International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 504 - 508, 2016, ISSN:- 2319–8656

Object Oriented Programming (OOP) Approach to the Development of Student Information Management System

Onu, Fergus U.
Computer Science Department,
Ebonyi State University, Abakaliki – Nigeria

Umeakuka, Chinelo V.
Computer Science Department,
Ebonyi State University, Abakaliki – Nigeria

Abstract: It is not in doubt that good data management is essential to the success of every organization. Most instituions in Nigeria still adopts the use of Relational Database Management System (RDBMS) platform for the management of their students' information hence they grapple with the enormous and overwhelming challenges facing the RDBMS technique. This paper presents the Object Oriented programming concepts in the development of the system for student data management. The research used Object oriented analysis and design (OOA&D) and Agile methodology to realise a simple and easy to manage approach to data representation, storage and retrieval. The resulting model of Students Registration System at the departmental level drastically reduced maintenance cost and increased productivity. The system generally reduced the burden of data management, report generation and presentation and hence brought efficient resources utilisation to the institutions.

Keywords: Object Oriented Database, software maintenance cost, UML, data retrieval, Object Oriented Models.

1.0 INTRODUCTION

The measurement of success of any Institution is largely dependent on its record keeping capability and management process. To enhance the process, there is the need to deploy Object Orientation in student Information system in order to provide better speed and efficiency in the representation of students' data, at the development and maintenance stage. The use of the concept of object orientation (OO) in the analysis and design of the application would prove beneficial in terms of cost, energy and time [1]. For over a decade, Relational databases (RDB) have been the accepted model for storage and retrieval of huge volumes of data. This technique is faced with a lot of problems thus:

- Complexity of Application: Applications often require large amount of code to produce many varied reports, the level of complexity as measured by the interactions between modules is relatively low. So, too limited to handle large class of applications[15].
- Hierarchy and Relationship: Database model is less expressive and flexible in terms of network of connected pieces of information[15]
- Query languages (Invocation of Operations): Relational query languages are not computationally complete and programming environment can be less uniform. As a result, the bulk of the application code may not reside in the database and not managed by all of the database facilitates (e.g., concurrency, recovery, and version control). Hence, data needs to be copied into its virtual memory[15]
- Automatic type checking: Type failures are only detected at the end of a transaction when the new values are checked back in, so does not support automatic type checking. [15]

Object Oriented Database (OODB) model provided solution to many of the problems associated with RDB. Based on the concept of abstraction and generalization, object oriented models capture the semantics and complexity of the data.[3]. Many authors have stated that OODBs are optimized to provide support for object oriented applications, different types of data structures including trees, composite objects and complex data relationships. The OODB system utilizes the concepts of object-oriented languages and has the capability to handle complex databases efficiently and it can allow the users to define a database, with features for creating, altering, dropping tables and establishing constraints.

The principal strength of OODB is its ability to handle applications involving Complex and interrelated information [2], but in the current scenario, the existing Object-Oriented Database Management System (OODBMS) technologies are not competing in the market with their RDB counterparts [7]. Also, there are numerous applications built on existing relational database management systems (RDBMS). It is difficult, if not impossible, to move off those RDBs. Hence, [7] felt that there is need to incorporate the object-oriented concepts into the existing RDBMSs.

Student Management information system (SMIS) is a computer-based system used within an Institution of higher learning. It is designed to be a secure, confidential collection of data about students which help in proper administration and management of students at the departmental level in higher Institutions. It is a system to handling objects and object identity by deploying the concepts of encapsulation, classes, and inheritance in an efficient way.

The system became very desirable dut to the ever increasing need to manage the data of students with a more robust technique. That was the motivation for this work. So we aimed at transforming the Relational database systems of managing student data which are prevalent within the university systems in Nigeria into Object Oriented systems, and show efficiently how administration is made easier through this effort.

The identifying benefits of Object Oriented model (OOM) in Student Information System in addition to the analysis of how OOP concepts is applied in complex system to make it easier and safer to implement in information systems showed that the use of OOP in the system should be embraced to effectively check the excesses of students.

2.0 LITERATURE REVIEW

A lot of related work focused on features of Object Oriented Programming and systems relating it to Relational Database Systems while some worked on its application areas models

Through interrelated works on Object oriented analysis and modeling, [1] explored the basics and advancement of OOA, its utilization and implementation. The use of concepts of objects in analysis and application design to prove its benefits, identified the shortcoming and its solution at the project phase. In a reviewed by [4], the concepts of Class, Object and inheritance, based on extent of coupling and cohesion in OOS, and their effect on results produced, taking into consideration the run-time properties of programming Languages were presented.

[5] Surveyed the discretional access control issues and mechanism of both structural and behavioural aspects of subject to Object, Inter-object and Intra-Object and their effects on object oriented design (OOD). They also explored other authorization mechanisms beneficial to OODBMS. Object Oriented modeling using Inheritance and propagation properties for complex systems was analysed by [6]. They highlighted how OO approach has powerful tools for data structuring in terms of generalization, classification, Aggregation and Association. The importance of inheritance and propagation to model dependencies of property operations and values as well as in the implementation were seriously considered.

It was the focus of [7] to design an object-oriented database, through incorporation of object-oriented programming concepts into existing relational databases. The presented approach of the Object oriented programming concepts such as inheritance and polymorphism aids showed the efficiency in data mining. The experimental results demonstrated the effectiveness of the presented OO approach in successful reduction of implementation overhead. There was a considerable reduction in the amount of money paid for memory space required for storing databases that grow in size in the design of an OODB when compared to the traditional databases.

2.1 Concept of Objects, Classes and Inheritance in Object Oriented Systems

In the modern computing world, the amount of data generated and stored in databases of organizations is vast and continuing to grow at a rapid pace [8], the OODMS which employs the use of OOPL should be incorporated to be able to handle the system efficiently. The concepts of OOP are Object, Class and Inheritance is reviewed.

Concept of Object and Class

Object is a central abstraction that models a real world entity. Every object encapsulates some state and is further uniquely identified by an object-identifier.[5]. The word object is used for a single occurrence (instantiation) of data describing something that has some individuality and some observable behaviour. The terms object type, sort, type, abstract data type, or module refers to types of objects, depending on the context [6]. In designing an application, objects should conceptualize the design using the real world components as objects. The state of an object is made of the values of its attributes (that describe the real world entity modelled). In behaviourally object-oriented database, the object state is accessible only through the operations (methods) supported by its interface(s). Every operation (method) is associated with a method body that contains some piece of executable code that models the behaviour of the corresponding real world entity. Every object belongs to a type that is determined by its class, and is thus considered to be an instance of the class [5].

A class is thus akin to an abstract data type definition. Classes can be organized into class hierarchies enabling the sharing of structure and behaviour through the mechanism of inheritance [5].

In OO design, the coupling of a class means the measurement of the interdependence of class with the other classes. In a design of reasonable size design size is ten classes normally classes do not exist in absolute isolation [4].

Concept of Inheritance

Inheritance is the transitive transmission of the properties from one super class to all related subclasses, and to their subclasses [6]. Inheritance is beneficial in terms of high reduction rate of data redundancy and maintains high integrity of data, consistence and modularity.

The most important object-oriented concept employed in an OODB model includes the inheritance mechanism and composite object modelling [13].

An inherited class is the base class or super class or parent class, whereas derived class is the subclass or child class. Defined operations on super class apply to other objects of its subclass. Defined operations on subclass are not related to the super class objects.

We can have single or multiple inheritances, while single inheritance restricts relations to a strict hierarchical structure, multiple inheritances allows properties defined on several super classes to be accessed by one or more

subclass thereby, should be considered to achieve desired goal. Good design decision creates better relations among interacting objects and their properties.

2.2 Unified Modelling Language (UML)

A **UML** is a standard modeling Language to model the real world in the field of software engineering. A UML diagram is a partial graphical view of a model of a system under design, implementation, or already in existence. UML diagram is made up of graphical elements, UML nodes connected with edges (flows) that represent elements system model. The UML model of the system might also contain other documentation such as use cases written as texts.

The kind of the diagram is defined by the primary graphical symbols shown on the diagram. Many UML diagrams exists but we look at the properties of Class diagram as we will be using it in the model (Table 1).

Table 1: UML Class and Object diagram properties

Diagram	Purpose	Elements
Class diagram	Shows structure of the designed system, subsystem or component as related classes and interfaces, with their features, constraints and relationships - associations, generalizations, dependencies, etc.	class, interface, feature, constraint, association, generalization, dependency.
Object diagram	Instance level class diagram which shows instance specifications of classes and interfaces (objects), slots with value specifications, and links (instances of association).	instance, specification, object, slot, link.
Use case diagram	Describes a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors) to provide some observable and valuable results to the actors or other stakeholders of the system(s).	use case, actor, subject, extend, include, association.

Source: http://www.uml-diagrams.org/uml-25-diagrams.html

3.0 OBJECT ORIENTED STUDENT INFORMATION MANAGEMENT SYSTEM (OOSIMS) SYSTEM METHODOLOGY

The Student Information system is a server-based system that uses an Object-Oriented approach to manage student registration at the department level. It consists of good data integrations features, good GUI to enhance user experience and flexible reporting features to deliver value to users.

In our design, we present the approach that extends the relational database system of managing student registration to an Object Database Management System (ODBMS) incorporated by utilizing the OOP concepts like Objects, classes, inheritances, and encapsulations. The modeled system makes use of the OOP relationship feature to show interaction among objects and classes, and these include Association, inheritance and Generalization This ability to represent classes in hierarchy is one of the OOP beauties.

Object Oriented Students Information Management System (OOSIMS) designed and modeled is for yearly record of new students and update of existing student information at the departmental level in higher institutions through the interfaces provided. A sample of the interface for the capturing of student bio data information is shown in Figure 1. It facilitates access to the information about a student at anytime, registered courses, and an id card is generated for every registered student to check impersonation during an examination

The System serves as a repository in the department, showing information about students

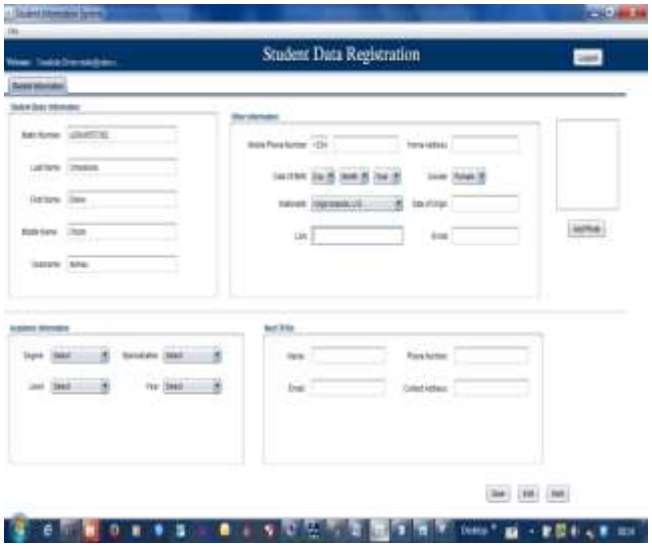


Fig. 1: Interface of Student Data Registration

3.1 Agile Methodology

The Agile Methodology employed, with the use of UML Class Diagram tool shows the structure of the Object Oriented Student Information Management system (OOSIMS). Its component as relates to entity type and responsibility, classes. The featured relationships include Inheritance, associations, generalizations.

UML 2 class diagram is one of the tools for representing object-oriented analysis and design. UML class diagrams in figure 2 shows the classes of the system, their interrelationships (including inheritance, association and Generalization, and the operations and attributes of the classes.

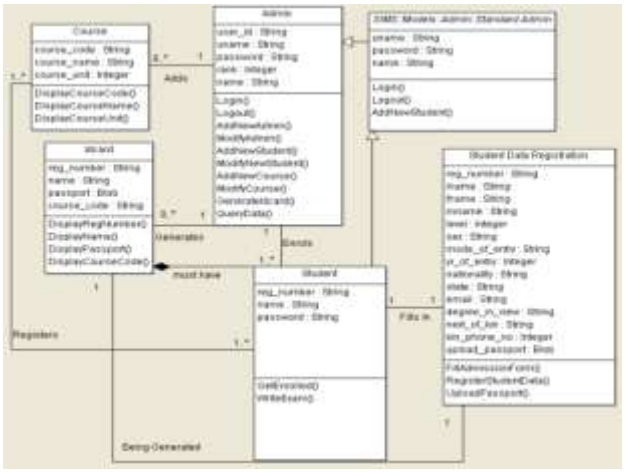


Fig. 2: Class Diagram of the OOSIMS Structure

3.2 Generalization and Inheritance Analysis

Classes and objects often show some similarities in attributes and methods. To maximize re-use of attributes and methods, Inheritance mechanism which "is a" and "is like" relationships, is deployed to avoid repetition of codes. These relations show high dependency nature to achieve a desired goal. These dependencies include generalization and association relationship types to reveal effective administration in student registration. Relationship Analysis among Classes is shown in table 2.

Table 2. Class - Relationship Analysis.

Relationship / Class (Modifier)	Inheritance type	Inheritance Property	Generalization	Association
Admin (Root)	Super class	-	-	Must create 'Student' with 0..1 multiplicity
Standard Admin *(Abstract)*	Sub class	all* of 'Admin' except rank, all # of 'Admin' except create, modify Admin	Can be both 'Admin' and 'Student'	-
Student	Super class	-	-	-Must be created by 'admin' to register, -Register 'Course with 1..* multiplicity
Student Data Registration *(Active)*	Sub class	all* of 'Student' except password,	-	Must have 'Id card' to write exam
Course	Super class	-	-	Each 'Course' must be registered by many 'Student' with with 0..1 multiplicity city
Id Card *(Leaf)*	Sub Class	Multiple- some * and # of 'Course', 'Student Data Registration', 'Student'.		must be generated for 'Student' with 1..* multiplicity

* denotes Attribute, # denotes Method

3.3 Process design in OOSIM System

The Functionality of the system is described using a Use Case Diagram illustrating the sequence of actions / interaction between the agents/actors and the database. Figure 4 illustrates the activities of actors in the system. The Actors are Admin, Database, and Student.. The generalization link indicates that the abstract '*Standard Admin'* can be an 'Admin' and a 'Student'. The actor 'Student' has direct use cases to the database indicating his actions in the system. The abstract use case '*include*' indicates that the student will always update the course as he moves from one level to another. The 'admin' creates student using reg number, student name and password. The 'Student' then login to fill the registration form, uploads passport, and register his courses

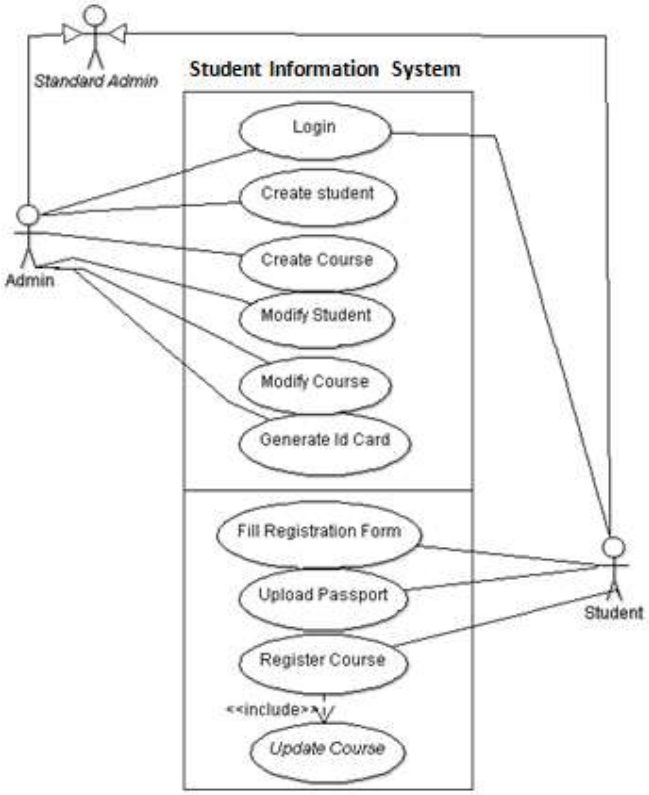


Fig.4: Use Case Diagram of OOSIMS Behaviour

4 DISCUSSION AND EVALUATION

Introducing object orientation through the OOSIMS in student information management described the main features of OOP in a database. Adopting Object Oriented Paradigm in student information management brings out the system functionality. It also showed objects and classes with their properties and operations. With OODB, the research has Identified the relationships and associations, as well as the coordination constraints among interacting objects and functional classes. All these have helped in the following:

1. Reduction of codes for developing an application and modification of similar functionality if already existed.
2. Enable the re-use of design and code function with minimal modification to suite a need (i.e. inheritance).
3. Improved Maintainability of OOSIMS by allowing complex systems broken into smaller manageable units.
4. Reduced cost and time of developing Student Information Management System due to the encouragement of team work.
5. Maintains data integrity by hiding access from unauthorized objects and users (i.e. encapsulation).

5.0 CONCLUSION

The OODB platform of student information system has been shown to be the most secured and flexible in the development of Student Information system. The object

oriented concepts impact in the areas of resource reuse, extensibility, maintenance, and scalability cannot be ignored. Inheritance and other relationship types employed in the structural definition makes the system concise. Its beneficial features like redundancy reduction, improved cost effective, data integrity and easy maintenance are to be considered by authority to embrace OODMS in administration of students to improve productivity.

6.0 REFERENCES

[1] Clara Kanmani A, S. Mohan Kumar, and Abhishek Y S S,. "Interrelated Research Works And Importance Of Object Oriented Analysis And Modeling", *ISOR Vol.5 No. 2, .785, ISSN: 2277-965, 2016.*

[2] Mansaf Alam, Siri Krishan Wasan, "Migration from Relational Database into Object Oriented Database," *Journal of Computer Science, Vol. 2, No. 10, pp. 781784,* 2006.

[3] Joseph Fong, "Converting Relational to Object Oriented Databases," *SIGMOD Record 1997, Vol. 26, No. 1,* 1997.

[4] Narendra Pal Singh Rathore1, Ravindra Gupta (). "A Novel Class, Object and Inheritance based Coupling Measure (COICM) to Find Better OOP Paradigm using JAVA", *International Journal of Advanced Computer Research Vol 1 , No1, ISSN : 2249-7277, 2011.*

[5] Roshan .K.Thomas and Ravi .S.Sandhu , " Discretionary access Control in object-Oriented databases: Ssues and research direction", *Proc.ofthe16thNIST-NCSC National Computer Security Conference, Baltimore, pages63-74, September 1993.*

[6] Max J. Egenhofer and Andrew U. Frank, " Object-Oriented Modeling in GIS: Inheritance and Propagation", *Res. National Center for Geographic Information and Analysis and Department of Surveying Engineering University of Maine USA.*

[7] Ajita Satheesh and Ravindra Patel, "Use Of Object-Oriented Concepts In Databases For Effective Mining", *International Journal on Computer Science and Engineering ,Vol.1, No. 3, pp. 206-216, 2009.*

[8] Satchidananda Dehuri, "Genetic Algorithms For MultiCriterion Classification And Clustering In Data Mining", *International journal of computing and information sciences, Vol. 4, No. 3, pp. 143-154, 2006.*

[9] Shermann Sze-Man Chan, and Qing Li, "Supporting Spatio-Temporal Reasoning in an object-Oriented Video Database System", *1999.*

[10] Satchidananda Dehuri, "Genetic Algorithms For MultiCriterion Classification And Clustering In Data Mining", *International journal of computing and information sciences, Vol. 4, No. 3, pp. 143-154, 2006.*

[11] Urban, S.D. and S.W. Dietrich, "Using UML Class Diagrams for a Comparative Analysis of Relational, Object-Oriented, and Object-Relational Database Mappings." *ACM SIGCSE Bulletin. 35(1):21-25, 2003.*

[12] Kelly Nunn-Clark, Lachlan Hunt, Teo Meng Hooi and Balachandran Gnanasekaraiyer, "Problems of Storing Advanced Data Abstraction in Databases," *In Proceedings of the First Australian Undergraduate Students' Computing Conference, pp. 59-64, 2003.*

[13] Cristian Seech et. Al. Object Oriented Modeling of Complex Mechatronic Components for the manufacturing Industry", IEEE/ASME Transactions on Mechatronics Vol. 12 2007, Pg. No. 696

[14] Jun Zhu et. Al. "Application of Design Patterns for Object Oriented Modeling of Power Systems", *IEEE Transactions on Power Systems, Pg. No. 532, 1999.*

[15] *K*aren E. Smith, Stanley B. Zdonii, "Intermedia: A Case Study of the Differences Between Relational and Object-Oriented Database Systems" *OOPSLA '87 Proceedings, ACM O-8979 l-247-0/87/00 lo-0452 $ I .50, 1987.*

[16] Victor T Sarinho et. Al. "Combine Feature Modeling with Object Oriented concepts to manage software viability", *IEEE IRI, Pg. No. 344, 2010.*

[17] Ulrich Frank, "Delegation: An Important Concept for the Appropriate Design of Object Models," *Journal of Object-Oriented Programming, Vol. 13, No. 3, pp. 1318, 2000.*

[18] Joseph Fong, "Converting Relational to ObjectOriented Databases," *SIGMOD Record, Vol. 26, No. 1, 1997.*

[19] Kitsana Waiyamai, Chidchanok Songsiri and Thanawin Rakthanmanon, "Object-Oriented Database Mining: Use of Object Oriented Concepts for Improving Data Classification Technique", *Lecture Notes in Computer Science, Vol: 3036, pp: 303-309, 2004.*

[20] J. Blakeley, "Object-oriented database management systems," *Tutorial at SIGMOD, Minneapolis, MN, May 1994.*

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 509 - 515, 2016, ISSN:- 2319–8656

A Gene Structure Prediction Model using Bayesian Algorithm and the Nearest Neighbor

Elham Naseh
Department of Computer Engineering, Qeshm international Branch, Islamic Azad University
Qeshm ,Iran

*Ali Asghar Safaee
Department of Medical Informatics,
Faculty of Medicine,
Tarbiat Modarres University
Tehran, Iran

Abstract: Basically genetic disorders include general problems and issues that are caused by the failure of one or more of the genome, and usually appear at birth; although they sometimes occur later. Genetic diseases may not be inherited and they may be caused as new mutations in the genome of embryos. Like many other diseases, diagnosis, treatment, and prognosis of genetic diseases is very important and sometimes complex. One of the best ways of treating genetic diseases is its diagnosis in a fetus. All gene structures of a fetus should be available in order to diagnose genetic disease. This structure can be achieved when the fetus is seven months and there are just 5% to 30% of gene sequence structure before seven months. To solve this shortcoming and fix the obstacle in the diagnosis of diseases, 5 to 30% of the gene sequence structure of the whole structure of the fetus is predicted with the help of parents' gene structure. In previous studies, gene structure prediction using machine learning algorithms has achieved a maximum accuracy of 95%.

Keywords: Gene structure prediction, nearest neighbor algorithm, Bayesian algorithm, blended learning methods, genetic disease diagnosis.

1. INTRODUCTION

Genetic diseases are one of the most serious issues in the field of health and medicine and researchers' efforts in related sciences and various fields still continues. Genetic diseases include diseases that are caused by failure or mutations in the genes or genetic material of humans. These diseases often occur at birth, but can also occur years later. Genetic diseases may not be inherited, for example they may be created as a result of new mutations in the genome of the fetus.

If genetic diseases are diagnosed before birth, they are treatable. One of the ways to diagnose genetic disease is based on the availability of fetus' genetic sequence. But the challenge is that only about 10% of genes have been identified in the embryonic time and 90% remains uncertain. In other words, the entire structure of genes is completed at birth, when almost all genetic diseases are no longer curable. Therefore, the remaining uncertain 90% of fetus' genetic structure should be predicted with the help of parents' chromosome structure and 10% of fetus' cell sequence in order to overcome this challenge [8, 9]. As it seems, predicting something that much of it is not available based on a much smaller proportion (predicting remaining 90% based on 10%) is very difficult and acceptable accuracy in the prediction will be of great importance.

To learn more about the issue that we face, some additional details are provided here. Deoxyribonucleic Acid (DNA), discovered in 1869 by Friedrich Miescher [1], is a chemical structure that creates chromosomes. The part of chromosome that has unique characteristics is called a gene.

DNA has a spiral structure and is formed by a double-stranded genetic material, wrapped around each other in a spiral form. Each strand contains a base composition that is called nucleotide. Each basic composition is formed by four structures (adenine A, guanine G, cytosine C, thymine T). Human cells have twenty-three pairs of chromosomes [2, 3].

Human cells inherit two different groups of chromosomes, one group from father and the other from mother. Each group of chromosome is made up of 23 single chromosome (22 asexual chromosomes and one sexual chromosome, either in the form of Y or in the form of X). The group of chromosome that can be seen in Figure 1 is for father's chromosomes (since it has chromosome X (XY) and chromosome Y). If the group of chromosome is for mother, the group will include chromosome X and again chromosome X.

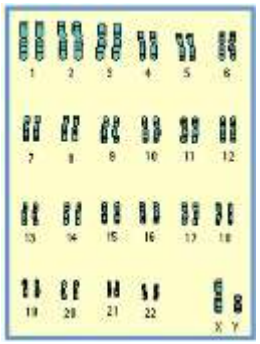


Figure 1: Father's chromosome structure [4]

In principle, every living entity has many attributes of: color, size, horned or polled. These are a few of the traits that are passed from parents to offspring. These traits are controlled by the genes. Genes are small and complex molecules that are on chromosomes. Any existing entity contains several thousand pairs of genes, half of which is inherited from father and the other half is inherited from mother, and these genes together form genotype of that existing entity [5].

Genotype determines the approximate function of an organism on different characteristics. Each pair of genes control some of the characteristics. Some characteristics have a very high heritability, but some others have a very low heritability [6, 7].

In this case, the gene structure of a fetus should be predicted with the help of parents' genes. Gene structure is a series of

DNA coding sequence. DNA has a basis code of two. DNA structures can be seen in Table 1.

Table 1: DNA code structure

Gene name	Equivalent code	Equivalent binary code
Guanine (G)	0	00
Cytosine (C)	1	01
Adenine (A)	2	10
Thymine (T)	3	11

Human beings have a sequence of DNA. An example of sequence can be seen in Figure 2.

DNA code	C	A	C	C	T	T	G	G	C	T	T	C	C
Equivalent code	1	2	1	1	3	3	0	0	1	3	3	1	1
Equivalent binary code	01	10	01	01	11	11	00	00	01	11	11	01	01

Figure 2: Gene sequence structure

Gene sequence of a fetus for any DNA is one of the following three conditions:

• Father's code has been copied.

• Mother's code has been copied.

• None of mother's or father's code; a mutation has occurred in this case.

In Figure 3, the gene structure of a fetus, father, and mother has been shown using DNA code.

	1	2	3	4	5	6	7	8	9	10	11	12	13
Father's DNA code	C	A	C	C	T	T	G	G	C	T	T	C	C
Mother's DNA code	A	A	C	T	T	A	T	G	A	C	G	C	T
Fetus' DNA code	A	A	C	C	T	T	G	G	A	A	T	C	C

Figure 3: Fetus' DNA structure based on parents' DNA

As can be seen in Figure 3, each DNA contains 13 gene sequence. Each gene sequences that include father, mother, and fetus' code is called the Trio. In Figure 3, 13 Trio can be seen, In Trios one to three, father's DNA code is copied, mother's DNA code is copied in Trios eight to nine, gene mutation has occurred in Trio ten, and father's DNA is copied in Trios eleven to thirteen.

The change of DNA code from father to mother and vice versa is called crossover. Crossover has occurred in trios three to four, seven to eight, nine to ten, and ten to eleven.

Up to 5 to 30% of a fetus' DNA codes are determined up to 7 month and the remaining DNA codes are changing. An example of codes excluded from DNA structure can be seen in Figure 4.

	1	2	3	4	5	6	7	8	9	10	11	12	13
Father's DNA code	01	10	01	01	11	11	00	00	01	11	11	01	01
Mother's DNA code	10	10	01	11	11	10	11	00	10	01	00	01	11
Fetus' DNA code	10	?	?	?	?	11	?	?	?	?	?	?	01

Figure 4: Equivalent binary code of sequence structure of fetus and father and mother in figure 3 that just 3 trios are determined

As can be seen in Figure 4, only 3 Trios of the fetus have been found, but fetus' parents' codes are all determined. Now all fetus' codes should be identified with the help of prediction models (such as the proposed algorithm).

The rest of the article would be as follows: the research papers and related work are introduced in section 2. The proposed model to predict gene structure is presented in section 3 and section 4 empirically evaluates the proposed model. Conclusions and recommendations for future studies are provided in sections 5 and 6.

2. RESEARCH BACKGROUND

In a study conducted by Rutkoski et al. (2013) entitled as "imputation of unordered markers and the impact on genomic selection accuracy", it was shown that genomic selection is a breeding method to accelerate the rate gene gain. In this study, using four empirical datasets, four imputation methods including k-nearest neighbors, singular value decomposition, random forest regression, and expectation maximization imputation were evaluated and characterized in terms of their imputation accuracies and the factors affecting accuracy. It was shown that SMV has a high accuracy of 93%.

In 91 years, Mrs. Maryam Ali in his thesis on DNA and predicted protein structure, and neuro-genetic algorithm used to predict [10], the results show that these methods increase the overall accuracy, quality improvement predict and solve the problem of unbalanced data, the prediction accuracy of 92% have been reported in this study.

Alireza (2012) predicted protein structure on DNA using Neuro-genetic algorithm in her thesis [10]. The results show that these methods increase the overall accuracy, improve prediction quality, and solve the problem of unbalanced data. The prediction accuracy has been reported to be 92% in this study.

In this study, we set out to achieve higher accuracy in the proposed algorithm; therefore, a combination of Bayesian algorithm and nearest neighbor has been used.

Bayesian algorithm is a statistical method that is based on the conditional probability, i.e. by classifying we seek to define the class of a new sample, which is the main goal of classification [17].

In this method, which is based on probability and statistical methods, it is found that how probable a sample belongs to a specific class and it is finally concluded that which class does the sample belong based on the highest obtained probability. Finding out the probability is the main challenge of Bayesian-based methods [18].

K-nearest neighbor learning algorithm is one of the most famous algorithms in the field of learning, the performance and features of which will be discussed [10].

In sample-based methods or sample-based learnings, classification of a new sample is done in a way that the new sample is compared with all the samples of a class and then the samples that are more similar to the new sample are extracted as the ones with the potential to have the same class with the new sample and then the class of the new sample is determined on the basis of some methods.

In the nearest neighbor, K of the nearest neighbors are determined from the training set. Now if K is equal to one, one of the nearest neighbors is extracted for the new sample and its class is investigated. The value of the nearest neighbor will be the same as the class of the new sample.

In [10], 90% of gene structure is predicted using a combination of two algorithms of support vector machine and neural networks, and structure of the parents' chromosomes as well as the 10% of the fetus' cell sequence. Support vector

machine algorithm has a good performance in dealing with linear problems and artificial neural network is useful for nonlinear problems. Thus, since predicting 90% of the structure is both linear and non-linear, the combination of two algorithms was used [10].

The proposed model to predict gene structure is presented in the next chapter and them its performance will be evaluated.

3. THE PROPOSED MODEL TO PREDICT GENE STRUCTURE

In order to provide the proposed prediction model, the problem should be first categorized using the nearest neighbor and Bayesian approach, then a new method is provided to combine two nearest neighbor method and the Bayesian approach to solve the problem, which will be explained in part 3 -3 in detail.

In predicting fetus' gene structure, the data, in principle, contains parents' and fetus' features that should be predicted with the help of parents' information, fetus' status, and his genetic structure. In order to achieve a category with a better accuracy with both methods, features of the table number are added to the information. Features of table number starts from 1 and has a rising rate. The reason for adding features of the table number is to determine the difference between each row. For example, first row is closer to the second row than the tenth row, because the difference between first row and tenth row is ten units, but the difference between the first row and the second row is two units. The information by adding the numbers of the table can be seen in Figure 5.

No.	Father	Mother	(fetus) Class
1	01	10	**10**
2	10	10	**10**
3	01	01	**01**
4	01	11	**01**
5	11	11	**11**
6	11	10	**11**
7	00	11	**00**
8	00	00	**00**
9	01	10	**10**
10	11	01	**10**
11	11	00	**11**
12	01	01	**01**
13	01	11	**01**

Figure 5: Adding features of number to each Trio

3.1 Prediction model based on the nearest neighbor

Since in this method values are binary, there is no need for pre-processing such as dimension reduction or numbers' transformation. To implement the nearest neighbor algorithm, the following steps are implemented:

1. First, the data are divided into two sets.

   1.1. The data set that its class label is determined (the training data set).

   1.2. The data set that its class label is not determined (the test data set).

2. The following steps are taken for each of the unlabeled data.

   2.2 Euclidean distance of each data from all labeled data is calculated.

   2.2. K nearest distance is selected as the most similar.

3. The class label is determined from the most similar Trios using majority voting rule.

4. The accuracy of the labels specified in Step 2 is calculated.

5. Algorithm runs from 1 to 10 for K, and the best value of K in terms of accuracy is determined as output.

Dividing data into two training and test sets is shown in Figure 6. Then the Euclidean distance of each test Trio from training Trio is calculated. Given that the labeled class data is discrete, majority voting method is used to select K-nearest training data and then test data class label is determined using voting method.

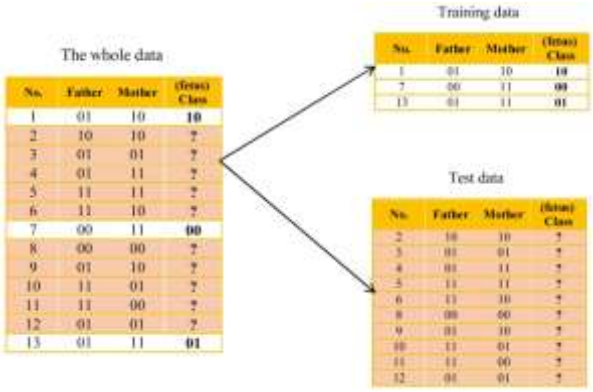


Figure 6: Dividing data into two training and test sets

3.2 Prediction Model Based on Bayesian Method

The issue of detecting fetus has four different classes. The following steps are taken to run the algorithm:

1. First, data is divided into two sets.

   1.1. The data set that its class label is determined.

   1.2. The data set that its class label is not determined.

2. The following steps are taken for each of the unlabeled data.

   2.1. The extent to which data belongs to each one of the four classes is calculated.

   2.2 The maximum amount is selected as the label.

3. The accuracy of the labels specified in Step 2 is calculated.

The first step in Bayesian algorithm is shown in Figure 6. The following equation is used to calculate the value of four classes.

$$\begin{cases} P(00|X) = P(x_1|00) \times P(x_2|00) \times P(x_3|00) \times P(00) \\ P(01|X) = P(x_1|01) \times P(x_2|01) \times P(x_3|01) \times P(01) \\ P(10|X) = P(x_1|10) \times P(x_2|10) \times P(x_3|10) \times P(10) \\ P(11|X) = P(x_1|11) \times P(x_2|11) \times P(x_3|11) \times P(11) \end{cases}$$

Then the probability of each above relationships is calculate and the highest probability is selected as the label.

3.3 Hybrid model

In order to combine the two Bayesian and the nearest neighbor methods to achieve a higher accuracy than either methods, a hybrid model is presented as follows.

The proposed algorithm is in a way that data is first divided into two training and test sets. Then, the nearest neighbor and Bayesian algorithms are implemented. The outputs of the test data are investigated and if both algorithms have the same outputs, the set is added to the training data, but if the two algorithm does not have the same output, the set will remain in the test data. Then both algorithms are run with new data of training set and the previous trend is done for test data. This procedure is repeated so that one of the two following states happens.

• Test data will end.

• No element of the training set is added to the test set.

If the first case occurs, then the algorithm has completed and the initial test set is evaluated. But if the latter occurs, the nearest neighbor method will be chosen as the output. The nearest neighbor method is chosen because the accuracy of the nearest neighbor method is higher than the Bayesian approach.

The proposed algorithm is shown in Figure 7.

```
Function Accuracy=Classification(input =Gen)
    1.  Cross Validation Gen To Training & Test
    2.  While (True)
    3.      X ← Classify Bayesian (Training)
    4.      Y ← Classify KNN (Training)
    5.      A ← Evaluation X (Test)
    6.      B ← Evaluation Y (Test)
    7.      If (The number of outputs A and B are the same)
    8.          The same output will be removed from the test set and the
                training set is added.
    9.      Else
    10.         Exit While
    11.     End IF
    12. End While
    13. If No output is not same
    14.     Classify KNN (Training)
    15. End if
    16.     Accuracy ← Evaluation (Test)
End
```

Figure 7: Pseudo code of the proposed algorithm

As can be seen in figure 7, data is divided into two training and test sets in the first line of the proposed pseudo code without any pre-processing, because the data is binary and there is no need for transformation. That is, data is complete and there is no missing or extreme data. It should be noted that data has four features of No, father's code, mother's code, and fetus' code (Class Label) and there is no need for dimension reduction.

Lines 2 to 12 are the main loops of this program. Two new models are developed using the nearest neighbor and Bayesian approach in lines 3 and 4 of the training data and then these two models are evaluated in lines 5 and 6 using test data. Finally, in line 7 it will be checked whether the answer to the two models is the same or not?

If a part of the test elements is similar, then line 8 is run. At this stage, a number of test data elements that have the same answer are removed from the test data sets and added to the training data set. However, if all the elements are the same or none of the elements are the same, the program exits the main loop on line 10.

If the output of all the test data is not the same in both the nearest neighbor and the Bayesian approaches in lines 13 to 15, the nearest neighbor is taken into account and then the accuracy of test data is sent as output in line 16.

4. EXPERIMENTAL EVALUATION

One of the most important parts of a theory is doing the experiments and proving their results. In order to test the proposed theory, some programs were created using MATLAB that will be explained and displayed here. The experiments were conducted on a computer with a 4GH processor and 6GB ram. The real data has been used to test data, which was collected from medical sciences of Mashhad and included 2 million records from parents' and fetus' gene. The data was stored on a 2 basis and a view of the data is shown in Figure 8.

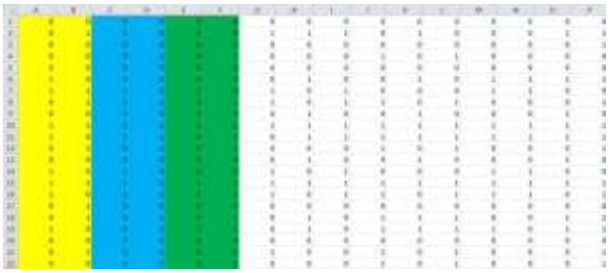


Figure 8: gene sequence data to evaluate the proposed algorithm

As shown in Figure 8, the first two columns are father's code, the second two columns are mother's code, and the last two columns are fetus' code. The missing data of fetus' gene is a number between 70 to 95%; therefore, 70 to 95% of the data must be removed.

The four parameters of Confusion Matrix, Sensitivity, Specificity and Accuracy are examined, then the time for their implementation will also be examined. The output can be seen in Table 1.

Confusion Matrix shows the performance of the algorithms. Usually this performance is used for algorithms such as decision tree. Each column of the matrix shows an example of the predicted value. If each row has an actual (right) sample, the structure of Confusion Matrix is seen in Figure 9.

		Predicted Class: Yes	Predicted Class: No
Actual Class	Yes	TP	FN
	No	FP	TN

Figure 9: Confusion Matrix structure

The formulas for Sensitivity, Specificity, and Accuracy are calculated with the help of Confusion Matrix that can be seen in equations (1), (2), and (3).

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (1)$$

$$\text{Specificity} = \frac{TN}{FP+TN} \quad (2)$$

$$\text{Accuracy} = \frac{TN+TP}{FP+TN+TP+FN} \quad (3)$$

Missing value of data ranges between 70 and 95% [1]. If we call the missing value K, K% of data should first be deleted randomly. Then, both the nearest neighbor algorithm and

Bayesian algorithm are run once, and then confusion matrix is calculated for both algorithms. With the help of confusion matrix, the three parameters of Sensitivity, Specificity and Accuracy are calculated. With the exception of the confusion matrix, execution time of both algorithms is calculated. As can be seen in Table 2, the nearest neighbor algorithm has a higher accuracy than Bayesian algorithm, but the nearest neighbor algorithm's execution time is three times more than Bayesian algorithm.

4.1 Comparison of the proposed hybrid algorithm and the nearest neighbor approach

Given that the nearest neighbor algorithm has better accuracy than the Bayesian approach; therefore, the proposed algorithm has been evaluated with the nearest neighbor method that can be seen in Table 3 and Figure 12. As can be seen in Table 3, the measure of the accuracy of the proposed algorithm has been better than the nearest neighbor algorithm. In all the missing values, the accuracy has increased; the minimum accuracy increase was in 73% of missing values in a way that accuracy had increased .17% and the maximum accuracy increase was in 95% of missing values in a way that accuracy had increased 1.68%. On average, an increase of .7% has happened.

Table 3: Comparison of the nearest neighbor algorithm and the proposed algorithm based on accuracy

ID	Missing value	Nearest Neighbor	proposed algorithm
		Accuracy	Accuracy
1	70	98.00%	98.54%
2	71	98.03%	98.32%
3	72	97.78%	98.12%
4	73	97.81%	97.89%
5	74	97.57%	97.93%
6	75	97.60%	97.91%
7	76	97.50%	97.90%
8	77	97.41%	97.81%
9	78	97.19%	97.80%
10	79	97.22%	97.73%
11	80	97.00%	97.62%
12	81	97.04%	97.53%
13	82	96.83%	97.34%
14	83	96.63%	97.32%
15	84	96.42%	97.31%
16	85	96.24%	97.01%
17	86	96.04%	96.86%
18	87	95.75%	96.74%
19	88	95.46%	96.49%
20	89	95.17%	96.12%
21	90	94.78%	95.32%
22	91	94.30%	95.31%
23	92	93.82%	95.07%
24	93	93.33%	94.24%
25	94	92.55%	93.72%
26	95	91.85%	93.53%

5. CONCLUSION AND SUGGESTIONS FOR FURTHER STUDIES

Gene structure prediction is one of the most important ways to diagnose genetic diseases. These diseases often occur at birth, but can also occur years later. Genetic diseases may not be inherited, for example they may be created as a result of new mutations in the genome of the fetus. In this study, it was shown that 95% of missed genes is predictable. It was also found that a model composed of the nearest neighbor algorithm and Bayesian algorithm is used to predict missing data of genes.

The proposed model is a combination of the nearest neighbor algorithm and Bayesian algorithm. The structure of this algorithm is based on voting and sameness of the two algorithms' output and it also has a higher accuracy than the nearest neighbor algorithm and the Bayesian algorithm.

In the proposed nearest neighbor algorithm, it was showed that the algorithm has high capability in solving this problem, and it was also revealed that the implementation of this algorithm is very time consuming, but has a high accuracy.

Bioinformatics algorithms must be appropriate in terms of Sensitivity and Specificity. If the value of one of these two parameters is low, the algorithm is not acceptable. In the present study, it was shown that a combination of several algorithms can be used to optimize both parameters.

Past methods have an accuracy of about 90% through which lots of diseases were predictable, but the proposed algorithm has reached an accuracy of 98%, which is more accurate than other methods. Therefore, it can give a better prediction in the structure of the gene sequence.

To continue the work done in identifying gene sequences, some suggestions are put forward:

• Parents' gene sequence was only used in the proposed system, but if the gene sequence of grandparents or siblings is used, better results can be achieved.

• Non-linear algorithms such as decision tree can be used instead of Bayesian algorithm.

• Reinforcement learning algorithms can be used to adjust the algorithm parameters (such as the K value in the nearest neighbor).

• The use of neuro-fuzzy networks can probably bring about interesting results in diagnosis.

6. REFERENCES

[1] Langman's Medical Embryology, Thirteenth, North American Edition edition. Philadelphia: LWW, 2014.

[2] Larsen's Human Embryology, 5e, 5 edition. Philadelphia, PA: Churchill Livingstone, 2014.

[3] BRS Embryology, Sixth edition. LWW, 2014.

[4] High-Yield Embryology, Fifth edition. Philadelphia: LWW, 2013.

[5] The Developing Human: Clinically Oriented Embryology, 10e, 10 edition. Philadelphia, PA: Saunders, 2015.

[6] Developmental Biology, Tenth Edition, 10 edition. Sunderland, MA, USA: Sinauer Associates, Inc., 2013.

[7] J. E. Rutkoski, J. Poland, J.-L. Jannink, and M. E. Sorrells, "Imputation of unordered markers and the impact on genomic selection accuracy," G3 Genes Genomes Genet., vol. 3, no. 3, pp. 427–439, 2013.

[8] H. Pang, M. Hauser, and S. Minvielle, "Pathway-based identification of SNPs predictive of survival," Eur. J. Hum. Genet., vol. 19, no. 6, pp. 704–709, 2011.

[9] J. He, J. Zhang, G. Altun, A. Zelikovsky, and Y. Zhang, "Haplotype tagging using support vector machines.," in GrC, 2006, pp. 758–761.

[10] D. Gutierrez, Machine Learning and Data Science: An Introduction to Statistical Learning Methods with R, First edition. Technics Publications, 2015.

[11] Machine Learning in Action, 1 edition. Shelter Island, N.Y: Manning Publications, 2012.

[12] M. ACI, DEVELOPMENT OF TWO HYBRID CLASSIFICATION METHODS FOR MACHINE LEARNING: Using Bayesian, K Nearest Neighbor Methods and Genetic Algorithm. LAP LAMBERT Academic Publishing, 2011.

[13] J. Kleissl, Solar Energy Forecasting and Resource Assessment, 1 edition. Academic Press, 2013.

[14] Doing Data Science: Straight Talk from the Frontline, 1 edition. Beijing ; Sebastopol: O'Reilly Media, 2013.

[15] "Data Mining in R - Learning with Case Studies" taught by Dr. Luis Torgo - 4 week online course - 6/28/13 to 7/26/13. .

[16] Machine Learning, 1 edition. New York: McGraw-Hill Education, 1997.

[17] P. Bessiere, E. Mazer, J. M. Ahuactzin, and K. Mekhnacha, Bayesian Programming, 1 edition. CRC Press, 2013.

[18] M.-H. Chen, L. Kuo, and P. O. Lewis, Eds., Bayesian Phylogenetics: Methods, Algorithms, and Applications. CRC Press, 2014.

[19] P. Domingos, The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World. Basic Books, 2015.

[20] Pattern Recognition and Machine Learning. New York: Springer, 2007.

[21] A First Course in Bayesian Statistical Methods, 1st ed. 2009 edition. London ; New York: Springer, 2009.

[22] Bayesian Analysis for the Social Sciences, 1 edition. Chichester, U.K: Wiley, 2009.

[23] Machine Learning for Healthcare, 1 edition. O'Reilly Media, 2015.

[24] Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference, 1 edition. New York: Addison-Wesley Professional, 2015.

[25] Bayesian Estimation and Tracking: A Practical Guide, 1 edition. Hoboken, N.J: Wiley, 2012.

[26] Bayesian Reasoning and Machine Learning, 1 edition. Cambridge ; New York: Cambridge University Press, 2012.

Table 2: Comparison of the two the nearest neighbor and Bayesian algorithms

		The nearest neighbor algorithm						Bayesian algorithm					
Number	**Missing value**	Confusion Matrix		Sensitivity	Specificity	Accuracy	Time/sec.	Confusion Matrix		Sensitivity	Specificity	Accuracy	Time/sec.
1	70	434	7	98.00%	97.99%	98.00%	55.762006	302	49	97.78%	98.02%	97.90%	2.083503
		7	342					48	300				
2	71	348	7	98.03%	98.02%	98.03%	55.251918	307	49	97.78%	98.02%	97.90%	2.022653
		7	347					49	305				
3	72	352	8	97.78%	97.78%	97.78%	56.160841	311	51	98.09%	97.74%	97.92%	1.895628
		8	352					49	309				
4	73	357	8	97.81%	97.80%	97.81%	57.503536	315	51	97.57%	97.49%	97.53%	1.862098
		8	356					50	313				
5	74	362	9	97.57%	97.57%	97.57%	59.087744	320	52	97.59%	97.34%	97.46%	1.749257
		9	361					51	318				
6	75	366	9	97.60%	97.60%	97.60%	55.781443	323	52	97.59%	97.34%	97.46%	1.911674
		9	366					62	322				
7	76	371	10	97.63%	97.37%	97.50%	55.171607	328	53	97.35%	97.12%	97.24%	1.927079
		9	370					53	326				
8	77	376	10	97.41%	97.40%	97.41%	55.587305	331	53	97.40%	97.71%	97.55%	1.884540
		10	375					55	332				
9	78	381	11	97.19%	97.18%	97.19%	55.481798	336	54	97.15%	97.46%	97.31%	1.749130
		11	379					55	335				
10	79	385	11	97.22%	97.22%	97.22%	55.030003	341	55	97.51%	97.74%	97.62%	1.726044
		11	384					55	340				
11	80	389	12	97.01%	97.00%	97.00%	54.942187	345	56	97.51%	97.74%	97.62%	1/948337
		12	388					56	344				
12	81	394	12	97.04%	93.03%	97.04%	54.555110	349	57	96.85%	96.30%	96.58%	1.566272
		22	392					57	348				
13	82	397	13	96.83%	96.82%	96.83%	54.143132	353	58	96.84%	96.81%	96.83%	1.589120
		13	396					57	352				
14	83	402	13	96.63%	96.62%	96.63%	54.337591	357	58	96.60%	96.45%	96.52%	1.513309

		14	400					58	356				
15	84	406	15	96.44%	96.42%	96.43%	54.253094	361	59	96.14%	96.01%	96.07%	2.011353
		15	404					60	361				
16	85	410	16	96.24%	96.23%	96.25%	53.734557	366	60	96.47%	96.47%	96.47%	1.459371
		16	408					60	364				
17	86	413	17	96.05%	96.04%	96.04%	53.359145	370	61	96.32%	96.00%	96.16%	1.273876
		17	412					60	368				
18	87	417	18	95.64%	95.85%	95.75%	59.260171	374	62	95.19%	95.85%	95.52%	1.519282
		19	416					62	373				
19	88	421	20	95.46%	95.45%	95.46%	55.870253	378	62	95.02%	95.68%	95.25%	1.367650
		20	429					63	377				
20	89	425	22	95.29%	95.06%	95.17%	56.284392	382	64	95.23%	95.56%	95.39%	1.246092
		21	423					64	381				
21	90	428	24	94.90%	94.67%	94.78%	50.550362	386	64	95.24%	95.33%	95.34%	1.336561
		23	246					65	386				
22	91	431	26	94.31%	94.29%	94.30%	50.081723	390	66	94.30%	94.27%	94.29%	1.076476
		26	429					66	389				
23	92	433	28	93.72%	93.91%	93.82%	54.954328	393	67	93.32%	94.96%	94.13%	0.935527
		29	432					68	393				
24	93	435	31	93.35%	93.32%	93.33%	57.330422	397	69	93.36%	92.04%	92.70%	0.898224
		31	433					69	396				
25	94	436	35	92.57%	92.54%	92.55%	59.419878	400	70	92.75%	91.95%	92.35%	0.792762
		35	434					71	399				
26	95	436	40	91.79%	91.58%	91.85%	56.490077	403	72	92.05%	92.75%	91.90%	0.714929
		39	435					72	402				

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 516 - 518, 2016, ISSN:- 2319–8656

The Presentation of a Genetic Algorithm to Solve Steiner Tree

Elham Naseh
Department of Computer Engineering, Qeshm international Branch, Islamic Azad University
Qeshm ,Iran

*Ali Asghar Safaee
Department of Medical Informatics,
Faculty of Medicine,
Tarbiat Modarres University
Tehran, Iran

Abstract: The problem of Minimal Steiner Tree is a classical and known one (NP-Complete). This problem has been used a lot in navigation of Networks. Since finding Minimal Steiner Tree is NP-Complete, there's no Algorithm for it in multi-nominal time and we should use other Algorithms like Approximate Algorithm or Random Algorithm. This study presents a new genetic Algorithm. In the conclusion, this proposed Algorithm will be evaluated.

Keywords: Navigation, Steiner Tree, NP-Complete, Genetic Algorithm

1. INTRODUCTION

The problem of Minimal Steiner Tree is a classical and known problem (NP-Complete). The directionless graph G(V,E) is given; the change of each Edge crest is non-negative and its vertexes are divided into two parts: A set of required vertexes and Steiner's vertexes .The purpose is to find the least expensive tree in G in a way that it involves all the vertexes of the required set and a subset of Steiner's vertexes .The problem of Steiner tree is defined as the following:

Graph , and a set of vertexes are given. The purpose is to find a tree in G with the least expenses, which includes all the vertexes of R set. Finding a tree with less expenses causes increase in the speed of navigation of computer Networks. [1]

If we consider figure 1 as the entrance graph and R, including colorful vertexes and MST Algorithm on R (without considering the vertex which is drawn hollow) is performed, it gives the tree in figure 2, while the optimum Steiner Tree will be tree number 3.

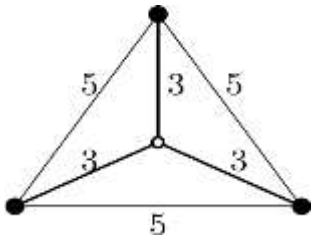


Figure 1: graphic example with three terminal vertexes and Steiner vertex [2]

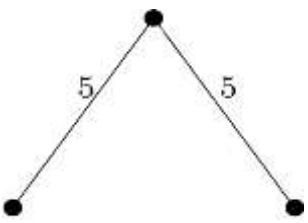


Figure 2, the created tree by Algorithm tree [3]

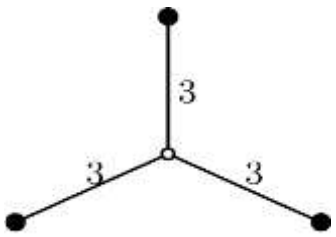


Figure 3: Steiner Tree

2. THE STAGES OF GENETIC ALGORITHM

In this article, genetic Algorithm is used to solve Steiner Tree whose structure is as follows [5]:

2.1 Displaying Gene Solution

The first step is a way to display the solution or Gene. In most applications, a range of bits is used. A range of zero and one (0 and 1) with the length of |V|; i.e. the number of vertexes of graph G. Each bit in the range is equal to one vertex of graph. Whenever it is the ith in the range, it means the ith vertex in graph or answer graph or Steiner Tree doesn't exist. For example, look at example 4. The Corresponding range for figure 5 is 11111110. It means vertexes 1 to 7 exist in answer graph, but vertex 8 doesn't. [6]

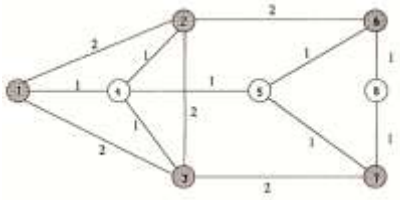


Figure 4: Graph G with 8 vertexes (5 Terminal vertexes and 3 Steiner vertexes) [7]

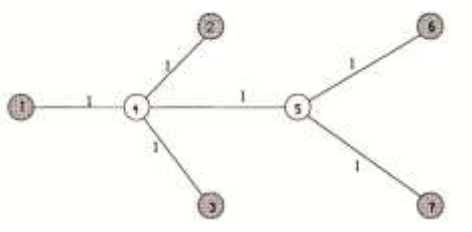


Figure 5: Steiner Tree for graph 4[7]

Not all the genes created by this method are a part of the solution to the problem. The genes in which the bits corresponding with terminal vertexes are zero (0) are not a part of the answer. These genes mustn't remain in the genetic reservoir, and they are omitted.

2.2 The Primary Generation (Population)

The second step is to create a primary population. To create a population, based on the experiments that are done, different techniques can be used. In action, first the population is created randomly and in the second stage, this randomly created population along with a possible answer that is equal to minimum Covered tree is placed for the whole graph; i.e. a whole range of 1 [8]. In the third stage, a detected answer from the remaining of the answer of minimum Covered Tree in the second stage is derived. In this article, 50% of the population is used in the first method and 50% in the third method.

After this stage, the number of population must be determined. It means how many chromosomes must exist in the gene reservoir. The number of population is considered as on important factor in Algorithm efficiency. If the population is too small, a small portion of the answer space will be searched for and the answer will probably be converted into a local optimum, and if the population is too large, a lot of calculations will be required, which is imbalanced in relation to the gained answer; therefore, performance time will be too long. In this article, the number of population is considered 100.

2.3 Fitness Function Calculation

Fitness function evaluates the suitability and performance of every member of the population. To solve the Steiner Tree problem, a function must also be defined to measure the rate of suitability of the gained answers; i.e. chromosomes. To measure the suitability of chromosome, a sub-graph of G is formed based on the combination of Terminal vertexes and Steiner vertexes – the Corresponding bit in the mentioned chromosome is 1. It is supposed that this sub-graph has K components. Minimum Covered Tree for each component is calculated by prim Algorithm and the total weight of all trees is considered as the rate of suitability. If $k \geq 2$, then a big punishment is considered for that chromosome.

2.4 Selection Methods

To choose the parents, we use Ranked Base Selection Method .The reason of this choice is that in the beginning of Algorithm; we prevent hasty convergence of the Algorithm. (The exploration of capability is carried out in Algorithm.) After several generations, the Algorithm must reach convergence. (The number of generations is gained by Trial and Error Software.) To do this, selection Algorithm is switched to Roulette Wheel Algorithm.

2.5 Crossover Combination

For fertilization in Ranked Base Selection, we use Uniform Crossover whose structure is as the following[9]:

In the open operation of Smooth Combination, the number of gene in the baby is chosen according to number of Corresponding gene from both parents. In this method, the number of genes from each parent has an equal chance of presence in the baby's Corresponding gene. In the open operation of Smooth combination, based on a random binary distribution, we can recognize according to which Corresponding gene of which parent, the number of the baby's genes must be chosen. One example is given in figure 6.

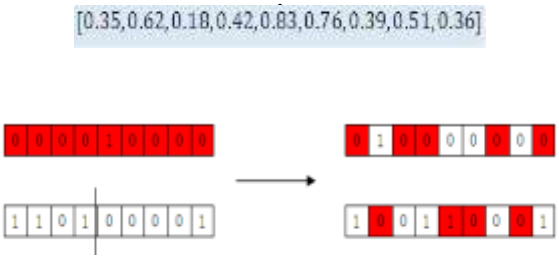


Figure 6: the presentation of an example of Smooth Combination

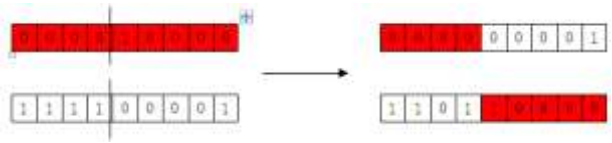
In Roulette Wheel Selection Method, we use one-point crossover whose structure is as the following:

In the operation of one-point Crossover, first a random point in the sequence of the parents' chromosomes is chosen, and then from the selected point, the chromosomes of both parents are sectioned. The second child consists of the first section of the second parent and the second section of the first parent. One example is in figure 7.

Figure 7: the presentation of an example of one-point crossover

2.6 Mutation

In the Mutation Operation with a change in the bit, one



gene is randomly selected. Whatever its amount is , it changes. Its structure is shown in figure 8.It's obvious that because of not utilizing the existing information in the population, this operation has complete congruity with the definition of Mutation operation, and it tried to explore the Algorithm[10].

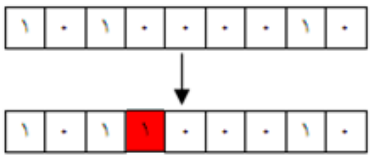


Figure 8- The presentation of an example of Mutation

In the beginning of Algorithm, the percentage of mutation (pm) must be raised so that the rate of exploration of Algorithm will be large. Then after several generations, Pm must be lowered so that it will keep up with the convergence of chromosomes.

2.7 Replacement

Here, we use generational replacement. In the beginning of the Algorithm, 50% of parents and 50% of children are transferred to the next generation by Generational Replacement, which then causes exploration in the problem. After each generation, the percentage of children will decrease

and that of parents will increase so that it will end in convergence. The rate of percentage increase of parents is calculated by Trial and Error Method[11].

3. EVALUATION OF THE PRESENTED ALGORITHM

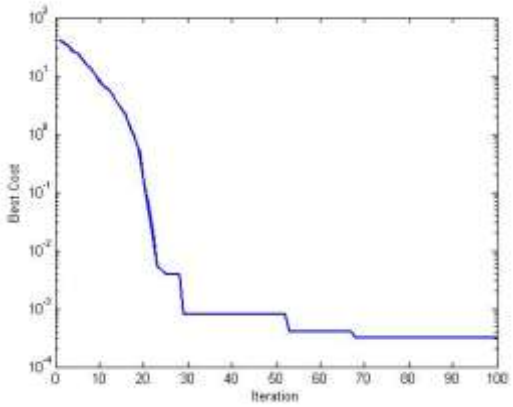


Figure 9: Algorithm with five Steiner nodes and production of 100 generations

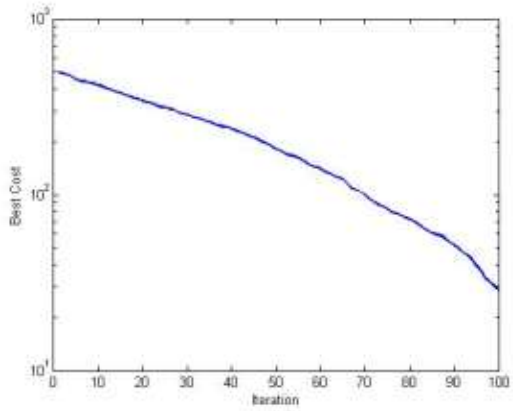


Figure 10: Algorithm with 20 Steiner nodes and production of 100 generations

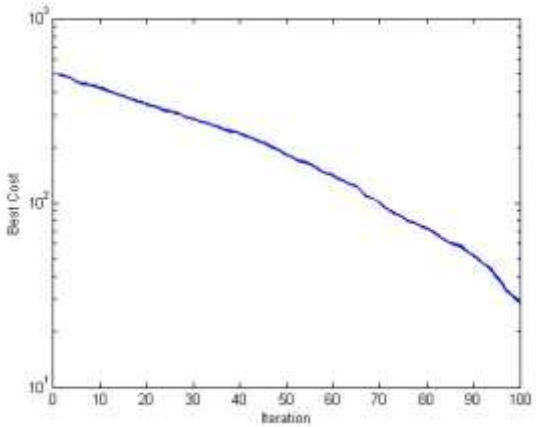


Figure 11: Algorithm with 5 Steiner nodes and production of 500 generations

4. Conclusion

In this study, using Genetic Algorithm, a solution was presented to solve the problem of Steiner Tree, which has reached optimum solution after 500 generations.

5. REFERENCES

[1] Richard M. Karp (1972). "Reducibility Among Combinatorial Problems". In R. E. Miller and J. W. Thatcher (editors).Complexity of Computer Computations. New York.

[2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. (2011),"Introduction to Algorithms", Second Edition. MIT Press and McGraw-Hill.

[3] Vazirani, Vijay V. (2003). Approximation Algorithms. Berlin: Springer.

[4] vafader, Safar (2010) "Approximation Algorithms for the Steiner Tree", Master of Science, Tehran University.

[5] Shirazi, Hussein, Behcet, Shahab (2007) "Introduction to the Analysis and Design of Algorithms" First Edition, Press Malek-ashtar.

[6] P. Berman and V. Ramaiyer, Improved approximations for the Steiner tree problem J. Algorithms 17 . 1994 , 381]408.

[7] Neil Moore, Computational complexity of the problem of tree generation under fine-grained access control policies, information and Computation, Volume 209, Issue 3, March 2011, Pages 548-567.

[8] AZ Zelikovsky An 11/6-approximation algorithm for the network steiner problem , Algorithmica, 1993 – Springer.

[9] L. Kou, G. Markowsky, and L. Berman, “A fast algorithm for Steiner trees,” Acta Informatica no. 15, pp. 141–145, 1981.

[10] M. Doar and I. Leslie, “How bad is naive multicast routing?” IEEE INFOCOM’ 1993, pp. 82–89,

[11] D.E. Goldberg ,1988,. Genttic Algorithms in Search, Optimization MachinLearning. Addision Wealey, New York.

[12] L Luyet, S Varone, N Zufferey , "An ant algorithm for the steiner tree problem in graphs" , Springer-Verlag Berlin Heidelberg 2007

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 519 - 524, 2016, ISSN:- 2319–8656

A Multi-Objective Optimization Algorithm to Predict Information in Mobile Databases

Leila Alaei Sheini
Department of Computer Engineer, Khomein Branch, Islamic Azad University
Khomein, Iran

*Hamid Paygozarh
Department of Computer Engineer, Khomein Branch, Islamic Azad University
Khomein, Iran

Mohammad Khalily Dermany
Department of Computer Engineer, Khomein Branch, Islamic Azad University
Khomein, Iran

Abstract: Recent advances in wireless data networking, satellite services and cell connections allow us to take advantage of mobile computing systems. Here, a problem is the mobility of computing process which needs to get information from databases. In fact, system movement during the transaction interrupts the mobile network connection to server wireless systems, and the disconnection causes failing in the running transaction so the mobile system has to run the transaction again. One way to avoid transaction failure is doing prediction process until connecting to the next wireless network. The objective of this study is to use particle swarm optimization (PSO) algorithm in order to provide a solution for prediction of the information sent to the mobile system for using the memory. Generally, PSO algorithms result in the best or an acceptable solution for optimized selection and design in many scientific and technological problems. Here, the objective of PSO is to select the best information to send to the mobile system when the connection is interrupted in order to prevent transaction failures. This study focuses only on information sent to the mobile system.

Keywords: mobile database, cache operations, mobile customer, cell connection, particle swarm optimization, multi-objective.

1. INTRODUCTION

Recent technical advances in portable computers as well as the wireless technology led to the emergence of portable computers with wireless data networking. Such systems allow users to act in distributed computing environments even if moving. Mobile users needs access to private or cooperative databases stored in mobile or fixed hosts and provide queries and updates of information for wired and wireless networks [1].

Such services include constraints such as limited power supply in mobile system, expensive services, limited bandwidth in mobile systems, and limited service area. Limited service area causes interruption in mobile network connection to server wireless systems which leads to request rejection in the running transaction. This makes the system rerun the process which is costly and time-consuming [2].

To avoid request rejection, an approach is to use cache operation in the mobile system. When receiving a request from the mobile system, mobile service provider predicts the information surplus to what is required by the request and sends it to the mobile system. The mobile system stores the surplus information into cache memory. If the data requested by the current or the next transaction is available in the cache memory, the mobile system will use it. In this way, there is no need to a connection to send the request and receive information. This prevents the failure of the running transaction while reducing costs and response time. The most important issue in this approach is accurate and optimized prediction of data which requires forecasting information available on server database. To this end, this paper takes advantages of the multi-objective particle swarm optimization algorithm [4].

The rest of this paper is structured as follows: Section 2 defines the mobile database; Section 3 studies PSO algorithm; the proposed algorithm is provided in Section 4; and the proposed algorithm is evaluated in Section 5.

2. MOBILE DATABASE

Traditionally, we had large-scale commercial databases that were developed as centralized database systems. However, this trend changed as more and more distributed applications started to emerge. Distributed database applications usually involved a strong central databases and powerful network administrations. However, trends of new technologies experienced changes due to emerging technologies such as:

- The notebook and laptop computers are being used increasingly among the Business Community
- The development and availability of a relatively low-cost availability of wireless digital communication infrastructure.

The rapid development of wireless communication and computer miniaturization technologies enables users to use computer resources anywhere on the computer network. For example, you can connect the World Wide Web even in-flight. Mobile database is the database that allows the development and deployment of database applications for handheld devices, thus, enabling relational database based applications in the hands of mobile workers. Mobile database technology allows employees using handled devices to link to their corporate networks, download data, work offline and then connect back to the network to synchronize with the corporate database. For example, with a mobile database embedded in a handheld device, a package delivery worker can collect signatures after each delivery and send the information to the database at the end of the day.

2.1 Defining the Mobile Database

A mobile database is a database that can be connected to by a mobile computing device over a mobile network so that the client/server connection is of wireless type. It also contains a cache to store data and transactions in disconnections. Database

is a structure used to organize information, this information may be a list of contacts, prices, traveled distances and so on.

Laptops, mobile phones, and pocket PCs are widely used and probably we will face with the increasing number of mobile systems applications in the future. Although analysts cannot accurately predict the most popular and most applicable application of the future, it is quite clear that a significant percentage of applications will require the use of the database. Many future applications will require the ability of downloading information from the database and perform operations on data even if located out of the target range or the connection is interrupted.

Today, with the advent of mobile databases, users can remotely exchange information with the mobile databases using smart phones or PDAs without worrying about time or distance. Mobile databases allow employees to enter information into databases even in flight. Information can be synchronized with the database server after a short time.

2.2 Mobile Database Requirements

Some factors that shall to be considered in mobile processing include:

- Mobile users should be able to continue their activities without a wireless connection (either because of poor communication or even when disconnected).
- User-time is the most valuable factor in most business applications.
- Since the connection is costly, reducing the connection time as much as possible should be highly considered.
- The number of transmitted bytes or packets is another factor that shall be considered in calculations.
- Applications interactions should be significant.
- Applications should be able to access local devices/hardware such as printers, bar code scanners, and GPS units (for mapping or navigator systems).
- The Bandwidth should be reserved (the minimum bandwidth required by wireless networks is a few megabytes)
- The user does not need access to data history, only the last modified data is required.
- A right time must be selected to update the information (during peak or off-peak periods)
- The limited lifetime of the power supply (battery)
- Changing the network topology

2.3 Mobile Database Systems Architecture

In any architecture, the followings shall be considered:

- Users are not located on a fixed geographical location;
- Mobile computing devices must be low power, low cost, and portable;
- Wireless networks;
- Computer limits;
- The mobile database structure

A mobile database normally consists of three components:

- Fixed host
- Mobile units
- Base stations

A mobile device is a device with the ability to connect to a fixed network through wireless connection. In fact, mobile unit is a communication device that operates with batteries and can move in a certain limited geographical area. The geographical area is limited due to limited bandwidth in wireless communication channels.

Fixed hosts perform the transaction and data management functions with the help of data base servers (DBS). Mobile units are portable computers that move around the geographic area that includes mobile network and are used for connecting to the base stations (note that mobile networks are not necessarily cell phone networks).

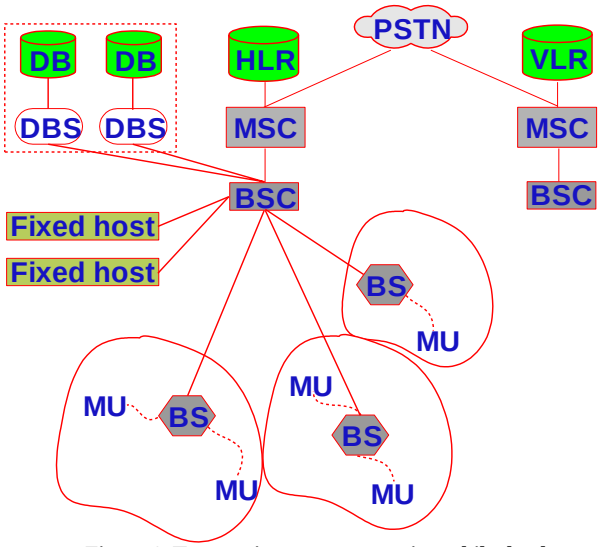


Figure 1. Transaction management in mobile database

Base stations are two-way radios installed at a fixed location establishing communication channels from the mobile unit to the fixed host and vice versa. They are usually low-power devices such as mobile phones and wireless routers.

3. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space. Hypotheses are plotted in this space and seeded with an initial velocity, as well as a communication channel between the particles. Particles then move through the solution space, and are evaluated according to some fitness criterion after each time-step. Over time, particles are accelerated towards those particles within their communication grouping which have better fitness values. The main advantage of such an approach over other global minimization strategies is that the large number of members that make up the particle swarm make the technique impressively resilient to the problem of local minima. Fig. 2 shows examples of particle movement in the search space. In Fig 2, the top left image represents the initial position of particles in a two-dimensional search space, and after some iterations, particles eventually converge as the down-right image.

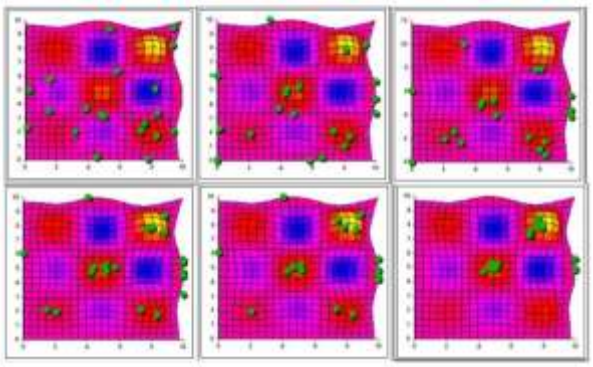


Figure 2 Movement of particles in a group [8]

Each particle has a position representing the coordinates in the multi-dimensional search space. The position of the particle changes as it moves. Here, xi (t) denotes the position of particle i at time-step t and vi (t) is the velocity of the particle i at time-step t. By considering the velocity, the current position of each particle can be calculated as Eq. 1.

$$x_i(t+1) = x_i(t) + v_i(t+1)$$
$$x_i(t) \sim U(x_{min}, x_{max}) \quad (1)$$

The fitness of the position of a particle in the search space is assessed through a fitness function. Particles have the ability to remember the best position during their lifetime. The best position personally found by a particle is yi (in some algorithm yi is also called pbest). In optimization process, the velocity vector reflects the particle empirical knowledge and the society information. Each particle moving in the search space considers two components:

- Cognitive component: $y_i(t) - x_i(t)$ is the best solution personally found by a particle.
- Social component: $\hat{y}_i(t) - x_i(t)$ is the best solution found by the group.

Two basic models are introduced for standard PSO algorithm: calculating the velocity vector based on cognitive and social components.

4. THE PROPOSED ALGORITHM

To solve an optimization problem, we must first define the position of each particle. In this problem, the position of each particle is considered as a [records number-table number] set. In other words, each particle is a set of records. The particle length is variable and is calculated as the difference between the user's request and the network packet length. Table 1 shows an example of a position for a request set.

Table 1 Set of particles for the proposed record set

Rec i-Table j		Rec i-Table j		....	Rec i-Table j		Fitness
1	65	4	245	....	3	567	

Each particle has a table number and a record number.

4.1 Generating the Initial Population

Here, 50% of the initial population is randomly selected and the remaining 50% using an intelligent approach based on the maximum entropy. Entropy is the maximum difference since it generates a fitted population. Entropy is calculated based on Eq. 2:

$$\max \sum_{\forall\ 1 \le i,j \le n\ \ i \ne j} (|a_i - a_j|) \quad (2)$$

At this stage, population size (the number of particles should exist in the population pool) is determined. This means that population size is an important factor affecting the algorithm efficiency. If its size is too small, only a small part of the solution space will be searched and the solution will quickly converge to probably a local optimum. And if the population size is very large, time-consuming calculations will be needed that is disproportionate to the solution. In Section 5 (algorithm evaluation), different population sizes are examined and the right size is found.

4.2 Calculating the Fitness Function

The fitness function evaluates the fitness and the performance of each particle of the population in problem-solving. The structure of each row is shown in Table 2.

Table 2: The structure of a row

Row A	
Table Code	TC
Record Code	RC
Last Record Sent Time	TR
Request Weight	WR
Usage Weight	WU
The Final Vote	Vote

So that:

Table Code (TC): Here, an ID is defined for each table in the database structure, and TC represents the table whose record is used.

Record Code (RC): According to the definitions of databases, the unique ID called key shall be assigned to each record (or sample entities). Here, RC is used.

Last Record Sent Time (TR): It is the last time that RC is sent from TC table to the mobile unit by the agent record. TR shows the agent which records have been (or have not been) recently used in the database. Thus, the agent removes the records that have not been long used since they probably are not interested by A.

Usage Weight (WU): It counts the times that RC from TC table is used by the mobile unit and modified or updated. The difference between WR and WU is that a user may request a record but does not change it.

Vote: Using the profile information, the agent can specify the most-used records (according to the request time and the weight). To this end, a coefficient (an importance factor which is calculated by trial and error) is considered for the mentioned variables as:

- Sent time × 0.65
- Request weight × 0.25
- Usage weight × 0.10

The Vote for i^{th} record is calculated as follows:

$$\text{Vote}_{(i)} = (\text{Tr} \times 0.65) + (\text{Wr} \times 0.25) + (\text{Wu} \times 0.10) \quad (3)$$

To solve the problem of mobile information prediction, we need to define a function measuring the fitness value for the obtained solutions. To measure the fitness of each particle, we can calculate Vote for each record, then consider the mean Vote as the fitness function.

4.3 Defining the Algorithm Parameters

4.3.1 Particles Initialization

The first step in a PSO algorithm is to initialize the swarm and control parameters. Usually, the particle's position is uniformly distributed to cover the search space. It should be noted that PSO performance is affected by the initial distribution of particles in the swarm (i.e. how much of the search space is covered and how the particles are distributed in the search space). If the optimal solutions are located somewhere in the search space that is not covered by the initial swarm, PSO will face difficulties to find them. In this case, PSO can find the optimal solutions only if the particles movements guide the search process toward these uncovered solutions.

Position initialization for each particle is defined as Eq. 4.

$$x(0) = x\_(min, j) + r\_j\ (x\_(max, j) - x\_(min, j)\ ), \forall j = 1,2 \quad (4)$$

Where xmin and xmax denote the minimum and maximum values for both table number and row number, respectively. Here, rj ~ U (0, 1).

4.3.2 Velocity C

To decide the efficiency and accuracy of a PSO algorithm, an important factor is the ability to achieve an optimal compromise between Explore and Exploit. Exploration is the ability to search for new solutions far from the current solution in the search space. On the other hand, exploitation tis he ability to concentrate the search around a promising area to refine a candidate solution. We found a solution achieving an optimal compromise between these two conflicting goals through PSO velocity updating which is shown in Eq. 5.

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)\left[y_{ij}(t) - x_{ij}(t)\right] + c_2 r_{2j}(t)\left[\tilde{y}_j(t) - x_{ij}(t)\right]$$
$$v_{ij}(t+1) = \begin{cases} v_{ij}(t+1) & if\ v_{ij}(t+1) < v_{max,j} \\ v_{max,j} & if\ v_{ij}(t+1) \geq v_{max,j} \end{cases} \quad (5)$$

where $v_{max,j}$ is the maximum allowed velocity in dimension j. The value of $v_{max,j}$ is very important since it guides the search by clamping the velocity. Large values of $v_{max,j}$ facilitate global exploration while small values encourage local exploitation. If $v_{max,j}$ is too small, the swarm may not explore sufficiently beyond locally good regions. Furthermore, the swarm may become trapped in a local optimum with no means of escape. On the other hand, too large values of $v_{max,j}$ are associated with the risk of missing a good region. The particles may jump over good solutions, and continue to search in fruitless regions of the search space. While large values do have the disadvantage that particles may jump over optima. In this case the particles move faster.

This leaves the problem of finding a good value for $v_{max,j}$ in order to balance between:

- Moving too fast or too slow
- Exploration and exploitation

Usually, the $v_{max,j}$ values are selected to be a fraction of the domain of each dimension of the search space. That is:

$$v_{max,j} = \delta(x_{max,j} - x_{min,j}) \quad (6)$$

where $x_{max,j}$ and $x_{min,j}$ are respectively the maximum and minimum values in dimension j, and $\delta \in (0,1]$. The value of δ is equal is first equal to 1 and varies in every generation according to Eq. 7. In each generation, the value of δ is 90% less than the previous generation.

$$\delta = 0.9^i \quad \text{i = generation number} \quad (7)$$

4.3.2.1 Algorithm Stopping Condition

The stopping condition is defined based on swarm radius. The algorithm terminates when the normalized swarm radius is close to zero. Normalized swarm radius is calculated as:

$$R_{norm} = \frac{R_{max}}{diameter(S)} \quad (8)$$

where diameter (S) denotes the diameter of the initial swarm and R_{max} is maximum radius calculated as:

$$R_{max} = \|x_m - \hat{y}\| \quad , \quad m = 1, \ldots, n\_s \quad (9)$$

The algorithm stops when R_{norm} is close to zero. Here, $\hat{y}$ is the minimum value of the fitness function.

4.3.3 Structure of the Proposed Algorithm

- Step 1: Initializing particles velocity and position in initial swarm according to Eq. 4;
- Step 2: Updating the best local position of i^{th} particle for all particles;
- Step 3: Updating the best global position of all particles;
- Step 4: Calculating the new velocity of all particles using Eq. 5;
- Step 5: Calculating the new position of all the particles using Eq. 10.

$$x\_i\ (t+1) = x\_i\ (t) + v\_i\ (t+1) \quad (10)$$

- Step 6: Repeating steps 2 through 5 considering the stopping condition discussed Section 4.3.3.

4.4 Multi-Objective Solutions

A drawback of the proposed method is failing to consider the previous needs of the user based on the vote, since it considers the overall needs of all users. To tackle this, the problem is defined as a multi-objective problem simultaneously considering needs of all users and needs of the current user. For this multi-objective algorithm, the code structure shown in Fig. 3 is proposed.

5. EVALUATION OF THE PROPOSED ALGORITHM

An important issue for a theory is testing and proving it. To test the presented theory, we developed codes in MATLAB environment which will be discussed here. The tests were conducted using a system with 4GH processor and 6GB memory. The data used in tests included information obtained from Sama University of Mahshahr.

Figure 3: the pseudo code proposed for multi-objective PSO

We ran the algorithm with initial populations of 125, 250, 500, and 1000. The algorithm was evaluated based on the processing time, as shown in Fig. 4.

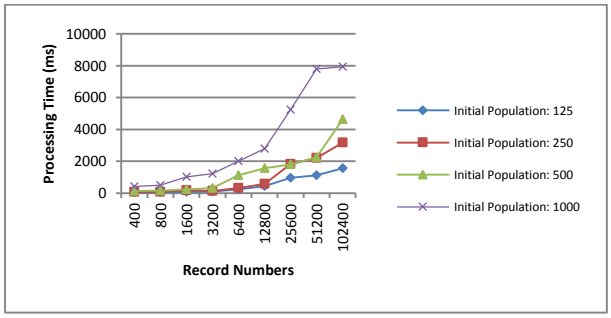


Figure 4 Processing time vs. record numbers for different initial populations

As seen in Fig. 4, the initial population of 125 is the optimal population in terms of running time.

Then, we run the algorithm for 200 users with 20 buffer. The results are listed in Table 4 and Fig. 5.

Table 4 Prediction percentage for 200 users with 20 buffer in the mobile unit and in terms of record number

	Number of Users = 200 Number of buffer = 20				
Total Record	200	400	800	1600	3200
Total Requests	2108	4419	5140	6703	8615
Predicted Records	775	1699	2084	2781	3943
Predicted Percent	36.76%	38.45%	39.84%	41.49%	45.77%

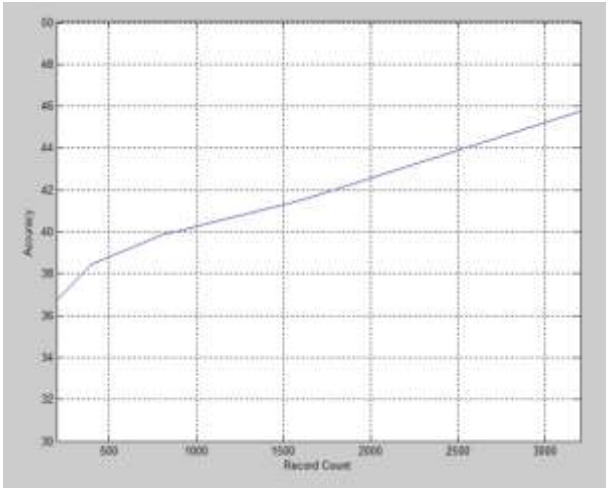


Figure 5 Prediction percentage for 200 users with 20 buffer in the mobile unit

5.1 Comparison with Other Algorithms

To compare the proposed algorithm with other algorithms, we applied the genetic algorithm using MATLAB. We also compared the algorithm with the results obtained in [3]. The output is shown in Table 5. Since we did not have access to standardized data, we evaluated the algorithms based on the prediction accuracy.

Table 5 Comparison between the proposed algorithm, genetic algorithm, and the proposed algorithm in [3]

	Algorithm	Accuracy
1	Proposed Algorithm	45.77%
2	Genetic Algorithm	35.76%
3	Proposed Algorithm in [3]	37.86%

6. CONCLUSIONS

In studies mentioned above, the only cashed information is the one that is now required, and there is no prediction for the information that may be used in the future. This study employed particle swarm optimization algorithm. To evaluate the proposed algorithm, some tests were conducted and the results indicated an efficiency of 25 to 45 percent for predictions that is reasonable.

7. REFERENCES

[1] Tarafdar M, Haghjoo M. Processing Spatial-Prediction Queries in Mobile Database using Timed Sequential Patterns. Tabriz; 2011. p. 234-41.

[2] Mohammadi M, Habibi A, Mohammadi F. Using Fuzzy Database to Predict Data and Improve Cache Operations in Mobile Databases. Science and Research University, East Azerbaijan; 2011.

[3] Beigi H. Investigation Transactions in Mobile Database and Using Genetic Algorithm to Predict Data in Cache Operations. MS's Thesis, 2008.

[4] Xuan K, Zhao G, Taniar D, Rahayu W, Safar M, Srinivasan B. Voronoi-based Range and Continuous Range Query Processing in Mobile Databases. J Comput Syst Sci. 2011; 77(4):637–51 .

[5] Srikant R, Agrawal R. Mining Sequential Patterns: Generalizations and Performance Improvements [Internet]. Springer; 1996 [cited 2014 Oct 2]. Available from: http://link.springer.com/chapter/10.1007/BFb0014140

[6] Fernando N, Loke SW, Rahayu W. Mobile Cloud Computing: A survey. Future Gener Comput Syst. 2013;29(1):84–106 .

[7] Yavaş G, Katsaros D, Ulusoy Ö, Manolopoulos Y. A Data Mining Approach for Location Prediction in Mobile Environments. Data Knowledge. 2005; 5 4(2):121–46 .

[8] Goldberg DE. Genetic Algorithms in Search, Optimization, and Machine Learning. 1 edition. Reading, Mass: Addison-Wesley Professional; 1989. 432 p .

[9] Mitchell M. An Introduction to Genetic Algorithms. Reprint edition. Cambridge, Mass.: A Bradford Book; 1998. 221 p .

[10] Memetic algorithm [Internet]. Wikipedia, the free encyclopedia. 2014 [cited 2014 Oct 28]. Available from: http://en.wikipedia.org/w/index.php?title=Memetic_algorithm&oldid=626348976

[11] Moscato P, Cotta C, Mendes A. Memetic algorithms. New Optimization Techniques in Engineering [Internet]. Springer; 2004 [cited 2014 Oct 28]. p. 53–85. Available from:http://link.springer.com/chapter/10.1007/978-3-540-39930-8_3

[12] Neri F, Cotta C, Moscato P, editors. Handbook of Memetic Algorithms. 2012 edition. S.l.: Springer; 2013. 370 p .

[13] Euclidean [Internet]. Wikipedia, the free encyclopedia. 2014 [cited 2014 Oct 28]. Available from: http://en.wikipedia.org/w/index.php?title=Euclidean&oldid=630468632

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 525 - 529, 2016, ISSN:- 2319–8656

New Genetic Algorithm to Predict Data In Mobile Databases

Leila Alaei Sheini
Department of Computer Engineer, Khomein Branch, Islamic Azad University
Khomein, Iran

*Hamid Paygozarh
Department of Computer Engineer, Khomein Branch, Islamic Azad University
Khomein, Iran

Mohammad Khalily Dermany
Department of Computer Engineer, Khomein Branch, Islamic Azad University
Khomein, Iran

Abstract: Modern enhancement in wireless network technology, satellite services and cell concoctions has been provided the possibility to use mobile computing systems. This movement for conducting mobile computations which needs receiving data from databases is problematic. System movement while performing transaction will lead to disconnection of mobile system with wireless network server. This disconnection will lead to fall running transaction and mobile system will be obliged to run transaction again from the beginning. One of the techniques to prevent from this transaction falling is to predict till to connect to the next mobile network. Our purpose is to present a solution to predict information to send to the mobile system in order to be used in the memory using Memetic algorithm. Generally, Memetic algorithms in optimal selection and design in many technical and scientific problems causes producing the best product or acceptable solution. The main purpose to use these kinds of optimization algorithms is to choose the best information to be sent to the mobile system while disconnection to prevent from transaction falling. This research has concentrated only on information sent to the mobile system.

Keywords: Mobile Database, Cache Operation, Mobile Client, Cell Connection, Memetic Algorithm..

1. INTRODUCTION

Recently, technical enhancement relevant to the portable PCs development and also development of wireless technologies causes emerging portable PCs with capability to carry wireless connections and gives users a possibility to do actions in distributed calculated environment even while moving. Mobile users are willing to access private or corporate databases which have been saved in mobile or fixed hosts and present their information update and queries in wire and wireless networks [1].

In this kind of service giving, limitation such as mobile system power supply, expensive service and limitation of mobile systems' bandwidth and limitation of service giving space will lead to disconnection of mobile system with service giving center. This disconnection will lead to reject running transaction and mobile system will be obliged to run transaction again from the beginning. This rerunning requires costs such as connection to the network and time [2]. One of the methods to prevent from this transaction rejection is to use cache operation in mobile system that mobile service giver center predicts extra information more than what have been requested while receiving request from mobile system and sends these information to the mobile system. Mobile system stores extra information in Cache memory and as a following of transaction running or next transactions, if there is requested information in local memory, it uses Cache information and there is no need to connect and send request and receive information. This issue prevents from transaction rejection due to disconnection while running and helps with decreasing costs and increasing speed to response to mobile system user. The most important issue in this technique is correct and optimal prediction of information in order to be sent which needs a kind of data mining in service giver database table. In this paper, Memetic algorithm has been applied for data mining [3].

2. MOBILE DATABASE

Along with achieved enhancement in the context of wireless communication technology and portable computation devices called mobile computation. So that, users move their portable devices and meanwhile these users have a possibility to access information and services without considering physical place and behavioral movements. In fact, with appearing computer networks especially internet network and simultaneously emerging portable calculative tools such as qualitative PCs and PDAs and new need of users to access internet network, this possibility has been created that people in anywhere and anytime can do their actions through wireless networks. Due to growth speed of wireless devices and mobile calculative devices, in a very close future, millions users will be using portable computers and wireless communication devices and their main demand is to connect World Wide Web and resolve calculative and information needs. Wireless technology gives this opportunity to the users to keep their connection with network while moving [4]. In such architecture due to wireless nature and environment and mobile users, distributed environment is proposed in a new way. In such environment, mobile users due to mobile environment are more affected by disconnection. Mobile calculations are more affected by disconnection. Mobile calculations due to mobile users and their computers have been distinguished from fixed connections. Mobile service giver/taker database system is as [5]:

- A set of weak or strong reliable information services which have been connected via a fast-paced immobile network.
- A set of unreliable service takers is defined on wireless network.

Therefore, architecture includes mobile and immobile components. Mobile sector is known as Mobile Unit (MU).

MU is a mobile device which is able to connect with immobile network through wireless system. In figure 1, architecture of mobile database has been shown.

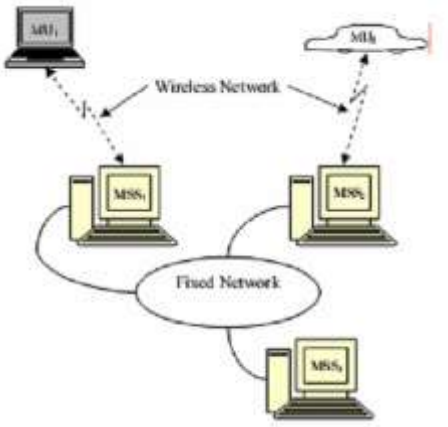


Figure 1: Mobile database architecture [1]

In fact, computer is a communication tool which is working with battery and it can move freely in a certain and limited geographical area (area covered by workstation). Restriction of this geographical area is due to restriction of bandwidth in wireless communication channels.

Immobile components of the architecture are connected through fast-paced immobile network (speed of this network can be in Mbps/Gbps range). Also, data and databases have been distributed throughout components of this fixed network. These fixed components are in 2 kinds:

- Fixed Host (FH)
- Workstation

In this environment, fixed hosts don't have capability to connect mobile unit, but they can propose their requests to the network like MU. Each workstation is capable to connect MU is equipped with a wireless communication to make a connection. Workstation is also known as a Mobile Support Station (MSS), because they act as a connector between mobile computers and fixed hosts. Geographical area which has been covered by is called cell. In workstations, information like users' identification file, network log files and data access rules are kept.

Mobile architecture due the cell which is now located, it only can connect to one MSS. Based on each MU; MU can change its connective and location position with network moving in wireless network. During this movement, MU might be running an application. Running this application/ transaction requires data which have been obtained by MU through connection with MSS. As a result of MU movement, it might be disconnected with the network for a moment. After reconnection, it might be inserted into a cell belongings to MSS. Running an application /transaction by mobile unit can be done in 2 ways:

- Doing transaction on mobile unit
- Doing transaction on fixed network

3. CACHE OPERATION

In mobile database systems, caching can play a considerable role in decreasing connection with services givers and takers. In a mobile computing environment, data are kept on mobile support stations (MSS). Primary control of data and doing any kind of operation over data are investigated by MSSs and after verification, it is recorded in database. Caching data items which are repeatedly and highly being used by MUs can be an effective method to decrease connective costs related between MSSs and MUs and also to use mobile units' resource better. MUs can cache information in their database which is similar to MSSs' database. In this way, accessibility to data required in MU is being done faster and number of requests compared with data receiving in network is decreasing. When a mobile unit requests for a data, if this data is available, it can be used and results will be sent to the service giver, otherwise if required data is unavailable or a part of it is available, mobile unit sends request for required data to service giver [6].

4. MEMETIC ALGORITHM

Gene is a part of biological information which is transferred from a generation to another generation. Genes determine physical features such as face, body shape and all of features which have been inherited from their parents. Meme has been introduced in 1979 by Dowkins with a gene comparison in the context of genetic evolution and in the context of cultural evolution [7, 8]. Meme is a cultural and behavioral element which is being transferred from a generation to another generation via non-genetic factors. In fact, Meme includes each feature and characteristic which has been learnt through experience and imitation during a life of a creature and it is propagated among organisms. This propagation has never had a genetic nature and reproduction operators don't have impact on it. Lingual definition of Meme is: apart of civilization which are not inherited through genes. As biologists consider gene as a unit to transfer physiologic features such as eyes color, hair color and etcetera from parents to children, psychologists also know gene as a unit to transfer behavioral characteristics such as Irritability, traditionalism and etcetera from parents to children. Based on psychologists' viewpoints, a person who is born in an illiterate family, doesn't have to stay illiterate to the end of his life and he can be promoted with gaining some of skills form his environment while biologists consider chromosomes genes unchanged from the birth to the death. Basis of Memetic algorithm (MA) is based on this theory. Unlike genetic algorithm which considers people unchanged from birth to death (participating in reproduction process for the next generation), a person in Memetic algorithm can promote his competency in a generation with behavior known mimic [9].

Idea relevant to use of Meme conception to design a Meta-heuristics algorithm was firstly introduced by Moscato [10]. He recommended that with applying a local searching operator in the genetic algorithm body (after mutation operator), an opportunity is given to each organism (child) to live. Memetic algorithm has lots of similarities with genetic algorithm. The underlying difference of Memetic algorithm with Genetic algorithm is optimization of population of each generation after doing mutation and combination operators. To achieve this purpose, for each person who is living in the population, a local search with predetermined neighborhood radius around chromosome relevant to the produced child is done in problem state space.

Memetic algorithm like genetic algorithm has been employed to solve problems related to continuous and discrete optimization and a wide range of real world issues [11].

5. MYTHOLOGY

In database architecture, each MU in order to use data existed in database must be connected to one of MSSs and gives it its request and receives its required information from it. Of

course, sending and receiving data by MU is done via wireless network. In this architecture, database has been dispensed to MSSs. Of course, each of mobile units has a structure similar to the main database which is being used for information caching and local data reception. Each MSS is responsible to manage connection with MUs. Also, MSSs are responsible of accuracy of conducted transactions and recording transactions' results in database. When MU is connected with MSS, it is responsible to create an agent to control caching and managing MU's relocation. In figure 2, agents within MSS have been shown.

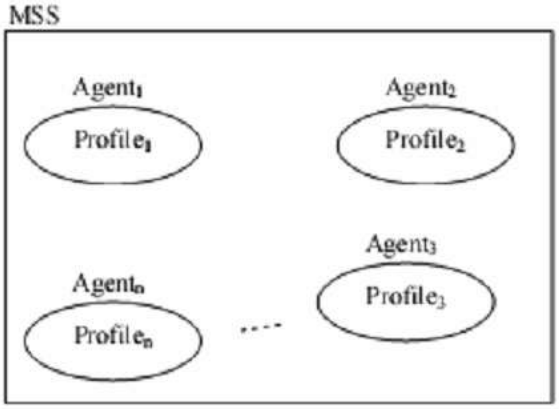


Figure 2: Agent within MSS [2]

In fact, for each MU, an agent is created. Having each agent within MSS shows MUs which have been connected to that MSS. Agent is applied to manage MU mobility. That is, with moving MU from a MSS to another MSS, all of information of agent related to this MU is transferred from origin MSS to the new MSS and new MSS is controlled via this agent and will take authority to manage MU.

Each agent is the maintenance of a profile of information which is being used to predict data which have to be cached. Agent can share their information to other agents. Of course, this action is being done to predict information required for the future. Profile information of each agent is created and completed by agent and based on data which has been used by MU so far and agent per se makes decision to eliminate these data.

Agent has to response each request from MU and record information relevant to profile completion in profile. Then, agent takes an effort to anticipate information required for MU in the future and after completing anticipation, sends obtained information to MU. In this process, 2 following stages are being done:

- Determining MU neighbors
- Findings information category related to the data which is being followed by user more and then choosing data having reasonable relation with used information.

How to perform these two steps will be described as following

5.1 Operation of Mobile site removing

Each MU when is connected to MSS, if it is a first time which connects to MSS, MSS creates a new agent for this MU and prepares agent to control and manage MU. While creating agent, in order to help MU identify its agent, MSS gives each agent a unique number. This number is unique in other MSSs. This number is seen in relation (1).

Agent ID=MSS Number+(Count Agent+1) (1)

Here, MSS Number is service giver number, because each service giver is assigned a unique number. Count Agent determines number of MSS agents. This number is incremented one unit for new agent. After determining agent number, a unit is added. After determining agent number, this number is sent to MU and MU is able to identify its agent based on this number. From now on, while sending request to MSS, request is given to the respective agent and agent makes decision to send information to MU and with each request, it completes its profile to make better decision. If MU is disconnected with MSS, after reconnection, first MU must identify MSS and its agent. For this purpose, MU sends identification number of its agent to MSS. If MU is not evicted from scope of previous MSS, MSS identifies number and connection is established. If number is unknown for MSS, new MSS is responsible to determine previous MSS via identification number and sends a request based on receiving information related to that agent to the previous MSS. New MSS, after receiving agent information, changes agent identification number based on its own number and sends this number to the MU entitled new number and makes connection between MU and its agent.

5.2 Profile structure

Based on aforementioned contents, agent has profile which includes information to predict data required for the future. Information of this profile is created based on earlier used data. Each profile has a structure as table 1.

Table 1: profile structure (gene) of A chromosome agent

Agent(A)	
T_C	Table code
R_C	Record code
T_R	Request time
W_R	Request weight
W_U	Usage weight
Vote	Vote

So that:

A: From here, mobile unit a place where anticipation is done for that is determined with A.

Table code (TC): Here, for each table in database structure, an address is considered and TC shows a table which its record has been used.

Record code (RC): Based on definitions of database, each record (or entity sample) must have unique address named key. Here, is RC being used.

Last request time (TR): It is the last time which record agent sends RC from TC to MU. This time for the record determines which record has been newly used or hasn't newly

used and agent eliminates records in the profile which have a certain difference with the current time, because these records are probably not interesting to A and are not being used.

Usage weight (WU): It shows number of times which MU uses RC from TC and changes/ updates this record. Difference between WU and WR is that a user requests for a record, but doesn't change it.

Vote: Agent, with having its profile information, can determine records which have been used more than others (based on request time and usage weight). To reach this purpose, a coefficient as an importance is assigned to each of above values as following:

- Sending time × 0.6
- Request weight × 0.25
- Usage weight× 0.15

Now, vote relevant to profile's ith record is calculated as following:

$Vote_{(i)}=(Tr\times0.6)+(Wr\times0.25)+(Wu\times0.15)$

In fact, each time that information is sent to MU, some of information of profile (Wu, Wr, Tr) are being changed and value of vote relevant to each profile's line with changes of that line are again obtained. Value of vote can display opinion/ interest of users about RC. With creating agent and profile relevant to the A mobile unit, agent completes receiving of each request from MU profile and tries to determine working scope of MU.

5.3 Defining and implementing Memetic space:

To use Memetic algorithm, the most important part is to define and implement genetic space which has been expressed as following:

1- Population: All of agents in all of MSSs connected to the wireless and fixed network are considered population.
2- Gene: Records which are located in tables of each agent are called gene which is equivalent to profile structure.
3- Chromosomes: Each agent is considered as chromosome.
4- Crossover: Finding a packet of information (records), the most optimal information to anticipate and send to MUs, is called crossover.
5- Fitness function: A function which is used to fit the best records to anticipate.
6- Local search: It causes that problem is explored sooner.

5.4 Prediction

As it was expressed, prediction algorithm is two-step algorithm. How to do these 2 steps and prediction algorithm have been illustrated as following

5.5 Neighbors selection

Here, neighbor doesn't mean near MUs in physical and place terms, but also it means closeness and adjacency in terms of information and data which have been used by respective MUs and other similar MUs. As it was discussed before, chromosomes can give their gens' information to other chromosomes. This issue is also true for chromosomes in MSSs. This can be done through MSSs. To determine neighbors, genes of other chromosomes have to be investigated and choose the nearest neighbors, because with the number of chromosomes in MSSs, comparison is being done. After selecting chromosomes, comparison between 2 genes can be done using Euclidean distance (12) (of course something that is changed) which is working on profile records. Euclidean (A, B) is similarity of active user A and user B.

$$\text{Euclidean}(A,B)=\sqrt{\sum_{i=1}^{k} TR_{A,i}\times(Vote_{A,B}-Vote_{B,i})^2}$$

So that:

K: It determines number of similar records which have been used by A, B chromosomes.

TRA,i: It is a time that chromosomes A has sent record i for MU.

VoteA,B: It is a vote which has been computed based on gene of ratio of chromosome A usage from gene B.

Time to run this equation is for n O(nk) chromosomes which is $K=Average(K_1,K_2,\ldots,K_n)$, here that Ki shows number of genes similar to chromosome A and chromosome i. Obtained distance is more or equal to zero. Chromosomes which have the least distance are required and out of N selected chromosomes, M chromosomes which have the least distance can be selected to be used to predict.

5.6 Prediction algorithm

Prediction is being done when MU proposes a request and information needed for this request are existed in MU. In this case, MU sends request to the respective MSS to obtain its required information. In this time, chromosomes existed in MSS has duty to predict other information in addition to information required for requested fixed genes and sends to MU. To predict, we can:

Based on gene information, record number (RV) and table number (TC) relevant to the line which has the most votes are obtained.

Among genes in selected chromosomes, those which are requested are eliminated, because they have previously located in new crossover chromosome.

After determining genes which don't have to participate in crossover (those which have been selected before), the nearest neighbor has to be determined as it was mentioned above. Now, genes which have been sent to MU have to be used by more number of nearest neighbors and also haven't been used by MU so far (they have been the best genes in fitness function).

After completing first selection operation, we have to select next gene which has the most value and perform Memetic algorithm for that gene. A certain number of Votes or just Votes which are more than a certain limit can be considered.

6. CONCLUSION

In researches which have been called above, only information are cached which are needed now and no more prediction is being done for information which will probably be used in the future. In this way, one of the best optimization

algorithm (Memetic algorithm) has been used which can guarantee an appropriate efficiency during a certain time

7. REFERENCES

[1] Tarafdar, M , Hagh joo Saniij, M. location prediction query processing in mobile database to discover patterns in timed sequence. Tabriz; 2011. p. 234-41.

[2] Saqayshy Mohammadi, M, Habibi Zadnovin, A., F. Mohammadi Saqayeshy, M. Using fuzzy database to predict and improve operations in mobile databases. Science and Research, East Azerbaijan; 2011. p. 65-71.

[3] Xuan K, Zhao G, Taniar D, Rahayu W, Safar M, Srinivasan B. Voronoi-based range and continuous range query processing in mobile databases. J Comput Syst Sci. 2011;77(4):637–51.

[4] Srikant R, Agrawal R. Mining Sequential Patterns: Generalizations and Performance Improvements [Internet]. Springer; 1996 [cited 2014 Oct 2]. Available from: http://link.springer.com/chapter/10.1007/BFb0014140

[5] Fernando N, Loke SW, Rahayu W. Mobile Cloud Computing: A survey. Future Gener Comput Syst. 2013;29(1):84–106 .

[6] Yavaş G, Katsaros D, Ulusoy Ö, Manolopoulos Y. A Data Mining Approach for Location Prediction in Mobile Environments. Data Knowledge. 2005; 5 4(2):121–46 .

[7] Goldberg DE. Genetic Algorithms in Search, Optimization, and Machine Learning. 1 edition. Reading, Mass: Addison-Wesley Professional; 1989. 432 p .

[8] Mitchell M. An Introduction to Genetic Algorithms. Reprint edition. Cambridge, Mass.: A Bradford Book; 1998. 221 p .

[9] Memetic algorithm [Internet]. Wikipedia, the free encyclopedia. 2014 [cited 2014 Oct 28]. Available from: http://en.wikipedia.org/w/index.php?title=Memetic_algorithm&oldid=626348976

[10] Moscato P, Cotta C, Mendes A. Memetic algorithms. New Optimization Techniques in Engineering [Internet]. Springer; 2004 [cited 2014 Oct 28]. p. 53–85. Available from:http://link.springer.com/chapter/10.1007/978-3-540-39930-8_3

[11] Neri F, Cotta C, Moscato P, editors. Handbook of Memetic Algorithms. 2012 edition. S.l.: Springer; 2013. 370 p .

[12] Euclidean [Internet]. Wikipedia, the free encyclopedia. 2014 [cited 2014 Oct 28]. Available from: http://en.wikipedia.org/w/index.php?title=Euclidean&oldid=630468632

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 530 - 532, 2016, ISSN:- 2319–8656

Clustering Students By K-means

* Mohammad Farzizadeh
Sama Technical and Vocational Training College
Islamic Azad University, Mahshahr Branch
Mahshar,Iran

Ali Abdolahi
Sama Technical and Vocational Training College
Islamic Azad University, Mahshahr Branch
Mahshar,Iran

Abstract: In typical assessment student are not given feedback, as it is harder to predict student knowledge if it is changing during testing. Intelligent Tutoring systems, that offer assistance while the student is participating, offer a clear benefit of assisting students, but how well can they assess students? What is the trade off in terms of assessment accuracy if we allow student to be assisted on an exam. In a prior study, we showed the assistance with assessments quality to be equal. In this work, we introduce a more sophisticated method by which we can ensemble together multiple models based upon clustering students. We show that in fact, the assessment quality as determined by the assistance data is a better estimator of student knowledge. The implications of this study suggest that by using computer tutors for assessment, we can save much instructional time that is currently used for just assessment.
Keywords: Clustering, Educational Data Mining,

1. INTRODUCTION

Feng et al.[1] reported the counter-intuitive result that data from an intelligent tutoring system could better predict state test scores if it considered the extra measures collected while providing the students with feedback and help. These measures included metrics such as number of hints that students needed to solve a problem correctly and the time it took them to solve. That paper [1] was judged as best article of the year at User Modeling and User-Adapted Interaction and was cited in the National Educational Technology plan. It mentions a weakness of the paper concerning the fact that time was never held constant. Feng et al. go one step ahead and controlled for time in following work [2]. In that paper, students did half the number of problems in a dynamic test setting (where help was administered by the tutor) as opposed to the static condition (where students received no help) and reported better predictions on the state test by the dynamic condition, but the difference was not statistically reliable. This present work starts from Feng et al. [2] and investigates if the dynamic assessment data can be better utilized to increase prediction accuracy over the static condition. We use a newly introduced method that clusters students, creates a mixture of experts and then ensembles the predictions made by each cluster model to achieve a reliable improvement.

2. LITERATURE REVIEW

The Bayesian knowledge tracing model [3] and its ariants [4] [5] have become the mainstay in the Intelligent utoring System (ITS) community to track student knowledge. his knowledge estimate is used for calibrating the amount of training students require for skill mastery. One of the most important aspects of such modeling is to ensure that performance on a tutoring system is transferred to actual post tests. If this is not the case, then that implies over-training within the tutoring system. In fact, it is reasonable to say that one of the most important measures of success of a tutoring.

Traditionally, performance on a post-test is predicted by using practice tests. Practice tests based on past questions from specific state tests can give a crude estimate of how well the student might perform in the actual state test. Improving this estimate would be highly beneficial for educators and students. For improving such assessment, dynamic assessment [6] has long been advocated as an effective method. Dynamic assessment is an interactive approach to student assessment that is based on how much help a student requires during a practice test. Campione et al. [7] compared the traditional testing paradigm, in which the students are not given any help, with a dynamic testing paradigm in which students are given graduated hints for questions that they answer incorrectly. They tried to measure learning gains for both the paradigms from pre-test to post-test and suggested that such dynamic testing could be done effectively with computers. Such assessment makes intuitive sense as standard practice tests simply measure the percent of questions that a student gets correct. This might not give a good estimate of a student's knowledge limitations. If a student gets a question wrong, it might not necessarily imply absence of knowledge pertaining to the question. It is likely that the student has some knowledge related to the question but not enough to get it correct. It is thus desirable to have a fine grained measure of the knowledge limitations of the student during assessment. Such a measure might be obtained by monitoring the amount of help the student needs to get to a correct response from an incorrect response. ITS provide the tools for doing dynamic assessment more effectively as they adapt while interacting with individual students and make it easier to provide interventions and measure their effect. Fuchs et al. [9] studied dynamic assessment focusing on unique information, such as how responsive a user is to intervention. Feng et al. [1][2] used extensive information collected by the ASSISTments tutor [13] to show that the dynamic assessment gives a relatively better prediction as compared to static assessment. This work effectively showed that dynamic assessment led to better predictions on the post test. This was done by fitting a linear regression model on the dynamic assessment features and making predictions on the MCAS test scores.system is its ability to predict student performance on a post-test. Since such a transfer is dependent on the quality of assessment, a tension exists between focusing on quality of assessment and quality of student assistance.

They concluded that while dynamic assessment gave good assessment of students, the MCAS predictions made using those features lead to only a marginally statistically significant improvement as compared to the static condition. In this paper we explored the dynamic assessment data to see if we could make significantly better predictions on the MCAS test score.

A significant result would further validate the use of ITS as a replacement to static assessments.

3. DATA

The dataset that we considered was the same as used by Feng et al.[2]. It comes from the 2004-05 school year, the first full year when ASSISTments was used in two schools in Massachusetts. ASSISTments is an e-learning and e-ssessing research platform [10] developed at Worcester Polytechnic Institute. Complete data for the 2004-05 year was obtained for 628 students. The data contained the dynamic interaction measures of the students and the final grades obtained in the state test (MCAS) taken in 2005. The dynamic measures were aggregated as students used the tutor.

3.1 Metrics

The following metrics were developed for dynamic testing by Feng et al. [2] and were used in these experiments. They try to incorporate a variety of features that summarize a student's performance in the system.The features were as follows: 1) the student's percent correct on the main problems 2) number of problems done 3) percent correct on the help questions 4) average time spent per item 5) average number of attempts per item and 6) average numbers of hints per item. Out of these, only the first was as a static metric and was used to predict the MCAS score in the static condition. The other five and a dynamic version of student's percent correct on the main problems were used to make predictions in the dynamic condition.

The predictions were made on the MCAS scores. The MCAS or the Massachusetts Comprehensive Assessment System is a state administered test. It produces tests for English, Mathematics, Science and Social Studies for grades 3 to 10. The data set we explore is from an 8th grade mathematics test.

4. METHODOLOGY

The data was split into randomly selected disjoint 70% train and 30% test sets. Feng et al.[2] fit a stepwise linear regression model using the dynamic assessment features on the training set to make a prediction on the MCAS scores on the test set. They reported an improvement in prediction accuracy with a marginal statistical significance relative to the predictions made only using data from the static condition. Fitting in a single linear regression model for the entire student data might be a bad idea for two reasons. First, the relationship between the independent variables (dynamic assessment features) and the dependent variables (MCAS test scores) might not be a linear one. If so, training a linear model would have high bias for the data and no matter how much data is used to train the model, there would always be a high prediction error. The second conceivable source of error is related to the first. A student population would have students with varying knowledge levels, thus requiring different amounts of assistance. Thus it might be a bad idea to fit the entire population in a single model. Students often fall into groups having similar knowledge levels, assistance requirements, etc. It is thus worth attempting to fit different models for different groups of students. It, however, must be noted that while such groups could be identified using clustering, the groups obtained may not be easily interpretable.

4.1 Clustering

The previous section mentions that it might not be a good idea to fit in a single model for the entire student population and that there might exist groups of students having similar knowledge levels and nature of responses to interventions. A natural method to find such patterns in the data is by clustering. If data was generated by a finite set of distinct processes, then clustering methods are maximum likelihood methods to identify such underlying processes and separating them. The idea in this work is to fit in a linear regression model for each such group in the training set. The prediction for the MCAS score for each student from the test set would thus involve two steps: identification of the cluster to which the student from the test set belongs and then using the model for that cluster to make the prediction of the MCAS score for the student.

We used K-means clustering for the identification of K groups. The initialization of cluster centroids was done randomly and the clusters were identified by using Euclidean distance. K-means finds out the best separated clusters by trying to minimize a distortion function. The distortion function is a non-convex function and thus implies that K-means is susceptible to getting stuck in local optima. This means that when K-means is run with random cluster centroids; we might not reach the best solution possible. To reduce the chances of getting a sub-optimal clustering we restarted K-means 200 times with random initialization.

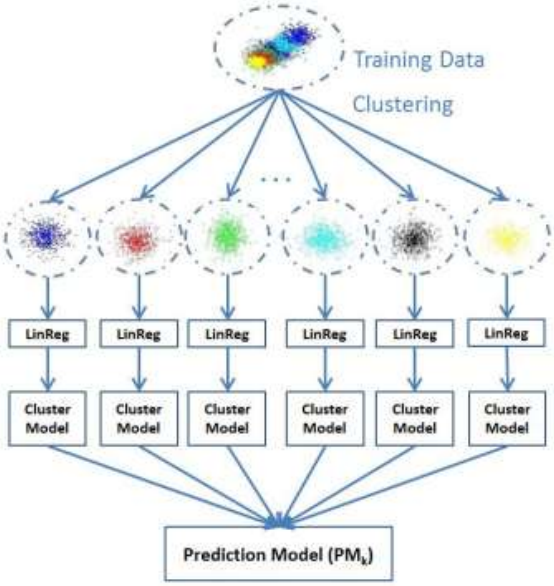


FIGURE 1 Schematic illustrating the steps for obtaining a prediction model (PMK). There would be one such prediction model for each value of K chosen (1 to K would give K prediction models)

For each cluster identified we trained a separate linear regression model (Fig. 1). We call such a linear regression model (for each cluster) a cluster model. For data separated into K clusters there would be K cluster models. All of these K cluster models taken together make predictions on the entire test set. These K cluster models together can be thought to form a more complex model. We call such a model a prediction model i.e. PMk , with the subscript K identifying the number of cluster models in the prediction model. Feng et al. [2] used the prediction model PM1 , since only a single linear regression model was fit over the entire data-set. The value of K can be varied from 1 to K to obtain K prediction models. For example: if K = 1, 2 and 3, there would be three prediction models - PM1 having a single cluster model (K=1), PM2 having two different cluster models (K=2) and PM3 , that is the prediction model with three different cluster models (K=3). It is noteworthy that the cluster models in different prediction models would be different.

4.2 Ensemble Learning

Section 3.1 described how, by using K as a controllable parameter, we can obtain a set of K prediction models and K corresponding predictions. The training data is first clustered by K-means and K clusters are obtained. For each of the clusters we fit a linear regression model, which we called the cluster model. The cluster models together are referred to as a prediction model. This prediction model makes a prediction on the entire test set. But since K is a free parameter, for each value of K we get a different prediction model and a different set of predictions. For example

While we are interested in looking at how each prediction model performs. It would also be interesting to look at ways in which the K predictions can be combined together to give a single prediction. Such a combination of predictors leads to ensembling. Ensemble methods have seen a rapid growth in the past decade in the machine learning community [12][13][14].

An ensemble is a group of predictors each of which gives an estimate of a target variable. Ensembling is a way to combine these predictions with the hope that the generalization error of the combination is lesser than each of the individual predictors. The success of ensembling lies in the ability to exploit diversity in the individual predictors. That is, if the individual predictors exhibit different patterns of generalization, then the strengths of each of the predictors can be combined to form a single stronger predictor. Dietterich [12] suggests three comprehensive reasons why
nsembles perform better than the individual predictors. Much research in ensembling has gone into finding methods that encourage diversity in the predictors.

4.2.1 Methodology for Combining the Predictions

We have a set of K predictors. The most obvious way of combining them is by some type of averaging. The combination could also be done using Random Forests [15], but they have not been explored in this work as we are extending work that simply used linear regression. We explored two methods for combining these predictors.

1. Uniform Averaging: This is the simplest method for combining predictions. The K predictions obtained (as discussed in section 3.1) are simply averaged to get a combined prediction. In addition to averaging all predictions we could also choose to average just a subset of the predictions together.

2. Weighted averaging: In uniform averaging, each predictor is given the same weight. However, it is possible that the predictions made by some model are more important than the predictions made by another model. Thus, it is reasonable to combine the models by means of a weighted average. Such weighted averaging could be done by means of a linear regression. Since we did not find an improvement with weighted averaging, the methodology and results are not discussed in detail.

5. CONTRIBUTIONS

This paper makes one clear contribution. This is the first paper we know of that clearly demonstrates that not only can an Intelligent Tutoring System allow students to learn while being assessed but also indicates a significant gain in assessment accuracy. This is important, as many classrooms take away time from instruction to administer tests. If we can provide such a technology it would save instruction time and give better assessment and would thus be highly beneficial to students and instructors. The second contribution of this paper is the application of clustering student data and ensembling predictions that we are introducing to the field in a KDD paper [16]. In that paper we applied this approach to a number of datasets from the

6. REFERENCES

[1] Feng, M., Heffernan. N. T. & Koedinger. K. R. (2009). Addressing the assessment challenge in an online system that tutors as it assesses. User Modeling and User-Adapted Interaction: The Journal of Personalization Research. 19(3). 2009.

[2] Feng. M., Heffernan. N. T., (2010). Can We Get Better Assessment From A Tutoring System Compared to raditional Paper Testing? Can We Have Our Cake (better assessment) and Eat it too (student learning during the test). Proceedings of the 3rd International Conference on Educational Data Mining?, 41-50.

[3] Corbett. A. T. & Anderson, J. R. (1995). Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. User Modeling and User Adapted Interaction, 4, 253-278.

[4] Pardos, Z.A., Heffernan, N. T. In Press (2011) Using HMMs and bagged decision trees to leverage rich features of user and skill from an intelligent tutoring system dataset. Journal of Machine Learning Research C & WP.

[5] Baker, R. S. J. d., Corbett. A. T., Aleven, V. (2008) More Accurate Student Modeling Through Contextual Estimation of Guess and Slip Probabilities in Bayesian Knowledge Tracing. Proceedings of the 14th International Conference on Artificial Intelligence in Education. Brighton, UK, 531-538

[6] Grigerenko, E. L. & Steinberg, R. J. (1998). Dynamic Testing. Psychological Bulletin, 124, 75-111

[7] Campione. J. C. & Brown. A. L. (1985). Dynamic Assessment: One Approach and some Initial Data. Technical Report. No. 361. Cambridge, MA. Illinois University, Urbana, Center for the Study of Reading. ED 269735

[8] Fuchs. L. S., Compton. D. L. Fuchs. D., Hollenbeck. K. N., Craddock. C. F., & Hamlett. C. L. (2008). Dynamic Assessment of Algebraic Learning in Predicting Third Graders' of Mathematical Problem Solving. Journal of Educational Psychology , 100(4), 829-850.

[9] Fuchs. D., Fuchs L.S., Compton. D. L., Bouton. B., Caffrey. E., & Hill. L., (2007). Dynamic Assessment as Responsiveness to Intervention. Teaching Exceptional Children. 39(5), 58-63.

[10] Razzaq. L., Feng M., Nuzzo-Jones. G., Heffernan. N. T., Koedinger. K. R., Junker. B., Ritter. S., Knight A., Aniszczyk. C., Choksey. S., Livak. T., Mercado. E., Turner. T. E., Upalekar R., Walonoski. J.A., Macasek. M. A., & Rasmussen. K. P. (2005). The Assistment Project: Blending Assessment and Assisting. In C. K. Looi, G. McCalla, B. Bredeweg, & J. Breuker (Eds). Proceedings of the 12th International Conference on Artificial Intelligence in Education, Amesterdam. ISO Press, pp 555-562.

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 533 - 538, 2016, ISSN:- 2319–8656

Harnessing Object–Oriented Programming Techniques for Transaction Processing

Onu Fergus U. (PhD)
Department of Computer Science,
Ebonyi State University,
Abakaliki - Nigeria

Akpan Abasiama G.
Department of ICT,
Ritman University,
Ikot Ekpene, Nigeria

Nnanna Emmanuel E.
Department of Computer Science
Ebonyi State University,
Abakaliki- Nigeria

Abstract: This study focused on harnessing object-oriented programming techniques for Transaction Processing. In this way, object-oriented programming reintroduces techniques for managing software components. On the other land, Transaction processing are means of managing classic software structure for concurrent accesses to global data and for maintaining data consistency in the presence of failures. Hence, in this study, object-oriented programming techniques are examined in a view to structure units that encapsulate complex behaviour and embrace groups of objects and method calls. We collected data from both primary and secondary sources, to elicit information from stakeholders in software development industry. It was unraveled that there is a high positive relationship between object-oriented programming techniques and Transaction processing.

Keywords: Object based programming, Abstraction, Transaction Processing, Modularity, Concurrency.

1.0 INTRODUCTION

In the words of [1], software technology is making a transition from the view that programs are action sequences to the view that programs are collections of interacting software components. Object – oriented programming is a form of component – based software technology whose software components are objects and classes. In the way, programmers can create relationships between one object and another. For example, objects can inherit characteristics from other objects [2].

Kienzle [3] posits that complex systems often need more elaborate concurrency features than the ones offered by concurrent object – oriented programming languages. The existing single method approaches do not scale well, since they deal with each single operation separately, hence, there is a need for structuring units that encapsulate complex behavior and embrace groups of objects and method calls. These units should represent dynamic systems execution as opposed to the static declaration of objects inside objects. A Transaction processing is an interaction in the real world, usually between an enterprise and a person or another enterprise, where something is exchanged. It requires the execution of multiple operations, it must run in its entirety, it must be incrementally scalable, and records of transactions, once completed, must be permanent and authoritative. Object – oriented programming presents these collaborative techniques and hence poses the tendency to provide great success in Transaction processing.

Meyer [4] defined, object-oriented programming in a distinct manner, as a type of programming in which programmers define not only, the data type of a data structure, but also the types of operations (functions) that can be applied to the data structure. Objects provide an ideal mechanism for implementing abstract data types, which includes stacks, trees and hash tables. It offers multi threading system in building control within applications developed as compared to conventional language; thereby solving real life problems by making everything (process included) an object and having control reside within each object, i.e. at anytime multiple objects could be executing an operation and communicating with other objects with the concept of message passing; which is simply an invocation of an operation in another object.

In the other hand, transaction processing has been an important software technology for 40 years. Large enterprises in transportation finance, retail, telecommunications, manufacturing, governments, and the military are utterly dependent on transaction processing applications for electronic reservations, banking, stock exchanges, order processing, music and video services, shipment tracking, government services, telephone switching, inventory control and command control [5]. Transaction processing systems have to handle high volumes efficiently, avoid errors to concurrent operation, and producing partial results, grow incrementally, avoid downtime, never lose results, offer geographical distribution, be customizable, scale up gracefully, and easy to manage.

Transaction processing requires the execution of multiple operations and if runs, it must run in its entirety and for high performance, transactions must execute concurrently. It is now interesting to focuses on the harnessing object-oriented processing techniques for Transaction processing.

The objectives of this study among others are:

i. To identify object-oriented programming techniques for transaction processing.
ii. To identify base principles and concepts that are related to object – oriented programming.
iii. To relate the object – orientation mechanisms to transaction processing.
iv. To focus on preserving and guaranteeing important properties of the data objects (resources) accessed during a transaction.

The relevance of this study lies in structuring dynamically created objects with object-oriented notions with extends to include temporary association of data and operations. This aid in handling high volumes of transactions efficiently, avoid errors, avoid partial results, downtime and ultimately enhance the ease with which we manage transaction operations.

2.0 REVIEW OF RELATED LITERATURE

2.1 Object-Oriented Programming (Definition and Evolution)

Many scholars have defined object-oriented programming in unique way, but we will consider its definition by [6] which said that "A programming language is said to be object-based if it supports objects as a language feature, and is said to be object-oriented if, additionally, objects are required to belong to classes that can be incrementally modified through inheritance"

The definition above clearly depicts Object-oriented programming being based on the principles of object – orientation and is based on the object-oriented concepts. It is a new way of organizing and developing programs and has nothing to do with any particular language. This study paper defines object – oriented programming has encapsulation of data as well as operations applicable to that data into objects.

[3] argued that the early programmers thought of programs as instruction sequences. Procedure-oriented languages introduced procedural abstractions that encapsulate sequences of actions into procedures. The procedure-oriented paradigm has strong organizing principles for managing actions and algorithms, but has weak organizing principles for managing shared data.

Typed languages introduced the notion of a data type. A type characterizes a group of values, and a set of operations applicable to those values. During that period, objects provided define interface and hiding the internal implementation in what is termed abstract data types - ADT.

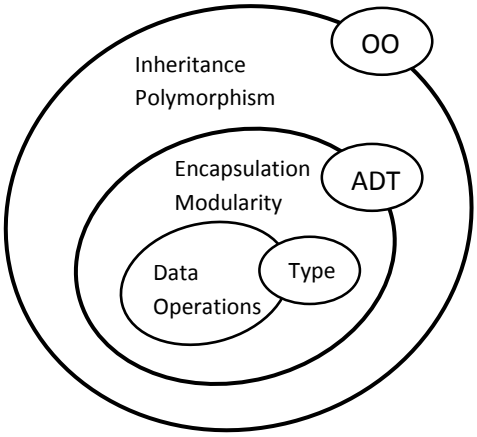


Fig 1: Evolution of Programming Language Concept.

Table 1: Evolution of Object – Oriented Programming Languages [7]

S/n	OOP Languages	Year	Description
1.	Simula	1967	Developed for simulating discrete Systems
2.	Smallalk	1970s	Developed for simulating graphics-oriented systems.
3	Ada	1980	A structured, statically typed, imperative, wide spectrum, and object oriented high-level computer programming language built to support extremely strong typing, explicit concurrency, offering tasks, and synchronism easy eg. Passing, protected objects and non-determinism.
4	C++	Early 198	A simpler version of c++ developed by sun Java. It is meant to be a programming language for video-on-demand applications and internet applications development.

Features and Concepts of Object Oriented Programming

The features of object – oriented programming offers robust techniques that give adequate support to Transaction processing. It creates a lot of ease in Transaction processing. It includes the concept of:

- **Abstraction:** The process of picking at (abstractly) common features of objects and procedures.
- **Class:** A category of objects. The class defines all the common properties of the different objects that belong to it.
- **Encapsulation:** The process of combining elements to create a new entity. A procedure is a type of encapsulation because it combines a series of computer instructions.
- **Information Hiding:** The process of hiding details of an object or function. Information undying is a powerful programming technique because it reduces complexity.
- **Inheritance:** A feature that represents the relationship between different classes.
- **Interface:** The languages and codes that the applications use to communicate with each other and with the hindrance.
- **Messaging:** Message passing it a form of communication used in parallel programming and object – oriented programming.
- **Object:** A self contained entity that consists of both data and procedures to manipulate the data.
- **Polymorphism:** A programming language's ability to process objects differently depending on their data type or class.
- **Procedure:** A section of a program that performs a specific task.

2.2 Transaction Processing (TP)

Definition and Evolution

Gray [8] defined a Transaction processing system (TPS) is an information processing system for business transactions involving the collection, modification and retrieval of all transaction data. Characteristics of a TPS include performance, reliability and consistency. TPS is also known as transaction processing or real-time processing. [5] x-rayed the origin of Transaction processing as shown in table 2.

Table 2: Evolution Of Transaction Processing – TP [5],

Year	Name	Manufacturers	Purpose
1960	IBM transaction Processing Facility (TPF)	IBM	Airtime control program (ACP)
1966	IBM information Management System (IMS)	IBM	Joint hierarchical data base with extensive capabilities
1969	IBM customer in frustration control system (CKS)	IBM	used for rapid, high volume online processing
1980s	Tuxedo Transactions for Unix, extended for distributed operative.	Oracle TPS	A Cross-platform
1970S	UNIVAC Transaction Interface package (TIP)	UNIVAC	A Transaction processing monitor
1980s	Burroughs corporation MCP	Burroughs	generalized message control system
1985	(DEC) Application control and management system (ACMS)	DEC	Creating and Controlling online transaction processing (OLTP).
1991	Transare Encina	IBM	A Transaction system

2.3 Features of Transaction Processing

The following features are considered important in evaluating transaction processing systems.

i. **Performance:** Fast performance with a rapid response time is critical. Transaction processing systems are usually measured by the number of transaction they can process in a given period of time.

ii. **Continuous Availability:** The system must be available during the period when the users are entering transactions. Many organizations rely heavily on their TPS; a breakdown will disrupt operations or even stop the business.

iii. **Data Integrity:** The system must be able to handle hardware or software problems without corrupting data. Multiple users must be protected from attempting to change the same example two operators cannot sell the same seat on an airplane.

iv. **Ease of User:** Often users of transaction processing systems are casual users. The system should be simple for them to understand, protect them from data entry errors as much as possible, and allow them to easily correct their errors.

v. **Modular Growth:** The system should be capable of growth at incremental costs, rather than requiring a complete replacement. It should be possible to add, replace, or update hardware and software components without shutting down the system.

2.4 Properties of Transaction

In conclusion, [5] said that a transaction is a unit of work that has the following properties:

i. **Atomicity:** A transaction should be done or undone completely and unambiguously.

ii. **Consistency:** A transaction should preserve all the invariant properties (such as integrity constraints) defined on the data.

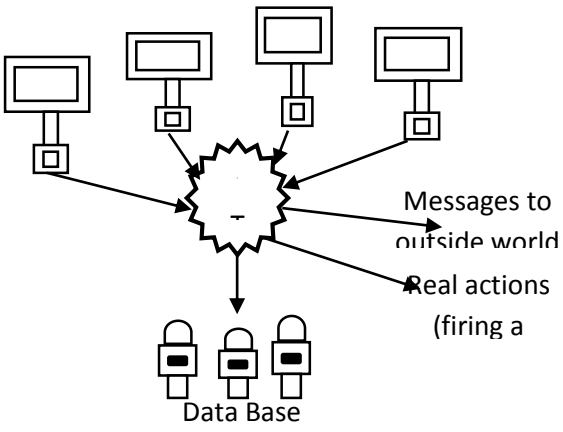
iii. **Isolation:** Each transaction should appear to execute independently of other transactions that may be executing concurrently in the same environment.

iv. **Durability:** The effects of a completed transaction should always be persistent.

2.5 Types of Transaction Processing

i. **Processing in batch:** Transactions may be collected and processing. Transactions will be collected and later updated as a batch when it is convenient or economical to process them. Historically, this was the most common method as the information technology did not exist to allow real-time processing.

ii. **Processing in real-time**: This is the immediate processing of data. It provides instant confirmation of a transaction. It may involve a large number of users who are simultaneously performing transactions which change data. Because of advances in technology (such as the increase in the speed of data transmission and larger band-with), real-time update in possible.



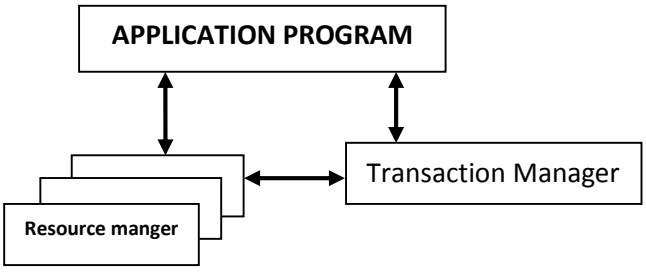


Fig. 2: Simplified explanation of TPS

Fig.3: Issues in building transaction applications:

To understand the issues involved in building Transaction applications, consider an order process application with the architecture shown in figure 3.

2.5 Standard Operations in Transaction Processing:

Transaction processing scheme is supported by the following standard primitive operations: Begin, Comit, and Abort.

After beginning a new transaction, all update operations on transaction objects are done on behalf of that transaction. At any point during the execution of the transaction if can abort and once a transaction has completed successfully is committed, the effects become permanent and visible to the outside [8].

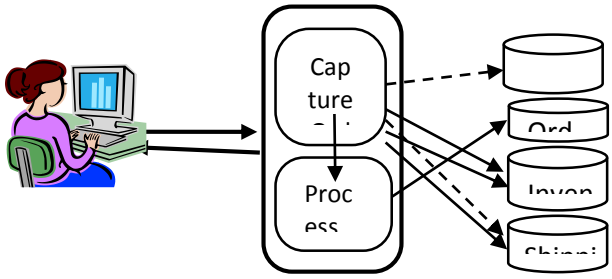
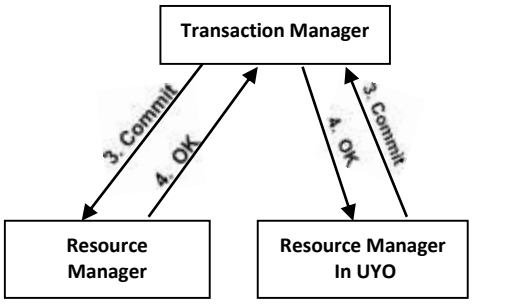
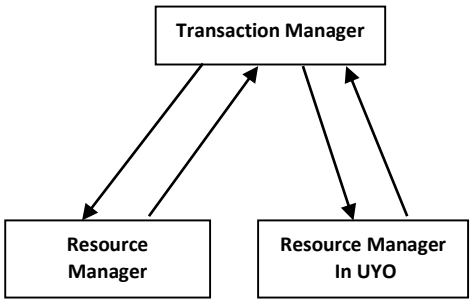


Fig. 4: ***X/Open Transaction model (XA).***

The transaction manager process starts, commit, and abort. It talks to resource managers to run Two Phases commit [5].



Phase 1



Phase 2

Fig. 5: ***The Two-phase commit protocol***.

In phase 1, every resource manager durably saves the transaction's updates before replying "I am prepared". Thus, all resources managers have durably stored the transaction's updates before any of the commits in phase 2 [5].

3.0 METHODOLOGY

3.1 Data Collection

We studied harnessing object – Oriented programming techniques for Transaction processing with data from two main sources thus:

a. Primary Source: We carried out a study using a questionnaire with a 10- point items. The 10-point items were structured to achieve the operational objectives of the study using the modified 5 Likert Scale of Strongly Agreed (SA), Agreed (A), Undecided (UD), Disagreed (D), Strongly Dis agreed. A total of 125 respondents out of which 10 were lecturers, 15 were freelance programmers in Port Harcourt and 100 were final year students of computer science. These respondents were selected from three different universities namely: University of Portharcourt, Portharcourt, University of Calabar, Calabar, University of Uyo, Uyo all in Nigeria, and the questions sought the views of the above named groups of persons' on harnessing object – oriented programming techniques for Transaction processing.

b. Secondary Source: We extract information from existing from existing computer science journals, text books, laboratory manuals and manuscripts, etc. The internet was a major source of the secondary data. People views were equally considered.

3.2 Data Analysis and Results Presentation

Table 3 shows the occupational distribution of the interviewee. The opinion of 125 respondents were sampled and responses collected and analyzed on a 5 – point likert type scale as shown in table 4.

Table 3: Occupation distribution of interviewed respondents

S/n	Respondents Occupation	No.	Percentage (%)
1.	Software Developers	15	12%
2.	Lecturers	10	8%
3.	Final Year Students	100	80%
	Total:	**125**	**100%**

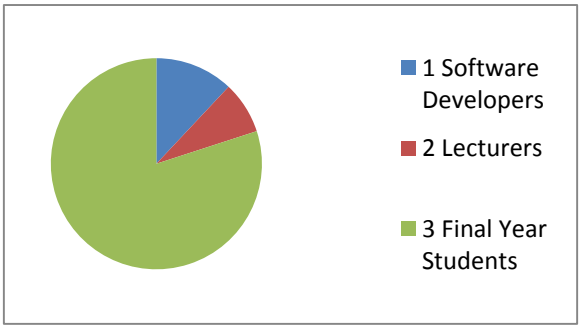


Fig. 6: Pie Chart showing the occupational distribution of interviewed respondents.

Table 4: Shows the opinion of the respondent to the questions presented by the interviewer. These questions were presented to freelance programmers, Lecturers and final year students of computer science department selected from the three Nigerian Universities mentioned in section 3.1.

Table 4: Questions and responses by respondents

S/N	QUESTIONS	X	F	FX	*X* (MEAN)	%
1.	Objects oriented programming languages encourage programming in modules?					
	• Strongly Agree	5	100	500	4.80	80.00
	• Agree	4	25	100		20.00
	• Undecided	3	0	0		0.00
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	0	0		0.00
2.	Object oriented programming languages have concept of objects, class, data abstraction, inheritance, and polymorphism?				4.62	
	• Strongly Agree	5	83	417		66.67
	• Agree	4	34	136		26.67
	• Undecided	3	8	25		6.67
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	0	0		0.00
03.	Program can be divided into modules of task and tested separately in OOP?					
	• Strongly Agree	5	100	500		80.00
	• Agree	4	17	68	4.74	13.33
	• Undecided	3	8	24		6.67
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	0			0.00
4.	OOP has Internals Process Concurrency?					
	• Strongly Agree	5	83	415		66.67
	• Agree	4	25	100	4.53	20.00
	• Undecided	3	17	51		13.33
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	0	0		0.00
5.	OOP has Inter-Process Communication?					
	• Strongly Agree	5	17	85		13.33
	• Agree	4	17	68	3.41	13.33
	• Undecided	3	91	273		73.33
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	0	0		0.00
6.	OOP supports shared data structure?					
	• Strongly Agree	5	25	125		20.00
	• Agree	4	42	168	3.34	33.33
	• Undecided	3	17	51		13.33
	• Disagree	2	33	66		26.67
	• Strongly Disagree	1	8	8		6.67
7.	Transactions allow shared data structures?					
	• Strongly Agree	5	67	335		53.33
	• Agree	4	58	232	4.54	46.67
	• Undecided	3	0	0		0.00
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	0	0		0.00
8.	Conventional objects bind data structures to operations in a permanent union?					
	• Strongly Agree	5	67	335	4.34	53.33
	• Agree	4	42	168		33.33
	• Undecided	3	8	24		6.67
	• Disagree	2	8	16		6.67
	• Strongly Disagree	1	0	0		0.00
9.	Are transactions object-based?					
	• Strongly Agree	5	58	290		46.67
	• Agree	4	42	168	4.14	33.33
	• Undecided	3	17	51		13.33
	• Disagree	2	0	0		0.00
	• Strongly Disagree	1	8	8		6.67

10.	Transactions may be viewed as extending traditional object-oriented notions to include dynamically created objects?					
	• Strongly Agree	5	50	250	3.86	40.00
	• Agree	4	33	132		26.67
	• Undecided	3	17	51		13.33
	• Disagree	2	25	50		20.00
	• Strongly Disagree	1	0	0		0.00

4.0 DISCUSSION AND EVALUATION

Most of the questions (8 – in number) listed for the opinion of the respondents had high arithmetic mean as shown in Table 4. This showed that greater number of the respondents agreed that OOP has a positive impact on Transaction processing. Questions five (5) and six (6) have arithmetic mean of 3.41 and 3.34 respectively which depicts the fact that the respondents do not see the tested parameters to have a positive impact that harness OOP techniques for Transaction processing.

About 80% of the respondents opined that Transaction processing is object-based hence the use of OOP has brought an increase in the production of millions of Transaction processing packages. Harnessing object – oriented programming techniques for Transaction processing would ensure the development of better and more efficiet transaction processing packages. In a nutshell, object – oriented programming has brought the dawn of a new epoch in Transaction processing. The key techniques of OOP (namely: abstraction, classes, encapsulation, information hiding, inheritance, interface, messaging, objects, polymorphism and procedure) has been shown to enhance transaction processing.

5.0 CONCLUSION

The discourse in this study centered on harnessing object – oriented programming techniques for transaction processing. In doing this, we espouse different object – oriented programming techniques in line with the objectives of the study. Data from respondents were presented and analyzed with statistical tools and charts. The analysis showed that object-oriented processing techniques are very suitable for transaction processing. The base principles and concepts as it is related to object – oriented programming, relate the object – orientation mechanisms to transaction processing and focus on preserving, and guaranteeing important properties of the data objects (resources) accessed during a transaction.

6.0 REFERENCES

[1] Wegner, P. (1991): Perspectives on Object – Oriented Design. *Technical Report No. CS -91 – 01*; Rhode Island, USA, 1991.

[2] Booch, G. (1994): *Object-oriented Analysis- 2nd Edition*. Cummings Publishers, Redwood city, USA.

[3] Kienzle, J. (2001): Open Multi-threaded transaction: A Transaction Model for Concurrent Object Oriented Programming. Cannes, France.

[4] Meyer, B.: *Object – oriented software construction, 2nd edition*. ISE, Santa Barbara, California, USA.

[5] Bernstein, P.A; Newcomer, E. (2009): *Principles of transaction processing,* 2nd edition. Morgan Kanfamann Publishers, USA.

[6] Wegner, P. (1990): Concepts and paradigms Expansion of Oct 4 OOPSLA – *89 Keynote Talk*, USA.

[7] Onu, F. U, Osagie, S.U, John- Otumu. M.A, Igboke M. E (2015): OOP and its calculated measures in Programming Interactivity. *Journal of Mobile Computing & Application (IOSR- JMCA), vol.2, pp.26-34*

[8] Gray, J. and Reuter, A. (1993): *Transaction processing: concepts and Techniques*. Morgan Kanfamann Publishers, USA.

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 539 - 542, 2016, ISSN:- 2319–8656

Image Processing Approach for INR Currency Note Number Recognition System for Automated Teller Machines

Rashmi C
High Performance Computing
DOS in CS
University of Mysore
Mysuru, India

Dr. Hemantha Kumar G
High Performance Computing
DOS in CS
University of Mysore
Mysuru, India

Abstract*:* In this paper an algorithm is proposed for real time application to an existing automated teller machine (ATM) using image processing in currency note number recognition. Sometimes getting fake currency notes from ATM nowadays has become a major issue leading to an ultimate loss to common people. Common people are afraid to complain regarding this to respective banks and live in hope that may recover their money from respective banks. An algorithm is developed for automatically noting and saving the currency note number to server at the time of withdrawal, so that customers are benefitted in recovering their currency back. The proposed algorithm makes the ATMs more reliable, user friendly and efficient usage to the customers.

Keyword: Character recognition, Currency note number, Image Processing, Fake Currency, ATM.

1. INTRODUCTION

Fake currency notes from ATMs are becoming a major problem across the country, which has created an alarming situation for the banking system. Such kind of currency notes from ATMs shows the strong flow of fake notes. Common man who is drawing notes from ATMs is in loss. Banking system is not affording to take the responsibility of such fake currency from ATM which does not produce effective evidence. Banks are rejecting such situations that customers are getting fake notes from their ATM machines. Since, it's very difficult to prove the source of fake notes, so at last the person gets nothing even after a complaint. This can be overcome by noting the currency note numbers at customer side and at bank databases. As manual process of noting down the INR (Indian Rupee) currency note numbers both at the customer side and by bank authority is tedious. An automated system is proposed in this paper for reading the INR note numbers at the ATM machines, using image processing. As we all know that character recognition, online/offline handwriting recognition and printed character recognition are the research areas that have been received a lot of attention since 1960's which plays an important role in industrial applications and financial transactions. Many research articles have been proposed which results in the fields of bank cheque
Processing [1], Zipcode [2], car license plate recognition [3].Minimal studies are carried out on automatic recognition of bank note serial numbers [4][5][6],which is beneficial in reducing financial crime.

The proposed system investigates INR serial numbers, which are printed at right top corner and bottom left corner on Indian currency note numbers consists of 9 characters first three are prefix and the last six are the serial number. The first character of the prefix is numeral and next two are alphabets. Currency note number will be extracted and recognized from the image and is noted, stored in database of respective banks.

The flow of research article is Section II presents the existing system; Section III is the proposed system for currency note number and reading mechanism which will be followed by sequence diagram for transactions in ATM machines. Section IV describes experimental results.

2. EXISTING SYSTEM

ATMs are one of the real-time front terminals which are supported by the central bank server and a centralized account database [7].The existing machine is simple in operating which follows basic steps that include insertions of ATM card to machine, enter PIN, select money to be withdraw and exit.ATM machine also offers cheque deposit facility through cheque box, where the customer can do transactions as needed. People also use ATM cards for online bill payments such as electricity and telephone bills. ATM is the most convenient to access the accounts and financing transactions. The given existing ATM algorithm and flow chart describes the operation of ATM as in figure 1 during withdraw of notes from the ATM machine. However the machine doesn't know that the note is fake or original notes.

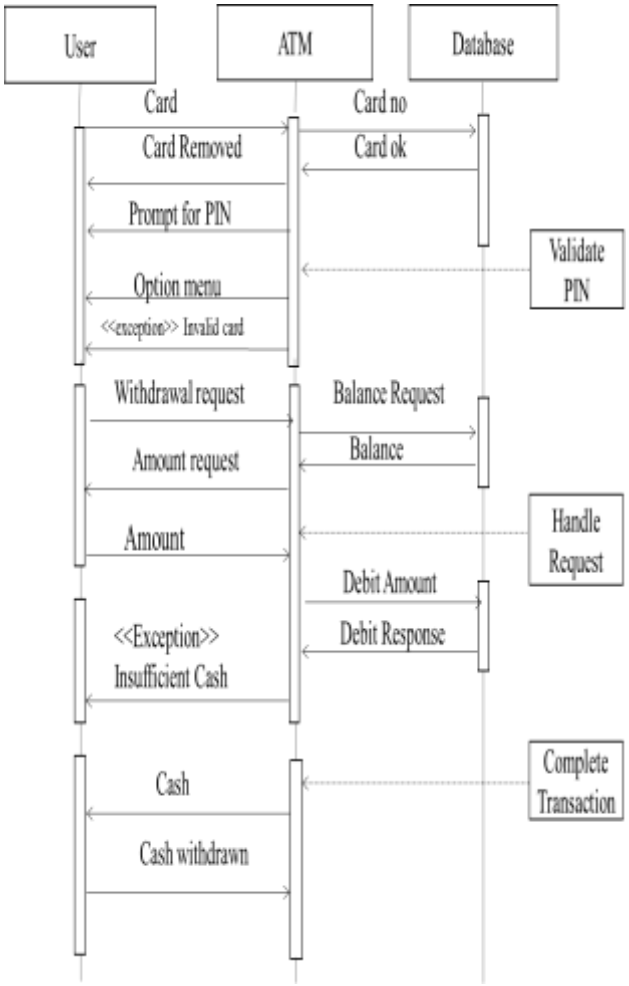


Fig. 1. Existing ATM Sequence Diagram using Pin

1. Customer swipes the ATM card in the Machine.
2. System will validate card number prompts for entering the PIN by customer.
3. Customer enters PIN number and system validates PIN.
4. The System provides an option to the customer whether to withdrawal or to check balance etc.
5. Customer will enter the amount and selects the submit Option on the cash withdrawal screen.
6. The amount entered by the customer will be verified by the machine for availability of cash requested by the customer and asks for the acknowledgement receipt of the transaction.
7. The customer selects 'Yes' on the Screen.
8. The system provides the cash, prints the receipt on customer request.

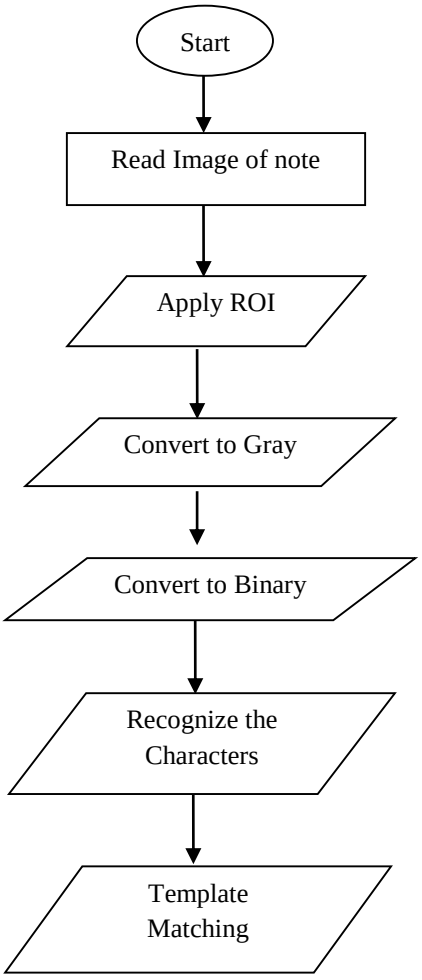
3. PROPOSED SYSTEM

The research article is proposed to design and enhance security and satisfaction of the customer by avoiding any fake note during withdrawals from the ATM. For an existing ATM system we are proposing and image processing based algorithm for reading mechanism of INR currency note number which is capable of storing that number in the banks database and as well as customer acknowledgement receipt during the time of withdrawals from the ATMs.

A simple algorithm is designed in image processing for reading the INR note number.The DIP is an area characterized by the need for extensive experimental work to establish the validity of proposed solutions to a given problem [9]. It encompasses processes whose inputs and outputs are images and encompasses processes that extract attributes from images up to and including the recognition of individual objects. This proposed system can be applied for Fake currency detection and counting machines respectively.

The proposed work is suitable for real time application system which will work on an image captured either by scanner or camera. The algorithm for discussed currency note number is implemented as follows.

a. Image of paper currency will be acquired by simple scanner or digital camera.
b. The image acquired is RGB image and then it will be converted into gray scale.
c. Resize whole image and apply ROI for INR note number.
d. Convert to Binary
e. Recognise the characters.
f. Load Template
g. Extract letter
h. Resize letter same as the template
i. Match the extracted letters with the template.
j. Display numbers and print.



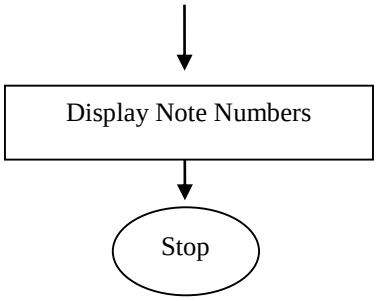


Fig. 2. Flow Chart for Currency Note no. recognition

The proposed ATM algorithm and Sequence diagram [10] describes the operation of ATM during withdrawal of currency notes by the customer from the ATM machine.

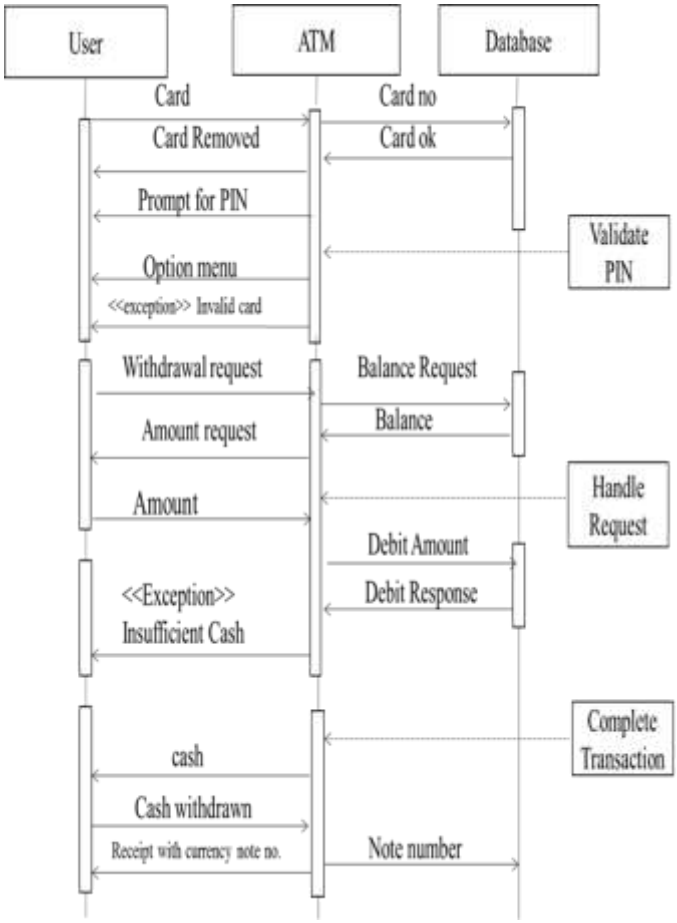


Fig. 3. Cash Withdrawal from ATM by Customer

The following describes the procedure stated in the flow diagram as illustrated in Figure 2:

1. Customer swipes the ATM card in the Machine.
2. System will validate card number prompts for entering the PIN by customer.
3. Customer enters PIN number and system validates PIN.
4. The System provides an option to the customer whether to Withdrawal or to check balance etc.
5. Customer will enter the amount and selects the submit Option on the cash withdrawal screen.
6. The amount entered by the customer will be verified by the machine for availability of cash requested by the customer and asks for the acknowledgement receipt of the transaction.
7. The customer selects 'Yes' on the Screen.
8. The system asks the customer to collect the cash, prints the receipt with currency note numbers and updates the account balance of the customer and also in their respective bank database.

Advantages

Makes the banking system more reliable and user friendly especially to customers.

4. EXPERIMENTAL RESULTS

The experimental results for reading the Indian currency note number is conducted offline as per the algorithm proposed in section III using image processing. The results has been tested on an Intel core i3 CPU, 3.20GHz, 2GB RAM. For experimental purpose ten notes of one thousand rupee currency note is considered for reading its serial number which is printed at right top corner of currency note. Following figure illustrates the experimental results.

Fig. 4. Input Image

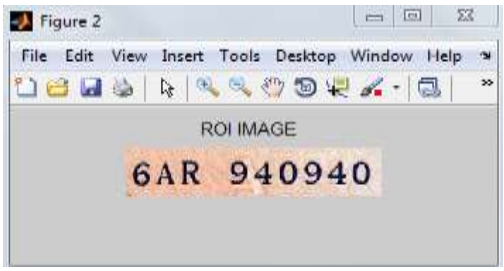


Fig. 5. ROI Extracted

Fig. 6. Binary Image

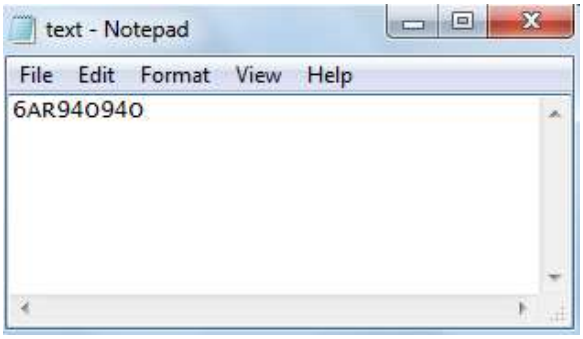


Fig. 7. Number Panel display of INR note

5. CONCLUSION AND FUTURE WORK

The proposed system mainly focuses on reading mechanism of Indian currency note number recognition using image processing for the existing ATM machines. This proposed system make the machines more efficient and user friendly especially to customers at the time of withdrawing money from ATM machines. The algorithm developed is tested for 1000 rupee notes which provides an accuracy of 86% for serial number extraction of Indian rupee currency note number and takes 0.568079 seconds for its execution.

Reading Mechanism for recognizing the currency note number for 500,100 Indian currency denomination has to be enhanced. This proposed system can be implemented to fake currency detecting machines and counting machines respectively. Additional study should be done on ATM network and server according to proposed system in this paper using parallel computation.

6. REFERENCES

[1] QizhiXu,LouisaLam,ChingY.Suen,"Automatic segmentation and recognition System for hand written dates on Canadian bank cheques" in proceedings of the 7th International Conference on Document Analysis and Recognition,Edinburg, UK, 2003,pp.704-708.

[2] YannLeCun,O.Matan,B.Boser,J.S.Denker,D.Henderson,R.E.Howard,W. Hubbard,L D Jacket, L.D.Jacket,H.S.Baird "Hand written zip code recognition with multilayer networks" in proceedings of the 10th International Conference on Pattern Recognition ,vol.2,Atlantic City,USA,1990,pp.35-40.

[3] Shyang-LihChang,Li-ShienChen,Yun-Chung Chung,Sei-WanChen ,"Automatic license plate recognition", IEEETrans.Intell.Transp.Syst.(2004)42–53.

[4] Ting-tingZhao,Ji-yinZhao,Rui-ruiZheng,Lu-lu Zhang ,"Study on RMB number recognition based on genetic algorithm artificial neural network", in: Proceedings of the 3rd International Congress on Image and Signal Processing, vol.4, Yantai,China, 2010, pp.1951–1955.

[5] WenhongLi,WenjuanTian,XiyanCao,ZhenGao, "Application of support vector machine(SVM)on serial number identification of RMB", in:Proceedings of the 8th World Congress on Intelligent Controland Automation, Jinan,China,2010,pp.6262–6266.

[6] Bo-Yuan Feng ; Dept. of Comput. Sci., Nanjing Univ. of Sci. & Technol., Nanjing, China ; Mingwu Ren ; Xu-Yao Zhang ," Extraction of Serial Numbers on Bank Notes".

[7] Yingxu Wang, Yanan Zhang, "The Formal Design Model of an Automatic Teller Machine (ATM)" International Journal of Software Science and Computational Intelligence, 2(1), 102-131, January-March 2010.

[8] Binod Prasad Yadav, C. S. Patil, R. R. Karhe, P.H Patil. "An automatic recognition of fake Indian paper currency note using MATLAB",International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 3, Issue 4, July 2014

[9] Shailesh N. Sisat, "Duplicate and fake currency note tracking in Automated teller machine(ATM), International Journal of Electronics and communication engineering and technology(IJECET),Volume 5,Issue 1,January 2014,pp. 11-15.

[10] Software Engineering : Seventh Edition Ian Sommerville Pearson Education India,2004.

[11] Rubeena Mirza, Vinti Nanda,"Paper currency verification system based on characteristic extraction using image processing", International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-3, February 2012

[12] Woods and Gonzalez (2008), Digital Image Processing (Third Edition), Pearson Education, New Delhi, 110092.

[13] Rubeena Mirza, Vinti Nanda ,"Paper Currency Verification System Based on Characteristic Extraction Using Image Processing",International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-3, February 2012.

[14] H. Hassanpour ,A. Yaseri, G. Ardeshiri ―Feature Extraction for Paper Currency Recognition‖, IEEE Transactions, 1-4244-0779-6/07,2007.

[15] Rajesh Kannan Megalingam, Prasanth Krishna, Pratheesh somarajan, Vishnu A Pillai, Reswan Hakkim ―Extraction of License Plate Region in Automatic License Plate Recognition‖, International Conference on Mechanical and Electrical Technology, IEEE Transactions, 978-1-4244-8102-6/10

[16] Adam Coates, Blake Carpenter, Carl Case, Sanjeev Satheesh, Bipin Suresh, Tao Wang, David J. Wu, Andrew Y. Ng Computer Science Department Stanford University 353 Serra Mall Stanford, CA 94305 USA,"Text Detection and Character Recognition in Scene Images with Unsupervised Feature Learning".

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 543 - 550, 2016, ISSN:- 2319–8656

Simulation of N-Way Traffic Lights Using Arduino Uno Environment

Ogwo Eme
Department of Computer Science, Akanu Ibiam Federal Polytechnic Unwana, Ebonyi State, Nigeria

Chibuike .E. Madubuike
Department of Computer Science, Akanu Ibiam Federal Polytechnic, Unwana, Ebonyi State, Nigeria

Joseph .O. Idemudia
Department of Computer Science, Akanu Ibiam Federal Polytechnic Unwana, Ebonyi State, Nigeria

Akaninyene Udo Ntuen
Department of Computer Science, Akanu Ibiam Federal Polytechnic Unwana,Ebonyi State, Nigeria

Abstract: Traffic is a major challenge in many cities in the world today. In many instances traffic flow is dictated by certain devices such as traffic lights. Traffic lights have become an integral part of human day-to-day life. This work focuses on the simulation of traffic light system with a microcontroller programmed on the Arduino Uno board. This system was designed to handle road traffic control as well as assisting pedestrians to move freely without auto crashes. The traffic lights simulation system was implemented using an Arduino Uno microprocessor connected to electronic circuit board. The programming platform for the simulation of the traffic light system was done using C++ programming language while the electronic circuit used was designed with some semiconductor components such as transistors, microprocessor, resistors and light emitting diodes (LEDs) to achieve optimal performance of the traffic lights.

Keywords: Embedded System, Light Emitting Diode (LED), Microcontroller, Simulation, Traffic Light.

1.0 INTRODUCTION

The tremendous increase of the number of vehicles on our roads and the over-whelming array of pedestrians on road side ways call for great concern. The numerous avoidable accidents at intersections be it T-junction or 4-way-junction is acknowledged not only by government but also other users of the roads. This situation calls for remedy or assistance of some sort not only to save lives but also to ensure orderliness in our everyday life. Electronic devices, which can do services round the clock, throughout the whole season of the year, are call to play this indispensable role.

Traffic is a problem in many urban areas worldwide. In more civilized cities, traffic flows are mostly dictated by certain devices such as traffic lights. These traffic lights are often held on poles at intersections of road.

Furthermore traffic light has become an integral part of human day to day life. With the advancement in semiconductors technology and a parallel rise in innovation, embedded technology has brought about a lot of change in lighting engineering. With this motivation in mind, this work aims at designing and implementing a running model of traffic light controller which will help to control the density of vehicular movements on the road.

Traffic lights - also known as traffic signals, traffic lamps, traffic semaphore, signal lights, stop lights, and (in technical parlance) traffic control signals- as signaling devices positioned at road intersections, pedestrian crossings, and other locations to control flows of traffic [1]. Traffic lights alternate the right of way accorded to road users by displaying lights of a standard color (red, yellow/amber, and green) following a universal color code. In the typical sequence of color phases:

- The green light allows traffic to proceed in the direction denoted, if it is safe to do so and there is room on the other side of the intersection.
- The amber (yellow) light warns that the signal is about to change to red. In a number of countries – among them the United Kingdom – a phase during which red and amber are displayed together indicates that the signal is about to change to green. Actions required by drivers on an amber light vary, with some jurisdictions requiring drivers to stop if it is safe to do so, and others allowing drivers to go through the intersection if safe to do so.
- A flashing yellow indication is a warning signal. In the United Kingdom, a flashing amber light is used only at pelican crossings, in place of the combined red–amber signal, and indicates that drivers may pass if no pedestrians are on the crossing.
- The red signal prohibits any traffic from proceeding.
- A flashing red indication is treated as a stop sign.

In some countries traffic signals will go into a flashing mode if the controller detects a problem, such as a program that tries to display green lights to conflicting traffic. The signal may display flashing yellow to the main road and flashing red to the side road, or flashing red in all directions. Flashing operation can also be used during times of day when traffic

is light, such as late at night. This is the case in many American States [2].

This work focuses on the simulation of traffic light system with a microcontroller programmed on the Arduino Uno board. This system was designed to handle road traffic control as well as assisting pedestrians to move freely without auto crashes. The traffic lights simulation system was implemented using an Arduino Uno microprocessor connected to electronic circuit board.

2.0 REVIEW OF RELATED LITERATURE

This section discusses related literature and contributions of other authors as it relates to the topic under consideration.

The first electric traffic light was developed in 1912 by Lester Wire, a policeman in Salt Lake City, Utah, who also used red-green lights. On 5 August 1914, the American Traffic Signal Company installed a traffic signal system on the corner of East 105th Street and Euclid Avenue in Cleveland, Ohio [3]. It had two colors, red and green, and a buzzer, based on the design of James Hoge, to provide a warning for color changes. The design by James Hoge allowed police and fire stations to control the signals in case of emergency. The first four-way, three-color traffic light was created by police officer William Potts in Detroit, Michigan in 1920.

Los Angeles installed its first automated traffic signals in October 1920 at five locations on Broadway. These early signals, manufactured by the Acme Traffic Signal Co., paired “Stop” and “Go” semaphore arms with small red and green lights. Bells played the role of today's amber or yellow lights, ringing when the flags changed—a process that took five seconds. By 1923 the city had installed 31 Acme traffic control devices. The Acme semaphore traffic lights were often used in Warner Bros. Looney Tunes and Merrie Melodies cartoons for comedic effect due to their loud bell [4].

According to [5], the first interconnected traffic signal system was installed in Salt Lake City in 1917, with six connected intersections controlled simultaneously from a manual switch. Automatic control of interconnected traffic lights was introduced March 1922 in Houston, Texas.

According to [1], traffic control systems are the most visible element of the urban infrastructure. They are not just physical systems like telephones or sewers or streets, although their technological elements, traffic lights, signs, and painted pavements, fit that description. Rather, they are systems that attempt to impose a strong social control over the most fundamental of human behaviors, whether to move or be still. Traffic engineers must control police, drivers, and pedestrians. For most other elements of the urban infrastructure, controlling the behavior of users did not constitute the primary goal of designers. For traffic engineers, understanding and manipulating the behavioral patterns of drivers and pedestrians (a group that included not just walkers, but people using the street for play, social gatherings, and commerce) proved to be a more important problem than the control mechanisms themselves. Traffic engineers learned rapidly to pay careful attention to ergonomics, the interface between people and machines.

Reference [6], advocated for the use of clever traffic light as a solution for traffic control for pedestrian crossing. According to them the sequence of switching signals of usual traffic light simplistically may be represented in the following ways:
• drivers go;
• both (drivers and foot-passengers) wait;
• foot-passengers go;
• both (drivers and foot-passengers) wait;
• repeating cycle.

Clever traffic-light works not in the least like that its possession of information about cars on the road and foot-passengers on pedestrian crossing. In presence of some foot-passengers and cars, traffic-light works like common traffic-light. If there are no cars, but foot-passengers are waiting on pedestrian crossing, it will indicate the green light for the pedestrian until at least one car appears, and vice versa. If there is neither car no foot-passengers, green light will be indicated for cars because they need more time to slow down and to pick up speed to continue the motion.

Reference [7] viewed optics and lighting as traditionally incandescent and observed that halogen bulbs where used in constructing traffic light. Because of the low efficiency of the light output and a single point of failure (Filament burnout) municipalities are increasing retrofitting traffic signals with LED arrays. Unlike incandescent and halogen bulbs, which generally get hot energy and melt, LEDs consume less power which have increased light output and in the event of an individual LED failure, still operate with a reduced light output. With the used of optics, the light pattern of LED array can be comparable to the pattern of an incandescent or halogen bulb.

According to [8], the conventional light system such as traffic signal lighting which is still common in some areas, utilizes a standard light bulb typically 67 watts, 69 watts or 115 watts medium-based light bulbs (house hold lamp in the US). Light bulb provides the illumination light then bounces off a mirrored glass or polished aluminum reflector bowl, and out through a polycarbonate plastic or glass signal lens. In some signals, these lenses were cut to include a specific refracting pattern. Crouse-Hinds is one notable company for this practice between the 1930s and 1950s, they utilized a beaded prismatic lens with a “Smiley” pattern embossed into the bottom of each lens.

Traffic light design in the United States, that traffic lights are currently designed with lights approximately 12 inches (300mm) in diameter. Previously the standard has been 8inches (200mm), however, those are slowly being out in favor of the larger and more-visible 12inch lights. Variations used have also include a hybrid design which had one or more 12inch light alone with one or more lights of 8inch (200mm) on the same light for example, those “12-8-8” (along with 8-8-8) light are standard in most jurisdictions in Ontario, Manitoba and British Columbia (i.e., the red light is 12 and others 8, making the read more prominent) [9].

It had been observed that as technological advancement with technologies in developed countries continuing to

advance, there is now an increasing move to develop and implement smart traffic light on the roads. These are basically intelligent systems that try to communicate with cars to alert drivers of impending light changes and reduce motorists waiting time considerably. Trails are currently being conducted for the implementation of these advanced traffic light but there are still many hurdles to widespread use that need to be address, one of the fact that not a lot of cars yet have the required system to communicate intelligently with these lights [10].

A microcontroller (abbreviated with µC or MCU) is a small computer on a single integrated circuit containing a processor core, memory and programmable input and output peripherals. The program memory in the form of NOR flash or EPROM is also included on the chip, as well as a typically small amount of RAM. Microcontroller is designed for embedded application, in contrast to the microprocessor used in personal computers or other general purpose applications. Microcontrollers are used in automatically controlled products and devices, such as automobile engine control system, implantable medical devices, remote controls, office machines electronic appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that use a separate microprocessor, memory input and output devices, microcontrollers make it economical to digitally control even more devices and processes. Mixed signal microcontrollers are common integrating analog components needed to control non-digital electronic system [11].

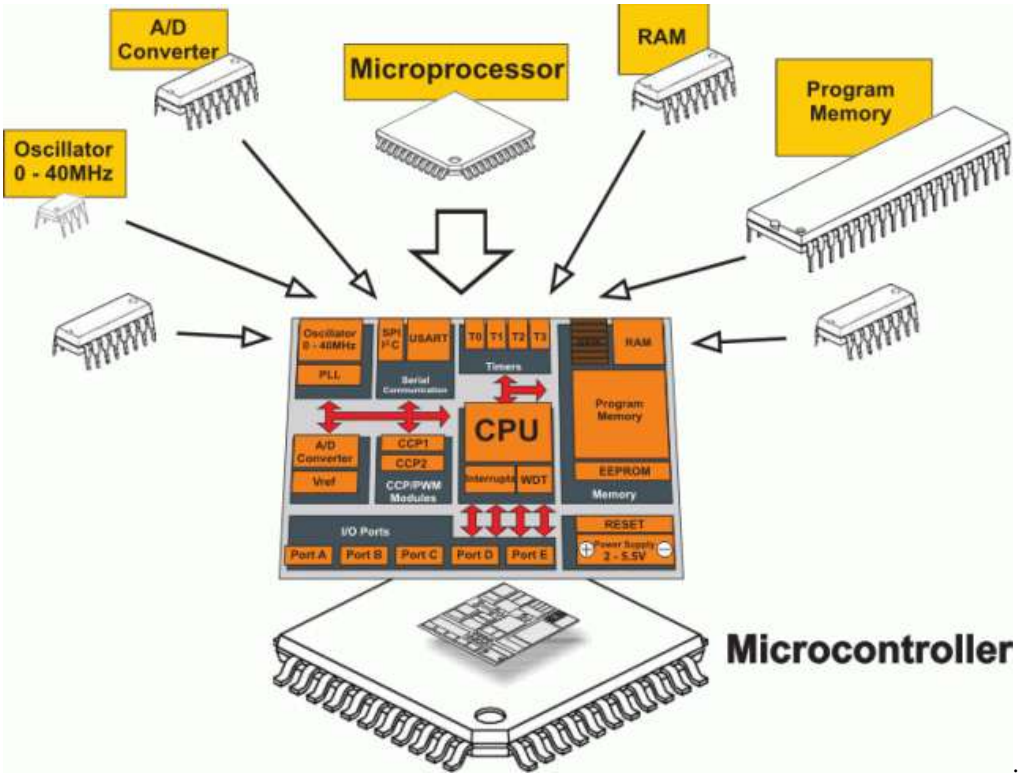


Fig.1: Block Diagram of a microcontroller
Source: [12]

As advancement in embedded systems and microcontroller continues to improve the existing road traffic system, it has become imperative for different regions and countries to develop their low cost indigenous system which will support their local traffic policy and will be easy to maintain. The need to integrate local content in traffic light system has necessitated this Arduino uno microcontroller based traffic light simulation.

3.0 METHODOLOGY

In this work, Arduino Uno microcontroller was used to simulate the flow of signals in the traffic light. Arduino, in general is an open source platform targeting hobbyists. An open source is a method where the developers of a software/hardware give the end user access to their end products design and implementation. This means that the end users have right to modify/change the way the software looks/works and redistribute it. Examples of open source software are Linux operating system, free PBC, and open BSD etc. Furthermore, Arduino is an open source hardware and software platform which uses Atmel microcontroller as core hardware component and C or C++ as core software language and which is used to control LEDs, motors, displays and any hobby projects.

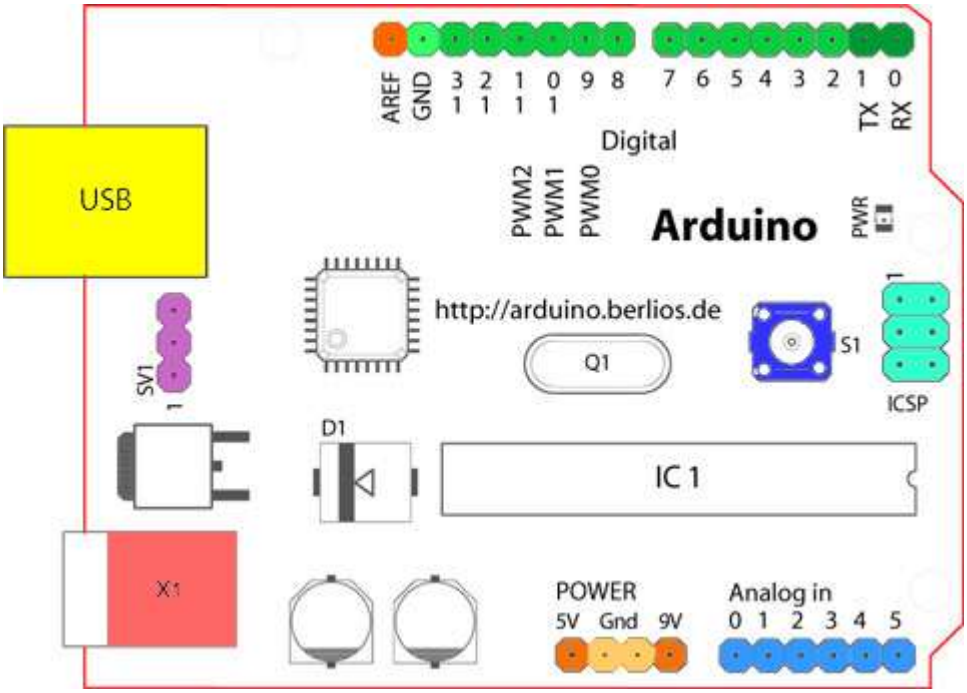


Fig.2: Pin Diagram of an Arduino Uno Board
Source: [13]

Starting clockwise from the top center:

- Analog Reference pin (orange)
- Digital Ground (light green)
- Digital Pins 2-13 (green)
- Digital Pins 0-1/Serial In/Out - TX/RX (dark green) - These pins cannot be used for digital i/o (digitalRead and digitalWrite) if you are also using serial communication (e.g. Serial.begin).
- Reset Button - S1 (dark blue)
- In-circuit Serial Programmer (blue-green)
- Analog In Pins 0-5 (light blue)
- Power and Ground Pins (power: orange, grounds: light orange)
- External Power Supply In (9-12VDC) - X1 (pink)
- Toggles External Power and USB Power (place jumper on two pins closest to desired supply) - SV1 (purple)
- USB (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)

The Arduino software consists of two parts: Arduino Boot-Loader and Arduino IDE. Arduino Boot-Loader is a piece of code residing inside the microcontroller which make the controller special and gives it the power of integration to the Arduino IDE and the Arduino board. The Arduino IDE has a compiler, serial monitor etc. Arduino language is a variant of C++ at least it looks like C++ Programs.

Apart from the basic Arduino, other tools that were used include:

- A red, yellow and green LED.
- A breadboard.
- Resistors for the LEDs (220 Ohms).
- Connecting wires.
- A pushbutton switch.
- A high value resistor (10k).

The output terminal can either be the source or the sink current. The maximum sink or source current is about 40MA. The high output is about 0.5v below V_{cc} while the low output voltage is about 0.1V about ground for load current below 25mA. The positive voltage supply terminal can take any voltage between +5v and +18v.

4.0 SYSTEM DESIGN

This is aimed at producing a specification that will enable the controller to keep accurate implementation of the new system. The design is based on the simulation of a workable traffic light control system using an Arduino Uno Microcontroller as illustrated by the schematic diagram in Fig. 2.

4.1 Input Design

The input to the system is through sending of digital signals in the Arduino IDE. This is done by implementing digital write and the delay of signals using the delay time in milli-seconds since the Arduino understands this. Therefore, the Green was delay for 30,000m_s (i.e 15s), then the yellow will blink for about five times each in 250m_s (i.e. 0.25s) which will display at the N- cross Intersection.

The Assembly was done by programming the Arduino board which is used to display the LEDs at each function. The LEDs were soldered on a PBC which are hung on the plastic with connectors passing through to the Arduino output pins.

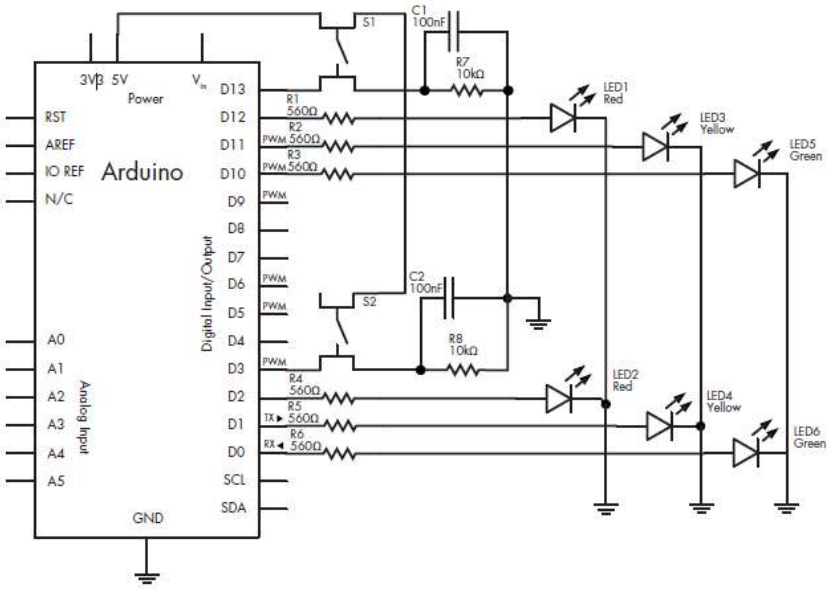


Fig.3: The schematic design of the Arduino uno microcontroller traffic system

4.2 Simulation of the Traffic Light LEDs

For purpose of the design, the display panel is made up of Light Emitting Diodes (LEDS). The display is used to indicate the status of the decoded counter and for performing the purpose for which it is meant to perform i.e. controlling the movement of traffic. Each status of the display contains three LEDs. Each lane has one status facing it, the RED indicting "STOP" the yellow means "READY" and the Green means "MOVE" or "GO".

The duration of the various sections of the simulated traffic lights was done through the programming of the Arduino Uno embedded microprocessor using C++ programming language. The sample codes below show the sample source codes for the LEDs of the traffic lights.

Προγραμ Σαμπλε Χοδεσ

```
//ινιτιαλιζινγ τηε πινσ
ιντ γρεεν1=13;
ιντ ψελλοω1=12;
ιντ ρεδ1=11;
ιντ γρεεν2=10;
ιντ ψελλοω2=9;
ιντ ρεδ2=8;
ιντ γρεεν3=7;
ιντ ψελλοω3=6;
ιντ ρεδ3=5;

//σεττινγ υπ τηε στατυσ οφ τηε πινσ
ωοιδ σετυπ()
{
 πινΜοδε(γρεεν1, ΟΥΤΠΥΤ);
 πινΜοδε(ψελλοω1,ΟΥΤΠΥΤ);
 πινΜοδε(ρεδ1,ΟΥΤΠΥΤ);
 πινΜοδε(γρεεν2, ΟΥΤΠΥΤ);
 πινΜοδε(ψελλοω2,ΟΥΤΠΥΤ);
 πινΜοδε(ρεδ2,ΟΥΤΠΥΤ);
 πινΜοδε(γρεεν3, ΟΥΤΠΥΤ);
 πινΜοδε(ψελλοω3,ΟΥΤΠΥΤ);


 πινΜοδε(ρεδ3,ΟΥΤΠΥΤ);
}

//ρεπεατινγ τηε αχτιονσ οφ τηε πινσ
ωοιδ λοοπ()
{
 //τυρνινγ τηε γρεεν λιγητ οφ τραφφιχ 1 ΟΝ ανδ ιτσ ψελλ
οω ανδ ρεδ ΟΦΦ
 διγιταλΩριτε(γρεεν1,ΗΙΓΗ);
 διγιταλΩριτε(ψελλοω1,ΛΟΩ);
 διγιταλΩριτε(ρεδ1,ΛΟΩ);

 //ρεδ λιγητ οφ τραφφιχ 2 ισ τυρνεδ ΟΝ
 διγιταλΩριτε(γρεεν2,ΛΟΩ);
 διγιταλΩριτε(ψελλοω2,ΛΟΩ);
 διγιταλΩριτε(ρεδ2,ΗΙΓΗ);

 //ρεδ λιγητ οφ τραφφιχ 3 ισ τυρνεδ ΟΝ
 διγιταλΩριτε(γρεεν3,ΛΟΩ);
 διγιταλΩριτε(ψελλοω3,ΛΟΩ);
 διγιταλΩριτε(ρεδ3,ΗΙΓΗ);

 //τηε εντιρε αχτιον ισ δελαψεδ φορ 30 σεχονδσ
 δελαψ (30000);

 //ψελλοω λιγητ οφ τραφφιχ 2 ισ τυρνεδ ΟΝ ανδ δελαψεδ
φορ 10 σεχονδσ
 διγιταλΩριτε(ψελλοω2,ΗΙΓΗ);
 δελαψ(10000);

 //τηε ψελλοω λιγητ ισ μαδε το βλινκ 10 τιμεσ
  φορ(ιντ ι=1;ι<10;ι++)
{
   διγιταλΩριτε(ψελλοω2,ΗΙΓΗ);
   δελαψ(250);
   διγιταλΩριτε(ψελλοω2,ΛΟΩ);
```

```
  δελαψ(250);
}
  //γρεεν λιγητ οφ τραφφιχ 2 ισ τυρνεδ ON
διγιταλΩριτε(γρεεν2,ΗΙΓΗ);
διγιταλΩριτε(ψελλοω2,ΛΟΩ);
διγιταλΩριτε(ρεδ2,ΛΟΩ);
  //ρεδ λιγητ οφ τραφφιχ 1 ισ τυρνεδ ON
διγιταλΩριτε(γρεεν1,ΛΟΩ);
διγιταλΩριτε(ψελλοω1,ΛΟΩ);
διγιταλΩριτε(ρεδ1,ΗΙΓΗ);

//ρεδ λιγητ οφ τραφφιχ 3 ισ τυρνεδ ON
διγιταλΩριτε(γρεεν3,ΛΟΩ);
διγιταλΩριτε(ψελλοω3,ΛΟΩ);
διγιταλΩριτε(ρεδ3,ΗΙΓΗ);
```

1 = ON
0 = OFF

Key: R = Red, Y = Yellow, G = Green, N = North, E = East,

W = West

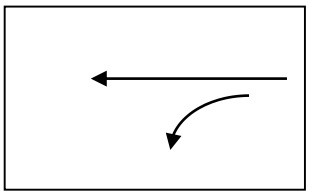


Fig.4: North Lane evaluation

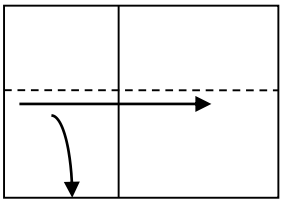


Fig.5: East lane evaluation

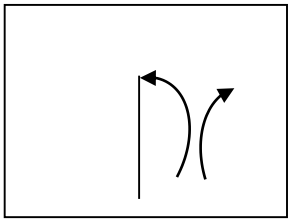


Fig.6: East lane evaluation

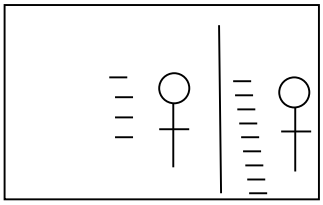


Fig.7: Pedestrians lane of evaluation

4.3 Results of the Simulation

After the assembling of the various components and the programming of the microcontroller traffic light control system, the following results were achieved:

(1) Automatic ON and OFF operation of the traffic light at the specified time

(2) Movement assignment to the busiest lane.

The performance evaluation of the traffic system was carried out, the model testing was performed at different occasions of light changing and the results below were achieved.

Initial time of operation of all the lanes = 550000ms (i.e. 65 seconds)

- Red duration =15,000ms
- Yellowing duration of blinking ten times each of = 10,000ms
- Green duration =30,000ms

Table 1: North Lane evaluation

	N	W	E
R	0	1	1
Y	0	1	0
G	1	0	0

Table 2: West lane evaluation

	N	W	E
R	1	0	1
Y	0	0	1
G	0	1	0

Table 3: East lane evaluation

	N	W	E
R	1	1	0
Y	1	1	0
G	0	0	1

Table 4: Pedestrians lane of evaluation

	N	W	E
R	1	1	1
Y	0	0	0
G	0	0	0

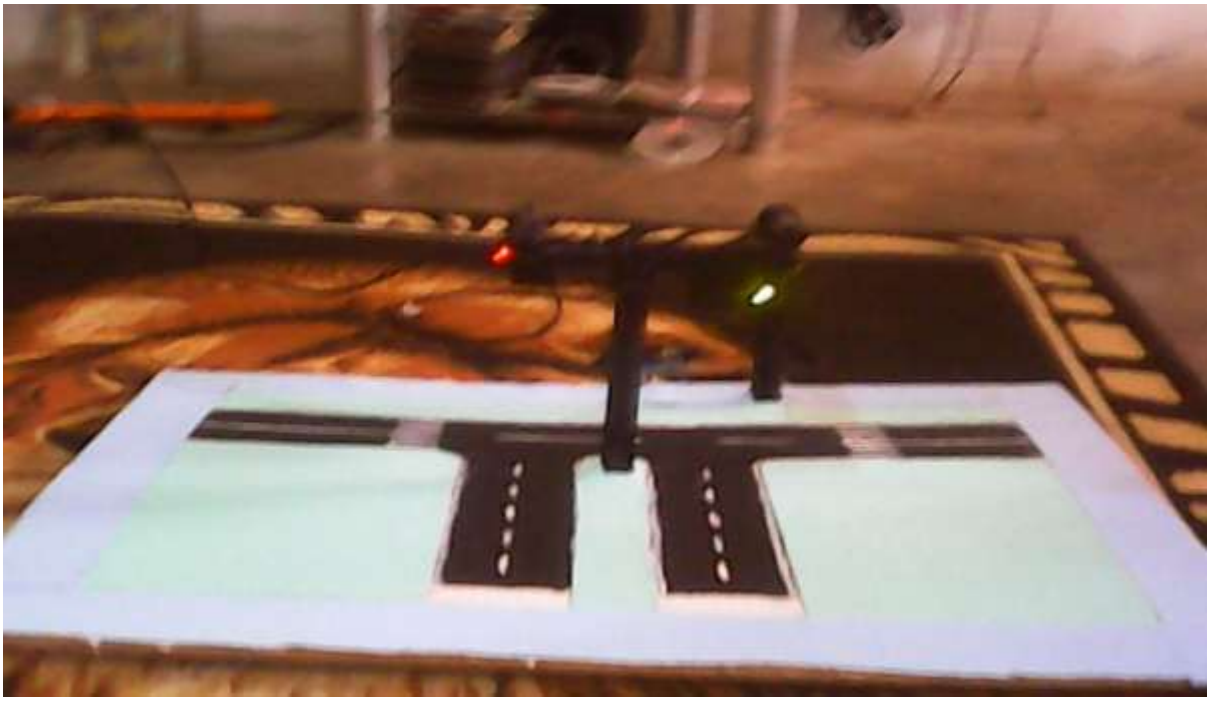


Fig.8: Output stand from the east (stand three)

Fig. 9: Output view from the north (stand one)

5.0 CONCLUSION

This work showed the practical application of simulation of traffic lights to improve traffic conditions. Traffic is a challenge in many cities of the world. Traffic flow is dictated by certain devices such as traffic lights. These traffic lights signal when each lane is able to pass through the intersection. The purpose of this work was to find a way to make intersections to be easily controlled with traffic lights. This goal was accomplished through the simulation of traffic lights using Arduino Uno environment and programmed with C++ programming language. From the design and construction of a simulated N-way traffic light controller carried out in the course of this work, it is obvious that such a system which if fully implemented will reduce the human stress of standing under favourable or unfavourable weather condition and controlling the movement of vehicles and pedestrians at N-way traffic intersections. Generally, this work also encourages the producing of an indigenous traffic control system which is easy to maintain,affodable and efficient in operation.

REFERENCES

[1] Mcshane, C (1999). *The Origins and Globalization of Traffic Control Signals*, Journal of Urban History / March, 1999. Retrieved from http://sites.tufts.edu/carscultureplace2010/files/2010/09/McShane-traffic-signals-1999.pdf

[2] Anonymous (2014). *Driving In America*. Retrieved from http://whatisusa.info/driving-in-america/

[3] Bellis, M (2014). *"The History of Roads and Asphalt"*. Retrieved from http://inventors.about.com/od/rstartinventions/a/History-Of-Roads.htm

[4] Masters, N (2013*). CityDig: Should I Stop or Should I Go? Early Traffic Signals in Los Angeles,* Los Angeles Magazine.

[5] Robert L., Gordon, P.E and Warren Tighe, P.E (2005). *Traffic Control Systems Handbook*,Office of Transportation Management Federal Highway Administration Room 3404 HOTM 400 Seventh Street Washington, D.C. 20590.

[6] Magomedov, T. G and Ostrovskiy, A. B (2006). *Simulation of Smart Traffic Lights*, Saint-Petersburg State University of Information Technologies, Mechanics and Optics Computer Technologies Department.

[7] Ferando, E. (2009). *Microcontrollers Fundamentals and Applications*, New York: CRC Press & Francis Group.

[8] Greenfield, J. (2000). *Digital Design using Integrated Circuits*, New York: Willey and Sons Incorporation.

[9] Mehta, V. (2008). *Principles of Electronics.* New Delhi: Chand Company.

[10] Morris, M. (2007). *Digital Logic and Computer Design*, New Delhi: Practice Hall of India.

[11] Mayank (2011). Getting Started, Microcontrollers, http://maxembedded.com/2011/06/mcu-vs-mpu/

[12] http://maxembedded.com/2011/06/mcu-vs-mpu/

[13] **https://www.arduino.cc/en/Reference/Board**

International Journal of Computer Applications Technology and Research
Volume 5– Issue 8, 551 - 555, 2016, ISSN:- 2319–8656

An Optimized Search Engine for Academics

Abbas Fadhil Mohammed Ali AL-Juboori
Department of Computer Science
University of Kerbala
Kerbala, Iraq

Abstract: Search engines are among the most useful and high-profile resources on the Internet. The problem of finding information on the Internet has been replaced with the problem of knowing where search engines are, what they are designed to retrieve, and how to use them. The main function of An Optimized Academic Search Engine is to allow its users to search for academic files. It also allows the users to specify query for searching phrases. The ranking and optimization was achieved for the result by the most website visit. The system have been designed by using PHP, MYSQL, and WAMP server.

Keywords: Search engine; Optimization; Academic; Information retrieval; Ranking

1. INTRODUCTION

Search Engine technology was born almost at the same time as the World Wide

Web [1], and has certainly improved dramatically over the past decade and become an integral part of everybody's Web browsing experience, especially after the phenomenal success of Google.

At the first glance, it appears that Search Engines have been studied very well, and many articles and theories including the paper by the founders of Google [2] have been published to describe and analyze their internal mechanisms.

1.2 The Basic Components of a Search Engine

All search engines includes:

1. A Web crawler.

2. A parser.

3. A ranking system.

4. A repository system.

5. A front-end interface.

These components are discussed individually below.

The starting point is a Web Crawler (or spider) to retrieve all Web pages: it simply traverses the entire Web or a certain subset of it, to download the pages or files it encounters and save for other components to use. The actual traversal algorithm varies depends on the implementation; depth first, breadth first, or random traversal are all being used to meet different design goals. The parser takes all downloaded raw results, analyze and eventually try to make sense out of them. In the case of a text search engine, this is done by extracting keywords and checking the locations and/or frequencies of them. Hidden HTML tags, such as KEYWORDS and DESCRIPTION are also considered. Usually a scoring system is involved to give a final point for each keyword on each page. Simple or complicated, a search engine must have a way to determine which pages are more important than the others, and present

them to users in a particular order. This is called the Ranking System. The most famous one is the Page Rank Algorithm published by Google founders [2].

A reliable repository system is definitely critical for any application. Search engine also requires everything to be stored in the most efficient way to ensure maximum performance. The choice of database vendor and the schema design can make big difference on performance for metadata such a URL description, crawling date, keywords, etc. More challenging part is the huge volume of downloaded files to be saved before they are picked up by other modules. Finally, a front-end interface for users: This is the face and presentation of the search engine. When a user submits a query, usually in the form of a list of textual terms, an internal scoring function is applied to each Web page in the repository [3],and the list of result is presented, usually in the order or relevance and importance .Google has been known for its simple and straight forward interface, while some most recent competitors, such as Ask.com, provide much richer user experience by adding features like preview or hierarchy displaying.

1.3 Search Engines Available Today

Other than well-known commercial products, such as Google, Yahoo and MSN, there are many open source Search Engines, for example, ASPSeek, BBDBot,Datapark Search, and ht://Dig. Evaluating their advantages and disadvantages is not the purpose of this thesis, but based on reviews and feedbacks from other people [4], they are either specialized only in a particular area, or not adopting good ranking algorithms, or have not been maintained for quite a while. Another important fact is that while most current search engines are focused on text, there is an inevitable trend that they are being extended to the multi-media arena, including dynamic contents, images, sounds and others [5]. None of the open source engines listed above has multimedia searching modules, and none of them is flexible enough to add new ones without significant effort.

1.4 Issues in Search Engine Research

Design of Web crawlers: Web crawler, also known as robot, spider, worm, and wanderer, is no doubt the first part of any search engine and designing a web crawler is a complex endeavor. Due to the competitive nature of the search engine business, there are very few papers in the literature describing the challenges and tradeoffs inherent in web crawler design [6]. Page ranking system: Page Rank [2] .is a system of scoring nodes in a directed graph

based on the stationary distribution of a random walk on the directed graph. Conceptually, the score of a node corresponds to the frequency with which the node is visited as an individual strolls randomly through the graph. Motivated largely by the success and scale of Google's Page Rank ranking function, much research has emerged on efficiently computing the stationary distributions of Web-scale Markov chain, the mathematical mechanism underlying Page Rank. The main challenge is that the Web graph is so large that its edges typically only exist in external memory and an explicit representation of its stationary distribution just barely fits in to main memory[7]. Repository freshness: A search engine uses its local repository to assign scores to the Web pages in response to a query, with the implicit assumption that the repository closely mirrors the current Web [3]. However, it is infeasible to maintain an exact mirror of a large portion of the Web due to its considerable aggregate size and dynamic nature, combined with the autonomous nature of Web servers. If the repository is not closely synchronized with the Web, the search engine may not include the most useful pages, for a query at the top of the result list. The repository has to be updated so as to maximize the overall quality of the user experience. Evaluating the feedback from users: Two mechanisms have been commonly used to accomplish this purpose: Click Popularity and Stickiness [8]. Click Popularity calculates how often a record in the returned list is actually clicked by the user, and promote/demote its rank accordingly. Stickiness assumes the longer an end user stays on a particular page, the more important it must be. While being straightforward, the implementation of these two algorithms can be quite error prone. The data collecting the most difficult part, as the server has to uniquely identify each user. This has been further complicated by the fact that many people want to manually or programmatically promote their own Web sites by exploiting the weaknesses of certain implementations [9]. Two graduate students at UCCS [10][11] have been working on an Image search engine and a text search engine, respectively. Part of their work is to adopt the published Page Rank algorithm [2], and the results are quite promising. However, giving the experimental nature of these two projects, they are not suitable for scaling up and not mature enough to serve as a stable platform for future research. A complete redesign and overhaul is needed.

1.5 The Original Page Rank algorithm

Google is known for its famous Page Rank algorithm, a way to measure the importance of a Web page by counting how many other pages link to it, as well as how important those page themselves are. The published Page Rank algorithm can be described in a very simple manner:

PR(A) = (1-d) + d (PR(T1)/C(T1) + ... + PR(Tn)/C(Tn))

In the equation above: PR(Tn): Each page has a notion of its own self-importance. That's "PR(T1)" for the first page in the web all the way up to PR(Tn) for the last page. C(Tn): Each page spreads its vote out evenly amongst all of its outgoing links. The count, or number, of outgoing links for page 1 is C(T1), C(Tn) for page n, and so onfor all pages.PR(Tn)/C(Tn): if a page (page A) has a back link from page N, the share ofthe vote page A gets is PR(Tn)/C(Tn). d: All these fractions of votes are added together but, to stop the other pages having too much influence, this total vote is "damped down" by multiplying it by 0.85(the factor d). The definition of d also came from an intuitive basis in random walks on graphs. The idea is that a random surfer keeps clicking on successive links at random, but the surfer periodically "gets bored" and jumps to a random page. The probability that the surfer gets bored is the dampening factor.(1 - d): The (1 – d) bit at the beginning is a probability math magic so the "sum of all Web pages" Page Rank is 1, achieved by adding the part lost by the d(....) calculation. It also means that if a page has no links to it, it still gets a small PR of 0.15 (i.e. 1 – 0.85). At the first glance, there is a paradox. In order to calculate the PR of page A, one must first have the PR of all other pages, whose Page Rank is calculated in the same way. The algorithm solves it by first assuming all pages to have the same PR of 1, and at each iteration PR is propagated to other pages until all PR stabilize to within some threshold. Because the large dataset PR algorithm deals with, measuring the stabilization of the PRs can be a difficult job itself. Research indicates that in some cases PR can be calculated in as few as 10 iterations [12], or it may take more than 100 iterations [13]. Another important fact is that when a page does not have outgoing links, the C(Tn), this page becomes a dangling URL, and must be removed from the whole picture. If such "pruning" was not done, the dangling may have critical implications in terms of computation. First, Page Rank values are likely to be smaller than they should be, and might become all zero in the worst case. Second, the iteration process might not converge to a fixed point [14] .

1.6 Crawler

A primitive implementation was written at very early stage of the project to retrieve some data for other modules to work with. While functioning correctly, this version rather is plain in terms of features: it is single threaded and does not have retrying, repository refreshing, URL hashing, smart checking on dynamic URLs, smart recognizing on file types, and avoiding crawler traps, etc. Its speed is also quite questionable and can only retrieve about 2000 URLs per hour on a fast network in the UCCS lab. Improvements can be made to add the features above and improve its speed. Fortunately two UCCS graduate students are already working on this area [14] .

1.7 Parsers

Same as the crawler, a simple functional text parser was written to glue the whole system together. It only parses certain selected areas of a document such as Meta data, title, anchor text, three levels of headers, and a short part at the beginning of each paragraph. A complete full text parser with satisfactory performance is in immediate need. Image processing is not currently implemented [14] .

2. INFORMATION RETRIEVAL AND RANKING

Web search engines return lists of web pages sorted by the page's relevance to the user query. The problem with web search relevance ranking is to estimate relevance of a page to a query. Nowadays, commercial web-page search engines combine hundreds of features to estimate relevance. The specific features and their mode of combination are kept secret to fight spammers and competitors. Nevertheless, the main types of features at use, as well as the methods for their combination, are publicly known and are the subject of scientific investigation.

Information Retrieval (IR) Systems are the predecessors of Web and search engines. These systems were designed to retrieve documents in curated digital collections such as library abstracts, corporate documents, news, etc. Traditionally, IR relevance ranking algorithms were designed to obtain high recall on medium-sized document collections using long detailed queries. Furthermore, textual documents in these collections had little or no structure or hyperlinks. Web search engines incorporated many of the principles and algorithms of Information Retrieval Systems, but had to adapt and extend them to fit their needs. Early Web Search engines such as Lycos and AltaVista concentrated on the scalability issues of running web search engines using traditional relevance ranking algorithms. Newer search engines, such as Google, exploited web-specific relevance features such as hyperlinks to obtain significant gains in quality. These measures were partly motivated by research in citation analysis carried out in the biblio metrics field. For most queries, there exist thousands of documents containing some or all of the terms in the query. A search engine needs to rank them in some appropriate way so that the first few results shown to the user will be the ones that are most pertinent to the user's need. The interest of a document with respect to the user query is referred to as "document relevance." this quantity is usually unknown and must be estimated from features of the document, the query, the user history or the web in general. Relevance ranking loosely refers to the different features and algorithms used to estimate the relevance of documents and to sort them appropriately. The most basic retrieval function would be a Boolean query on the presence or absence of terms in documents. Given a query "word1 word2" the Boolean AND query would return all documents containing the terms word1 and word2 at least once. These documents are referred to as the query's "AND result set" and represent the set of potentially relevant documents; all documents not in this set could be considered irrelevant and ignored. This is usually the first step in web search relevance ranking. It greatly reduces the number of documents to be considered for ranking, but it does not rank the documents in the result set. For this, each document needs to be "scored", that is, the document's relevance needs to be estimated as a function of its relevance features. Contemporary search engines use hundreds of features. These features and their combination are kept secret to fight spam and competitors. Nevertheless, the general classes of employed features are Publicly known and are the subject of scientific investigation. The main types of relevance features are described in the remainder of this section, roughly in order of importance. Note that some features are query-dependent and some are not. This is an important distinction because query-independent features are constant with respect to the user query and can be pre-computed off-line. Query-dependent features, on the other hand, need to be computed at search time or cached [15].

3. SYSTEM DESIGN

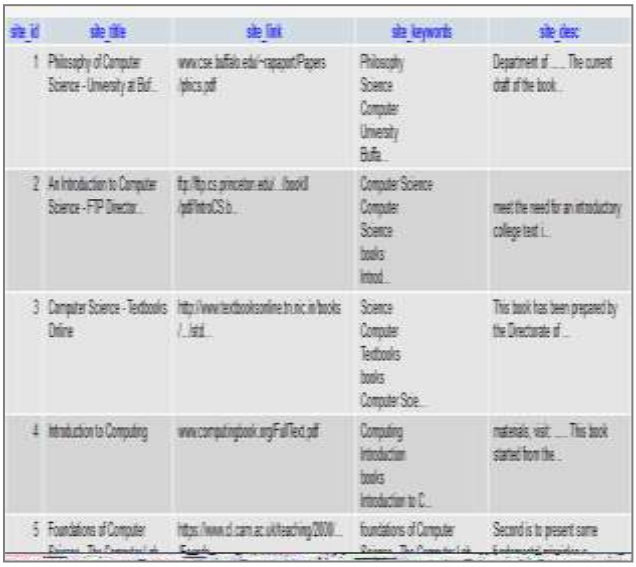
The displayed search results based on the number of visits .The system designed by using HTML, PHP and MYSQL. And WampServer.

Our system divided into two sides, client side and server side which contain the database of the system.

3.1. Database

Our of System consists of Database which is built in MYSQL. The type of data entered is (PDF, DOC, and PPT).it contains six fields which are explained in the table (1) blow:-

Table (1): Database of the system



site_id	site_title	site_link	site_keywords	site_desc
1	Philosophy of Computer Science - University at Buf...	www.cse.buffalo.edu/~rapaport/Papers /phics.pdf	Philosophy Science Computer University Buffa...	Department of .... The current draft of the book...
2	An Introduction to Computer Science - FTP Director...	ftp://ftp.cs.princeton.edu/.../bookB /pdf/IntroCS.b...	Computer Science Computer Science books Introd...	meet the need for an introductory college text i...
3	Computer Science - Textbooks Online	http://www.textbooksonline.tn.nic.in/books /...text...	Science Computer Textbooks books Computer Scie...	This book has been prepared by the Directorate of ...
4	Introduction to Computing	www.computingbook.org/FullText.pdf	Computing Introduction books Introduction to C...	materials, visit: .... This book started from the...
5	Foundations of Computer	https://www.cl.cam.ac.uk/teaching/2009/...	foundations of Computer	Second is to present some

The description of the table above explained as follows:

1-site_id: it is the primary key of database.
2-site_title: contain Web addresses.
3- Site_link: contain URLs.
4- site_keywords : It contains reserved words that are on the basis of which Search.
5- Site_desc: It has a simple description of the sites.
6- site_counter : A dynamic where it calculates the number of visits to the site.

4. SYSTEM IMPLEMENTATION

The following three steps in the process are:

1. Entering the word, or phrase of the file to be searched.
2. Getting the search results, or receiving the list of found documents back to terminal.
3. Finding the right file, or the information you were looking for and downloading to our own terminal.

This system can be implement by opening the first page of the system site as shown in figure(1) :

Figure (1):the Home page

We can write the keyword we need in search bar (any words or phrase) to search about, for example (Computer Science) and after that click on (Search) , the search results for that keyword will appear as shown in figure(2):

Figure (2): (computer science)Search results

Another example (Thesis) as shown in figure (3):-

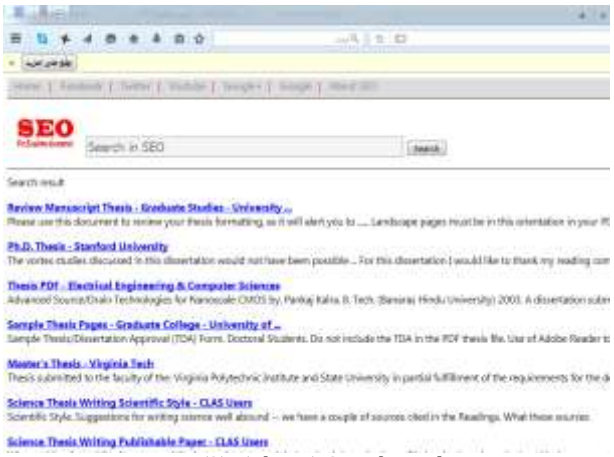


Figure (3): (Thesis) Search results

Ranking used in this system depending on the most Visit of any website included in the database. For example when we write (Articles) in the search bar, after that the results appear. If we enter the link time (Read the 5 Most Download Articles in 2011 for Free!) as shown in figure below (4):

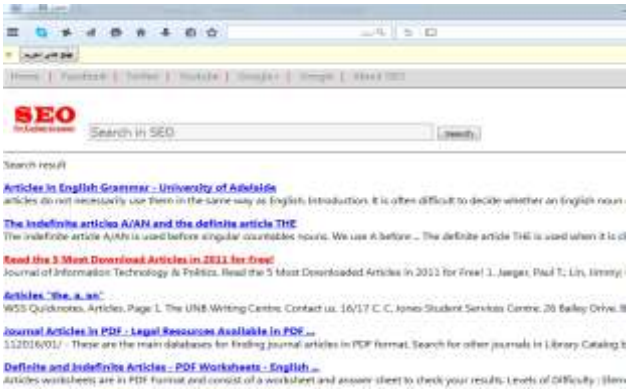


Figure (4): Search results before ranking

for the first time, and then visit this link many times more than the other links, the ranking of the search result will be changed when we write the same keyword in the search bar as shown in figure (5)below:

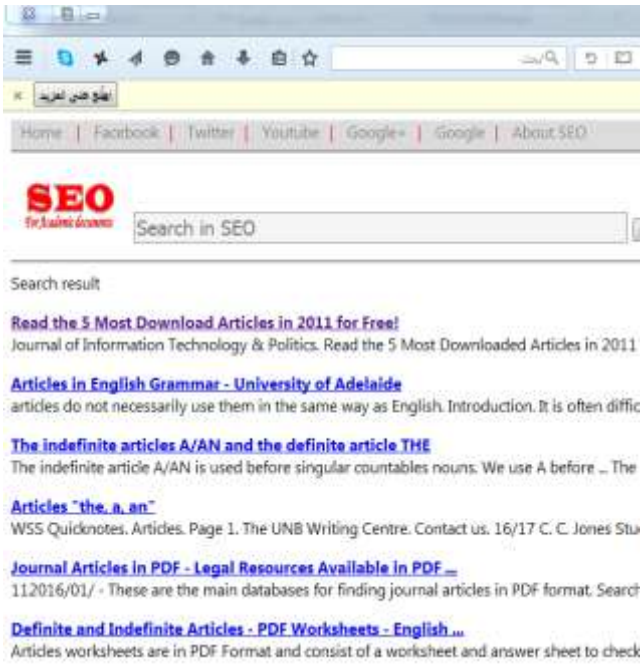


Figure (5): Search results after ranking

5. CONCLUSIONS

1-In this paper we conclude that there is an ability to search for information has already been entered into the database. The ranking in our search engine was achieved by using the most visit of any website included in the database. The suggestions we recommend to be achieved in the future works is to add Boolean operators to help in the search and increase the size of the database, also we can recommend to choose other Ranking algorithms to include the system.

6. REFERENCES

[1] Wall, Aaron. History of Search Engines & Web History. Visited November, 2005.

[2] Brin, Sergey and Lawrence Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. Proceedings of the Seventh International Conference on World Wide Web 7, Brisbane, Australia, Pages 107 – 117, 1998

[3] Pandey, Sandeep and Christopher Olston, User-Centric Web Crawling, International World Wide Web Conference, Chiba, Japan, May 10-14, 2005.

[4] Morgan, Eric. Comparing Open Source Indexers. O'Reilly Open Source Software Conference, San Diego, CA, July 23-27, 2001.

[5] Wall, Aaron. Future of Search Engines. Visited November, 200**5.**

[6] Allan Heydon, Marc Najork, Mercator: A scalable, extensible Web Crawler, World Wide Web 2, Pages 219-229, 1999

[7] McSherry, Frank, A Uniform Approach to Accelerated Page Rank Computation, International World Wide Web Conference, Chiba, Japan, May 10-14, 2005.

[8] Nowack, Craig. Using Topological Constructs To Model Interactive Information Retrieval Dialogue In The Context Of Belief, Desire, and Intention Theory. Dissertation of Ph.D. Pennsylvania State University, Pennsylvania, 2005.

[9] Harpf, Lauri. Free website promotion tutorial. Visited Nov, 2005.

http://www.apromotionguide.com/

[10] Jacobs, Jing. CatsSearch An Improved Search Engine Design For web pages in the UCCS Domain. University Of Colorado at Colorado Springs, December, 2005.

[11] Kodavanti, Apparao. Implementation of an Image Search Engine. University Of Colorado at Colorado Springs, December, 2005.

[12] Haveliwala, Taher H. Effcient Computation of Page Rank. Technical Report. Stanford University, California, 1999

[13] Kamvar Sepandar D, Taher H. Haveliwala, Christopher D. Manning, and Gene H. Golub. Extrapolation methods for accelerating Page Rank computations. The 12th International. Conference on the World Wide Web, pages 261–270, 2003.

[14] Kim, Sung Jin and Sang Ho Lee. An improved computation of the Page Rank algorithm. The European Conference on Information Retrieval (ECIR), pages 73–85, 2002.

[15] Hugo Zaragoza and Marc Najork Web Search Relevance Ranking. Yahoo Research Group, Barcelona, Spain. 2009.