance Applications

Muhammad Naveed[1] Sameer Qazi[2]

Abstract

It is always beneficial to look for latest technology introduced so far to ensure proper surveillance. In recent times, unmanned aerial vehicles (UAVs) represent a new potential market and have appealed a lot of attention. UAVs serve a lot for law enforcing agencies for constant video surveillance at remote locations in rural areas where otherwise required a lot of resources. Airborne monitoring through camera mounted UAVs assists to capture the real time streaming video from different vantage points to a single commanding and controlling unit. We suggest a UAV based framework that would lead to a timely action in any criminal activity or disaster prevention. In this paper we analyzed the throughput of indoor UEs and outdoor UEs of one such video streaming system by using a model of wireless propagation, multipath propagation loss, shadowing and fading.

1 Introduction

For the constant and proper video surveillance of any area in a remote location, it is always a good practice to use modern technology instead of human physical interaction. In recent times the modern electronic spy's, camera mounted Unmanned Aerial Vehicles (UAVs) can be used to achieve this objective [1]. UAVs serve in surveillance of farms, floods, highway traffic monitoring [2], terrorism, natural disaster prevention etc. We cannot ignore the probabilities of terrorist attacks especially in the buildings positioned at distant urban or open areas; here we are concerned with the performance of video surveillance of such buildings.

There are several vantage points that exist in indoor and outdoor of such sites whose surveillance is supposed to be carried out. We have suggested a two tiered design for closed circuit aerial monitoring for the kind of scenario. For this work we used Wireless 4G LTE technology based on cellular infrastructure (Figure 1).

For real time video streaming, several indoor and outdoor cells are used which we called femto cells and macro cells respectively in this structural design to construct a closed circuit monitoring framework.

The topological concept and related terminologies are adopted from 3GPP R4-092042 standard. There are several camera mounted UAVs in this architecture. As far as indoor framework is concerned; we have certain UAVs inside the building that are part of femto cells referred to as home UEs responsible for the transmission of real time video to their relevant base stations called home ENBs. In the similar way outdoor framework consists of outdoor macrocells where UAVs are flying constantly, referred to as macro UEs whom duty is to convey video in real time

[1] *Karachi Institute of Economics & Technology, Karachi | naveed@pafkiet.edu.pk*
[2] *Karachi Institute of Economics & Technology, Karachi | sameer.qazi@pafkiet.edu.pk*

to their corresponding base stations mentioned as macro ENBs. For Enhanced Node we used the word ENB whereas for User Equipment we used the term UE.

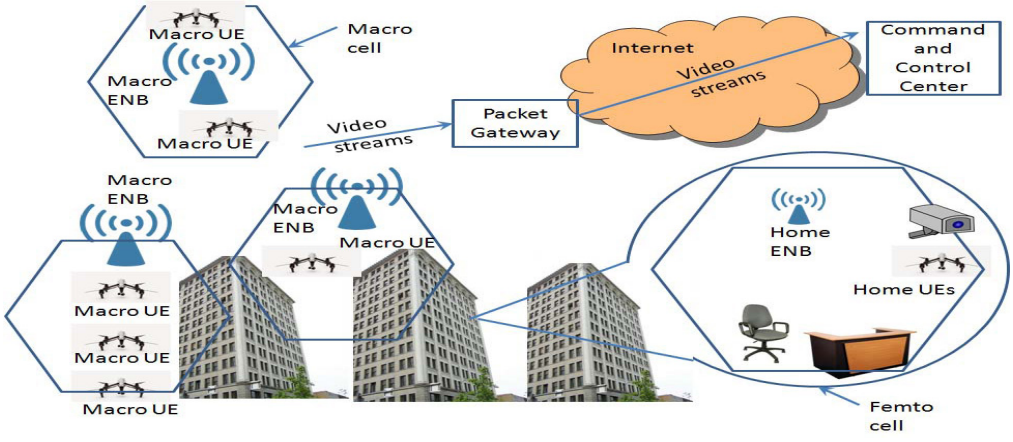


Figure 1: Two tiered Building Surveillance System comprising both Macro and Femto Cells

Indoor and outdoor captured streams of video are simultaneously transmitted to a single command and control center over Internet link. Topological structure of such kind would lead to well-timed action in wrongdoing or disaster avoidance.

Following are our contributions in this work.

(1) For appropriate real time video surveillance of different vantage spots outside and inside the buildings located at remote location in urban or open areas, we suggested a two tiered architecture using multi UAVs that are camera mounted.
(2) Investigating the consequences of wireless propagation set-ups to examine the end results of distortion in wireless signal because of multipath propagation at applied flying speeds of UAVs and signal propagation loss also known as shadowing loss over towering building infrastructures and; (3) Analysis of the Quality of Service (QoS) of such a multi-UAV framework supporting real time video streaming with video format encoded in MP4.

Under the 4G LTE standards, for macro cell we operated models supporting flight patterns of UAV. To accomplish this we took into our consideration the actual frequency band (2.1 GHz) and handover algorithms for smooth mobility for the proper communication of wireless nodes. At the end, we used Network Simulator-3 (version 3.26) to endorse our work by simulating our experiments [3].

2 Related Work

Very important survey paper on Flying Adhoc Networks (FANETs) by Becmezci et al [4] highlighted the capability of FANETs in recent applications. Another new direction given by Eckert et al [5] explored new propagation models for newly introduced application in Aerial Adhoc Networks like communication among paragliders during flight; prior to that several researchers have supposed simplistic LoS propagation models in their research because of the high flying altitudes of UAVs.

For maximum network connectivity, Haibo et al [8] have considered the Airborne Networks nodes placement; though in their study UAVs are serve as the relay point in the sky for mobile nodes while on ground it is impossible that UAVs reach each other without the flying UAVs; in contrast in our study we did not focus the placement of nodes on the ground and supposed all nodes to be mobile in air in constant flight.

Previously in recent research, researcher like Sahingoz in his paper [6] [7] discussed the issues involved in networks support mobility using multiple UAVs in flying communication and raised various interesting queries on path planning, QoS and protocol suitability but did not concentrate on propagation model in his research. We tried in this paper to explore some of the unanswered issues of the performance of routing and real propagation models using a simulated environment in NS3.

3 Wireless Propagation Models Used

Multi-path propagation and shadowing are the major hurdles faced by wireless signal propagation as compared with wired signal propagation which corresponds to a deterministic model differs as per distance. We have used Friis propagation model under Line of Sight (LoS) signal propagation as shown in equation 1 mentioned below.

If we are talking about shadowing loss, it is the phenomenon when wireless signal has to go through hurdles (for instance walls); this condition occurs as there is no clear line of sight in between transmitter and receiver. On the other hand, the second major issue is fading or multipath propagation which is the phenomenon in which the wireless signals look discontinuous reception of LoS signals and reflected forms of signals from additional hurdles. Here we are interested to investigate the effects of shadowing loss in our infrastructure.

A Shadowing Loss

We have used Hybrid Buildings Propagation model as the standard model for indoor wireless signal propagation. This model is embedded in latest versions of ns-3 and its beauty is its multiple properties as it is the combination of ITU-R P.1238 for indoor communications, ITU-R P.1411 model for small range communications, Hata model and COST231.

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi d)^2} \qquad (1)$$

$P_r(d)$ = Received power at distance d

P_t = Transmit power

$G_t G_r$ = Transmit and Receive Antenna Gains

Under different setups we used this model to examine the path loss. Table 1 given below represents the parameters for shadowing loss we used in this work as standard; where we had to introduce any variation in the simulation settings for any specific experiment we mentioned that explicitly in its experimental results in this paper.

Table 1. Building Propagation Model Parameters

LOS TO NLOS THRESHOLD	200 M
INTERNAL WALL LOSS	5 DB
STANDARD DEV OF THE NORMAL DISTRIBUTION TO CALCULATE SHADOWING DUE TO EXTERNAL WALLS	7
STANDARD DEV OF THE NORMAL DISTRIBUTION TO CALCULATE SHADOWING FOR MACROUES	8
STANDARD DEV OF THE NORMAL DISTRIBUTION TO CALCULATE SHADOWING FOR HOMEUES	5

4 Mobility Model for MacroUEs- Gauss Markov

In this work to mimic the actual real world flight pattern of outdoor UE which we called as macroUE, Gauss-Markov mobility model is used [9] [10]. This model uses three variables to update its pitch, direction and speed. The motion along z-axis is examined by the pitch variable while the direction and speed variable command the new direction and speed in the x-y plane. Equations 2, 3 and 4 are given below to show this.

$$s_n = \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2)}s_{x_n-1} \quad (2)$$

$$\theta_n = \alpha\theta_{n-1} + (1-\alpha)\bar{\theta} + \sqrt{(1-\alpha^2)}\theta_{x_n-1} \quad (3)$$

$$p_n = \alpha p_{n-1} + (1-\alpha)\bar{p} + \sqrt{(1-\alpha^2)}p_{x_n-1} \quad (4)$$

Parameters set for the simulation of Gauss Markov Model for our framework are given in Table 2.

As we have already discussed, our infrastructure is based on two types of user equipment's and these are home UEs and macro UEs. As macro UEs are constantly flying around the buildings, these nodes are considered being mobile; because of their mobility these camera mounted UAVs are frequently detached and attached from one macro ENB to another; the phenomenon is called handover as it happens normally in consistent cellular systems. The criterion of handover algorithms depend on the supreme received signal from the macrocell base stations.

Table 2: Gauss Markov Model Parameters to simulate mobility of macro UEs

Default PITCH	GAUSSIAN RV(BOUND=0.04, MEAN=0, VAR =0.02)
Default DIRECTION	GAUSSIAN RV (BOUND=0.4,MEAN=0, VAR=0.2)
Default VELOCITY	GAUSSIAN RV (BOUND=0, MEAN=0, VARIANCE=0)
Average PITCH	UNIFORM RV (Minimum =0.05, Maximum =0.05)
Average DIRECTION	UNIFORM RV (Minimum=0, Maximum=6.28)
Average VELOCITY	VARIABLE 1-10 M/S
Value of ALPHA	0.85
Assigned TIME STEP	0.5 SECONDS

In contrast the home UEs are assumed to be stationary in this work which means either these are miniature UAVs that choose resting position at any point inside the building (on a cabinet or window sill) to discover detection or these are fixed wireless IP cameras embedded discretely into walls of the rooms. Home UEs are connected with their base stations or access points that we referred as home ENBs. Inside the building remote host connectivity is established by an internet link over a protected and secure channel.

5 Simulation Environment and Parameters

As mentioned in Figure 1, we have adopted the network topology from 3GPP R4-092042 standard. NS-3 version 3.26 is used for performing all simulations. The standard parameters taken for this study are mentioned in Table 3.

6 Performance Evaluation

We analyzed the throughput of home UEs and macro UEs by changing different parameters. We did four experiments to judge the increase or decrease in throughput; two experiments for home UEs and two for macro UEs. We were also interested to find out the causes behind such an increase or decrease in throughput.

For the simulations we used Evalvid Client-Server application developed by GERCOM Group for ns3 [11]. Using this application the MPEG-4 encoded video was used for surveillance recordings of the buildings.

In contrast with the original GERCOM code we introduced some amendments; In the original GERCOM code macroUEs use a random waypoint mobility model while we used Gauss-Markov mobility model to mimic macroUEs, second in original code the video flows from remote hosts to the macroUEs/homeUEs but in our situation the videos are run by using UDP from the macroUE/homeUE to a remote host.

A Impact of Line of Sight to Non Line of Sight (Los2NLoS) threshold on throughput of MacroUEs

For this experiment, we simulated by varying the LoS2NLoS threshold from 200 to 300 with stepping 10 as shown in Figure 2. We observed very interesting trend of increase in average throughput of macroUEs; and it is natural phenomenon as more we go with larger LoS value, we found greater throughput as there is no obstacle found within the LoS range; hence the increase in throughput from urban towards open areas was analyzed. All the parameters are same as given in Table 3 except the values of LoS2NLoS threshold which we gradually increased for this simulation.

Table 3: Simulation parameters as per 3GPP R4-092042 specification

Rooms/apartments	4
Number of floors	4
Total number of femtocells	2-8
Total number of macroenb sites	1-4
Area margin factor	0.5
Density of macroue (number of macroues per square meter)	0.00002
Deployment ratio of homeenb	0.2
Activation ratio of homeenb	0.5
Homeues to homeenb ratio	1
Macroues/homeues	20 / 10
Macroenb /homeenb tx power	46 / 20 dbm
Macroenb dlearfcn / homeenb dlearfcn	100
Macroenb ulearfcn / homeenb ulearfcn	18100
Homeenb bandwidth (in terms of resource Blocks rbs)	100
Macroenb bandwidth (in terms of resource Blocks rbs)	100
Numbers of bearers per ue	1
Srs periodicity	80
Scheduler	Proportional Fair

B Impact of number of MacroENB sites on throughput of MacroUEs

By configuring in the settings of our simulation, we kept the maximum velocity of macroUEs constant i.e. 2m/s.

Increasing the number of macroENB sites decreased the average throughput of macroUEs as shown in Figure 3, the reason behind this decrease is handover among macroENB sites; as in limited distance of 500 meters we increased the number of macroENB sites, the faster switching of macroUEs occurred in handover from one macroENB site to another that resulted in the decrease of overall macroUEs throughput. So with this analysis we figured out that in limited range of 100 meters it is useless to increase the number of macroENB sites because of higher frequency of handover.

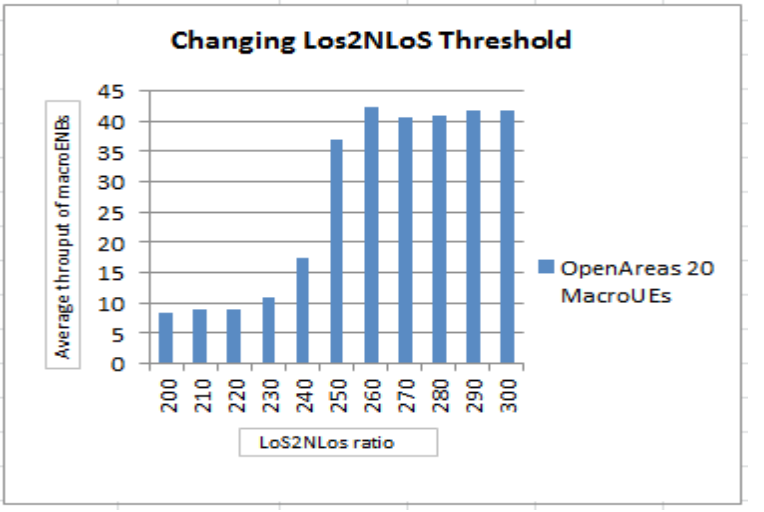


Figure 2: Changing LoS2NLoS threshold from 200 to 300 m

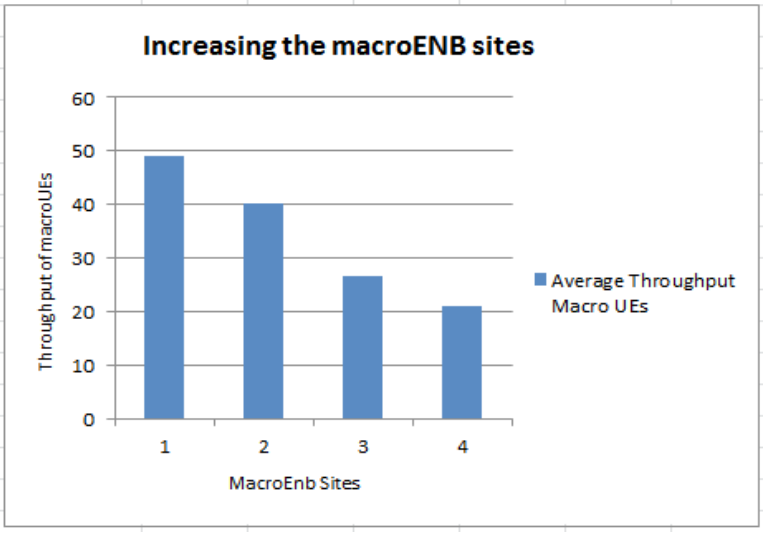


Figure 3: Increasing the number of macroENB sites from 1 to 4

C Impact of Internal Wall Loss of different materials on throughput of homeUEs

For this experiment; under our simulation settings we allocated our homeUEs in random rooms and homeUEs per homeENB ratio 2. We took different material into consideration for this analysis in our simulation as shown in Table 4. We found decreasing trend in average throughput of homeUEs that is obvious effect of attenuation in signal as seems in Figure 4; so throughput also depends on the material of building structure.

D Impact of homeUEs per homeENBs ratio on average throughput of homeUEs

For this experiment, under our simulation settings, we allocated our homeUEs in random rooms and increased the homeUEs per homeENBs ratio from 0.5 to 2 as shown in Figure 5. We found decreasing trend in average throughput of homeUEs; and it seemed to be obvious because

gradual increase in number of homeUEs per homeENBs decreases the average throughput as burden increases on homeENBs; the second reason of this decreasing trend in average throughput is that for constant number of homeUEs if we varied the homeUE to homeENBs ratio, we experienced the change in average throughput of homeUEs. A higher homeUE to homeENBs ratio implies less probability of homeENBs and homeUEs to be in close proximity of each other so higher propagation loss introduced; hence in this analysis we can analyzed that the optimal ratio should be adjusted for simulation otherwise we have to face the decrease in average throughput as penalty.

Table 4. List of materials to analyze internal wall loss

Material	Thickness	Wall loss (db)
Glass	13 mm	2
Lumber	76 mm	2.8
Brick	267 mm	7
Reinforced Concrete	89 mm	27
Concrete	305 mm	35

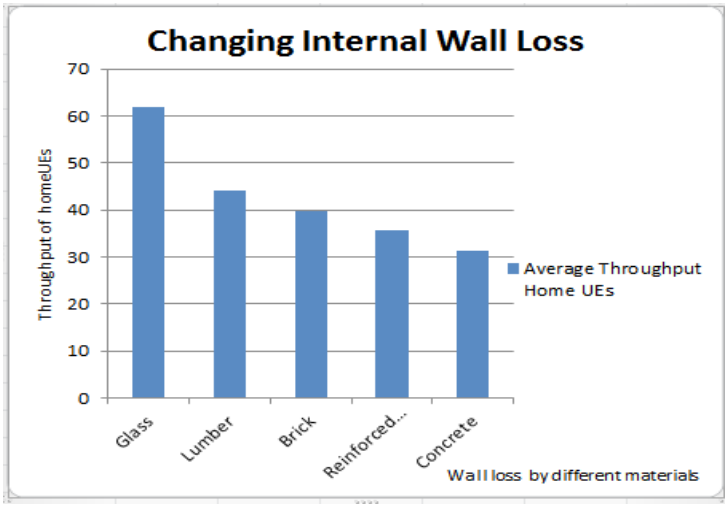


Figure 4: Changing Internal Wall Loss (db.) for different materials

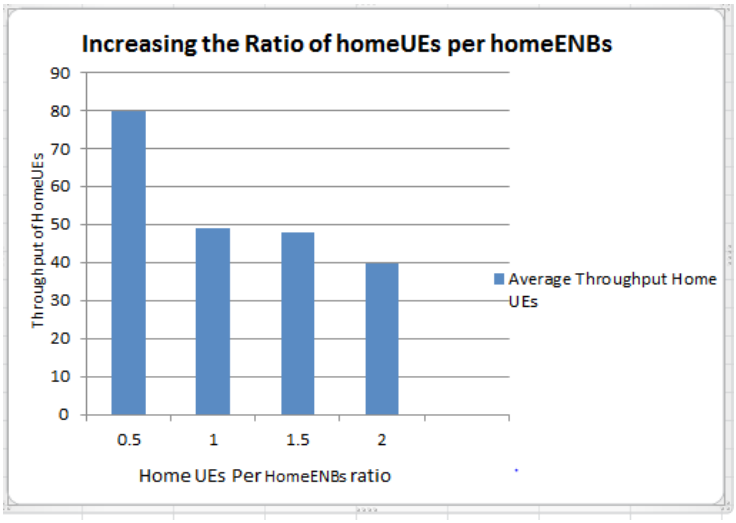


Figure 5: Increasing the ratio of homeUEs per HomeENBs from 0.5 to 2

7 Conclusion and Future Work

The overall motive of this study was to optimize the Quality of Service (QoS) of building surveillance networks based on Aerial UAVs by analyzing the effects on data throughput using realistic mobility and propagation models to mimic real world UAV based frameworks. In this paper we tried to highlight some important aspects in this area. As future work, we will consider the performance of different types of encoded video streaming by analyzing the average Picture Signal to Noise Ratio (PSNR) to maximize the end user Quality of Experience (QoE) for video viewing.[12].

References

[1] A. Merwaday and I. Guvenc, "UAV assisted heterogeneous networks for public safety communications," presented at Wireless Communication and Networking Conference (WCNCW) 2015, pp. 329–334.

[2] S. Srinivasan, H. Latchman, J. Shea, T. Wong, and J. McNair, "Airborne traffic surveillance systems: video surveillance of highway traffic," Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks 2004, p. 131.

[3] "NS3 Network Simulator," Oct-2017. [Online]. Available: http://www.nsnam.org.

[4] İ Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying Ad-Hoc Networks (FANETs): A survey," Ad Hoc Netw., vol. 11, no. 3, pp. 1254–1270, May 2013.

[5] J. Eckert, D. Eckhoff, and R. German, "A deterministic radio propagation model for inter-paraglider communication," Wireless on demand Network Systems and Services (WONS) 11th Annual Conference 2014, pp. 138–142.

[6] O. K. Sahingoz, "Mobile networking with UAVs: Opportunities and challenges," Unmanned Aircraft Systems (ICUAS) International Conference 2013, pp. 933–941.

[7] O. K. Sahingoz, "Networking Models in Flying Ad-Hoc Networks (FANETs): Concepts and

Challenges," J. Intell. Robot. Syst., vol. 74, no. 1–2, pp. 513–527, Apr. 2014.

[8] H. Wang, D. Huo, and B. Alidaee, "Position Unmanned Aerial Vehicles in the Mobile Ad Hoc Network," J. Intell. Robot. Syst., vol. 74, no. 1–2, pp. 455–464, Apr. 2014.

[9] D. Broyles, A. Jabbar and J.P.G. Sterbenz, "Design and Analyis of a 3-D gauss-markov mobility model for highly-dynamic air bourne networks," presented at the International Telemetering Conference ITC, San Deigo, 2010.

[10] J. P. ROHRER, E. K. CETINKAYA, H. NARRA, D. BROYLES, K. PETERS, AND J. P. G. STERBENZ, "AERORP PERFORMANCE IN HIGHLY-DYNAMIC AIRBORNE NETWORKS USING 3D GAUSS-MARKOV MOBILITY MODEL," Military Communications Conference MILCOM 2011, PP. 834–841.

[11] "GERCOM,." [Online]. Available: http://www.gercom.ufpa.br.

[12] D. Ong and T. Moors, "Improving video quality in congested networks through deferred discard," Telecommunication Networks and Applications Conference (ATNAC) 2014, pp. 123–128.

Fault Prediction in Self-Healing Telecommunication Systems

Ahmad S. Kazmi [1]

Abstract

Telecommunication systems are heterogeneous networks with parts supplied by many vendors. Such complex systems face a number of faults that may deny services to the end users resulting in revenue losses to the telecommunication companies. Best case scenario is to avoid these faults completely, or failing that, correct the faults as soon as possible. Therefore there is a need for self-healing networks that can proactively predict and correct faults automatically. In this paper a fault prediction technique is presented that is useful in a self-healing network. The proposed technique first trains an artificial intelligence technique on the historical alarm data to find correlations and then uses these correlations to predict future alarms. The artificial intelligence techniques being used are, Artificial neural network, support vector machine, Kalman filter and hidden Markov model. In this paper we reported on artificial neural network. The proposed technique is applied on the alarm data from a real telecommunication company and prediction accuracies of the proposed technique are calculated. The details of the proposed fault prediction technique and results that suggest optimal parameters are presented. The proposed technique is effective in a proactive self-healing network.

Keywords: Self-healing Networks; Alarm; Faults; Neural Networks; Telecommunication system

1 Introduction

Today's telecommunication networks are complex heterogeneous systems consisting of components from multiple vendors. Recurring faults in a telecommunication network is a major reason for degradation and refusal of service. A faster fault resolution results in quick resumption of service. Therefore the need for self-healing networks have arisen [1] [2] [3] [4]. In a self-healing network faults are detected and corrected automatically. In a more proactive self-healing network faults are predicted and corrected before happening resulting in less network failures and degradation of the service. There are two parts to a proactive self-healing network: one is prediction of impending faults and second is the root cause analysis of the impending faults [5]. This paper deals with the first part, fault prediction, and suggests how to do the second part, root cause analysis. The reported self-healing networks [1] [2] use current state of the network to make a prognosis of an impending fault. One approach models the degrading factor as additive faults and fault estimate sequence is treated as time series [3]. Another approach uses compensating service degradation by readjustment of parameters using fuzzy logic in a wireless healing

network [4]. We have taken a different approach of using historical alarms to find correlations between components of a telecommunication network and then predicting faults.

[1] *University of Central Punjab, Lahore | ahmad.kazmi@ucp.edu.pk*

Any telecommunication network management system must have a mechanism of collection and reporting of network faults in the form of alarms. These alarms contain a lot of information about the health of the system over time. Unfortunately this valuable information is hidden in a large volume of alarms. Therefore use of an automated tool is a must to make sense of the alarm data and develop algorithms to find hidden patterns. There are two types of correlations that exist in the alarm data: one is spatial relation between network components and second is the temporal relation between sequence of alarms that can be considered as a time series. ANN has been proven to be valuable in recognizing patterns not found by other techniques in a time series [10] [11].Various applications of ANN have been reported for pattern recognition [12] [13]. ANN has the inherent property of making sense of the correlations found in nonlinear data. The addition of the hidden neuron layer and positive or negative feedback mechanism is the main reason for ANN's success. The fault prediction, from historical alarm data, can be formulated as a pattern recognition (or a classification) problem.

In the proposed technique the artificial neural network (ANN) is trained to learn to find these correlations and use this learning to predict future alarms. The proposed ANN based technique predicts faults by formulating an input matrix based on historical alarm data. The input (feature) matrix contains spatial and temporal features that help ANN in finding patterns and correlations. We believed that the formation of the input matrix and our application of the ANN are new, unique and do better fault predictions. In this paper we reported the details of the proposed ANN based fault prediction technique. A number of experiments have been conducted to calculate prediction parameters and accuracy of the prediction. The accuracy of prediction is calculated by comparing the predicted fault occurrences with the actual fault occurrences. The optimal parameters are reported that produce the best prediction accuracies of a single event. It is believed that the same approach is useful in any network to find optimal parameters for accurate faults prediction using ANN. Furthermore an algorithm is suggested on how to proactively selfheal a telecommunication network.

This paper is arranged in five sections as follows. First section is the introduction. Second section is about the related work. Third section presents the proposed ANN based fault prediction technique in detail. 4th section presents the results and optimal parameters. Last section is about conclusions and possible future research directions.

2 Related Work

A Self-healing Networks

Telecommunication networks are becoming more complex as more functionalities are being added. This complexity is further enhanced due to multivendor components servicing highly heterogeneous networks. It is very common that a typical telecommunication network supports various communication channels (wired, wireless, and a combination) and technologies (SONET, GSM, CDMA, 3G, 4G etc.). The Telecommunication Management Network paradigm of FCAPS (what is FCAPS) needs changes to become more autonomous. An automatically managed network will be self-organizing requiring self-configuration, self-optimization and self-healing

[2]. Such an automated network is called a Self-Organizing Network (SON) (www.fp7-socrates.org). An important feature of SON is self-healing.

A self-healing network will proactively and automatically monitor, diagnose and correct anomalies and faults. A self-healing network can be considered as one that can automatically manage faults. Although that automatic fault management must also include a way to look at the system as a whole and not just rely on alarms from network elements. A number of techniques have been proposed on how to achieve this holistic approach to automatic fault management.

[1] proposes a hybrid fault prediction model that supports automatic self-healing networks. Based on the system situation the fault prediction model selects one of these methods: ID3 algorithm, fuzzy interference, fuzzy neural network and Bayesian network. The proposed self-healing system consists of five modules: system monitoring module collects system data, resource level evaluator module decides the level of detail in the system, prediction model selector module selects a suitable algorithm that can be applied for the current state, prediction model algorithm executor applies the selected prediction algorithm and model updater module feedbacks the resulting prediction to the four modules.

[2] presents a unified self-healing network model that identifies tasks to be performed automatically. First task is automatic information collection from these sources: configuration parameters, alarms, network counters, network traces, real time monitoring, drive tests, Key Performance Indicators (KPI) and context information. The second task is fault detection that includes identification of out of service components and service degradation components. Of course identification of out of service components is easy, but the identification of degrading components will require looking at alarms and KPIs. The third task is diagnosis requiring identification of faults and then the corresponding remedial actions. The fourth task is fault compensation that requires changing system parameters for graceful degradation of service rather than abrupt drop of service. The last task is reporting and storing the current steps for future use.

[3] Presents a fault prediction based self-healing approach. First equation for a discrete time dynamic system is provided and then any degradation of the system is modeled as another input to the system, basically a fault. Based on this model, fault prediction and estimation is made. After fault prediction system reliability is predicted. Particle filtering is used to estimate faults. The fault estimates result in a time series that is processed using exponential smoothing.

[4] Uses Mamdani [14] type fuzzy logic to adjust system parameters to compensate the degrading network. For example antenna downward tilt can be controlled by a bisector defuzzyfication. The proposed system consists of three types of modules: Fuzzy Forward Module that applies the changes in the forward (increase) direction, Fuzzy Backward Module that applies the changes in the reverse (decrease) direction and Monitor module that monitors the wireless network performance. The proposed algorithm keeps applying changes via Fuzzy Forward module and monitors the performance. As soon as the performance is targeted degrading the Fuzzy Backward module acts in the opposite direction until optimal values are achieved.

[5] Targets the self-healing part of a SON and presents a framework for automatic detection (of what) and diagnosis of mobile communication systems. The detection part of the framework uses KPI from the network by a unified KPI interface that compares a KPI with a reference or profile and returns a level (0,1) indicating conformance. The KPI profile is built using sampled values of the KPI and using Cumulative Distribution Function (CDF). The diagnosis part of the framework targets either a root cause or a corrective action. In order to avoid false alarm/wrong diagnosis, a built in null target is used. The heuristic knowledge from an operator is stored in the form of KPI level deviation that indicates a fault and corrective steps that will restore the desired KPI level. The relationship between a KPI level and target root cause is stored in a structure called Report. A report consists of a KPI subset and diagnosis target of the report. A scoring system is used to identify the most suitable target from a list of targets (root causes) contained in the reports. The scoring system assigns a score to a target based on reports that best match deviations from KPI levels. Since a target may have different reports containing KPI subsets, a particular likelihood value is used for the KPI. Basically expert knowledge is used to find out the high level of KPI against a given target. A KPI may consistently happen with a target, therefore a consistency score between (0,1) is used.

This paper addresses the issue of fault detection in a self-healing network. The approach is to go one step further and predict a fault before happening. Furthermore, historical alarm data is used for fault prediction rather than other information sources. The reason is that active information collection from various resources of an active real network impacts on network performance and overhead of the mechanism needed for information collection. Furthermore, historical alarms over a period of time contain a lot of information about the network health and there is no need to collect information from other sources. The main problem with historical alarms is that the network health information is not easily accessible and identifiable. Fortunately the information in historical alarms can be considered as a classification or pattern matching problem and a number of artificial intelligence techniques are available for pattern matching or classification. Furthermore features in the historical alarm data are used such a way that leads to better fault predictions.

B Artificial Intelligence Techniques for Fault Prediction

There is a wealth of information hidden in the historical alarm data. The alarm data collected at various network points contain the state of the system health at any point in time. Any number of simple statistical techniques can reveal valuable facts about a system e.g. the most occurring faults and where and how these happen; how these faults are caused and how to eliminate these causes; average down time due to these faults; order of redundancy needed to keep the system running etc. Artificial intelligence techniques are needed to find or classify various patterns available in the alarm logs.The fault prediction can be considered as an event prediction problem. More specifically an event prediction can be considered as a pattern recognition or classification problem for a time series. A number of event prediction techniques, in general and fault prediction in particular, are reported in literature.

Intelligent prediction systems are expert systems that use a variety of different prediction (AI and non AI) techniques to make a hybrid system that can predict more accurately than individual prediction techniques. One of the studies [1] presents a hybrid intelligent system for learning about data. The architectures for Neural Networks (ANN), Fuzzy Intelligent systems (FIS), Evolutionary Computing (EC) and probabilistic reasoning (PR) are discussed along with the integration issues of these techniques.

Another research [2] presents a four layered FMAS (Fuzzy multi agent system) for stock market prediction. These four layers use ANN (Artificial Neural Networks), GA (Genetic Algorithm) and SOM (Self Organizing Maps) to develop a hybrid prediction model. The model is based on the co-ordination of intelligent agents that perform the functions of data processing and learning. GA has been used to automate the design process of ANN in some studies [3] and the process is proved to give better time series forecasting results than statistical models like ARIMA. Hence there has been huge focus on designing this kind of hybrid prediction systems for various predictions in different industries. Time series prediction systems have also been explored many times in the past for efficient and intelligence based forecasting of time series data.

3 Proposed Fault Prediction Using ANN

We present a new fault prediction technique that uses ANN on real historical alarm data of a telecommunication company. The proposed technique first statistically analyzes the alarm data and then formulates an input matrix for the ANN system. The formulation of input matrix is central to our approach. Therefore here we have presented some basis and basic terminologies that will be helpful in describing the proposed technique.

A Alarm Data

The historical alarm data of a telecommunication company is used to apply our proposed fault prediction technique. A typical alarm is shown in fig. 1.

Alarm ID	Alarm Type	Severity	Probable Case	Managed Object
42193	Link Down	Major	ALM_IMA_LINK_LCD	EMS T2000

Managed Element	Rack	Shelf	Slot	Reason	Date
PTP	1	1	7	Loss of Signal	2/09/2013:4:27

Figure 1: A typical Telecommunication Alarm

Alarms in telecommunication are messages describing some sort of abnormality or malfunctioning in the network. This malfunctioning might not be visible to the end user. The Network Operational Center (NOC) of the wireless network daily receives thousands of alarms generated by BTSs and BSCs. The receiving system stores these alarms in a database and displays these alarms to the operator. Later these alarms can be analyzed and converted into

useful data by a correlation system. The generation time and generation date field together make up the alarm time. The information about the location of the network element, which has sent this alarm, is scattered in several fields. These are slot number, BTS number, Link number etc.

The process of alarm correlation is difficult because of the complexity of network elements. These network elements produce different types of alarms and usually there are large variations in the order of alarms in a given time frame. A model based analysis of the data can be carried out with the assistance of a network management expert, who knows what patterns to expect. However the statistical analysis, as proposed here, gives us more insight into network behavior and we can detect patterns that enable us to predict future faults. Another important reason for statistically analyzing the data is that different telecommunication management experts may not agree on same set of rules. The historical alarm data is in chronicle order. The basic idea behind a correlation system is to discover recurrent patterns from alarm database. It usually consists of discovering alarm rules within a time window in the alarm database.

1) Analysis of the Real Time Alarm Data

Here we present some statistical analysis of the historical alarm data [9]. This analysis is essential for the formulation of the input matrix for the ANN application. The alarm data is analyzed to categorize alarms according to alarm predicates like severity, type and sub rack and slot numbers. Our goal in fault prediction is to predict one of the critical alarms as these alarms are actually faults.

Table 1: Percentages Of Alarms With Different Severities And Types

	Trunk	Running (define)	Comm.	SW	Signal
Warn.	0.0000	1.9107	0.0000	0.0014	0.0000
Minor	2.9367	0.0000	0.0014	0.0070	0.0000
Major	54.5566	4.5594	0.0000	2.7186	0.3131
Critical	0.0000	0.0000	25.7128	7.2822	0.0000

From the table 1 we can see the types and severities of faults that actually occur. Furthermore 11 categories from a total of 20 appear with non-zero percentage and some of these, three in this case, have less than 1% chance of appearing.

B Input Matrix Formulation for ANN

One major contribution of this paper is the formulation of the input matrix that is applied to the ANN for fault prediction. This input matrix contains a particular state of the historical alarms over a given time period. There are two types of information contained in the matrix: frequency of occurrence of an alarm property (alarm predicates) and frequency of occurrence of sequence of alarms (event sequences). The alarm predicates are determined by analysis of

the alarm data. One predicate of an alarm means presence or absence of certain property of that particular alarm. The alarms being used here have five types and four levels of severities or stages. This makes a total of 5x4 possibilities. These 20 alarm categories can appear on different sub racks and slots. There are sixteen sub racks and each of these have 16 Slots. Apart from these there are some other important alarms that usually occur on BTS – BSC interface (called PCF define). These alarms can also have the 20 types and severity. Therefore this results in a total of (20 x 16 x 16) + 20= 5140 predicates. We have analyzed the given historical alarm data and eliminated those predicates that do not have significant (less than 1 %) contribution. We identify 21 predicates that are present in the given alarm data. Table 3 shows these 21 predicates. These predicates are allotted predicate numbers for ease of use in our ANN system.

Table 2: The 21 Predicates Present In The Alarm Data

Equipment	Param. 1 (what do you mean by parameter, it is not clear)	Param.2	Should be changed	Alarm type	Pred. No.
Subrack	Subrack=2	PCF alarm	Communication	Critical	1
Subrack	Subrack=3	Slot=0	Trunk	Major	2
Subrack	Subrack=4	Slot=0	Trunk	Minor	3
Subrack	Subrack=4	Slot=0	Trunk	Major	4
Subrack	Subrack=5	Slot=0	Trunk	Minor	5
Subrack	Subrack=5	Slot=0	Trunk	Major	6
Subrack	Subrack=5	Slot=12	Software	Critical	7
Subrack	Subrack=6	Slot=0	Trunk	Major	8
Subrack	Subrack=6	Slot=13	Software	Critical	9
Subrack	Subrack=7	Slot=0	Trunk	Major	10
Subrack	Subrack=8	Slot=0	Trunk	Minor	11
Subrack	Subrack=8	Slot=0	Trunk	Major	12
Subrack	Subrack=9	Slot=0	Trunk	Major	13
Subrack	Subrck=10	Slot=0	Trunk	Major	14
Subrack	Subrck=10	Slot=12	Software	Critical	15
Subrack	Subrck=11	Slot=0	Trunk	Major	16
Subrack	Subrck=12	Slot=0	Trunk	Minor	17
Subrack	Subrck=12	Slot=0	Trunk	Major	18
BTS	BTS	PCF	Software	Major	19
BTS	BTS	X	Running	Warning	20
BTS	BTS	X	Running	Major	21

Please note that the predicates in table 3 are not generic and are identified for the given 6 months of alarm data. For any other alarm data, an analysis (table 1, 2& 3) must be done to identify the predicates to be used in the input matrix. The 21 predicates form one side of the state of the alarm data and correlates the alarms in a spatial sense, the other side is the

frequency of sequence of events that correlates the alarms in a temporal sense. For a given time period, all occurrences of all the episodes (of the selected 21 predicates) are calculated. An episode is any sequence of 2 alarms occurring during a given time period one after the other. Hence there can be 21x21=441 possible episodes of alarms and we can find the frequency with which each of these episodes occur in a given time. Therefore frequency of an episode can be used to calculate the probability (relative frequency) of one alarm occurring after another alarm so that following type of statement can be made.
"If an alarm with predicate 1 occurs then another alarm with predicate 6 will follow within 20 conds with probability (relative frequency) 0.54"

An alarm A at time t1 can have one of the 21 predicate values. Suppose alarm A is followed by alarm B. Then alarm B can also have one of the 21 predicate values. Therefore (1) is the formula that is used to calculate the probability of the episode A→B (B follows A or A leads to B)

$$P(A \rightarrow B) = \frac{frequency(A \rightarrow B)}{\sum_{i=1}^{21} frequency(A \rightarrow X_i)} \quad \forall A, B \in [X_1 \ldots . X_{21}] \quad (1)$$

The formula states that the probability that alarm A leads to B is equal to the frequency with which alarm A leads to alarm B divided by the sum of frequencies with which alarm A leads to any alarm having each of the 21 predicates values. Here it is important to note the difference between an alarm and a predicate. Each entry in our alarm data base is called an alarm and these alarms are then (after statistical analysis of data) categorized into 21 predicates. Each predicate is a set of properties. For example if an alarm has predicate value 1 then it is a major communication alarm on sub track 6 slot no. 3 of a BTS. (2) shows that the sum of frequencies with which A leads to alarms with each of the 21 predicate values is equal to 1.

$$P(A \rightarrow B) = 1 \quad \forall A \in [X_1 \ldots . X_{21}] \quad (2)$$

The probability (relative frequency) values for all episodes contain the temporal information needed for the input matrix to the ANN system. The input matrix contains spatial and temporal information about the current state of the telecommunication system as contained in the alarm data. The input matrix is basically a 21x21 matrix of probabilities. These are the probabilities of occurrence of all the episodes (21x21=441) in the current state. This input matrix is one of the main contributions of this paper.

C Proposed Fault Prediction Methodology

We propose an ANN system (fig. 2) for the fault prediction from a given set of historical alarm data.

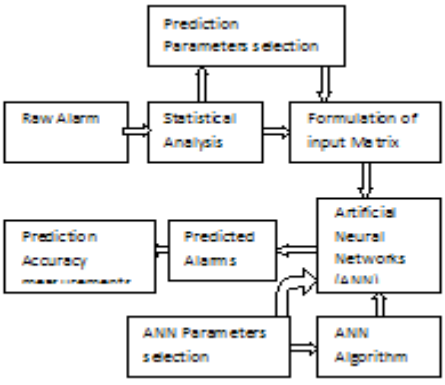


Figure 2: The Proposed ANN System for Fault Prediction

The raw alarm data is fed to a Statistical Analysis module to help select the prediction parameters. The statistical data is used to formulate the input matrix for the ANN module. The ANN module uses a particular algorithm and selected ANN parameters to output the predicted alarms. Finally prediction accuracy measurements are done by comparing the predicted alarms with the actually happened alarms. A number of experiments are run to find the optimal prediction and ANN parameters.

D Prediction Parameters

There a number of prediction parameters that can be varied to find an optimal mix.

Alarm Period is the total time duration for which we have the historical alarm data e.g. 6 months.
Training Period is the portion of the Alarm Period used for training the ANN.
Prediction Interval (PI) is the portion of the period of time that will be used for fault predictions.

Time Window (TW) is an important parameter that requires a bit of explanation. The overall training period is equally divided into time slots called time windows. First of all, statistical analysis is done for the first Time Window (TW) and that is the first sample of statistical data. Next the TW is shifted by an amount called Time shift (TS) and a second sample of statistical data is collected. Similarly the time window is successively shifted and statistical data is collected until the end of the training period is reached. The characteristics of these collected samples are used to formulate the input matrix.

The Time window and Time shift are 2 important parameters and have significant impact on the prediction accuracy. A large Time window will has more data in one sample but less number of samples. A wide window shift will has less overlapping data and less number of samples. A narrow window shift will has more overlapping data and more number of samples. The optimal window and window shift can only be obtained by experiments.

Fig. 3 shows the time windows. Also, it is to be noted that:

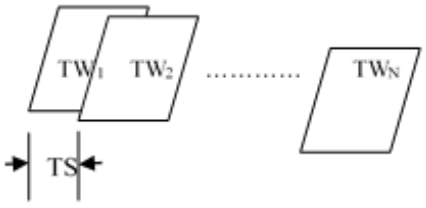


Figure 3: Windowing Technique for Correlation

$$time\ shift(seconds) \leq window(seconds)$$

In fig. 3 TW is time window and TS is time shift.Now for each window taken, the first task is to find all the 441 episodes. We have to calculate three values for each episode:

1) Frequency of an Episode: is the frequency of occurrence of an episode
2) Probability of an Episode: is the probability (relative frequency) that an episode will occur
3) Time bound (delta) of an Episode: is the maximum time period between 2 alarms of an episode.

In each window, frequency of episodes is a 21x21 matrix that stores the frequencies of each of the 441 episodes. Similarly the Probabilities and Time Bounds are also 21x21 matrices in each window. Each row of the 441 matrix represents a predicate which appears first in an episode and each column represents a predicate which follows the first one in the episode. Hence these predicate matrices store episode characteristics for all 441 episodes.

E ANN Input and outputs

Let us suppose that for a particular Window Size, there are total m windows in the Training Period. The 441 values for the episodes (frequencies, probabilities and delta) of the first window is used as an input to the neural network (window 2 in fig. 3). The target of the ANN for this input (three 21x21 matrices) will be three 21x21 matrices in the second window. This means that we are predicting future window (see window 3 in fig. 3) state from the current window state. The first window is only used once as an input. As there are m windows, the number of training examples we have are m-1.

F Evaluation of results

The input to the proposed ANN is three 21x21 binary matrices for frequency, probability and delta of episodes of the current window. The output of ANN is the predicted three 21x21 matrices for frequency, probability and delta of episodes of the next window. Our next task is to obtain a list of predicted alarms for the next window. Let us show an example to understand how this prediction is done.

Suppose Window Length = 2700seconds (45min) and Time Shift=900seconds (15min)

Note that the length of the two windows on both sides of the overlap area is equal. Because of the overlap area, we already know the initial alarms of the next window. These alarms (in overlap area) are used to predict the remaining alarms in the next window. Hence the remaining length of the next window will be: Window length - overlap area= time shift. So our prediction interval (during which we are doing the prediction) is equal to the time shift.

For example, if an alarm with predicate 18 occurs in overlap area, then we look at the 18th row of each of the three predicted matrices (21x21). If predicted frequency, probability and delta matrices all show a value greater than 0.5 for ith column of 18th row, then i is added in the list of predicted alarms. This procedure is done for all the alarms in the overlap area and a list of predicted alarms is obtained. Some of these predicted alarms actually occur(true positives), and the rest are wrongly predicted (false positives). Also some alarms that actually occur are not predicted at all (false negatives).

Next task is to define a measure for calculating the accuracy of this prediction. We calculate the accuracy in terms of two factors:

1) CPA(Percentage of correctly predicted alarms with respect to actual alarms)

For an input window, we find the CPA from actual and predicted alarms vectors as follows:

$$CPA = \frac{length(actual\ alarms\ \cap predicted\ alarms)}{length(actual\ Alarm)}\ x\ 100 \quad (4)$$

This factor tells us what percentage of actual alarms is correctly predicted and is the so called true positives in literature. If this process is done for all the windows we have in our data, and then a mean is calculated then it is called mean CPA.

$$\text{Mean CPA} = (\sum_{i=1}^{p} [\![CPA]\!]_i)/p \quad (5)$$

Where p is the total number of windows.

2) CPP(Percentage of correctly predicted alarms with respect to predicted alarms)

For an input window, we find CPP from actual and predicted alarm vectors as follows:

$$CPP = (length(actual\ alarms \cap predicted\ alarms))/(length(predicted\ Alarm))\ x\ 100 \quad (6)$$

This factor tells us what percentage of predicted alarms actually occurred. If this is low, then it may be that we end up predicting a lot of wrong alarms even if our CPA is quite high. Mean CPP is given as:

$$\text{Mean CPP} = (\sum_{i=1}^{p} [\![CPP]\!]_i)/p \quad (7)$$

Where p is the total number of windows. The optimal parameters will maximize both CPA and CPP.

4 Results

A number of experiments had been done on a six months of alarm data from a telecommunication company. The purpose of these experiments is to find optimal ANN and prediction parameters that provide the best Mean CPP and CPA values. These experiments are specific to the alarm data at hand and will have to be done for any other data.

A Optimal ANN parameters

There are a number of parameters available for the ANN. The following parameters are not dependent on the dynamic data and can be decided based on static knowledge about the data and prediction goals.

Number of inputs are 441 (matrix of 21x21 predicates).
Numbers of outputs are 441 (matrix of 21x21 predicates)
Number of neurons in first layer is equal to the number of inputs in the first layer and is equal to the number of outputs in the last layer.
Transfer functions are TANSIG (-1 to 1) and LOGSIG (0 to 1)
Training algorithm are Back propagation Gradient Descent (TRAINGDX) and Back propagation Conjugate Gradient Training algorithms (TRAINCGB).We have used these two training algorithms and also compared them for better results.
Data sets for training, validation and test data are data sets needed to control ANN. The training data set is used to update weights and biases, and to calculate error. The validation dataset is used to stop the training before the net starts over fitting. The test data just tells us at run time about how the network will perform in case of unseen data. We have used 20% training data as validation data, 10% as test data and 70% as training data
Maximum number of epochs is the maximum number of times all the inputs are iterated through the network. If the number of epochs reaches this number during training, the training stops. We have used a value of 1000 for maximum number of epochs.

The parameters, Learning Rate and (LR) and Number of Hidden Layer Neurons (NHLN), are dependent on the data and are decided through experiments using a window size of 2 days and a time shift of 1 day. In these experiments we calculate and analyze Mean Square Error (MSE) of prediction, the convergence rate, CPA and CPP values.

Table 3: Effect Of Learning Rate On Mean Square Error (MSE)

Learning Rates	Prob. net MSE	Freq. net MSE	Delta net MSE	Average MSE
0.1	0.0374	0.0348	0.0286	0.0336
0.3	0.0258	0.0336	0.0289	0.0294
0.5	0.0327	0.0311	0.0336	0.0325
0.7	0.0341	0.0456	0.0453	0.0417
0.9	0.0259	0.0297	0.0262	0.0273

It can be seen from table II that the best learning rate (minimum average of all three nets MSE) is 0.9. Although a learning rate of 0.9 will be very unstable (does not converge). Therefore we recommend using a learnig rate of 0.3 because it is close to the performance at 0.9 learning rate and convergence is better.

Table 4: Learning Rate And Prediction Accuracy

Learning Rate Analysis TRAINGDX on test data				
Learning Rate.	**CPA**	**WPA**	**CPP**	**WPP**
0.1	83.70	16.30	77.91	22.09
0.3	84.43	15.57	78.24	21.76
0.5	84.34	15.66	78.16	21.84
0.7	84.62	15.38	77.75	22.25
0.9	83.32	16.68	78.46	21.54

Table IV shows the accuracy parameters for different Learning Rates. Here CPA and CPP are as defined earlier. WPA is the percentage of actually occurring alarms that are not predicted by ANN and is equal to 100 – CPA. WPP is percentage of predicted alarms that actually never occurred and is equal to 100 - CPP.

In considering all these factors (MSE, convergence speed and prediction accuracy parameters) we conclude that a learning rate of 0.3 is a good compromising value.

Table 5: Number of Nhln on Mse

For GDX (Effect of increasing Hidden neurons)				
Hidden Neurons	**Prob. net MSE (least)**	**Freq. net MSE (least)**	**Delta net MSE**	**Mean MSE**
220 (inp/2)	0.0405	0.0361	0.0353	0.0373
441 (inp)	0.0327	0.0311	0.0336	0.0325
882 (inp*2)	0.0225	0.018	0.0208	0.0204
1764 (inp*4)	0.0193	0.0164	0.0201	0.0186

Table 6: Number of Nhln on Mse

For CGB (Effect of increasing Hidden neurons)				
Hidden Neurons	**Prob. net MSE (least)**	**Freq. net MSE (least)**	**Delta net MSE**	**Mean MSE**
220 (inp/2)	0.027	0.0321	0.0358	0.0316
441 (inp)	0.0286	0.0274	0.0245	0.0268
882 (inp*2)	0.0403	0.0193	0.0272	0.0289
1764 (inp*4)	0.0488	0.031	0.0222	0.0340

It can be seen from tables 5 and 6 that 441 values for number of NHLN is a good number. NHLN of 441 provides good convergence as the higher values are an over kill.

It can be seen from table 7 that 441 provides good enough prediction accuracy parameters. Furthermore a NHLN of 441 has a low mean training time. Therefore after due consideration we conclude that 441 is a good compromising number for NHLNs.

Table 7: Number of Nhln and Prediction Accuracy

Hidden Layer Neurons Analysis TRAINGDX on test data				
Hid. Neurons	CPA	WPA	CPP	WPP
220	82.87	17.13	78.86	21.14
441	84.34	15.66	78.16	21.84
882	84.27	15.73	77.77	22.23
1764	83.45	16.55	78.31	21.69
Hidden Layer Neurons Analysis TRAINCGB on test data				
Hid. Neurons	CPA	WPA	CPP	WPP
220	83.93	16.07	78.63	21.37
441	83.92	16.08	78.02	21.98
882	83.91	16.09	78.22	21.78
1764	83.75	16.25	78.12	21.88

B Optimal Prediction Parameters

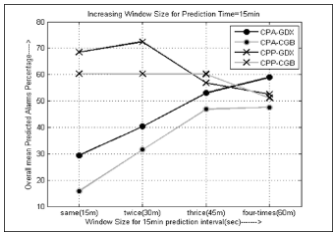
The proposed fault prediction algorithm depends on the Time Window (TW) and Time shift (TS) parameters (see Section III). Therefore search for optimal TW and TS parameters is an important contribution of this paper.

In our experiments we had used five different values for TS: 15 minutes, 1 hour, 5 hours, 1 day and 5 days. The TW is varied as equal to: TS, twice that of TS, thrice that of TS and four times that of TS. Total number of experiments is 20 and 20 neural net sets were trained (20 nets each for frequency, probability and delta) for analyzing the effect of increasing TW for each TS and the effect of increasing TS for each TW.

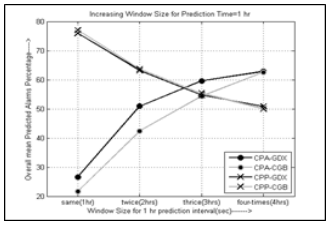
1) Optimal Time Window (TW) Size Analysis

The proposed fault prediction algorithm depends on the Time Window (TW) and Time shift (TS) parameters (see Section III). Therefore, search for optimal TW and TS parameters is an important contribution of this paper.

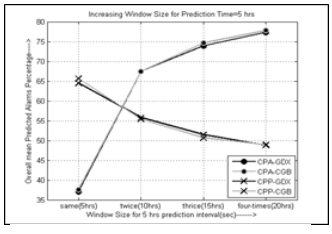
For the following experiments we have analyzed the values of CPA and CPP for the 2 training algorithms: GDX and CGB. These values of CPA and CPP are calculated for the training, validation and test data. First four months of training data is used and then 2 months of actual predictions are done.



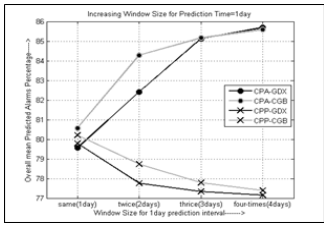
e. PI =15 Minutes



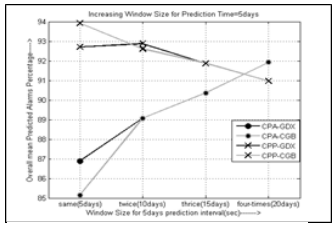
d. PI = 1 Hour



c. PI = 5 Hours

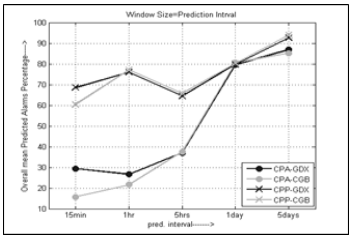


b. PI = 1 Day

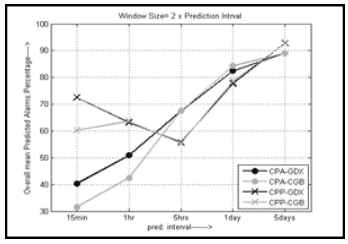


a. PI = 5 Days

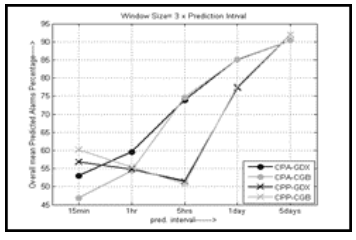
Figure 4: TS and TW Analysis for different PI



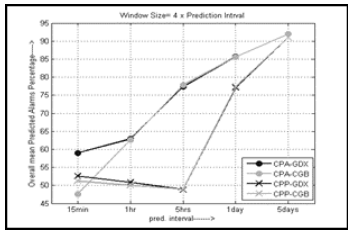
d. TW = TS



c. TW = 2*TS



b. TW = 3*TS



a. TW = 4*TS

Figure 5: TW and TS Analysis

Fig. 4 shows graphically the values of CPA and CPP for the 2 months of predictions on unknown test data as window size is increased. In fig.4a, fig.4b and fig.4e, the best accuracy is for a TW that is 3 times the PI. In fig.4c and fig.4d, the best accuracy is for TW of 4 times the PI. Note that fig. 4 shows the mean of the training, validation and test result values (CPA, CPP) for the two algorithms. Fig. 4 thus gives us an overall picture for choosing the best TW value for a particular TS value.

Fig. 5 shows graphically the values of CPA and CPP for the 2 months of predictions on unknown test data as window size is increased. Fig.5 show that for TW = TS, TW = 2*TS, TW = 3*TS and TW = 4*TS, the accuracy keeps getting better for increasing PI. It can be seen that a window size equal to PI is a good compromise. Based on our experiments we believe that our approach is useful for use in a self-healing network. Here is our proposed procedure for proactive fault prediction so that the faults can be predicted and corrected in self-healing network

Training Phase

1. Statistically analyze the raw historical alarm data to identify contributing Predicates
2. Use raw historical alarm data to formulate the input predicate matrix
3. Train the ANNs using the input matrix and the raw historical alarm data
4. Run experiments and select optimal ANN (learning rate and number of NHLN) and prediction parameters (TW and TS)
5. Update the input matrix for the prediction phase.

Prediction Phase

6. Use optimal parameters, resultant input matrix and ANN to predict future (within the prediction period) faults
7. Initiate corrective measures for the predicted future faults.

Please note that steps of the training phase are done one time only. Therefore the time spent in the training phase is not important. The prediction phase is rather fast and can predict faults in seconds.

Proposed Parameters selection for Real Time Fault prediction

Following are the recommended ANN and prediction parameters based on our experiments and analysis of the results.

1. Learning Rate: 0.5
2. Hidden Layer Neurons: Equal to number of inputs (441)
3. Prediction Interval (PI): Availability of 6 month alarm data.
4. Time Window (TW): As per need of the self-healing network operational requirements
5. Time shift (TS): 2*TW

5 Conclusion

A fault prediction technique had been presented that is useful in a self-healing network. The proposed fault prediction technique used historical alarm data and trained an Artificial Neural Network (ANN) for future alarm predictions. A thorough analysis is done to find optimal ANN and prediction parameters. The prediction results indicated that the proposed fault prediction technique will be useful in prediction and correction of faults before these faults occur.

For future work, other artificial intelligence techniques will be applied and compared using the proposed technique. Furthermore the proposed fault prediction technique will be tested on an actual Telecommunication network to develop a practical application for self-healing of the network.

References

[1] Giljong Yoo, Jeongmin Park and Eunseok Lee,"Hybrid Prediction Model for improving Reliability in Self-Healing System", Proceedings of the Fourth International Conference on Software Engineering Research, Management and Applications 2006 (SERA'06)

[2] Raquel Barco, Pedro Lázaro, and Pablo Muñoz, "A Unified Framework for Self-Healing in Wireless Networks", IEEE Communications Magazine • December 2012 p 134-142

[3] Zhengguo Xu, Yindong Ji, and Donghua Zhou, "A New Real-Time Reliability Prediction Method for Dynamic Systems Based on On-Line Fault Prediction", IEEE TRANSACTIONS ON RELIABILITY, VOL. 58, NO. 3, SEPTEMBER 2009 p 523-538

[4] Arsalan Saeed, Osianoh Glenn Aliu, Muhammad Ali Imran, "Controlling Self Healing Cellular Networks using Fuzzy Logic", IEEE Wireless Communications and Networking Conference: Mobile and Wireless Networks 2012

[5] Peter Szilagyi and Szabolcs Novaczki, "An Automatic Detection and Diagnosis Framework for Mobile Communication Systems", IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 9, NO. 2, JUNE 2012 p 184-197

[6] Marco Canini, Dejan Novakovic, Vojin Jovanovic, and Dejan Kosti, "Fault Prediction in Distributed Systems Gone Wild", LADIS 10 Zürich, Switzerland 2010

[7] K. Hatonen, M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen "Knowledge Discovery from Telecommunication. Network Alarm Databases".In 12th International Conference Data Engineering (ICDE 96'), pp 115-122, New Orleans Louisiana. Feb. 1996.

[8] P. Fröhlich, W. Nejdl, K. Jobmann, H. Wietgrefe "Model-based alarm correlation in cellular phone networks". InProc. of MASCOTS 1997 (1997)

[9] C. Lo, S. Chen, B. Lin. "Coding based schemes for fault identification in communication networks". In International Journal of Network Management; 10:157-164. 2000.

[10] Z.Q. Liu and F. Yan, "Fuzzy Neural Network in Case-Based Diagnostic System", IEEE Transactions on Fuzzy Systems, Vol. 5, No. 2, 1997.

[11] M. Klemettinen, H. Mannila, and H. Toivonen. "Rule discovery in telecommunication

alarm data".In Journal of Network and Systems Management, 1999.Vol 7 No.4: 395-423 (1999).

[12] M. Khalid, "Statistical techniques for fault prediction in telecom Networks", MS Thesis, University of Computer and Emerging Sciences, Pakistan, 2009

[13] O.P. Kogeda and J.I. Agbinya, "Prediction of Faults in Cellular Networks Using Bayesian Network Model". In the Proceedings of First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless 2006), Sydney, Australia, March 13'16, 2006.

[14] Mamdani, E.H. and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," International Journal of Man-Machine Studies, Vol. 7, No. 1, pp. 1-13, 1975.

Heart Diseases Prediction using Data Mining and its Techniques- A Survey

Muhammad Fahad [1] Sadiq Ur Rehman[2] Aqeel-ur-Rehman[3] Muhammad Kashif Alam[4]

Abstract

A process or way toward analyzing patterns of data as indicated by different points of view for classification into meaningful data, which is gathered and amassed in likely manner, e.g. data-warehouse for effective analysis, data mining algorithms, enabling business decisions to cut expenses and increase income. Areas including business, retail system, medical, sciences and engineering are indicating the worth of data mining. In this paper, ways to predict heart diseases using different algorithms/techniques are presented. To explore areas of data mining in health care is the key objective of this research. Medical industry is capable to produce data of different types i.e. non-real time or real-time and the amount of such data is increasing day by day. Due to the daily increment of medical data, medical industry is capable to provide huge contribution in the area of data mining which in result gives prediction of diseases and improves quality of services to the patients. This paper shows the combination and analysis of neural network and data mining, Fuzzy and genetic algorithm, data mining and machine learning.

Keywords: Data mining, Data analysis, Naïve Bayes, Heart disease, Data mining algorithms, Neural Networks, Decision Tree, Fuzzy- Logic, Machine Learning, Data mining.

1 Introduction

Data mining is one of the processes in which non-trivial data is being extracted. Data mining is a technique through which data can be gathered for further processing which is called knowledge/information. This technique has a key role in diseases prediction. There are multiple diseases which we can predict through data mining techniques, cancer, heart diseases, etc. are the major diseases which we can predict through data mining. In health industry data mining is very important. Since health industry has a lot of complex data for processing, this data can be in the form of hospital records, digital devices data, survey data of medical students, electronic gadget data of medical devices etc. The category of complex data used in health industry can be either real-time or non-real time. However, in both the cases error free data is of high importance as this data can be used in correct diagnosis and efficient treatment. Bad diagnosis is not acceptable in health care industry as it may result in death or big health hazard. Some of the main focused technologies of data mining are databank technology, machine learning and statistical analysis [1].

Heart is one of the mains of body [2]. Proper working or human body is mainly dependent on proper working of human heart. If due to any reason, working of heart gets disturbed then

[1] *Hamdard Institute of Engineering and Technology, Karachi | mfahad@hamdard.edu.pk*
[2] *Hamdard Institute of Engineering and Technology, Karachi | sadiqsr@gmail.com*
[3] *Hamdard Institute of Engineering and Technology, Karachi | aqeel.rehman@hamdard.edu.pk*
[4] *Hamdard Institute of Engineering and Technology, Karachi | kashif.alam@hamdard.edu.pk*

certain diseases may arise in human body.

Cardiovascular disease also known as heart disease is now a days most common disease in human body [3]. According to the statistics of 2012 world health report [4], there is a raised blood pressure complain in every one in three patients. Moreover, if we consider WHO (World Health Organization), around 17 million deaths were reported due to heart attack. There are various reasons which increase the probability of heart diseases, some of which are;

- Work load
- Mental stress
- Hypertension
- Physical inactivity
- Obesity
- Uncontrolled cholesterol
- Smoking
- Poor diet
- Blood pressure issue
- Genetic susceptibility to heart diseases

This paper is divided into five sections, Section I is related to the introduction of the topic. Details regarding heart diseases algorithms can be found in section II. Discussion about prediction can be found in Section III. Open source tools that are used in data mining are discussed in Section IV. Section V presents conclusion and future work.

2 Data Mining Techniques and Algorithms in Health Care

There are multiple techniques available to identify diseases of heart with the help of data mining. It includes classification, clustering, association, prediction etc. Classification is the machine learning technique and is responsible to categorize individual items into predefined groups. Statistics, decision trees, neural networks etc. Clustering is very beneficial to the group of substance having similar properties/features with the help of mechanical technique. Association is a technique which is considered to be the best data mining technique for the predication of heart diseases so far. In this technique, all the non-similar attributes that have been used for analyzing the heart disease are incorporated and patients with complete risk factors (important for prediction of disease) are sorted out. Prediction is a technique used in data mining to find the correlation between independent and dependent variables. Data mining algorithms and techniques are classified into the following sub categories,

A. Neural Network
B. Naïve Bayes
C. Decision Tree
D. Genetic Algorithm

Each category has different accuracy rates among which Neural Networks were found to be the most accurate classification technique having accuracy rate of 100%.

Intelligent Heart Disease Prediction System (IHDPS) which uses above mentioned data mining techniques can be seen in [5].

A Neural network

An artificial neural network (ANN) generally called as neural network (NN) is a model of mathematics based on biological NN. ANN works same like human brain. Neural network is composed of many parallel working nodes which are joint together with one-directional signal connections. Supervised learning and unsupervised learning are the two main categories of neural network

B Naïve Bayesian

Based on Bayes theorem, Naïve Bayes [6] is an algorithm that is used for classification. The key concept behind Bayes theorem is probability. In this theorem, probability of an on-going event is calculated given the probability of already occurred event.

C Decision Tree

Decision tree, as from its name, is a structure of large data set that gets divided into successive small data sets with the implementation of a sequence of decision rules. As increment takes place in successive division. The outcome of result will gets closer to other members of the set. There are various models of decision tree. Gain ratio decision tree is the most successful type of decision tree [7].

D Genetic Algorithm

In genetic algorithm, process of natural selection takes place by search method. This algorithm is responsible to provide optimization and solution to search problem by using advance techniques which includes mutation, crossover, inheritance and selection.

It would be a very powerful mechanism for efficient classification if genetic algorithm and fuzzy logic gets combined. Genetic algorithm helps in effectiveness while fuzzy logic helps to develop knowledge based system in health disease

Table 1: Data Mining Techniques with their accuracy rate. [18,20,30]

Reference	Classification Techniques	Accuracy	Recommended Data mining technique
Ahmed F et al. (2015)	Bayes Net	84.5%	SVM
	SVM(Support Vector Machines)	85.1%	
	FT(Functional Trees)	84.5%	
Salha M et al. (2014)	Neural Networks	91%	Decision Tree
	Decision Tree	99%	
	Naïve Bayes	96.5%	
Sivagowry et al. (2014)	Neural Networks	98%	Neural Networks
	Decision Tree	52%	
	Naïve Bayes	52.33%	
Rashe-Dur et al. (2013)	Decision Tree	75.5%	Fuzzy Logic
	Neural Network	79.19%	
	Fuzzy Logic	83.85%	
Indira S. FalDessai (2013)	BNN	80.4%	PNN
	PNN	94.6%	
	NB	84%	
	DT	84.2%	
Apte et al. (2012)	Naive Bayes	90.74%	Neural Networks
	Decision Trees	99.62%	
	Neural Networks	100%	
Jyoti K et al. (2012)	Neural Networks	100%	Neural Networks
	Decision Tree	99.62%	
	Naïve Bayes	90.74%	
Nidhi et al. (2012)	Naive Bayes	90.74%	Neural Networks
	Decision Trees	52.33%	
	Neural Networks	96.5%	
Chaitrali et al. (2012)	Neural networks	100%	Neural Networks
	Naive Bayes	90.74%	
	Decision Tree	99.62%	

Resul et al. (2009)	Neural networks	89.01%	Neural Networks
Resul et al (2009)	Neural networks	97.4%	Neural Networks
M. Anbarasi et al. (1999)	Classification through clustering Naive Bayes	88.3%	Decision Tree
	Decision Tree	99.2%	
	Naive Bayes	96.5%	
Matjaz et al. (1999)	Neural Networks	85%	Neural Networks
	ECG(Neural Networks)	74%	

Above mentioned table shows the comparison of different data mining techniques in which accuracy is measured. Different authors used different data sets for comparison with different techniques. After implementing techniques and algorithms the next phase is prediction that is discussed in the next section.

3 Prediction of Heart Diseases Using Data Mining

Many papers have been written related to heart diseases and data mining techniques for prediction of heart diseases. Different techniques like classification, dataset, algorithms are used to observe and show result which are efficient methods.

P.K Anooj [14], [17] has proposed CDSS for prediction and diagnosis of heart diseases based on fuzzy rules. The proposed system has two parts one was computerized and the other one was generalized. The process is simple it takes patient data automatically, and implements both phases. Proposed system is better than other systems. Result is better when applied fuzzy rules. Latha Parhiban [13], [14] also formulated the approach using co active nuero-fuzzy inference system (CANFIS).

Subbalakshmi, Ramesh and Chinarao [14], [15] proposed system which takes age, sex, blood pressure, cholesterol etc and other attributes as an input and then shows the result and predicts heart diseases. This model was good and predicts even complex queries with effective results. The system provides decision support in heart diseases using naïve Bayes data mining techniques.

Some of the solutions for heart diseases prediction are presented above which are based on Neural networks, Fuzzy logic, Decision Tree etc. Few open source tools that are available for data mining are discussed in detail in the next Section.

4 Open Source Tools for Data Mining Used In Health Care Applications

There are many available open source tools that are frequently being used in data mining especially in the case of heath care applications. Some of the most common open source tools are as follows,

A. Tanagra

It is also an open source software that is used for the purpose of academics and research. It covers that data mining concepts from several dimensions which include machine learning, exploratory data analysis, meta supervised learning, feature selection, database area etc. Tanagra is the graphical user interface based data mining tool [33]. Tanagra can be used to analyze both types of data (i.e. synthetic or real).

B. WEKA Tool

Weka [34] is a data mining software that has been developed in java language by University of Waikato (New Zealand). It is a set machine learning algorithm that is used for data mining. The beauty of algorithms used in Weka is that they are independent (can directly be practiced on data set or personal java code). Weka consists of specialized tools that are used for pre-processing of data, visualization, clustering, regression etc. Since Weka is an open source software, it facilitates developers to create new machine language techniques and also applications that are required to solve the data mining problems. The biggest plus point of Weka is the capability to be applied on big data. File format used in Weka is ARFF.

C. MATLAB

MATLAB [35] is highly recommended for the fast computation, visualization and for coding. Matlab is a GUI based software in which we can perform several complex tasks in efficient way. Matlab helps us in performing analysis of data, creating/ modifying complex algorithms, developing applications etc. The computation time of Matlab is faster than the computation time in C/C++ and other programming languages.

D. ORANGE

Orange [36], an open source machine learning and data mining suit used for the data analysis. It is a data mining tool in python. The use of Orange is very much simple, it can be used by professional programmers and also by beginners/ students who are working in the field of data mining. Library in this software is classified in hierarchical structure for the components of data mining.

E. Rapid Miner

Rapid Miner [37], is the top most open source software that is used for data mining. Rapid miner is most powerful and useful software for data mining. By using this software, data mining

as well as analysis can be done with integration of two products or a single product. It is used for data analyzing which includes environment for analytics, mining, deep learning, machine learning etc. Rapid miner shows result through visualization, optimization and validation. It uses client server model. It is very helpful and useful in data mining which optimizes, validates and visualizes results. There are various types of graphs which rapid miner shows after analyzing and mining which includes Pie Charts, Contour, 3-D, Density, Histogram, Survey Plots, and Quartiles etc.

Table 2: Open Source Tools Technical Overview [39,40,41,42,43]

S.No	Name	Version	OS Support	Reference
1	Tanagara	1.4.50	Windows	https://eric.univ-lyon2.fr/~ricco/tanagra/index.html
2	Weka	3.9	Cross Platform	https://www.cs.waikato.ac.nz/ml/weka/
3	MATLAB	9.0	Cross Platform	https://www.mathworks.com/
4	Orange	3.7	Cross Platform	www.orange.biolab.si
5	Rapid Miner	7.6	Cross Platform	www.rapidminer.com

Table 3: Open Source Tools Functions and Characteristics [38]

S.No	Name	Functions	Properties/Charactistics
1	Weka	Machine Learning	Visualization Classification Regression
2	MATLAB	Analysis, Computation and Visualization	Data Analysis Statics Integration
3	Orange	Machine Learning and Data mining/visualization	Interaction Visualization
4	Rapid Miner	Analysis and Data Mining	Shows real time macro values
5	Tanagara	Statistical Learning, Data Analysis and Machine Learning	Drag and drop Different Controls for functions.

5 Conclusion and Future Work

In this paper, each data mining technique had been shown with the result separately. Research also showed that when different techniques are used to predict heart diseases, there are some differences (in accuracy) in it. Same data and classifications show different result in different data mining techniques as discussed in table 1. The main purpose of the survey was to study and analyze different data mining techniques that are used to predict heart diseases. Result shows that Neural Network method provides more accuracy in different scenarios.

Text mining can also be performed on different data sets. Since, there is huge amount of data which is unstructured and we can utilize this data and apply above techniques an it.

References

[1] Singh, Kuldip, and Gurpreet Singh. "Alterations in Some Oxidative Stress Markers in Diabetic Nephropathy." Journal of Cardiovascular Disease Research 8, no. 1 (2017).

[2] Nagre, SurajWasudeo. "Mobile Left Atrial Mass–Clot or Left Atrial Myxoma." Journal of Cardiovascular Disease Research 8, no. 1 (2017).

[3] Bhatla, Nidhi, and KiranJyoti. "An analysis of heart disease prediction using different data mining techniques." International Journal of Engineering 1, no. 8 (2012): 1-4.

[4] Berenson, Gerald S., Sathanur R. Srinivasan, WeihangBao, William P. Newman, Richard E. Tracy, and Wendy A. Wattigney. "Association between multiple cardiovascular risk factors and atherosclerosis in children and young adults." New England journal of medicine 338, no. 23 (1998): 1650-1656.

[5] Palaniappan, Sellappan, and RafiahAwang. "Intelligent heart disease prediction system using data mining techniques." In Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on, pp. 108-115. IEEE, 2008.

[6] Ratnam, D., P. Himabindu, V. MallikSai, SP Rama Devi, and P. RaghavendraRao. "Computer-Based Clinical Decision Support System for Prediction of Heart Diseases Using Naïve Bayes Algorithm." Int. J. Comput. Sci. Inf. Technol. 5, no. 2 (2014): 2384-2388.

[7] Quilan, J. R. "Decision trees and multi-valued attributes." In Machine intelligence 11, pp. 305-318. Oxford University Press, Inc., 1988.

[8] JyotiKiran and BhatlaNidhi "An Analysis of Heart Disease Prediction using Different Data Mining Techniques" International Journal of Engineering Research & Technology (IJERT). ISSN 27278-0181, Vol. 1 Issue 8, October 2012.

[9] Dangare S. Chaitrali and S. ApteSulabha "Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques" International Journal of Computer Applications (0975-888) Vloum 47-No. 10 June 2012.

[10] M. AlzahaniSalha, AlthopityAfnan, AlghamdiAshwag, AlshehriBoushra and AljuaidSuheer "An overview of DataMining Techniques Applied for Heart Diseases Diagnosis and Prediction" lecture Notes on Information Theory Vol. 2 No 4, December 2014.

[11] NikharSonam and Karandikar A.M "Prediction of heart diseases using Data Mining

Techniques – A Review" International Research Journal of Engineering and Technology (IRJET) Volume 3 Issue : 2 Feb 2016.

[12] Singh Williamjeet and KaurBeant "Review on Heart Disease Prediction system using data Mining Techniques" International Journal on Recent and Innovation Trends in Computing and Communication Volume : 02 issue : 10 October 2014 ISSN : 2321-8169.

[13] OyedotunKayode and Olaniyi Ebenezer "I.J. Intelligent Systems and Applications" Published online November 2015 MECS.

[14] Dubey Rishi and ChandarakarSantosh "Review on Hybrid Data Mining Techniques for the Diagnosis of Heart Diseases in Medical Ground" Indian Journal of Applied Research Volume: 5, Issue: 8 August 2015, ISSN 2249-555X.

[15] S. Sivagowry, M Durairaj and A Persia "An Empirical Study on applying Data mining Techniques for the Analysis and Prediction of Heart Disease"

[16] V. Manikantan and S. Latha, "Predicting the analysis of heart disease symptoms using medicinal data mining methods", International Journal of Advanced Computer Theory and Engineering, vol. 2, pp.46-51, 2013. W.-K. Chen, Linear Networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[17] SellappanPalaniappan and RafiahAwang, "Intelligent heart disease prediction system using data mining techniques", International Journal of Computer Science and Network Security, vol.8, no.8, pp. 343-350,2008.

[18] K.Srinivas, Dr.G.Ragavendra and Dr. A. Govardhan, "A Survey on prediction of heart morbidity using data mining techniques", International Journal of Data Mining & Knowledge Management Process (IJDKP) vol.1, no.3, pp.14-34, May 2011.

[19] G.Subbalakshmi, K.Ramesh and N.ChinnaRao, "Decision support in heart disease prediction system using Naïve Bayes", ISSN: 0976-5166, vol. 2, no. 2.pp.170-176, 2011.

[20] K.Srinivas , B.Kavihta Rani, Dr. A.Govrdhan , Applications of Data Mining Techniques in Healthcare and Prediction of Heart Attacks , IJCSE Vol. 02, No. 02, 2010, 250-255.

[21] D. P. Shukla, ShamsherBahadur Patel and Ashish Kumar Sen, "A literature review in health informatics using data mining techniques," International Journal of Software & Hardware Research in Engineering, vol. 2, no. 2, February 2014.

[22] D. P. Shukla, ShamsherBahadur Patel and Ashish Kumar Sen, "A literature review in health informatics using data mining techniques," International Journal of Software & Hardware Research in Engineering, vol. 2, no. 2, February 2014.

[23] Promad Kumar Yadav, K. L. Jaiswal, ShamsherBahadur Patel, D. P. Shukla, "Intelligent heart disease prediction model using classification algorithms," UCSMC, vol. 3, no. 08, pp. 102-107, August 2013.

[24] R.Chitra and V. Seenivasagam, "Review of Heart Disease Prediction System Using Data Mining and Hybrid Intelligent Techniques", ICTACT Journal on Soft Computing, vol.3, pp. 605–609, July 2013.

[25] JyotiSoni, "Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction", International Journal of Computer Applications, vol.17, pp. 43–48, Mar. 2011.

[26] S. Vijayarani and S. Sudha, "An Efficient Clustering Algorithm for Predicting Diseases from Hemogram Blood Test Samples", Indian Journal of Science and Technology, vol.8, pp. 1–8, Aug. 2015.

[27] B.Venkatalakshmi, M.V Shivsankar, "Heart Disease Diagnosis Using Predictive Data Mining", International Journal of Innovative Research in Science, Engineering and Technology, vol.3, pp. 1873–1877, Mar. 2014.

[28] Sultana Marjia, Haider Afrin and ShorifUdding Mohammad "Analysis of Data Mining Techniques for Heart Disease Preditction" iCEEiCT 2016.

[29] Govardhan A., RoaTaghavendra, Srinivas K. "Analysis of Conronary Heart Diseases and Prediction of Heart Attack in Coal Mining Regions Using Data Mining Techniques" The 5th International Conference on Computer Science &Education Hefei, China August 24-27, 2010

[30] TanejaAbhishek "Heart Disease Prediction System Using Data mining Techniques" Oriental journal of Computer Science and Technology ISSN: 0974-6471 Vol 6 No. 4. Dec 2013.

[31] ChandnaDeepali "Diagnosis of Heart Disease Using Data mining Algorithm" International Journal of Computer Science and Information Technologies Vol. 5(2), 2014. (IJCSIT)

[32] Rani Usha K. "Analysis of Heart Disease Dataset using neural Network approach" International journal of Data mining & knowledge manangement process Vol 1, No.5, September 2011.

[33] Krishnaiah, V., G. Narsimha, and N. Subhash Chandra. "Heart Disease Prediction System using Data Mining Techniques and Intelligent Fuzzy Approach: A Review." Heart Disease 136, no. 2 (2016).

[34] Russell, Ingrid, and Zdravko Markov. "An Introduction to the Weka Data Mining System." In Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education, pp. 742-742. ACM, 2017.

[35] Nakamura, Shoichiro. Numerical analysis and graphic visualization with MATLAB. Prentice-Hall, Inc., 1995.

[36] Demšar, Janez, TomažCurk, AlešErjavec, ČrtGorup, TomažHočevar, MitarMilutinovič, Martin Možina et al. "Orange: data mining toolbox in Python." The Journal of Machine Learning Research 14, no. 1 (2013): 2349-2353.

[37] Hofmann, Markus, and Ralf Klinkenberg, eds. RapidMiner: Data mining use cases and business analytics applications. CRC Press, 2013.

[38] Rangra, Kalpana, and K. L. Bansal. "Comparative study of data mining tools." International journal of advanced research in computer science and software engineering 4, no. 6 (2014).

[39] https://eric.univ-lyon2.fr/~ricco/tanagra/index.html

[40] https://www.cs.waikato.ac.nz/ml/weka/

[41] https://www.mathworks.com/

[42] www.orange.biolab.si/

[43] www.rapidminer.com/

IoT: Protocols, challenges, and opportunities with agriculture perspective

Muhammad Azeem Abbas [1] Sharifullah Khan [2] Mushhad Gillani [3] Yaser Hafeez [4]

Abstract

The rustic regions in Pakistan face numerous and similar concerns in the fields of agriculture, connectivity, water, transport and others. The present work provides a roadmap to solve these issues potentially and pave a way for similar results to be focused and headed for explaining these concerns. In this regard, an extensive variety of industrial IoT applications specific to agriculture is evaluated and presented in this work. This study analyses the contemporary enquiries of IoT, significant and supporting technologies, key IoT protocols in agriculture and classifies research tendencies and experiments in Pakistan. A simulation of the state of the art IoT protocols was conducted for their evaluation.

Keywords: IoT, Agriculture, Protocols, Pakistan.

1 Introduction

IoT in its basic form can be explained as a network of physical devices connected to internet without requiring any human interaction for them to function. As per CISCO's predictions, over 50 billion devices will be connected to the internet by 2020. The impact and numbers are such which cannot be ignored; IoT will affect our lives in every aspect hence, making it worthwhile, in agriculture, just as in any other industry. Being the second largest sector of Pakistan's economy, it engages about 42.3% of the labor force.

Our country, Pakistan, has been blessed with abundant natural resources such as arable-land and water making agriculture as one of the most important part of its gross domestic product i.e. 19.8%. Roughly 60% of the country's population resides in rural areas which is directly or indirectly related to agriculture. Punjab, being the most cultivatable of all provinces, yields cotton and wheat in every year. Despite the considerable importance of the industry, it is suffering from an unstoppable decline. From 5.1% growth performance in 1960's, it has seen a steep drop to 3.2% in the 2000's. This can only be attributed to low investments and absence of advancements in conventional technologies or farming techniques to which the concept IoT can play a vital role in agriculture[1].

Not only does meeting the ever-growing local and international demands for food and other edible resources is becoming an up-hill task, the drying up water resources is worsening the situation as well, eventually, putting an immense burden on the agriculture sector. The rapid population increase and urban migration contributes mainly to that.

[1 3 4] *University Institute of Information Technology (UIIT), PMAS-Arid Agriculture University Rawalpindi*
[2] *National University of Sciences and Technology (NUST), Islamabad*
[1] *azeem.abbas@uaar.edu.pk* | [2] *mushhad@uaar.edu.pk* | [3] *yasir@uaar.edu.pk* | [4] *sharifullah.khan@seecs.edu.pk*

In order to tackle the challenges that we face today, we need to understand that there lies a strong connection between agriculture and climate. Factors such as temperature, moisture, precipitation and other aspects affect the agricultural production and eventually, our economy.

2 IoT in Agriculture

In smart farming [2]–[4] variety of things that you may connect like weather forecasting. You have to connect devices like drone [5] with mobile phone that forecast and signal to spray water on those days in which sun is hot and temperature is above the threshold value. Different monitoring and sensory equipment are the key precision in agriculture technology. Smart mapping is very essential where the drones move and give direction to them automatically. Crop monitor by taking the pictures randomly and find the growing rate. Figure 1 describes this process visually how smart farming connects with those features.

A Human Factor

The 21st century world favors automation over labor force. Farmers do the tiresome job of spraying their corps which at times badly affects their respiratory system, hence affecting their health. This can be done by a remotely controlled drone or robot which would perform the spraying task instead of the human. These drones could well be equipped with various agricultural sensors such as NDVI crop sensors to full spectrum and near infrared cameras. Drones can be useful in a way that they cover large areas of land in short time interval with:

- No safety risks
- Attached sensor data and real time imagery data that can't be retrieved quickly on foot or by vehicle
- Low cost and low human efforts

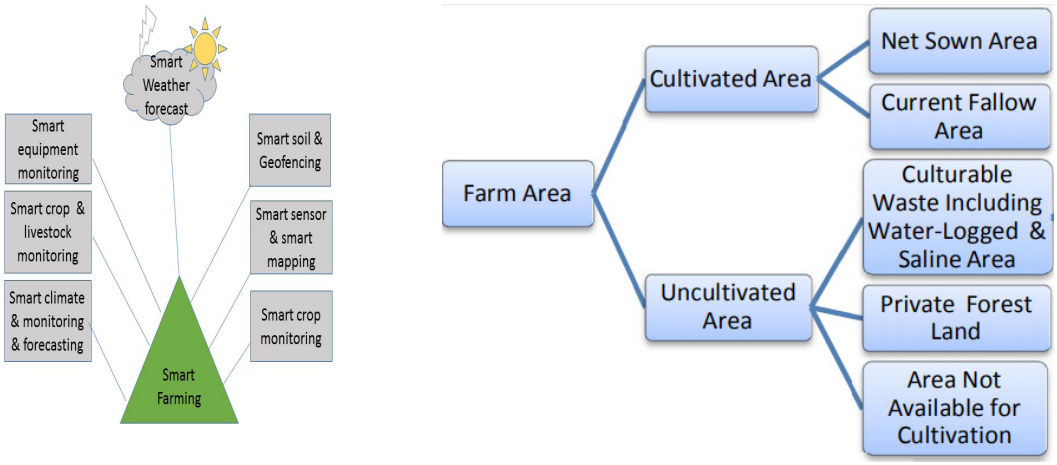
This can be categorized as Precision agriculture which requires real time data and the capability of UAV's to glide over crops and speedily gather its data. The improvements and developments in drone's batteries, guidance systems and control systems have enabled them to be both reasonable and practical. It is also important to note here that the center of those crops which are of considerable height and/or are dense are difficulty assessed on foot or by land vehicle. They get damaged in the assessment process which can be avoided by the usage of light weight drones.

B Intelligent Greenhouses

Greenhouses are used to grow plants in a controlled environment. Arduino based intelligent greenhouses could well maintain the suitable temperature for plants with minimum human interaction. Greenhouse control system hardware refers to the hardware circuit based on single chip design; the part of the hardware mainly includes 5 elements:

- The Microprocessor Control Unit (MCU)
- The sensor of temperature and humidity detection system

- The Arduino UNO controller
- Input and display system
- Network controller. [2]



(a) (b)

Figure 1: IoT in Agriculture (a) Smart farming (b) Farming area classification

C Pesticides and disease detection

Pests and insects not only affect the quality of corps but the quantity as well. Forecasting plant diseases [6] or insecticides attacks have become much more important. Machine learning based on detection and recognition of plant diseases can provide an early insight for us to start its treatment accordingly and in early stages. With digital image analysis being a technology that is firmly-established, can be used in plant disease assessment and detection.

As discussed, RGB (red, green and blue) imaging and/or digital photographic images are vital in evaluating plant health. The simple source of RGB digital images for disease detection, identification and quantification are digital cameras which are easy to handle. The technical attributes that these digital cameras are comprised of such as light sensitivity, spatial resolution or/and digital focus have improved significantly over the course of time [7].

D Animal Intrusion

Animals do intrude into fields and affect the crops yields. The most common techniques used for this are such fences which when applied pressure on, automatically triggers an alarm to the concerned farmer and let him/her know that an intrusion is in process. The digital image analysis including RGB co lor images, as already discussed above, with the red, green and blue channels are also vital for its use in detect biotic stress detection in plants[6].

E. Weather Monitoring and Water Irrigation

A farmer could take care of his/her farm in a better way if he/she is aware of the weather

conditions. Arduino can be used for weather monitoring which consists of 4 main sensors. These sensors detect and monitor the temperature, humidity, light and rain level. These readings are constantly sent to a web server where a farmer can login, inspect the reports and take actions accordingly. The same concept can be used for water irrigation purposes by detecting the soil moisture level and notifying the farmer when to stop the supply of water.

Implementing a change is not always as easy as it looks. In Pakistan's agriculture sector, the numbers of required professionals are low. The agriculture related jobs are not ample for the number of already limited professionals, making the introduction of IoT an issue amongst less educated farmers. Adoption of such devices that are connected is also a considerable issue. Machines that are commonly used in west would be hard to adopt in our scenario for which steps are needed on a broader level, farmers need to be made aware of the importance of these techniques and advancements.

While automaton farming is on the rise, it isn't seeing speedy progress. The main aspect behind it is the constraints of farm robots. Most robots are confined to do a single job and current technology is still away from developing robots that could run unmanned farms. As with all impending products, the costs are quite high due to limited global demand and technical expenditure. Once parts become inexpensive and are developed on a wider scale, the charges will drop and farm robots will see a much wider adoption.
Currently Pakistan has almost 80 million hectors of land out of which half is cultivable, yet we only have a mere 22 million hectors cultivated. With the use of IoT, the uncultivated land of can be brought to use. Considering the fast urbanization that is been in progress, the introduction can increase the current yield production using automation and machine learning.

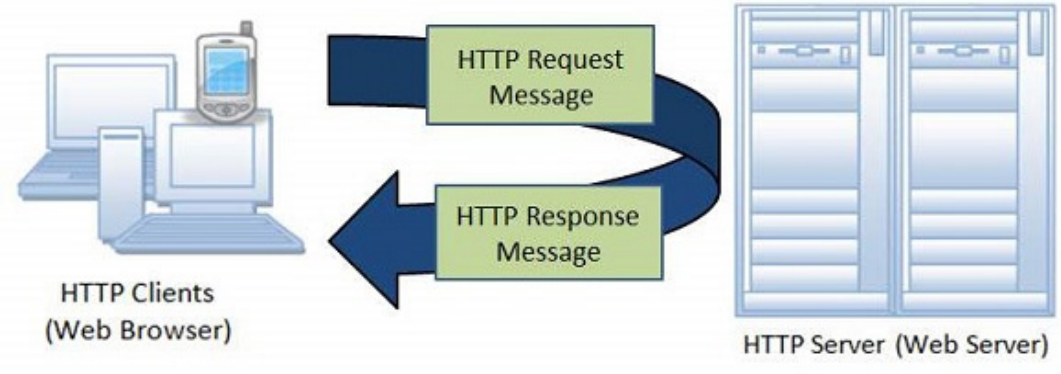


Figure 2: Hypertext transfer protocol (HTTP)

3 IoT Protocols

Common understandable communication is necessary for IoT implementation. Different devices communicate with each other to expedite a given problem. Traditional technologies such as HTTP, HTTP(S) and Web Services can be used for communication between devices. The communication payload can be represented as XML or JavaScript Object Notation (JSON). JSON provides a data representation and development of a stateful web application for persistent connection between clients and the web server.

A Hypertext transfer protocol (HTTP)

The widely used client/server model for web applications is the HTTP [8]. HTTP provides secure method when communication mainly initiated by the clients. Several security methods using private and public keys concept has been well matured in HTTP environment. For IoT, simply adding clients to the network initiates communication and secure transfer of data. Since HTTP works on client/server model, direct communication among clients is not possible. Architecture of HTTP is shown in Figure 2.

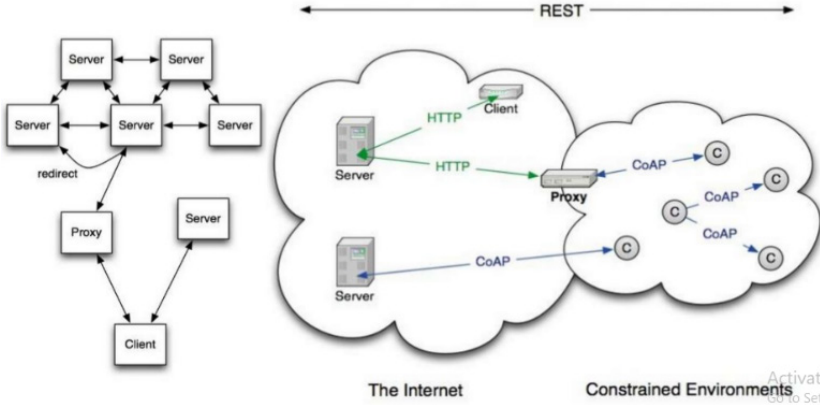


Figure 3: Constrained Application Protocol (CoAP)

B Constrained Application Protocol (CoAP)

Web protocols such as HTTP provide easy and secure solution for IoT devices. However, web protocols are resource intensive for many IoT applications. Internet Engineering Task Force (IETF) [9] has designed a protocol named Constrained Application Protocol (CoAP) for low-power devices working under constrained network. CoAP works over the traditional RESTful protocol. In addition, CoAP support one to one mapping with (to and from) HTTP. CoAP is preferred for devices using batteries as power supply or devices using energy harvesting strategies. The internal architecture of the CoAP works on UDP protocol with additional functions of the TCP as shown in Figure 3. These additional functions include acknowledgement for distinguishing messages. Furthermore, CoAP messages for request and response are communicated asynchronously. To reduce protocol overhead for making CoAP usable with low powered devices, the headers, status and underlying methods are encoded in binary representation. For caching, CoAP depend on response code similar to HTTP. In agriculture scenario, CoAP is a suitable choice as most of the IoT devices are lightweight and require a consistent connection. Implementing CoAP as a web service is a possible way for inducting it into agriculture IoT.

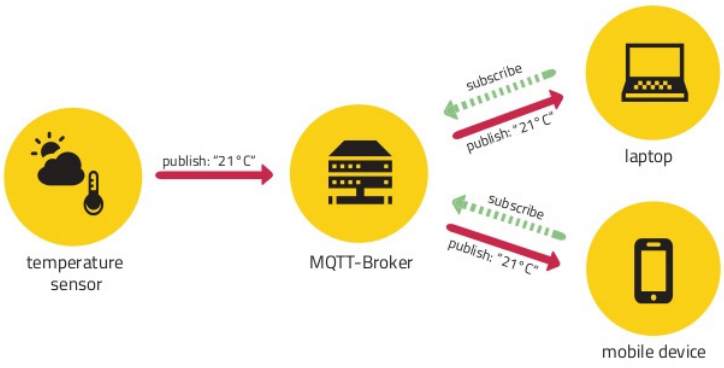


Figure 4: MQ Telemetry Transport (MQTT)

C MQ Telemetry Transport (MQTT)

Researchers at the IBM has invented the MQ Telemetry Transport (MQTT), an open source protocol [10]. Similar to CoAP, MQTT is designed for lightweight messaging among devices using publish/subscribe architecture. Figure 4 shows the architecture of the MQTT. MQTT main objectives are constrained and unreliable network with low-bandwidth communication. MQTT does not focus primarily on the security rather only assurance about successful message delivery between sender and the respondents. As compared to CoAP, MQTT provides a device to device communication, data agnostic and is suitable for mobile devices with low power and network requirements. However, MQTT does not support broadcast communication.

4 Evaluation and Results

Management of farming land, for the farmers and governments can play vital role in the economics and in the environment too. The application of IoT may incorporates decision support systems for entire farm administration with the objective of advancing profits for inputs while safeguarding assets, empowered by the across the board utilization of microcontrollers, GPS, various sensors, and the most importantly the communication infrastructure that supported by emerging protocols.

For these reasons, the performance of above mentioned protocols, i.e. HTTP [8], CoAP [9] and MQTT [10] is also evaluated in this work. First a test bed has been developed to perform performance measurement. A typical agriculture IoT network is realized. It is believed that in such network there will be thousands of sensing nodes spreading over the smart farm. These sensing nodes may have specific tasks or they might be useful for collective sensing.

Table 1: IoT Protocols Comparison

Protocol	Transport	Messaging	2G, 3G, 4G Suitability (10000s nodes)	LLN Suitability (10000s nodes)	Computational Resources	Data Format	Real Time Updates	Review
CoAP	UDP	Request / Response	Excellent	Excellent	10Ks RAM/Flash	Binary	With Observe Specifications	Field Area Networks
MQTT	TCP	Publish / Subscribe Request / Response	Excellent	Fair	10Ks RAM/Flash	Binary	Yes	Provision of enterprise level messaging between IoT applications
HTTP (RESTful)	TCP	Request / Response	Excellent	Fair	10Ks RAM/Flash	Text With Binary Support	With Custom protocol Over Websocket	Energy Management IoT Application

Since this is a preliminary work, it has some limitation, so in this study we have incorporated only few parameters. The sensing nodes that are considered in this simulation environment are assigned to measure temperature and humidity across the farm landscape. For which it is considered that the farm spread over hundreds of square meters. The sensors are installed densely to measure the farming-land more accurately. The various protocols that are compared in Table 1 are studied.

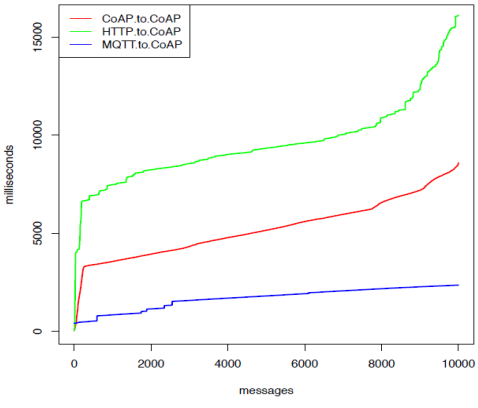


Figure 5: Messaging over the IoT system of multiple protocols

From the Figure 5, it is apparent that the behavior of HTTP-MQTT protocol is subject to higher latency and at second place CoAP is also facing similar problem. At both ends, when MQTT is implemented same gives some better results. Similar observation is found in Figure 6, where multiple clients are supported over the IoT system. However, MQTT based Mosquito is again

at leading position with minimum latency. This provides a comparison of the evaluation of the IoT protocols with respect to agriculture domain. The comparison parameters are transport method, messaging, scalability, adaptability, suitability and computation resources required by a device under a given protocol. Transport methods describe the transport layer implementation of the IoT devices within the network. Messaging parameter deals with methods or models for message passing between clients and server or client to client. Since agriculture is mainly rural area, high speed internet connectivity is not possible or available at many areas; adaptability parameter provides indication about different speed internet network supported by an IoT protocol.

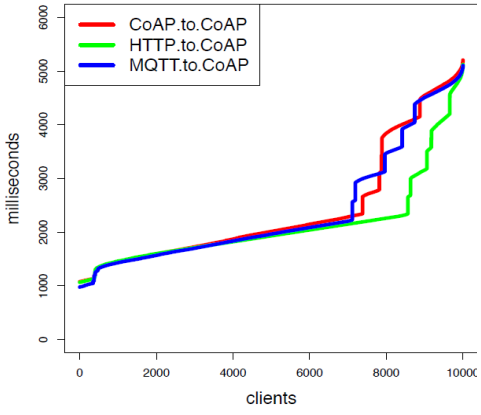


Figure 6: Multiple Clients support over the IoT system

For scalability the evaluation was carried out up to hundreds of nodes/devices. This can be increased to thousands in the future work. Even though ten thousand is sufficiently a big figure for agriculture domain, however the current evaluation was performed keeping in view a local IoT network rather than a global network. The sensors are installed densely to measure the farming-land more accurately. The various protocols that are compared in Table 1 are studied. Multiple Clients support over the IoT system.

Based on results, it can be concluded that MQTT is a viable solution for an environment where real time updates are not required. However, several tasks such as irrigation, spraying and animal intrusions require real time support that can be implemented using CoAP or HTTP. In addition, energy harvesting such as solar and wind powered techniques are required for implementing a real time solution. Latency of different evaluated IoT protocols remains same for number of clients less than 50K. However, it increases exponentially with increasing number of clients. The message communication provides a clear comparison among evaluated protocols. The communication between MQTT and CoAP shows a better performance over the other variants. MQTT and CoAP both use binary data format and support client's individual communication. However, HTTP variants work on client/server model that degrade the communication performance.

5 Conclusion

The application of IoT in the agriculture incorporates decision support systems for the entire farm administration with the objective of advancing profits for farmers while safeguarding assets, empowered by microcontrollers, GPS, various sensors, and the most importantly the communication infrastructure that is supported by emerging IoT protocols. This study develops the intuition smart agriculture systems that can help farmers and the regulating authorities to build and manage the farming activities more effectively. In this work various IoT protocols are discussed and also analyzed thoroughly. The HTTP, CoAP and MQTT are discussed and then compared. While for the development of smart farms a preliminary study work is also carried out in this paper. From the simulation work, it is find that the implementation of MQTT based Mosquito has far better performance than its counter parts. From these obvious reasons it has been proposed that the more suitable messaging protocol for the smart agriculture is the MQTT based Mosquito implementation. Results presented in the present work provide a comparison of the evaluation of the IoT protocols with respect to agriculture domain.

References

[1] R. Ni, "Design and implementation of intelligent greenhouses based on the internet of things," in Applied Mechanics and Materials, 2014, vol. 631, pp. 188–191.

[2] A. Abbas, M. T. Jilani, and M. K. Khan, "Comparative Analysis Of Wireless Technologies For Internet-Of-Things Based Smart Farm," Sci. Int., vol. 29, no. 2, pp. 373–378, 2017.

[3] S. Wolfert, L. Ge, C. Verdouw, and M.-J. Bogaardt, "Big Data in Smart Farming--A review," Agric. Syst., vol. 153, pp. 69–80, 2017.

[4] F. TongKe, "Smart agriculture based on cloud computing and IoT," J. Converg. Inf. Technol., vol. 8, no. 2, 2013.

[5] C. Malveaux, S. G. Hall, and R. Price, "Using Drones in Agriculture: Unmanned Aerial Systems for Agricultural Remote Sensing Applications," in 2014 Montreal, Quebec Canada July 13--July 16, 2014, 2014, p. 1.

[6] Y. Shi, Z. Wang, X. Wang, and S. Zhang, "Internet of Things Application to Monitoring Plant Disease and Insect Pests," in International Conference on Applied Science and Engineering Innovation (ASEI 2015), 2015, pp. 31–34.

[7] D. Georgakopoulos and P. P. Jayaraman, "Internet of things: from internet scale sensing to smart services," Computing, vol. 98, no. 10, pp. 1041–1058, 2016.

[8] Z. Shelby, "Embedded web services," IEEE Wirel. Commun., vol. 17, no. 6, 2010.

[9] C. Bormann, A. P. Castellani, and Z. Shelby, "Coap: An application protocol for billions of tiny internet nodes," IEEE Internet Comput., vol. 16, no. 2, pp. 62–67, 2012.

[10] M. Collina, G. E. Corazza, and A. Vanelli-Coralli, "Introducing the QEST broker: Scaling the IoT by bridging MQTT and REST," in Personal indoor and mobile radio communications (pimrc), 2012 ieee 23rd international symposium on, 2012, pp. 36–41.

Practices for Achieving Accuracy in Software Costing and Estimation

Mohammad Ayub Latif[1] Muhammad Yaseen Khan[2] Kashif Bashir[3]

Abstract

Predictability is the foremost choice of all stakeholders in software development; it could be custom software or a general solution. Many software costing models have been proposed and used in the software industry over the last 40 years. In this paper, we go down into the details of recent approaches in software estimation and besides we propose the mandatory steps which can lead towards accuracy in software cost estimation. The more mature an organization is in costing and estimation, the more accurate results it is expected to achieve. We believe that using the steps defined in this paper will lead to more accurate results in costing and estimation. Unlike the Capability Maturity Model (CMM), we don't propose any specific levels and designate key process areas to it; but specific list of procedures towards an accurate costing and estimation of software is clearly identified in this paper.

Keywords: software development, software costing, costs, estimation, software estimation techniques, project management

1 Introduction

Not everything in the world is free, people have to pay for their work to make several contributions in return, and it continues to function in a cyclic fashion. Estimating the price and cost of work involves a lot of things, especially for the intangible assets like softwares which pertains to change with the pacing technological changes and human skill-sets. Achieving accuracy in estimation is one of them. If we look at research in software costing and estimation, we will find that significant cost failures are reported in many studies: The Standish Report stands as a leading example [1], [2]. Boehm et al. in their classic paper [3], argue that a careful evaluation of estimation results generated through several techniques is most likely to produce a realistic estimate. It comes as a valid question that the mentions in the Standish Report are still justifiable or not? Many researchers have challenged the validity of the Standish Report [4], [5] and have provided evidences that much-exaggerated figures were promulgated in the report [6]. We have several software costing and estimating models. Function points [8] stay as the very classical regression approach for software costing. These estimating models based on regression techniques conventionally function the estimates based on the historical data, collected on the completed projects through equating various variables and relationships therein [9]. The other widely used parametric models for software cost estimation comprise COCOMO [10], COCOMO-II [11], SLIM [12], SEER-SEM [13], and ESTIMACS [14]. These software estimation models churn the tentative cost, duration and efforts required for completing the software development. They include the factors like the desired functional needs of the software and size of the product. Along with these regression and parametric approaches, software

[1] *Karachi Institute of Economics and Technology, Karachi | malatif@pafkiet.edu.pk*
[2] *Mohammad Ali Jinnah University, Karachi | yaseen.khan@jinnah.edu.pk*
[3] *Karachi Institute of Economics and Technology, Karachi | kashif@pafkiet.edu.pk*

engineering practitioners have also employed machine learning (ML) techniques for predicting the software cost estimates. Since we are targeting a continuous number as an output, therefore, in ML the software cost estimation will be considered as a regression problem [15], [16]. In this regard, a leading study was made by Krishnamoorthy et al. in [17] in which Neural Networks (NN) and Regression tree based ML approaches were employed. An adaptive learning model for estimating software costs is presented in [18], while other NN based ML models are employed by many researchers for software cost estimation [19]–[21].

Researchers have also dwelled on the identification of causes that lead to inaccurate estimation for a software [22]. Many have invested their energy to understand the exact cause. Todd, in his famous list has concluded that "frequent requests for changes by users" and "lack of understanding of their own requirements'' constitutes the top two causes of inaccurate estimates[23]. These surely cannot be considered as the only reasons; there are numerous other reasons as well in which errors come from the estimation process itself and also from many other areas. Right time of committing the cost of software is critically important to address. As an experience, it comes as a practical agreement that the variability factor decreases, and the cone of uncertainty shrinks as we continue with the software development [7]. Hence, keeping the cone of uncertainty in context, we can suitably find the ideal time for cost commitment.

Instead of proposing any new model for software cost estimation, we believe that a step by step procedure to attain estimation accuracy is not clearly proposed by the research community. Therefore, in this paper, we have proposed the necessary steps in a sequential order which can lead towards best estimation results. We have used the specified procedure in small projects, made professionally for the industry or for the academics.

The rest of the paper is structured as follows: In the next section, we discuss several studies pertaining to software costing and estimation and provide strategies for costing without the knowledge of which one can never be a good estimator. Section III provides the details of how our proposed methodology gives a step by step mechanism for accurate estimation of software. In section IV, we describe how our methodology is applied in the estimation of software for academics and industry and in section V, we discuss the result of proposed methodology followed by conclusion and direction of future research in section VI.

2 Strategies for Software Costing

A Time of cost commitment

Understanding of "the cone of uncertainty" [10], as it was coined by McConnell [24], guides us that we should not commit the cost and efforts required to complete the project before creating the prototypes of the software system. For information system, this means, the complete Graphical User Interface (GUI) of the software system should be prepared.

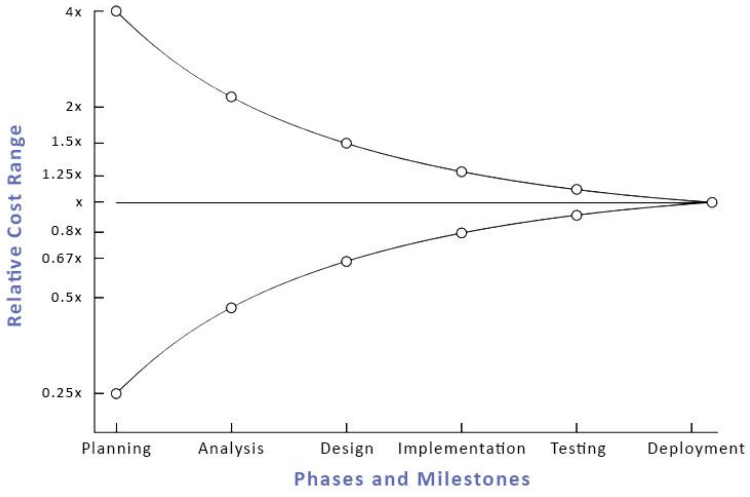


Figure 1: The cone of uncertainty with variability factor.

If we look at the figure 1, it is quite obvious that until the phase of designing is started (which is generally the time when requirements specifications are written) the cone is narrowed to 1.5 at the higher side and 0.6 at the lower side. This means the variability factor at this point has reduced from 16x to 2.5x. It could be probably a good time to commit the software cost estimate and effort.

B Understanding of Parkinson's Law

Parkinson's Law [25] stipulates that the assigned work will automatically acquire the available time, this implies that available time automatically finishes when a particular activity is finished, even if the work is much less to be done in allocated time. Chris, in [26], has explained the phenomenon of Parkinson's Law in a great detail. The irony of the fact is that even when the assigned task requires lesser time than the allocated time, the available time finishes. On the other hand, when the activity does not complete in available time, the time period overruns for that activity. Hence, stringent project control is required by the project manager and/or team lead to make sure that Parkinson's Law is not applied in the software project.

C Subjectivity

A good estimator needs to know that subjectivity engenders an element of doubt. Considering this, McConnell in his classical book [7] has identified and proved that the more controlling knobs we have in an estimation model, the more variance we can get in our estimates. This is the reason why the Intermediate and COCOMO II [11] models are criticized by researchers and experts because they have 15 and 17 effort adjustment factors respectively, and these adjustment factors can create subjectivity in estimation cases.

D Biasedness

An estimate can be said biased when it is the deliberate attempt to fudge estimation in one direction or the other, when pressure is applied to the project. It needs to be avoided at all costs.

E The Law of Large Numbers

The law guides us that the accuracies of empirical calculations can be achieved with number of trials [27]. Historically, it was named "Golden Theorem" or "Bernoulli's Theorem" [28], later in 1837, it was further explained by Simeon´ Poisson under the name "The Law of Large Numbers (LNN)" [29]. In the context of software cost estimates, whenever, we are to give an estimate which is based on multiple tasks, we should calculate the time frame and effort of the individual (near atomic) tasks, and then calculate their total. The best advantage of this repeating activity is that our estimates will cancel each other on both positive and negative sides, and therefore, we will encounter a much diminished error that could have happened if the total estimate was calculated in a consolidated manner.

F The Need of Data

Data for the purpose of estimation can be classified into three main categories: i) industrial data, ii) historical data and iii) project data. In many research papers, the different type of data is used. We know that the most accurate among the three is project data as it is the data of the ongoing project itself. The only disadvantage it poses is that for the use of project data our software development model has to be an iterative model.

G Avoiding arguments and let data do its job

We unknowingly enter into subjectivity when we evaluate people through assigning values of adjustment factors used in different costing models. COCOMO-II [11] provides us with people classification in its 15 adjustment factors where we need to select analyst capability, programmer capability etc. as low, marginal, high etc. Evaluating a person through these value can give rise to politically charged statements like, "my programmers are below average" etc. [7] A better way to say the same thing in front of management is, "We averages 300 LOC productivity in the previous project we have used the same average for the new project" [7].

H Accuracy, not Precision

In software costing and estimation, our goal is to be accurate, not precise [7]. For example, rather than saying a particular project will be completed in 27.5 staff months it is better to say that the project will take between 25 - 29 staff months. This will give us more chances of being accurate, so the trick of the trade is to commit your estimates in ranges, this way we will achieve accuracy.

I Flat or Dynamic Staffing Model

Broadly the costing models can be classified as the flat staffing model [3] or dynamic model [7]. A flat model is the one which assumes that same number of people will be working for the software from its early requirements phase to the implementation phase. A dynamic model assumes that team size can be changed with respect to phases of the software during development. For example, the team size can be of 2 people in the requirement phase and 10 people in the development phase.

3 Proposed Methodology for Software Estimation

In this section, we propose 7 steps which should be followed in a proper sequence which will surely lead towards the most accurate estimate.

1) **Not too early, even not too late to commit.** As an estimator is it very important that we should commit our estimate to the customer when the Requirements Specifications phase has finished. If we look at this with respect to the cone of uncertainty then by this time the 16x factor has reduced to 2.25 [23]. How can an estimator make sure that he/she has reached the desired point? The signed Requirements Specifications Document (SRS) will be a good proof of this that we have reached this level. For small projects, this point can even be until the completion of GUI where the factor can be between 1.6x to 1.8x.

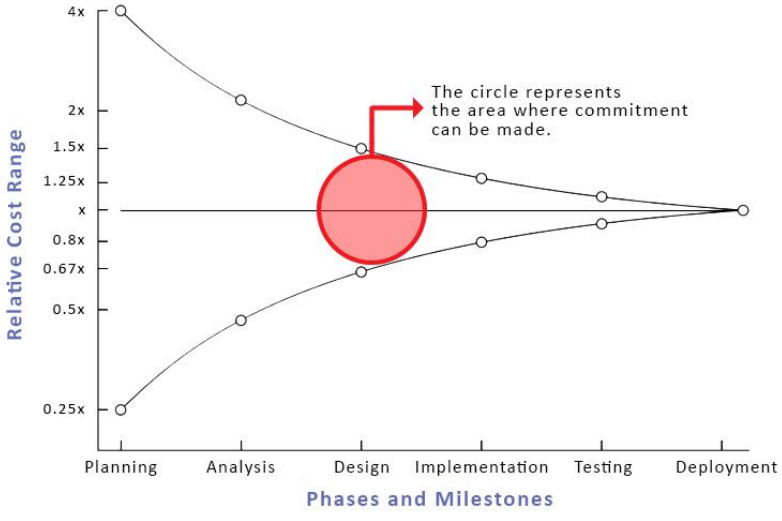


Figure 2: The circle in the cone suggesting the best time to make commitment.

In the figure 2, the circle shows the area where commitment can be made as the cone of uncertainty has narrowed to a point where variability is in a manageable state and going on either side will not create a huge problem. On the left edge of the circle the variability factor is 0.55 and 1.5 on the higher and the lower sides that makes it 3.1x and on the righter edge of the circle the variability factor is 1.2 and 0:8 that makes it 1.5x.

2) **Dynamic cost estimation models.** These are generally suggested as it helps in many cases. Initially when the customers are quickly looking for the estimate to give approval whether they would approve the software to be made or not, adding more people to the project investigating will help in an effective manner. Once the cone of uncertainty has narrowed to a point when the commitment can be made, the commitment can be made and immediately after the customer approval, the team size can be reduced during the post requirements and analysis phases. Again the

team size can be increased on "as needed" basis. This liberty is possible only when we use dynamic models for estimation.

3) **Avoid cloud of uncertainty.** We need to make sure that due to weak controls, cloud of uncertainty is not formed in our project. Because, with its formation we will head towards a disaster and our project can reach the referent point. The figure 3 shows the effect of cloud in the cone of uncertainty. The variability factor remains until the end of the project.

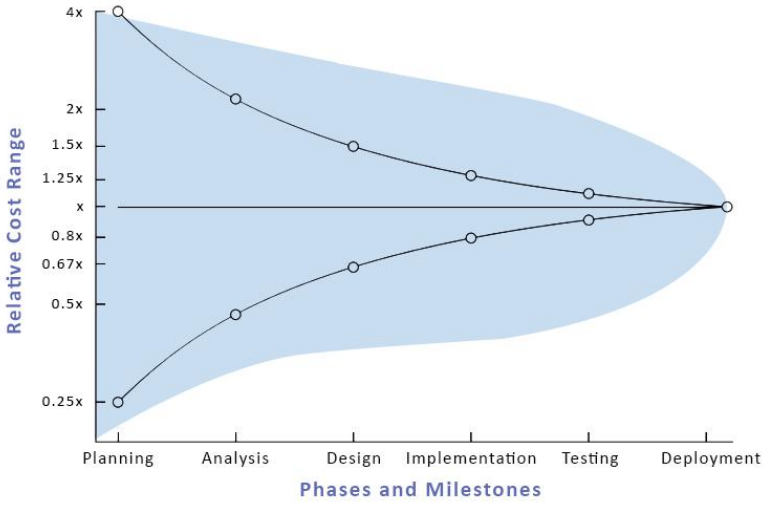


Figure 3: The cloud of uncertainty which appears because of weak project controls.

4) **Tasks up-to 2 weeks.** The overall task of the entire project must be broken down into small task list where each task must not take more than 2 weeks. We have already discussed in our strategy section II-E that how breaking the estimate can help us in taking advantage of the law of large numbers.

5) **Iterative development model.** Select an iterative model for the development of the software so immediately after the first iteration; we can use project data for later estimation instead of historical data. We have already seen in section II-F that project data is always the most accurate to use.

6) **Convergence in estimating techniques.** Always use multiple techniques for your estimation case and look for conver-gence among them. In next section, we have used multiple techniques in our estimation case and have shown how we have used convergence among two techniques to rule out the third technique that showed divergence from the other two.

7) **Standard deviation.** When there is convergence in multiple techniques we have a consensus to believe that we are near to a good estimate. At this time instead of committing the estimate as it is; a better way is to use standard deviation and, propose the estimate with specific confidence.

4 Case Studies

We demonstrate the usage of principles of our proposed methodology in student registration software for a small sized institute. At first, we didn't apply the principles discussed in this paper and just did a general estimate using the proxy-based technique of standard components. Standard Components is discussed in depth in [7] and the authors also discuss components as a modularity technique for Agile developments [30]. Later, we decided to apply the principles discussed in this paper to the estimation case of students' registration system and came up with the following results. In this section, we discuss the complete estimation case, a brief discussion of the problem statement is also given in this paper, but a detailed discussion is avoided as it is not necessary for the domain of this paper.

A Problem Statement of the Student Registration System

The 'student and course registration problem' is used to calculate the effort and cost required for the software. The process of assigning teachers to courses and the registration of students is a frustrating and time-consuming job for any institute.

In the new system, after the management and teachers of the institute have decided which course in which timings they are going to teach for the session, the administration office enters the information into the computer system. The registered students when they log in the system will see a report of course timings and teachers' information. They will register themselves for specific timings with a specific teacher. None of the course will have more than 12 students. Once 12 students have been completed for a particular course, the system will not show that timings and teacher option to the students for registration.

When the student has registered himself/herself, a message is prompted on their screen of successful registration and they can even see the monthly schedule of their classes. The system gives a tentative date for the start of the course and if a course is closed for registration, it gives out a specific date also. One day before the class, the system also generates a message to the student on his/her cell phone (short texts / SMS) so that they know that their class is starting from tomorrow.

Teachers can also log into the system and generate the list of students which they will be teaching in a particular time slot. Obviously, the teachers will generate the list when the registration of a particular slot is closed. The entire system is web-based and is developed using ASP.Net.

B Initial requirements and analysis of the problem domain

The following actors and use cases were identified for the problem statement:

1) Actors
   a) Teacher
   b) Academic/Admin Staff
   c) Student
2) Main use cases of the system with actors are shown in the table 1.

Table 1: The main use cases and actors of the problem statement

Use Case	Actor
Entering course and timings	Admin/Academic
Selecting and registering course	Student
Generating course schedule	Student
Generating Student list for a course	Teacher

C Costing Metric

Let C be the set of all type of components used in Student Registration System C = {c1, c2, … cn}, and L be the set of LOC required for completing individual components respective, such that, L = { ∫c1, ∫c2, … ∫cn} where ∫c1 indicates the LOC required for component c1. Hence by aggregating the productions of C and L accordingly, we can have the total LOC required for completing the whole Student Registration System. The function f(C; L) models the metric below under the condition that provides ||C|| = ||L||.

$$f(C + L) = \sum_{i=1}^{\|c\|} (c_i \,.\, l_{ci}) \quad (1)$$

Case 1: Without applying the principles to achieve estimation accuracy

The broader category of standard components is proxy based techniques and it is one of the types of the proxy-based technique. The core idea is that if you develop many programs that are architecturally similar to each other, you can use the standard components approach to estimate size.

For estimation purpose, we have chart of historical data as depicted in table II to apply estimation using the standard components technique.

Table 2: Components for Case Study I

Standard Component	LOC/Component
Dynamic Web Pages	417
Static Web Pages	78
Database Tables	1227
Reports	388

Table 3: Estimation for Case Study-I w.r.t Number of Components and LOC

Type of Component	Number of components in Case Study I	LOC/component	Total LOC for specific component
Dynamic Web Pages	14	417	417 x 14 = 5, 838
Static Web Pages	7	78	78 x 7 = 546
Databases	12	1,227	1, 227 x 12 = 14, 724
Reports	11	388	388 x 11 = 4, 268
Business Rules	3	4,237	4, 237 x 3 = 12, 711
Total LOC for the Students Registration System			38, 087

Table 4: Estimation for Case Study-II w.r.t Number of Components and LOC

Type of Component	Number of components in Case Study I	LOC/component	Total LOC for specific component
Dynamic Web Pages	18	367	367 x 18 = 6, 606
Static Web Pages	9	110	110 x 9 = 990
Databases	10	1, 037	1, 037 x 10 = 10, 370
Reports	14	270	270 x 14 = 3, 780
Business Rules	2	3, 100	3, 100 x 2 = 6, 200
Total LOC for the Students Registration System			27,946

As such systems are generally web-based information systems so this much of historical data with the specified LOC is sufficient enough for an early estimate of effort. For the three stated actors, we identified the number of components for the five stated types for each of the actor. Then we sum up the total number of components of different types to reach to an early effort estimate based on the Lines of Code (LOC). With the simple analysis, we made out that this number of components of each type will be used in our Students Registration System.

Table III shows that the basic standard components technique and applying the historical data, the first estimate using equation 1 was around 38, 000 LOC or ≈38 KLOC.

Case 2: Applying the estimation principles to discussed in this paper

For the simple estimation case, we tried to include maximum steps from our proposed methodology and came up with a visible accuracy of x%. The implementation of our step by step estimation accuracy, in this case of student registration system, is stated as below:

We made use of our step number 1, by delaying the estimate till we have not generated the GUI of the software. Step 2 was also followed as we deployed maximum people at the start of the project to get to the User Interface of the software and after getting the estimate when we were sure of an accurate estimate we lessened the number of members of the team for later phases. This was possible as we used dynamic costing models. Step 3 was also used as we had

made projects before and we knew that which wrong practices can lead us towards a cloud of uncertainty. Talking about our suggested step 4, we believe that in the first case also the project was nicely broken down in smaller tasks so we did not bother breaking it further as we believed that law of large numbers advantage was already applied in the first estimation case. Secondly, with so much smaller tasks there were lesser chances that the project estimate can get away with hidden work.

We already follow the rational unified approach for creating software in our software development unit that is why we followed step 5 also as Rational Unified Process (RUP) [31] is an iterative model and as we suggested earlier and we immediately started using project data instead of historical data as soon as we started with the next iteration.

When we applied the principles we were able to commit deadline for the completion of the project after 15 days of the inception phase, in case 1 the commitment was made after 5 days of the inception phase.

Please note that the historical data figures as used in case number 1 were changed as we used project data for this estimation and we derived to project data after performing the first iteration. The project data value chart used for the second estimation case is given in table V.

Table 5: Components for Case Study II

Standard Component	LOC/Component
Dynamic Web Pages	367
Static Web Pages	110
Database Tables	1, 037
Reports	270
Business Rules	3, 100

The summary of our estimation with equation 1 is shown in table IV. Through the basic standard components technique and using the steps for accurate estimates as identified in this paper and using project data, the second estimate was around 2, 8000 LOC or ≈28 KLOC.

5 Results

The project was completed in 2 months using a team of 4 dedicated people in the team, two working as analyst and designers and other two working as programmers; considering a productivity of 3,000 LOC per person per month, it shows clearly that project was completed in 24, 000 LOC or ≈24 KLOC. Considering the result it is clearly evident that following the estimation steps as specified in this paper we can reach towards more accurate results in software costing and estimation.

6 Conclusion

In this paper, we studied about the strategies used in software costing and estimation and then we proposed few steps which if implemented during the estimation process can lead toward more accurate estimate of the software under development. Our results show that using five out of the seven specified steps we were able to commit an estimate which was only deviating 16.66% from the actual outcome than compared to the other estimate which was 58.33% away from the actual outcome. It was completely unintentional that both the estimation cases were on the overestimate side.

There are few more observations which pave the way for future research; the first is that the more accurate estimate was given after 15 days of the inception phase, although the first estimate which was low in accuracy was given after 5 days of the inception phase. It needs to be investigated what out of the two is a bigger benefit, early commitment or accuracy. For this estimation case, we confined to the simple information system, the likes of which were previously made by the development organization. The future investigation can be done on different types of software and of unfamiliar areas where expertise lack.

References

[1] G. Standish, "The chaos report," The Standish Group, 1994.

[2] M. Jørgensen and K. Moløkken-Østvold, "How large are software cost overruns? a review of the 1994 chaos report," Information and Software Technology, vol. 48, no. 4, pp. 297–301, 2006.

[3] B. Boehm, C. Abts, and S. Chulani, "Software development cost estimation approachesa survey," Annals of software engineering, vol. 10, no. 1-4, pp. 177–205, 2000.

[4] R. L. Glass, "It failure rates-70% or 10-15%?" IEEE Software, vol. 22, no. 3, 2005.

[5] R. L. Glass, "The standish report: does it really describe a software crisis?" Communications of the ACM, vol. 49, no. 8, pp. 15–16, 2006.

[6] K. Moløkken and M. Jørgensen, "A review of software surveys on software effort estimation," in Empirical Software Engineering, 2003. ISESE 2003. Proceedings. 2003 International Symposium on. IEEE, 2003, pp. 223–230.

[7] S. McConnell, Software estimation: demystifying the black art. Microsoft press, 2006.

[8] J. E. Matson, B. E. Barrett, and J. M. Mellichamp, "Software development cost estimation using function points," IEEE Transactions on Software Engineering, vol. 20, no. 4, pp. 275–287, 1994.

[9] R. E. Fairley, "Recent advances in software estimation techniques," in Proceedings of the 14th international conference on Software engineering. ACM, 1992, pp. 382–391.

[10] B. W. Boehm et al., Software engineering economics. Prentice-hall Englewood Cliffs (NJ), 1981, vol. 197.

[11] B. W. Boehm, R. Madachy, B. Steece et al., Software cost estimation with Cocomo II with Cdrom. Prentice Hall PTR, 2000.

[12] L. H. Putnam and W. Myers, Five core metrics: the intelligence behind successful software management. Pearson Education, 2013.

[13] D. D. Galorath and M. W. Evans, Software sizing, estimation, and risk management: when performance is measured performance improves. CRC Press, 2006.

[14] H. A. Rubin, "Macroestimation of software development parameters: The estimacs system," in SOFTFAIR Conference on Software Development Tools, Techniques and Alternatives, 1983, pp. 109–118.

[15] N. M. Nasrabadi, "Pattern recognition and machine learning," Journal of electronic imaging, vol. 16, no. 4, p. 049901, 2007.

[16] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, Machine learning: An artificial intelligence approach. Springer Science & Business Media, 2013.

[17] K. Srinivasan and D. Fisher, "Machine learning approaches to estimating software development effort," IEEE Transactions on Software Engineering, vol. 21, no. 2, pp. 126–137, Feb 1995.

[18] A. Kaushik, A. Soni, and R. Soni, "An adaptive learning approach to software cost estimation," in Computing and Communication Systems (NCCCS), 2012 National Conference on. IEEE, 2012, pp. 1–6.

[19] N. Tadayon, "Neural network approach for software cost estimation," in Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on, vol. 2. IEEE, 2005, pp. 815–818.

[20] D. E. Neumann, "An enhanced neural network technique for software risk analysis," IEEE Transactions on Software Engineering, vol. 28, no. 9, pp. 904–912, 2002.

[21] G. E. Wittig and G. Finnic, "Using artificial neural networks and function points to estimate 4gl software development effort," Australasian Journal of Information Systems, vol. 1, no. 2, 1994.

[22] A. L. Lederer and J. Prasad, "Causes of inaccurate software development cost estimates," Journal of systems and software, vol. 31, no. 2, pp. 125–134, 1995.

[23] T. Little, "Schedule estimation and uncertainty surrounding the cone of uncertainty," IEEE software, vol. 23, no. 3, pp. 48–54, 2006.

[24] S. McConnell, Software project survival guide. Pearson Education, 1998.

[25] C. N. Parkinson and R. C. Osborn, Parkinson's law, and other studies in administration. Houghton Mifflin Boston, 1957, vol. 24.

[26] C. F. Kemerer, "An empirical validation of software cost estimation models," Communications of the ACM, vol. 30, no. 5, pp. 416–429, 1987.

[27] L. Mlodinow, The drunkard's walk: How randomness rules our lives. Vintage, 2009.

[28] J. Bernoulli, "Ars conjectandi: Usum & applicationem praecedentis doctrinae in civilibus," Moralibus & Oeconomicis, vol. 4, 1713.

[29] S. D. Poisson and C. H. Schnuse, Recherches sur la probabilite´ des jugements en matiere` criminelle et en matiere` civile. Meyer, 1841.

[30] S. K. T. Ziauddin and S. Zia, "An effort estimation model for agile software development," Advances in computer science and its applications (ACSA), vol. 2, no. 1, pp. 314–324, 2012.

[31] P. Kruchten, The rational unified process: an introduction. Addison-Wesley Professional, 2004.

Call for Papers/Authors Guideline

KIET Journal of Computing & Information Sciences (KJCIS) is biannual publication of College of Computing & Information Sciences, PAF-Karachi Institute of Economics and Technologies. It is published in January and July every year. We are lucky to have onboard prominent and scholarly academicians as part of Advisory Committee and reviewers.

KJCIS is a multi-disciplinary journal covering viewpoints/ researches / opinions relevant to the non exhaustive list of the topics including data mining, big data, machine learning, artificial intelligence, mobile applications, computer networks, cryptography and information security, mobile and wireless communication, adhoc and body area networks, software engineering, speech and pattern recognition, evolutionary computation, semantic web and its application, data base technologies and its applications, internet of things (IoT), computer vision, distributed computing, grid and cloud computing.

The authors may submit manuscripts abiding to following rules:-

- Certify that the paper is original and is not under consideration for publication in any other journal. Please mention so, in case it has been submitted elsewhere.
- Adhere to normal rules of business or research writing. Font style be 12 points and the length of the paper can vary between 3000 to 5000 words.
- Illustrations/tables or figures should be numbered consecutively in Arabic numerals and should be inserted appropriately within the text.
- The title page of the manuscript should contain the Title, the Name(s), email address and institutional affiliation, an abstract of not more than 200 words should be included. A footnote on the same sheet should give a short profile of the author(s).
- Full reference and /or websites link, should be given in accordance with the APA citation style. These will be listed as separate section at the end of the paper in bibliographic style. References should not exceed 50.
- All manuscripts would be subjected to tests of plagiarism before being peer reviewed.
- All manuscripts go through double blind peer review process .
- Electronic submission would only be accepted at kjcis@pafkiet.edu.pk
- All successful authors will be remunerated adequately.
- The Journal does not have any article processing and publication charges.

Submission is voluntary and all contributors will find a respectable acknowledgement on their opinion and effort from our team of editors. Submission of a paper will be held to imply that it contains original unpublished work. In case the paper has been forwarded for publication

elsewhere, kindly apprise in time if the paper has been accepted elsewhere. Manuscripts may be submitted before September and May to get published in Jan & July issues respectively. We encourage you to submit your manuscripts at kjcis@pafkiet.du.pk

Editorial Board KJCIS
College of Computing & Information Sciences
PAF-KIET Institute of Economics and Technology

PAF-KIET Institute of Economics and Technology
Korangi Creek, Karachi-75190, Pakistan
Tel: (9221) 3509114-7, 34532182, 34543280 Fax: (92221) 35009118
Email: kjcis@pafkiet.du.pk
http://kjcis.pafkiet.edu.pk

KARACHI INSTITUTE OF ECONOMICS
AND TECHNOLOGY

PUBLISHED BY:
College of Computing and Information
Sciences, PAF-KIET.
www.cocis.pafkiet.edu.pk
www.kjcis.pafkiet.edu.pk
kjcis@pafkiet.edu.pk

MAIN CAMPUS

PAF BASE Korangi Creek,
Karachi-75190.
Ph: (9221) 35091114-7
Fax: (9221) 35091118

CITY CAMPUS

28-D, Block 6, P.E.C.H.S,
Karachi-75400.
Ph: (9221) 34543280
Fax: (9221) 34383819

NORTH NAZIMABAD CAMPUS

F-98, Block B, (Near KDA roundabout)
North Nazimabad, Karachi-74700.
Ph: (9221) 36628381 or 36679314
Cell: 0336-2444191-92